

ArandanoIRT Backend
1.0

Generated by Doxygen 1.10.0

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	9
3.1 Class List	9
4 File Index	17
4.1 File List	17
5 Namespace Documentation	23
5.1 ArandanoIRT_Backend Namespace Reference	23
5.2 ArandanoIRT_Backend.Application Namespace Reference	23
5.3 ArandanoIRT_Backend.Application.DTOs Namespace Reference	23
5.4 ArandanoIRT_Backend.Application.DTOs.Auth Namespace Reference	24
5.5 ArandanoIRT_Backend.Application.DTOs.Objetcts Namespace Reference	24
5.6 ArandanoIRT_Backend.Application.DTOs.Person Namespace Reference	24
5.7 ArandanoIRT_Backend.Application.Interfaces Namespace Reference	25
5.8 ArandanoIRT_Backend.Application.Interfaces.Auditing Namespace Reference	25
5.8.1 Function Documentation	25
5.8.1.1 AuditMetadata()	25
5.9 ArandanoIRT_Backend.Application.Interfaces.IServices Namespace Reference	26
5.10 ArandanoIRT_Backend.Application.Interfaces.Utilities Namespace Reference	26
5.11 ArandanoIRT_Backend.Application.Mappings Namespace Reference	26
5.12 ArandanoIRT_Backend.Application.Services Namespace Reference	27
5.13 ArandanoIRT_Backend.Application.Utilities Namespace Reference	27
5.14 ArandanoIRT_Backend.Domain Namespace Reference	27
5.15 ArandanoIRT_Backend.Domain.Entities Namespace Reference	27
5.16 ArandanoIRT_Backend.Domain.Exceptions Namespace Reference	29
5.17 ArandanoIRT_Backend.Domain.IRepositories Namespace Reference	29
5.18 ArandanoIRT_Backend.Domain.ValueObjects Namespace Reference	30
5.18.1 Function Documentation	31
5.18.1.1 ClientInfo()	31
5.19 ArandanoIRT_Backend.Infrastructure Namespace Reference	31
5.20 ArandanoIRT_Backend.Infrastructure.Attributes Namespace Reference	32
5.21 ArandanoIRT_Backend.Infrastructure.Config Namespace Reference	32
5.22 ArandanoIRT_Backend.Infrastructure.Data Namespace Reference	32
5.23 ArandanoIRT_Backend.Infrastructure.Filters Namespace Reference	33
5.24 ArandanoIRT_Backend.Infrastructure.Interfaces Namespace Reference	33

5.25 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices Namespace Reference	33
5.26 ArandanoIRT_Backend.Infrastructure.Persistence Namespace Reference	34
5.27 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing Namespace Reference	34
5.28 ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors Namespace Reference	34
5.29 ArandanoIRT_Backend.Infrastructure.Repositories Namespace Reference	35
5.29.1 Function Documentation	35
5.29.1.1 StatusRepository()	35
5.30 ArandanoIRT_Backend.Infrastructure.Services Namespace Reference	36
5.31 ArandanoIRT_Backend.Infrastructure.Utilities Namespace Reference	37
5.32 ArandanoIRT_Backend.Tests Namespace Reference	38
5.33 ArandanoIRT_Backend.Tests.Application Namespace Reference	38
5.34 ArandanoIRT_Backend.Tests.Application.Services Namespace Reference	38
5.35 ArandanoIRT_Backend.Tests.Application.Utilities Namespace Reference	38
5.36 ArandanoIRT_Backend.Tests.Integration Namespace Reference	39
5.37 ArandanoIRT_Backend.Tests.Integration.Common Namespace Reference	39
5.38 ArandanoIRT_Backend.Tests.Integration.Repositories Namespace Reference	39
5.39 ArandanoIRT_Backend.Tests.Unit Namespace Reference	39
5.40 ArandanoIRT_Backend.Tests.Unit.Domain Namespace Reference	40
5.41 ArandanoIRT_Backend.Tests.Unit.Domain.Entities Namespace Reference	40
5.42 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects Namespace Reference	40
5.43 ArandanoIRT_Backend.Tests.Unit.Infrastructure Namespace Reference	40
5.44 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence Namespace Reference	41
5.45 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services Namespace Reference	41
5.46 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities Namespace Reference	41
5.47 ArandanoIRT_Backend.UI Namespace Reference	42
5.48 ArandanoIRT_Backend.UI.Controllers Namespace Reference	42
5.49 ArandanoIRT_Backend.UI.Middleware Namespace Reference	42
6 Class Documentation	43
6.1 ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto Class Reference	43
6.1.1 Detailed Description	44
6.1.2 Property Documentation	44
6.1.2.1 Email	44
6.1.2.2 FirstName	44
6.1.2.3 LastName	44
6.1.2.4 Password	44
6.2 ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext Class Reference	45
6.2.1 Constructor & Destructor Documentation	47
6.2.1.1 ApplicationDbContext() [1/2]	47

6.2.1.2 ApplicationDbContext() [2/2]	47
6.2.2 Member Function Documentation	47
6.2.2.1 OnConfiguring()	47
6.2.2.2 OnModelCreating()	48
6.2.3 Property Documentation	48
6.2.3.1 Auditcrop	48
6.2.3.2 Auditdatatable	48
6.2.3.3 Auditdevice	48
6.2.3.4 Auditperson	48
6.2.3.5 Auditsensitizeddata	48
6.2.3.6 Auditsystemtable	48
6.2.3.7 Changepassword	48
6.2.3.8 Crop	49
6.2.3.9 Cropinvitation	49
6.2.3.10 Deviceactivation	49
6.2.3.11 Devicedata	49
6.2.3.12 Devicelog	49
6.2.3.13 Devicetoken	49
6.2.3.14 Failedloginattempt	49
6.2.3.15 Person	49
6.2.3.16 Plantdata	50
6.2.3.17 Plantobservation	50
6.2.3.18 Plantstatehistory	50
6.2.3.19 RefreshToken	50
6.2.3.20 Sensordata	50
6.2.3.21 Status	50
6.2.3.22 Tablerelation	50
6.2.3.23 Thermaldata	50
6.3 ArandanoIRT_Backend.Infrastructure.Data.Auditcrop Class Reference	51
6.3.1 Property Documentation	51
6.3.1.1 Action	51
6.3.1.2 Columnname	52
6.3.1.3 Cropid	52
6.3.1.4 Iauditcrop	52
6.3.1.5 Newvalue	52
6.3.1.6 Oldvalue	52
6.3.1.7 Performedat	52
6.3.1.8 Performedby	52
6.3.1.9 Performedbyip	52

6.3.1.10 Recordid	52
6.3.1.11 Useragent	53
6.4 ArandanoIRT_Backend.Domain.Entities.AuditCropEntity Class Reference	53
6.4.1 Detailed Description	54
6.4.2 Constructor & Destructor Documentation	54
6.4.2.1 AuditCropEntity()	54
6.4.3 Property Documentation	55
6.4.3.1 Action	55
6.4.3.2 ColumnName	55
6.4.3.3 CropId	55
6.4.3.4 IdAuditCrop	56
6.4.3.5 NewValue	56
6.4.3.6 OldValue	56
6.4.3.7 PerformedAt	56
6.4.3.8 PerformedBy	56
6.4.3.9 PerformedByIp	56
6.4.3.10 RecordId	57
6.4.3.11 UserAgent	57
6.5 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator Class Reference	57
6.5.1 Detailed Description	59
6.5.2 Constructor & Destructor Documentation	59
6.5.2.1 AuditCropGenerator()	59
6.5.3 Member Function Documentation	59
6.5.3.1 CanHandle()	59
6.5.3.2 GenerateEntries()	60
6.6 ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository Class Reference	61
6.6.1 Member Function Documentation	63
6.6.1.1 Create()	63
6.6.1.2 Delete()	63
6.6.1.3 GetAll()	63
6.6.1.4 GetById()	63
6.6.1.5 Update()	63
6.7 ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable Class Reference	64
6.7.1 Property Documentation	65
6.7.1.1 Action	65
6.7.1.2 Columnname	65
6.7.1.3 Cropid	65
6.7.1.4 Iauditdatatable	65
6.7.1.5 Newvalue	65

6.7.1.6 Oldvalue	65
6.7.1.7 Performedat	65
6.7.1.8 Performedby	65
6.7.1.9 Performedbyip	65
6.7.1.10 Recordid	66
6.7.1.11 Tablename	66
6.7.1.12 Useragent	66
6.8 ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity Class Reference	66
6.8.1 Detailed Description	67
6.8.2 Constructor & Destructor Documentation	68
6.8.2.1 AuditDataTableEntity()	68
6.8.3 Property Documentation	68
6.8.3.1 Action	68
6.8.3.2 ColumnName	69
6.8.3.3 Cropld	69
6.8.3.4 IdAuditDataTable	69
6.8.3.5 NewValue	69
6.8.3.6 OldValue	69
6.8.3.7 PerformedAt	69
6.8.3.8 PerformedBy	70
6.8.3.9 PerformedByIp	70
6.8.3.10 RecordId	70
6.8.3.11 TableName	70
6.8.3.12 UserAgent	70
6.9 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator Class Reference	71
6.9.1 Detailed Description	73
6.9.2 Constructor & Destructor Documentation	73
6.9.2.1 AuditDataTableGenerator()	73
6.9.3 Member Function Documentation	73
6.9.3.1 CanHandle()	73
6.9.3.2 GenerateEntries()	74
6.10 ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository Class Reference	75
6.10.1 Member Function Documentation	77
6.10.1.1 Create()	77
6.10.1.2 Delete()	77
6.10.1.3 GetAll()	77
6.10.1.4 GetById()	77
6.10.1.5 Update()	78
6.11 ArandanoIRT_Backend.Infrastructure.Data.Auditdevice Class Reference	78

6.11.1 Property Documentation	79
6.11.1.1 Action	79
6.11.1.2 Columnname	79
6.11.1.3 Cropid	79
6.11.1.4 Idauditdevice	79
6.11.1.5 Newvalue	79
6.11.1.6 Oldvalue	79
6.11.1.7 Performedat	79
6.11.1.8 Performedby	80
6.11.1.9 Performedbyip	80
6.11.1.10 Recordid	80
6.11.1.11 Useragent	80
6.12 ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity Class Reference	80
6.12.1 Detailed Description	82
6.12.2 Constructor & Destructor Documentation	82
6.12.2.1 AuditDeviceEntity()	82
6.12.3 Property Documentation	83
6.12.3.1 Action	83
6.12.3.2 ColumnName	83
6.12.3.3 Cropld	83
6.12.3.4 IdAuditDevice	83
6.12.3.5 NewValue	83
6.12.3.6 OldValue	84
6.12.3.7 PerformedAt	84
6.12.3.8 PerformedBy	84
6.12.3.9 PerformedBylp	84
6.12.3.10 RecordId	84
6.12.3.11 UserAgent	84
6.13 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator Class Reference	85
6.13.1 Detailed Description	87
6.13.2 Constructor & Destructor Documentation	87
6.13.2.1 AuditDeviceGenerator()	87
6.13.3 Member Function Documentation	87
6.13.3.1 CanHandle()	87
6.13.3.2 GenerateEntries()	88
6.14 ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository Class Reference	89
6.14.1 Member Function Documentation	91
6.14.1.1 Create()	91
6.14.1.2 Delete()	91

6.14.1.3 GetAll()	91
6.14.1.4 GetById()	91
6.14.1.5 Update()	92
6.15 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService Class Reference	92
6.15.1 Detailed Description	95
6.15.2 Constructor & Destructor Documentation	95
6.15.2.1 AuditHelperService()	95
6.15.3 Member Function Documentation	95
6.15.3.1 GetCropIdValue()	95
6.15.3.2 GetPrimaryKeyValue()	96
6.15.3.3 GetValuesDictionary() [1/2]	96
6.15.3.4 GetValuesDictionary() [2/2]	97
6.15.3.5 SerializePropertyValueDictionary()	98
6.15.4 Member Data Documentation	98
6.15.4.1 UnknownTableName	98
6.16 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests Class Reference	98
6.16.1 Detailed Description	101
6.16.2 Constructor & Destructor Documentation	101
6.16.2.1 AuditHelperServiceTests()	101
6.16.3 Member Function Documentation	101
6.16.3.1 Dispose()	101
6.16.3.2 GetCropIdValue_ForDeletedEntityWithCropId_ShouldReturnCropId()	102
6.16.3.3 GetCropIdValue_ForEntityWithCropId_ShouldReturnCropId()	102
6.16.3.4 GetCropIdValue_ForEntityWithNullCropId_ShouldReturnNull()	102
6.16.3.5 GetCropIdValue_EntityWithoutCropIdProperty_ShouldReturnNull()	102
6.16.3.6 GetPrimaryKeyValue_ForAddedEntity_ShouldReturnNull()	102
6.16.3.7 GetPrimaryKeyValue_ForCompositePkEntity_ShouldReturnNull()	102
6.16.3.8 GetPrimaryKeyValue_ForDeletedEntity_ShouldReturnCorrectPk()	102
6.16.3.9 GetPrimaryKeyValue_ForModifiedEntity_ShouldReturnCorrectPk()	102
6.16.3.10 GetPrimaryKeyValue_StringPkEntity_ShouldReturnNull()	103
6.16.3.11 GetValuesDictionary_WithAdditionalExclusions_ShouldExcludeMoreProperties()	103
6.16.3.12 GetValuesDictionary_WithCurrentValues_ShouldReturnDictionaryWithoutExcluded()	103
6.16.3.13 GetValuesDictionary_WithNullPropertyValues_ShouldReturnNull()	103
6.16.3.14 GetValuesDictionary_WithOriginalValues_ShouldReturnDictionaryWithoutExcluded()	103
6.16.3.15 SerializePropertyValueDictionary_WithEmptyDictionary_ShouldReturnNull()	103
6.16.3.16 SerializePropertyValueDictionary_WithError_ShouldHandleGracefully()	103
6.16.3.17 SerializePropertyValueDictionary_WithNullDictionary_ShouldReturnNull()	103
6.16.3.18 SerializePropertyValueDictionary_WithValidDictionary_ShouldReturnJsonString()	104
6.17 ArandanoIRT_Backend.Infrastructure.Data.Auditperson Class Reference	104

6.17.1 Property Documentation	105
6.17.1.1 Action	105
6.17.1.2 Columnname	105
6.17.1.3 Cropid	105
6.17.1.4 Idauditperson	105
6.17.1.5 Performedat	105
6.17.1.6 Performedby	105
6.17.1.7 Performedbyip	105
6.17.1.8 Recordid	105
6.17.1.9 Useragent	105
6.18 ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity Class Reference	106
6.18.1 Detailed Description	107
6.18.2 Constructor & Destructor Documentation	107
6.18.2.1 AuditPersonEntity()	107
6.18.3 Property Documentation	108
6.18.3.1 Action	108
6.18.3.2 ColumnName	108
6.18.3.3 Cropld	108
6.18.3.4 IdAuditPerson	108
6.18.3.5 PerformedAt	108
6.18.3.6 PerformedBy	109
6.18.3.7 PerformedBylp	109
6.18.3.8 RecordId	109
6.18.3.9 UserAgent	109
6.19 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator Class Reference	110
6.19.1 Detailed Description	112
6.19.2 Constructor & Destructor Documentation	112
6.19.2.1 AuditPersonGenerator()	112
6.19.3 Member Function Documentation	112
6.19.3.1 CanHandle()	112
6.19.3.2 GenerateEntries()	113
6.20 ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository Class Reference	114
6.20.1 Member Function Documentation	116
6.20.1.1 Create()	116
6.20.1.2 Delete()	116
6.20.1.3 GetAll()	116
6.20.1.4 GetById()	116
6.20.1.5 Update()	117

6.21 ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor Class Reference	117
6.21.1 Detailed Description	118
6.21.2 Constructor & Destructor Documentation	118
6.21.2.1 AuditSaveChangesInterceptor()	118
6.21.3 Member Function Documentation	119
6.21.3.1 SavingChangesAsync()	119
6.22 ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata Class Reference	120
6.22.1 Property Documentation	121
6.22.1.1 Action	121
6.22.1.2 Columnname	121
6.22.1.3 Cropid	121
6.22.1.4 Idauditsensitizeddata	121
6.22.1.5 Performedat	121
6.22.1.6 Performedby	121
6.22.1.7 Performedbyip	121
6.22.1.8 Recordid	121
6.22.1.9 Tablename	122
6.22.1.10 Useragent	122
6.23 ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity Class Reference	122
6.23.1 Detailed Description	123
6.23.2 Constructor & Destructor Documentation	124
6.23.2.1 AuditSensitiveDataEntity()	124
6.23.3 Property Documentation	124
6.23.3.1 Action	124
6.23.3.2 ColumnName	125
6.23.3.3 Cropld	125
6.23.3.4 IdAuditSensitiveData	125
6.23.3.5 PerformedAt	125
6.23.3.6 PerformedBy	125
6.23.3.7 PerformedBylp	125
6.23.3.8 RecordId	126
6.23.3.9 TableName	126
6.23.3.10 UserAgent	126
6.24 ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository Class Reference	127
6.24.1 Member Function Documentation	129
6.24.1.1 Create()	129
6.24.1.2 Delete()	129
6.24.1.3 GetAll()	129

6.24.1.4 GetById()	129
6.24.1.5 Update()	130
6.25 ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable Class Reference	130
6.25.1 Property Documentation	131
6.25.1.1 Action	131
6.25.1.2 Columnname	131
6.25.1.3 Cropid	131
6.25.1.4 Idauditsystemtable	131
6.25.1.5 Newvalue	131
6.25.1.6 Oldvalue	131
6.25.1.7 Performedat	131
6.25.1.8 Performedby	132
6.25.1.9 Performedbyip	132
6.25.1.10 Recordid	132
6.25.1.11 Tablename	132
6.25.1.12 Useragent	132
6.26 ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity Class Reference	132
6.26.1 Detailed Description	134
6.26.2 Constructor & Destructor Documentation	134
6.26.2.1 AuditSystemTableEntity()	134
6.26.3 Property Documentation	135
6.26.3.1 Action	135
6.26.3.2 ColumnName	135
6.26.3.3 CropId	135
6.26.3.4 IAuditSystemTable	135
6.26.3.5 NewValue	136
6.26.3.6 OldValue	136
6.26.3.7 PerformedAt	136
6.26.3.8 PerformedBy	136
6.26.3.9 PerformedByIp	136
6.26.3.10 RecordId	136
6.26.3.11 TableName	137
6.26.3.12 UserAgent	137
6.27 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator Class Reference	137
6.27.1 Detailed Description	139
6.27.2 Constructor & Destructor Documentation	139
6.27.2.1 AuditSystemTableGenerator()	139
6.27.3 Member Function Documentation	139
6.27.3.1 CanHandle()	139

6.27.3.2 GenerateEntries()	140
6.28 ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository Class Reference	141
6.28.1 Member Function Documentation	143
6.28.1.1 Create()	143
6.28.1.2 Delete()	143
6.28.1.3 GetAll()	143
6.28.1.4 GetById()	143
6.28.1.5 Update()	144
6.29 ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter Class Reference	144
6.29.1 Detailed Description	145
6.29.2 Member Function Documentation	145
6.29.2.1 Apply()	145
6.30 ArandanoIRT_Backend.Application.Services.AuthPasswordService Class Reference	146
6.30.1 Detailed Description	148
6.30.2 Constructor & Destructor Documentation	148
6.30.2.1 AuthPasswordService()	148
6.30.3 Member Function Documentation	148
6.30.3.1 RequestPasswordResetAsync()	148
6.30.3.2 ResetPasswordAsync()	149
6.31 ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests Class Reference	149
6.31.1 Detailed Description	151
6.31.2 Constructor & Destructor Documentation	151
6.31.2.1 AuthPasswordServiceTests()	151
6.31.3 Member Function Documentation	151
6.31.3.1 RequestPasswordResetAsync_InvalidEmailFormat_ReturnsSuccessAndDoesNothing()	151
6.31.3.2 RequestPasswordResetAsync_TokenCreationFails_ReturnsFailure()	151
6.31.3.3 RequestPasswordResetAsync_UserNotFound_ReturnsSuccessAndDoesNotSendEmail()	151
6.31.3.4 RequestPasswordResetAsync_ValidEmailUserExists_ReturnsSuccessAndSendsEmail()	152
6.31.3.5 ResetPasswordAsync_InvalidInput_ReturnsFailure()	152
6.31.3.6 ResetPasswordAsync_InvalidToken_ReturnsFailure()	152
6.31.3.7 ResetPasswordAsync_PasswordProcessingFails_ReturnsFailure()	152
6.31.3.8 ResetPasswordAsync_PasswordUpdateFails_ReturnsFailure()	152
6.31.3.9 ResetPasswordAsync_TokenDeletionFails_StillReturnsSuccess()	152
6.31.3.10 ResetPasswordAsync_ValidRequestAndToken_ReturnsSuccess()	152
6.32 ArandanoIRT_Backend.Application.Services.AuthRegistrationService Class Reference	153
6.32.1 Detailed Description	155
6.32.2 Constructor & Destructor Documentation	155
6.32.2.1 AuthRegistrationService()	155
6.32.3 Member Function Documentation	155

6.32.3.1 RegisterAdminAsync()	155
6.32.3.2 RegisterUserAsync()	155
6.33 ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests Class Reference	156
6.33.1 Detailed Description	158
6.33.2 Constructor & Destructor Documentation	158
6.33.2.1 AuthRegistrationServiceTests()	158
6.33.3 Member Function Documentation	158
6.33.3.1 RegisterAdminAsync_CropCreationFails_ReturnsFailureWithException()	158
6.33.3.2 RegisterAdminAsync_PasswordProcessingFails_ReturnsFailure()	158
6.33.3.3 RegisterAdminAsync_PersonCreationFails_ReturnsFailureWithException()	158
6.33.3.4 RegisterAdminAsync_UserAlreadyExists_ThrowsNullReferenceException()	158
6.33.3.5 RegisterAdminAsync_ValidRequestAndDependenciesSucceed_ReturnsFailureWithException()	159
6.33.3.6 RegisterUserAsync_InvalidInvitationCode_ReturnsFailure()	159
6.33.3.7 RegisterUserAsync_UserAlreadyExistsAfterValidInvitation_ThrowsNullReferenceException()	159
6.33.3.8 RegisterUserAsync_ValidRequestAndInvitationAndDependenciesSucceed_ReturnsFailureWithException()	159
6.34 ArandanoIRT_Backend.Application.Services.AuthSessionService Class Reference	159
6.34.1 Detailed Description	163
6.34.2 Constructor & Destructor Documentation	163
6.34.2.1 AuthSessionService()	163
6.34.3 Member Function Documentation	163
6.34.3.1 LoginAsync() [1/2]	163
6.34.3.2 LoginAsync() [2/2]	164
6.34.3.3 LogoutAsync()	164
6.34.3.4 LogoutEverywhereAsync()	165
6.34.3.5 RefreshTokenAsync() [1/2]	165
6.34.3.6 RefreshTokenAsync() [2/2]	166
6.34.4 Member Data Documentation	166
6.34.4.1 EmailError	166
6.35 ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests Class Reference	167
6.35.1 Detailed Description	169
6.35.2 Constructor & Destructor Documentation	169
6.35.2.1 AuthSessionServiceTests()	169
6.35.3 Member Function Documentation	169
6.35.3.1 Dispose()	169
6.35.3.2 LoginAsync_DecryptionFails_ReturnsFailureAndRecordsAttempt()	169
6.35.3.3 LoginAsync_FailedAttemptThresholdMet_SendsWarningEmail()	170
6.35.3.4 LoginAsync_IncorrectPassword_ReturnsFailureAndRecordsAttempt()	170
6.35.3.5 LoginAsync_TokenGenerationFails_ReturnsFailure()	170
6.35.3.6 LoginAsync_UserNotFound_ReturnsFailure()	170

6.35.3.7 LoginAsync_ValidCredentials_ReturnsSuccessPayload()	170
6.35.3.8 LogoutAsync_TokenServiceFails_ReturnsFailure()	170
6.35.3.9 LogoutAsync_ValidToken_ReturnsSuccessFromTokenService()	170
6.35.3.10 LogoutEverywhereAsync_RepoFails_ReturnsFailure()	170
6.35.3.11 LogoutEverywhereAsync_ValidSessionId_ReturnsSuccessFromRepo()	171
6.35.3.12 RefreshTokenAsync_TokenServiceFails_ReturnsFailure()	171
6.35.3.13 RefreshTokenAsync_ValidToken_ReturnsSuccessFromTokenService()	171
6.36 ArandanoIRT_Backend.Application.Utilities.AuthUtilities Class Reference	171
6.36.1 Detailed Description	173
6.36.2 Constructor & Destructor Documentation	173
6.36.2.1 AuthUtilities()	173
6.36.3 Member Function Documentation	173
6.36.3.1 FindUserByEmailAsync()	173
6.36.3.2 ProcessPassword()	173
6.37 ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests Class Reference	174
6.37.1 Detailed Description	175
6.37.2 Constructor & Destructor Documentation	175
6.37.2.1 AuthUtilitiesTests()	175
6.37.3 Member Function Documentation	175
6.37.3.1 FindUserByEmailAsync_InvalidEmailFormat_ReturnsFailureResult()	175
6.37.3.2 FindUserByEmailAsync_ValidEmailAndUserExists_ReturnsSuccessResultWithUser()	176
6.37.3.3 FindUserByEmailAsync_ValidEmailAndUserNotFound_ReturnsFailureResult()	176
6.37.3.4 ProcessPassword_DecryptionFails_ReturnsFailureResult()	176
6.37.3.5 ProcessPassword_PasswordPolicyFails_ReturnsFailureResult()	176
6.37.3.6 ProcessPassword_ValidEncryptedPassword_ReturnsSuccessResultWithHash()	176
6.38 ArandanoIRT_Backend.UI.Controllers.BaseApiController Class Reference	177
6.38.1 Detailed Description	178
6.38.2 Constructor & Destructor Documentation	178
6.38.2.1 BaseApiController()	178
6.38.3 Member Function Documentation	178
6.38.3.1 ValidateCaptchaIfNeededAsync()	178
6.39 ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher Class Reference	179
6.39.1 Detailed Description	180
6.39.2 Member Function Documentation	180
6.39.2.1 Hash()	180
6.39.2.2 Verify()	181
6.40 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests Class Reference	182
6.40.1 Detailed Description	183
6.40.2 Member Function Documentation	183

6.40.2.1 Hash_WithEmptyPassword_ShouldThrowArgumentException()	183
6.40.2.2 Hash_WithNullPassword_ShouldThrowArgumentException()	183
6.40.2.3 Hash_WithValidPassword_ShouldReturnNonEmptyHash()	184
6.40.2.4 Hash_WithWhitespacePassword_ShouldThrowArgumentException()	184
6.40.2.5 Verify_WithCorrectPassword_ShouldReturnTrue()	184
6.40.2.6 Verify_WithEmptyHash_ShouldReturnFalse()	184
6.40.2.7 Verify_WithEmptyProvidedPassword_ShouldReturnFalse()	184
6.40.2.8 Verify_WithIncorrectPassword_ShouldReturnFalse()	184
6.40.2.9 Verify_WithInvalidHashFormat_ShouldReturnFalse()	184
6.40.2.10 Verify_WithNullHash_ShouldReturnFalse()	185
6.40.2.11 Verify_WithNullProvidedPassword_ShouldReturnFalse()	185
6.41 ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException Class Reference	185
6.41.1 Detailed Description	186
6.41.2 Constructor & Destructor Documentation	186
6.41.2.1 BusinessRuleViolationException()	186
6.42 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests Class Reference	187
6.42.1 Detailed Description	190
6.42.2 Constructor & Destructor Documentation	190
6.42.2.1 CacheServiceTests()	190
6.42.3 Member Function Documentation	190
6.42.3.1 Dispose()	190
6.42.3.2 Get_WhenItemDoesNotExist_ShouldReturnNull()	190
6.42.3.3 Get_WhenItemExistsAndNotExpired_ShouldReturnItem()	191
6.42.3.4 Get_WhenItemExistsButExpired_ShouldReturnNull()	191
6.42.3.5 GetCollection_WhenCollectionDoesNotExist_ShouldReturnNull()	191
6.42.3.6 GetCollection_WhenCollectionExistsAndNotExpired_ShouldReturnCollection()	191
6.42.3.7 GetCollection_WhenCollectionExistsButExpired_ShouldReturnNull()	191
6.42.3.8 Remove_WhenItemDoesNotExist_ShouldNotThrowException()	191
6.42.3.9 Remove_WhenItemExists_ShouldRemoveItemFromCache()	191
6.42.3.10 Set_ShouldAddItemToCache_AndGetShouldRetrieveIt()	191
6.42.3.11 SetCollection_ShouldAddCollectionToCache_AndGetShouldRetrieveIt()	192
6.42.3.12 SetCollection_WithIEnumerable_ShouldStoreAsList()	192
6.43 ArandanoIRT_Backend.Infrastructure.Data.Changepassword Class Reference	192
6.43.1 Property Documentation	193
6.43.1.1 Idchangepassword	193
6.43.1.2 Passwordresettoken	193
6.43.1.3 Person	193
6.43.1.4 Personid	193
6.43.1.5 Resettokenexpiresat	193

6.43.1.6 TokenCreatedat	193
6.44 ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity Class Reference	194
6.44.1 Detailed Description	194
6.44.2 Constructor & Destructor Documentation	195
6.44.2.1 ChangePasswordEntity()	195
6.44.3 Property Documentation	195
6.44.3.1 IdChangePassword	195
6.44.3.2 PasswordResetToken	195
6.44.3.3 PersonId	195
6.44.3.4 ResetTokenExpiresAt	196
6.44.3.5 TokenCreatedAt	196
6.45 ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests Class Reference	196
6.45.1 Detailed Description	199
6.45.2 Constructor & Destructor Documentation	199
6.45.2.1 ChangePasswordRepositoryTests()	199
6.45.3 Member Function Documentation	199
6.45.3.1 Create_ShouldAddTokenAndDeleteExistingForPerson()	199
6.45.3.2 DeleteTokensByPersonIdAsync_ShouldRemoveAllTokensForPerson()	199
6.45.3.3 Dispose()	199
6.45.3.4 GetActiveByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure()	200
6.45.3.5 GetActiveByTokenAsync_WhenTokenExistsAndIsActive_ShouldReturnToken()	200
6.45.3.6 GetActiveByTokenAsync_WhenTokenExistsButExpired_ShouldReturnFailure()	200
6.45.3.7 GetById_WhenTokenDoesNotExist_ShouldReturnFailure()	200
6.45.3.8 GetById_WhenTokenExists_ShouldReturnToken()	200
6.46 ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto Class Reference	200
6.46.1 Detailed Description	201
6.46.2 Property Documentation	201
6.46.2.1 ChangeCode	201
6.46.2.2 ConfirmPassword	201
6.46.2.3 NewPassword	201
6.47 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests Class Reference	202
6.47.1 Detailed Description	203
6.47.2 Member Function Documentation	203
6.47.2.1 ToFormattedString_ShouldOmitDeviceFamilyOther_WhenAgentOrOSIsKnown()	203
6.47.2.2 ToFormattedString_ShouldReturnAgentAndDevice_WhenOSIsUnknown()	203
6.47.2.3 ToFormattedString_ShouldReturnAgentAndOSWithVersionsAndDevice_WhenAllProvided()	204
6.47.2.4 ToFormattedString_ShouldReturnAgentFamilyAndDevice_WhenAgentVersionAndOSAreMissing()	204
6.47.2.5 ToFormattedString_ShouldReturnDeviceFamily_WhenAgentAndOSAreUnknown()	204
6.47.2.6 ToFormattedString_ShouldReturnOSAndDevice_WhenAgentIsUnknown()	204

6.47.2.7 ToFormattedString_ShouldReturnOSFamilyAndDevice_WhenOSVersionAndAgentAreMissing()	204
6.47.2.8 ToFormattedString_ShouldReturnUnknownDevice_WhenAllInfoIsUnknown()	204
6.47.2.9 ToFormattedString_ShouldTrimWhitespaceAndIncludeDevice_WhenPresent()	205
6.48 ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService Class Reference	205
6.48.1 Detailed Description	206
6.48.2 Constructor & Destructor Documentation	206
6.48.2.1 CloudflareTurnstileService()	206
6.48.3 Member Function Documentation	207
6.48.3.1 VerifyCaptchaAsync()	207
6.49 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests Class Reference	208
6.49.1 Detailed Description	209
6.49.2 Constructor & Destructor Documentation	209
6.49.2.1 CloudflareTurnstileServiceTests()	209
6.49.3 Member Function Documentation	209
6.49.3.1 VerifyCaptchaAsync_WithEmptyToken_ShouldReturnFailure()	209
6.49.3.2 VerifyCaptchaAsync_WithHttpError_ShouldReturnFailure()	210
6.49.3.3 VerifyCaptchaAsync_WithInvalidJsonResponse_ShouldReturnFailure()	210
6.49.3.4 VerifyCaptchaAsync_WithNetworkError_ShouldReturnFailure()	210
6.49.3.5 VerifyCaptchaAsync_WithNullToken_ShouldReturnFailure()	210
6.49.3.6 VerifyCaptchaAsync_WithValidTokenAndFailureResponse_ShouldReturnFailureWithErrorCodes()	210
6.49.3.7 VerifyCaptchaAsync_WithValidTokenAndSuccessfulResponse_ShouldReturnSuccess() .	210
6.49.3.8 VerifyCaptchaAsync_WithWhitespaceToken_ShouldReturnFailure()	211
6.50 ArandanoIRT_Backend.Infrastructure.Services.CookiesService Class Reference	211
6.50.1 Detailed Description	212
6.50.2 Constructor & Destructor Documentation	212
6.50.2.1 CookiesService()	212
6.50.3 Member Function Documentation	212
6.50.3.1 GetRefreshToken()	212
6.50.3.2 GetStayLoggedIn()	213
6.50.3.3 RemoveCookies()	213
6.50.3.4 RenewAuthCookies()	213
6.50.3.5 SetAuthCookies()	214
6.51 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests Class Reference	214
6.51.1 Detailed Description	217
6.51.2 Constructor & Destructor Documentation	217
6.51.2.1 CookiesServiceTests()	217
6.51.3 Member Function Documentation	217
6.51.3.1 Dispose()	217
6.51.3.2 GetRefreshToken_WhenCookieDoesNotExist_ShouldReturnNull()	218

6.51.3.3 GetRefreshToken_WhenCookieExists_ShouldReturnTokenValue()	218
6.51.3.4 GetStayLoggedIn_WhenCookieDoesNotExist_ShouldReturnFalse()	218
6.51.3.5 GetStayLoggedIn_WhenCookieIsFalse_ShouldReturnFalse()	218
6.51.3.6 GetStayLoggedIn_WhenCookieIsNotValidBoolean_ShouldReturnFalse()	218
6.51.3.7 GetStayLoggedIn_WhenCookieIsTrue_ShouldReturnTrue()	218
6.51.3.8 RemoveCookies_ShouldAddCookiesWithPastExpiry()	218
6.51.3.9 RenewAuthCookies_WhenStayLoggedInWasFalse_ShouldRenewAsSession()	218
6.51.3.10 RenewAuthCookies_WhenStayLoggedInWasTrue_ShouldRenewWithExpiry()	219
6.51.3.11 SetAuthCookies_WhenStayLoggedInFalse_ShouldSetSessionCookiesWithOptions()	219
6.51.3.12 SetAuthCookies_WhenStayLoggedInTrue_ShouldSetPersistentCookiesWithOptions()	219
6.52 ArandanoIRT_Backend.Infrastructure.Data.Crop Class Reference	220
6.52.1 Property Documentation	221
6.52.1.1 Addresscrop	221
6.52.1.2 Adminuser	221
6.52.1.3 Adminuserid	221
6.52.1.4 Cityname	221
6.52.1.5 Createdat	221
6.52.1.6 Cropinvitation	221
6.52.1.7 Devicedata	221
6.52.1.8 Idcrop	221
6.52.1.9 Namecrop	221
6.52.1.10 Person	222
6.52.1.11 Plantdata	222
6.52.1.12 Sensordata	222
6.52.1.13 Thermaldata	222
6.52.1.14 Updatedat	222
6.53 ArandanoIRT_Backend.Domain.Entities.CropEntity Class Reference	222
6.53.1 Detailed Description	224
6.53.2 Constructor & Destructor Documentation	224
6.53.2.1 CropEntity()	224
6.53.3 Property Documentation	224
6.53.3.1 AddressCrop	224
6.53.3.2 AdminUserId	225
6.53.3.3 CityName	225
6.53.3.4 CreatedAt	225
6.53.3.5 IdCrop	225
6.53.3.6 NameCrop	225
6.53.3.7 UpdatedAt	225
6.54 ArandanoIRT_Backend.Application.DTOs.Objects.CropInfoDto Class Reference	226

6.54.1 Detailed Description	226
6.54.2 Property Documentation	226
6.54.2.1 AddresCrop	226
6.54.2.2 NameCrop	227
6.54.2.3 UbicationCrop	227
6.55 ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation Class Reference	227
6.55.1 Property Documentation	228
6.55.1.1 Accescode	228
6.55.1.2 Createdat	228
6.55.1.3 Createdby	228
6.55.1.4 Createdbynavigation	228
6.55.1.5 Crop	228
6.55.1.6 Cropid	228
6.55.1.7 Expiresat	229
6.55.1.8 Idcropinvitation	229
6.55.1.9 Status	229
6.55.1.10 Statusid	229
6.55.1.11 Usedby	229
6.55.1.12 Usedbynavigation	229
6.56 ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity Class Reference	230
6.56.1 Detailed Description	231
6.56.2 Constructor & Destructor Documentation	231
6.56.2.1 CropInvitationEntity()	231
6.56.3 Property Documentation	232
6.56.3.1 AccessCode	232
6.56.3.2 CreatedAt	232
6.56.3.3 CreatedBy	232
6.56.3.4 CropId	232
6.56.3.5 ExpiresAt	232
6.56.3.6 IdCropInvitation	232
6.56.3.7 StatusId	233
6.56.3.8 UsedBy	233
6.57 ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests Class Reference	233
6.57.1 Detailed Description	236
6.57.2 Constructor & Destructor Documentation	236
6.57.2.1 CropRepositoryTests()	236
6.57.3 Member Function Documentation	236
6.57.3.1 Create_ShouldAddCropToDatabase()	236
6.57.3.2 Delete_ShouldRemoveCropFromDatabase()	236

6.57.3.3 Dispose()	236
6.57.3.4 GetAll_ShouldReturnAllCrops()	236
6.57.3.5 GetById_WhenCropDoesNotExist_ShouldReturnFailure()	237
6.57.3.6 GetById_WhenCropExists_ShouldReturnCrop()	237
6.57.3.7 GetByNameAsync_WhenCropDoesNotExist_ShouldReturnFailure()	237
6.57.3.8 GetByNameAsync_WhenCropExists_ShouldReturnCrop()	237
6.57.3.9 Update_ShouldModifyCropInDatabase()	237
6.58 ArandanoIRT_Backend.UI.Controllers.CryptoController Class Reference	237
6.58.1 Detailed Description	239
6.58.2 Constructor & Destructor Documentation	239
6.58.2.1 CryptoController()	239
6.58.3 Member Function Documentation	239
6.58.3.1 GetPublicKey()	239
6.59 ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException Class Reference	240
6.59.1 Detailed Description	241
6.59.2 Constructor & Destructor Documentation	241
6.59.2.1 DatabaseUpdateException()	241
6.60 ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider Class Reference	241
6.60.1 Detailed Description	243
6.60.2 Member Function Documentation	243
6.60.2.1 GetUtcNow()	243
6.61 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests Class Reference	243
6.61.1 Detailed Description	244
6.61.2 Member Function Documentation	244
6.61.2.1 GetUtcNow_ShouldReturnDateTimeWithUtcKind()	244
6.61.2.2 GetUtcNow_ShouldReturnTimeCloseToActualUtcNow()	244
6.62 ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation Class Reference	245
6.62.1 Property Documentation	245
6.62.1.1 Activatedat	245
6.62.1.2 Activationcode	246
6.62.1.3 Activationstatus	246
6.62.1.4 ActivationstatusNavigation	246
6.62.1.5 Createdat	246
6.62.1.6 Device	246
6.62.1.7 Deviceid	246
6.62.1.8 Expiresat	246
6.62.1.9 Iddeviceactivation	246
6.63 ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity Class Reference	247
6.63.1 Detailed Description	248

6.63.2 Constructor & Destructor Documentation	248
6.63.2.1 DeviceActivationEntity()	248
6.63.3 Property Documentation	248
6.63.3.1 ActivatedAt	248
6.63.3.2 ActivationCode	249
6.63.3.3 ActivationStatus	249
6.63.3.4 CreatedAt	249
6.63.3.5 Deviceld	249
6.63.3.6 ExpiresAt	249
6.63.3.7 IdDeviceActivation	249
6.64 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository Class Reference	250
6.64.1 Member Function Documentation	252
6.64.1.1 Create()	252
6.64.1.2 Delete()	252
6.64.1.3 GetAll()	252
6.64.1.4 GetById()	252
6.64.1.5 Update()	253
6.65 ArandanoIRT_Backend.Infrastructure.Data.Devicedata Class Reference	253
6.65.1 Property Documentation	254
6.65.1.1 Crop	254
6.65.1.2 Cropid	254
6.65.1.3 Datacollectiontime	254
6.65.1.4 Descriptiondevice	254
6.65.1.5 Deviceactivation	254
6.65.1.6 Devicelog	254
6.65.1.7 Devicetoken	255
6.65.1.8 Iddevicedata	255
6.65.1.9 Namedevice	255
6.65.1.10 Plant	255
6.65.1.11 Plantid	255
6.65.1.12 Registeredat	255
6.65.1.13 Registeredby	255
6.65.1.14 Registeredbynavigation	255
6.65.1.15 Status	255
6.65.1.16 Statusid	256
6.65.1.17 Updatedat	256
6.65.1.18 Updatedby	256
6.65.1.19 Updatedbynavigation	256
6.66 ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity Class Reference	256

6.66.1 Detailed Description	258
6.66.2 Constructor & Destructor Documentation	258
6.66.2.1 DeviceDataEntity()	258
6.66.3 Property Documentation	259
6.66.3.1 Cropld	259
6.66.3.2 DataCollectionTime	259
6.66.3.3 DescriptionDevice	259
6.66.3.4 IdDeviceData	259
6.66.3.5 NameDevice	259
6.66.3.6 PlantId	260
6.66.3.7 RegisteredAt	260
6.66.3.8 RegisteredBy	260
6.66.3.9 StatusId	260
6.66.3.10 UpdatedAt	260
6.66.3.11 UpdatedBy	260
6.67 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository Class Reference	261
6.67.1 Member Function Documentation	263
6.67.1.1 Create()	263
6.67.1.2 Delete()	263
6.67.1.3 GetAll()	263
6.67.1.4 GetById()	263
6.67.1.5 Update()	263
6.68 ArandanoIRT_Backend.Infrastructure.Data.Devicelog Class Reference	264
6.68.1 Property Documentation	264
6.68.1.1 Device	264
6.68.1.2 Deviceid	264
6.68.1.3 Iddevicelog	264
6.68.1.4 Logmessage	265
6.68.1.5 Logtimestamp	265
6.68.1.6 Logtype	265
6.69 ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity Class Reference	265
6.69.1 Detailed Description	266
6.69.2 Constructor & Destructor Documentation	266
6.69.2.1 DeviceLogEntity()	266
6.69.3 Property Documentation	267
6.69.3.1 Deviceld	267
6.69.3.2 IdDeviceLog	267
6.69.3.3 LogMessage	267
6.69.3.4 LogTimestamp	267

6.69.3.5 LogType	267
6.70 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository Class Reference	268
6.70.1 Member Function Documentation	270
6.70.1.1 Create()	270
6.70.1.2 Delete()	270
6.70.1.3 GetAll()	270
6.70.1.4 GetById()	270
6.70.1.5 Update()	270
6.71 ArandanoIRT_Backend.Infrastructure.Data.Devicetoken Class Reference	271
6.71.1 Property Documentation	271
6.71.1.1 Createdat	271
6.71.1.2 Device	272
6.71.1.3 Deviceid	272
6.71.1.4 Deviceinfo	272
6.71.1.5 Expiresat	272
6.71.1.6 Iddevicetoken	272
6.71.1.7 RefreshToken	272
6.71.1.8 Revokedat	272
6.71.1.9 Revokedbyip	272
6.71.1.10 Token	272
6.71.1.11 Useragent	273
6.72 ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity Class Reference	273
6.72.1 Detailed Description	274
6.72.2 Constructor & Destructor Documentation	274
6.72.2.1 DeviceTokenEntity()	274
6.72.3 Property Documentation	275
6.72.3.1 CreatedAt	275
6.72.3.2 Deviceld	275
6.72.3.3 DeviceInfo	275
6.72.3.4 ExpiresAt	275
6.72.3.5 IdDeviceToken	276
6.72.3.6 RefreshToken	276
6.72.3.7 RevokedAt	276
6.72.3.8 RevokedByIp	276
6.72.3.9 Token	276
6.72.3.10 UserAgent	276
6.73 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository Class Reference	277
6.73.1 Member Function Documentation	279
6.73.1.1 Create()	279

6.73.1.2 Delete()	279
6.73.1.3 GetAll()	279
6.73.1.4 GetById()	279
6.73.1.5 Update()	280
6.74 ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility Class Reference	280
6.74.1 Detailed Description	280
6.74.2 Constructor & Destructor Documentation	281
6.74.2.1 EmailSenderUtility()	281
6.74.3 Member Function Documentation	281
6.74.3.1 SendEmailAsync()	281
6.74.3.2 SendEmailWithAttachmentAsync()	282
6.75 ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests Class Reference	282
6.75.1 Detailed Description	283
6.75.2 Member Function Documentation	283
6.75.2.1 ValidateEmailAsync_InvalidSyntaxEmails_ShouldReturnFailure()	283
6.75.2.2 ValidateEmailAsync_ValidSyntaxEmails_ShouldNotFailOnSyntax()	284
6.76 ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware Class Reference	284
6.76.1 Detailed Description	285
6.76.2 Constructor & Destructor Documentation	285
6.76.2.1 ExceptionHandlingMiddleware()	285
6.76.3 Member Function Documentation	285
6.76.3.1 InvokeAsync()	285
6.77 ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt Class Reference	286
6.77.1 Property Documentation	287
6.77.1.1 Attemptdate	287
6.77.1.2 Deviceinfo	287
6.77.1.3 Idfailedloginattempt	287
6.77.1.4 Ipaddress	287
6.77.1.5 Person	287
6.77.1.6 Personid	287
6.77.1.7 Useragent	287
6.78 ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity Class Reference	288
6.78.1 Detailed Description	289
6.78.2 Constructor & Destructor Documentation	289
6.78.2.1 FailedLoginAttemptEntity()	289
6.78.3 Property Documentation	289
6.78.3.1 AttemptDate	289
6.78.3.2 DeviceInfo	290
6.78.3.3 IdFailedLoginAttempt	290

6.78.3.4 IpAddress	290
6.78.3.5 PersonId	290
6.78.3.6 UserAgent	290
6.79 ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests Class Reference	291
6.79.1 Detailed Description	293
6.79.2 Constructor & Destructor Documentation	293
6.79.2.1 FailedLoginAttemptRepositoryTests()	293
6.79.3 Member Function Documentation	293
6.79.3.1 Create_ShouldAddAttemptToDatabase()	293
6.79.3.2 Delete_ShouldRemoveAttemptFromDatabase()	293
6.79.3.3 Dispose()	293
6.79.3.4 GetAll_ShouldReturnAllAttempts()	294
6.79.3.5 GetById_WhenAttemptDoesNotExist_ShouldReturnFailure()	294
6.79.3.6 GetById_WhenAttemptExists_ShouldReturnAttempt()	294
6.79.3.7 GetRecentAttemptsAsync_ShouldReturnAttemptsForPersonSinceDate()	294
6.79.3.8 GetRecentAttemptsByIpAsync_ShouldReturnAttemptsForIpSinceDate()	294
6.79.3.9 Update_ShouldModifyAttemptInDatabase()	294
6.80 ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto Class Reference	295
6.80.1 Detailed Description	295
6.80.2 Property Documentation	295
6.80.2.1 CaptchaToken	295
6.80.2.2 Email	296
6.81 ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto Class Reference	296
6.81.1 Detailed Description	297
6.81.2 Property Documentation	297
6.81.2.1 CaptchaToken	297
6.81.2.2 CropName	297
6.81.2.3 Email	297
6.81.2.4 Message	297
6.81.2.5 Name	297
6.81.2.6 Subject	298
6.82 ArandanoIRT_Backend.UI.Controllers.HelpRequestsController Class Reference	298
6.82.1 Detailed Description	300
6.82.2 Constructor & Destructor Documentation	300
6.82.2.1 HelpRequestsController()	300
6.82.3 Member Function Documentation	300
6.82.3.1 SendUnauthenticatedHelpRequest()	300
6.83 ArandanoIRT_Backend.Application.Services.HelpRequestService Class Reference	301

6.83.1 Detailed Description	302
6.83.2 Constructor & Destructor Documentation	302
6.83.2.1 HelpRequestService()	302
6.83.3 Member Function Documentation	303
6.83.3.1 SendUnauthenticatedHelpAsync()	303
6.84 ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests Class Reference	303
6.84.1 Detailed Description	306
6.84.2 Constructor & Destructor Documentation	306
6.84.2.1 HelpRequestServiceTests()	306
6.84.3 Member Function Documentation	306
6.84.3.1 Dispose()	306
6.84.3.2 SendUnauthenticatedHelpAsync_AdminUserNotFound_ReturnsFailure()	306
6.84.3.3 SendUnauthenticatedHelpAsync_CropHasNoAdmin_ReturnsFailure()	306
6.84.3.4 SendUnauthenticatedHelpAsync_CropNotFound_ReturnsFailure()	307
6.84.3.5 SendUnauthenticatedHelpAsync_EmailBodyGenerationFails_ReturnsFailure()	307
6.84.3.6 SendUnauthenticatedHelpAsync_EmailSendingFails_ReturnsFailure()	307
6.84.3.7 SendUnauthenticatedHelpAsync_MissingRequiredFields_ReturnsFailure()	307
6.84.3.8 SendUnauthenticatedHelpAsync_NullRequest_ReturnsFailure()	307
6.84.3.9 SendUnauthenticatedHelpAsync_ValidRequest_ReturnsSuccess()	307
6.85 ArandanoIRT_Backend.Domain.IRepositories.IAuditCropRepository Interface Reference	308
6.85.1 Detailed Description	309
6.86 ArandanoIRT_Backend.Domain.IRepositories.IAuditDataTableRepository Interface Reference	310
6.86.1 Detailed Description	312
6.87 ArandanoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository Interface Reference	312
6.87.1 Detailed Description	315
6.88 ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator Interface Reference	315
6.88.1 Detailed Description	316
6.88.2 Member Function Documentation	316
6.88.2.1 CanHandle()	316
6.88.2.2 GenerateEntries()	316
6.89 ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService Interface Reference	317
6.89.1 Detailed Description	318
6.89.2 Member Function Documentation	319
6.89.2.1 GetCropIdValue()	319
6.89.2.2 GetPrimaryKeyValue()	320
6.89.2.3 GetValuesDictionary() [1/2]	320
6.89.2.4 GetValuesDictionary() [2/2]	321
6.89.2.5 SerializePropertyValueDictionary()	321
6.90 ArandanoIRT_Backend.Domain.IRepositories.IAuditPersonRepository Interface Reference	322

6.90.1 Detailed Description	325
6.91 ArandanoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository Interface Reference	325
6.91.1 Detailed Description	328
6.92 ArandanoIRT_Backend.Domain.IRepositories.IAuditSystemTableRepository Interface Reference	328
6.92.1 Detailed Description	331
6.93 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService Interface Reference	331
6.93.1 Detailed Description	332
6.93.2 Member Function Documentation	332
6.93.2.1 RequestPasswordResetAsync()	332
6.93.2.2 ResetPasswordAsync()	333
6.94 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService Interface Reference	333
6.94.1 Detailed Description	335
6.94.2 Member Function Documentation	335
6.94.2.1 RegisterAdminAsync()	335
6.94.2.2 RegisterUserAsync()	335
6.95 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService Interface Reference	336
6.95.1 Detailed Description	337
6.95.2 Member Function Documentation	338
6.95.2.1 LoginAsync() [1/2]	338
6.95.2.2 LoginAsync() [2/2]	338
6.95.2.3 LogoutAsync()	339
6.95.2.4 LogoutEverywhereAsync()	339
6.95.2.5 RefreshTokenAsync() [1/2]	339
6.95.2.6 RefreshTokenAsync() [2/2]	340
6.96 ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities Interface Reference	341
6.96.1 Detailed Description	342
6.96.2 Member Function Documentation	342
6.96.2.1 FindUserByEmailAsync()	342
6.96.2.2 ProcessPassword()	342
6.97 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService Interface Reference	343
6.97.1 Detailed Description	344
6.97.2 Member Function Documentation	344
6.97.2.1 Get< T >()	344
6.97.2.2 GetCollection< T >()	344
6.97.2.3 Remove()	345
6.97.2.4 Set< T >()	345
6.97.2.5 SetCollection< T >()	345
6.98 ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService Interface Reference	346
6.98.1 Member Function Documentation	347

6.98.1.1 VerifyCaptchaAsync()	347
6.99 ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository Interface Reference	348
6.99.1 Detailed Description	350
6.99.2 Member Function Documentation	350
6.99.2.1 DeleteTokensByPersonIdAsync()	350
6.99.2.2 GetActiveByTokenAsync()	350
6.100 ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository Interface Reference	351
6.100.1 Detailed Description	353
6.100.2 Member Function Documentation	353
6.100.2.1 GetActiveByCodeAsync()	353
6.100.2.2 MarkAsUsedAsync()	353
6.101 ArandanoIRT_Backend.Domain.IRepositories.ICropRepository Interface Reference	354
6.101.1 Detailed Description	356
6.101.2 Member Function Documentation	356
6.101.2.1 GetByNameAsync()	356
6.102 ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider Interface Reference	356
6.102.1 Detailed Description	357
6.102.2 Member Function Documentation	358
6.102.2.1 GetUtcNow()	358
6.103 ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository Interface Reference	358
6.103.1 Detailed Description	361
6.104 ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository Interface Reference	361
6.104.1 Detailed Description	363
6.105 ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository Interface Reference	364
6.105.1 Detailed Description	365
6.106 ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository Interface Reference	366
6.106.1 Detailed Description	368
6.107 ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService Interface Reference	369
6.107.1 Member Function Documentation	371
6.107.1.1 GenerateAccountConfirmationBody()	371
6.107.1.2 GenerateHelpRequestBody()	371
6.107.1.3 GenerateInvitationBody()	371
6.107.1.4 GeneratePasswordChangeConfirmationBody()	373
6.107.1.5 GeneratePasswordResetBody()	373
6.107.1.6 GeneratePasswordResetTokenBody()	374
6.107.1.7 GenerateSuspiciousActivityBody()	374
6.107.1.8 GenerateWelcomeBody()	375
6.107.1.9 SendEmailAsync()	375
6.108 ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository Interface Reference	376

6.108.1 Detailed Description	378
6.108.2 Member Function Documentation	378
6.108.2.1 GetRecentAttemptsAsync()	378
6.108.2.2 GetRecentAttemptsByIpAsync()	378
6.109 ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService Interface Reference	379
6.109.1 Detailed Description	380
6.109.2 Member Function Documentation	380
6.109.2.1 SendUnauthenticatedHelpAsync()	380
6.110 ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher Interface Reference	381
6.110.1 Detailed Description	382
6.110.2 Member Function Documentation	382
6.110.2.1 Hash()	382
6.110.2.2 Verify()	382
6.111 ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository Interface Reference	383
6.111.1 Detailed Description	385
6.111.2 Member Function Documentation	385
6.111.2.1 GetByEmailAsync()	385
6.111.2.2 UpdatePasswordAsync()	385
6.112 ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository Interface Reference	386
6.112.1 Detailed Description	388
6.113 ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository Interface Reference	389
6.113.1 Detailed Description	391
6.114 ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository Interface Reference	391
6.114.1 Detailed Description	394
6.115 ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository Interface Reference	394
6.115.1 Detailed Description	396
6.115.2 Member Function Documentation	396
6.115.2.1 DeleteBySessionAsync()	396
6.115.2.2 DeleteExpiredTokensAsync()	396
6.115.2.3 GetActiveTokensByPersonIdAsync()	397
6.115.2.4 GetByTokenAsync()	397
6.115.2.5 RevokeBySessionIdAsync()	398
6.116 ArandanoIRT_Backend.Domain.IRepositories IRepository< T > Interface Template Reference	398
6.116.1 Detailed Description	399
6.116.2 Member Function Documentation	400
6.116.2.1 Create()	400
6.116.2.2 Delete()	400
6.116.2.3 GetAll()	400
6.116.2.4 GetById()	401

6.116.2.5 Update()	401
6.117 ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor Interface Reference	401
6.117.1 Detailed Description	403
6.117.2 Member Function Documentation	404
6.117.2.1 GetClientInfo()	404
6.117.2.2 GetCurrentCropId()	404
6.117.2.3 GetCurrentUserId()	404
6.117.2.4 GetFormattedDeviceInfo()	405
6.117.2.5 GetIpAddress()	405
6.117.2.6 GetUserAgent()	405
6.118 ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService Interface Reference	406
6.118.1 Detailed Description	407
6.118.2 Member Function Documentation	407
6.118.2.1 Decrypt()	407
6.118.2.2 Encrypt()	407
6.118.2.3 GetPublicKeyPem()	408
6.119 ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService Interface Reference	408
6.119.1 Detailed Description	409
6.119.2 Member Function Documentation	409
6.119.2.1 Sanitize()	409
6.120 ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository Interface Reference	410
6.120.1 Detailed Description	411
6.121 ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository Interface Reference	412
6.121.1 Detailed Description	414
6.121.2 Member Function Documentation	414
6.121.2.1 GetStatusesByTableNameAsync()	414
6.122 ArandanoIRT_Backend.Domain.IRepositories.ITableRelationRepository Interface Reference	414
6.122.1 Detailed Description	417
6.123 ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository Interface Reference	417
6.123.1 Detailed Description	420
6.124 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService Interface Reference	420
6.124.1 Detailed Description	422
6.124.2 Member Function Documentation	423
6.124.2.1 GenerateAndStoreTokensAsync() [1/2]	423
6.124.2.2 GenerateAndStoreTokensAsync() [2/2]	423
6.124.2.3 GetPrincipalFromExpiredToken()	424
6.124.2.4 GetSessionIdFromTokenAsync()	424
6.124.2.5 RefreshTokensAsync() [1/2]	425
6.124.2.6 RefreshTokensAsync() [2/2]	425

6.124.2.7 RevokeRefreshTokenAsync()	426
6.125 ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser Interface Reference	427
6.125.1 Member Function Documentation	428
6.125.1.1 Parse()	428
6.126 ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService Class Reference	428
6.126.1 Detailed Description	432
6.126.2 Constructor & Destructor Documentation	432
6.126.2.1 JwtTokenService()	432
6.126.3 Member Function Documentation	433
6.126.3.1 GenerateAndStoreTokensAsync() [1/2]	433
6.126.3.2 GenerateAndStoreTokensAsync() [2/2]	433
6.126.3.3 GetPrincipalFromExpiredToken()	434
6.126.3.4 GetSessionIdFromTokenAsync()	434
6.126.3.5 RefreshTokensAsync() [1/2]	435
6.126.3.6 RefreshTokensAsync() [2/2]	435
6.126.3.7 RevokeRefreshTokenAsync()	436
6.127 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests Class Reference	436
6.127.1 Detailed Description	438
6.127.2 Constructor & Destructor Documentation	439
6.127.2.1 JwtTokenServiceTests()	439
6.127.3 Member Function Documentation	439
6.127.3.1 GenerateAndStoreTokensAsync_WhenRepoCreateFails_ShouldReturnFailure()	439
6.127.3.2 GenerateAndStoreTokensAsync_WithNullPerson_ShouldReturnFailure()	439
6.127.3.3 GenerateAndStoreTokensAsync_WithValidUser_ShouldReturnSuccessResultWithTokens()	439
6.127.3.4 GetPrincipalFromExpiredToken_WithEmptyToken_ShouldReturnNull()	439
6.127.3.5 GetPrincipalFromExpiredToken_WithInvalidSignature_ShouldReturnNull()	440
6.127.3.6 GetPrincipalFromExpiredToken_WithMalformedToken_ShouldReturnNull()	440
6.127.3.7 GetPrincipalFromExpiredToken_WithNullToken_ShouldReturnNull()	440
6.127.3.8 GetPrincipalFromExpiredToken_WithValidToken_ShouldReturnPrincipalWithClaims()	440
6.127.3.9 GetSessionIdFromTokenAsync_WithExpiredToken_ShouldReturnFailure()	440
6.127.3.10 GetSessionIdFromTokenAsync_WithNonExistentToken_ShouldReturnFailure()	440
6.127.3.11 GetSessionIdFromTokenAsync_WithRevokedToken_ShouldReturnFailure()	441
6.127.3.12 GetSessionIdFromTokenAsync_WithValidActiveToken_ShouldReturnSessionId()	441
6.127.3.13 RefreshTokensAsync_WithExpiredToken_ShouldReturnFailure()	441
6.127.3.14 RefreshTokensAsync_WithNonExistentToken_ShouldReturnFailure()	441
6.127.3.15 RefreshTokensAsync_WithRevokedToken_ShouldReturnFailure()	441
6.127.3.16 RefreshTokensAsync_WithValidActiveToken_ShouldReturnNewTokensAndRevokeOld()	441
6.127.3.17 RefreshTokensAsync_WithValidTokenButUserNotFound_ShouldReturnFailureAndRevokeToken()	442
6.127.3.18 RevokeRefreshTokenAsync_WithAlreadyRevokedToken_ShouldReturnSuccess()	442

6.127.3.19 RevokeRefreshTokenAsync_WithNonExistentToken_ShouldReturnSuccess()	442
6.127.3.20 RevokeRefreshTokenAsync_WithValidActiveToken_ShouldReturnSuccessAndUpdateToken()	442
6.128 ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService Class Reference	443
6.128.1 Detailed Description	444
6.128.2 Constructor & Destructor Documentation	444
6.128.2.1 LogCleanupService()	444
6.128.3 Member Function Documentation	444
6.128.3.1 ExecuteAsync()	444
6.129 ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto Class Reference	445
6.129.1 Detailed Description	446
6.129.2 Property Documentation	446
6.129.2.1 CaptchaToken	446
6.129.2.2 Email	446
6.129.2.3 Password	446
6.129.2.4 RememberMe	446
6.130 ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto Class Reference	447
6.130.1 Detailed Description	447
6.130.2 Property Documentation	448
6.130.2.1 Email	448
6.130.2.2 IdCrop	448
6.130.2.3 IdUser	448
6.130.2.4 Role	448
6.130.2.5 Token	448
6.130.2.6 Username	448
6.131 ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload Class Reference	449
6.131.1 Detailed Description	449
6.131.2 Property Documentation	449
6.131.2.1 LoginDetails	449
6.131.2.2 RefreshToken	450
6.132 ArandanoIRT_Backend.Domain.Exceptions.NotFoundException Class Reference	450
6.132.1 Detailed Description	451
6.132.2 Constructor & Destructor Documentation	451
6.132.2.1 NotFoundException()	451
6.133 ArandanoIRT_Backend.UI.Controllers.PasswordController Class Reference	452
6.133.1 Detailed Description	454
6.133.2 Constructor & Destructor Documentation	454
6.133.2.1 PasswordController()	454
6.133.3 Member Function Documentation	454
6.133.3.1 RequestPasswordReset()	454

6.133.3.2 ResetPassword()	455
6.134 ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest Class Reference	455
6.134.1 Detailed Description	456
6.134.2 Property Documentation	456
6.134.2.1 Password	456
6.135 ArandanoIRT_Backend.Infrastructure.Data.Person Class Reference	457
6.135.1 Property Documentation	458
6.135.1.1 Allnotifications	458
6.135.1.2 Changepassword	458
6.135.1.3 Createdat	458
6.135.1.4 Crop	458
6.135.1.5 Cropid	458
6.135.1.6 CropinvitationCreatedbynavigation	458
6.135.1.7 CropinvitationUsedbynavigation	459
6.135.1.8 Cropnavigation	459
6.135.1.9 DevicedataRegisteredbynavigation	459
6.135.1.10 DevicedataUpdatedbynavigation	459
6.135.1.11 Email	459
6.135.1.12 Failedloginattempt	459
6.135.1.13 Firstname	459
6.135.1.14 Idperson	459
6.135.1.15 Isadmin	459
6.135.1.16 Lastloginat	460
6.135.1.17 Lastname	460
6.135.1.18 Lastpasswordchangeat	460
6.135.1.19 Password	460
6.135.1.20 Plantobservation	460
6.135.1.21 Plantstatehistory	460
6.135.1.22 Refreshtoken	460
6.135.1.23 Updatedat	460
6.136 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person Class Reference	461
6.136.1 Property Documentation	461
6.136.1.1 Password	461
6.137 ArandanoIRT_Backend.Domain.Entities.PersonEntity Class Reference	461
6.137.1 Detailed Description	463
6.137.2 Constructor & Destructor Documentation	463
6.137.2.1 Personentity()	463
6.137.3 Property Documentation	464
6.137.3.1 AllNotifications	464

6.137.3.2 CreatedAt	464
6.137.3.3 CropId	464
6.137.3.4 Email	464
6.137.3.5 FirstName	464
6.137.3.6 IdPerson	465
6.137.3.7 IsAdmin	465
6.137.3.8 LastLoginAt	465
6.137.3.9 LastName	465
6.137.3.10 LastPasswordChangeAt	465
6.137.3.11 Password	465
6.137.3.12 UpdatedAt	466
6.138 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests Class Reference	466
6.138.1 Detailed Description	467
6.138.2 Member Function Documentation	467
6.138.2.1 Constructor_ShouldCreateInstance_WhenOptionalCropIdIsNull()	467
6.138.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()	467
6.138.2.3 Constructor_ShouldThrowArgumentException_WhenFirstNameIsEmpty()	468
6.138.2.4 Constructor_ShouldThrowArgumentException_WhenFirstNameIsNull()	468
6.138.2.5 Constructor_ShouldThrowArgumentException_WhenFirstNameIsWhitespace()	468
6.138.2.6 Constructor_ShouldThrowArgumentException_WhenLastNameIsEmpty()	468
6.138.2.7 Constructor_ShouldThrowArgumentException_WhenLastNameIsNull()	468
6.138.2.8 Constructor_ShouldThrowArgumentException_WhenLastNameIsWhitespace()	468
6.139 ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests Class Reference	469
6.139.1 Detailed Description	471
6.139.2 Constructor & Destructor Documentation	471
6.139.2.1 PersonRepositoryTests()	471
6.139.3 Member Function Documentation	471
6.139.3.1 Create_ShouldAddPersonToDatabase()	471
6.139.3.2 Dispose()	471
6.139.3.3 GetByEmailAsync_WhenPersonDoesNotExist_ShouldReturnFailure()	471
6.139.3.4 GetByEmailAsync_WhenPersonExists_ShouldReturnPerson()	471
6.139.3.5 GetById_WhenPersonDoesNotExist_ShouldReturnFailure()	471
6.139.3.6 GetById_WhenPersonExists_ShouldReturnPerson()	472
6.139.3.7 Update_EFEntityDirectly_ShouldModifyPersonInDatabase()	472
6.140 ArandanoIRT_Backend.Infrastructure.Data.Plantdata Class Reference	472
6.140.1 Property Documentation	473
6.140.1.1 Crop	473
6.140.1.2 CropId	473
6.140.1.3 Devicedata	473

6.140.1.4 Idplantdata	473
6.140.1.5 Nameplant	473
6.140.1.6 Plantobservation	473
6.140.1.7 Plantstatehistory	474
6.140.1.8 Registeredat	474
6.140.1.9 Sensordata	474
6.140.1.10 Status	474
6.140.1.11 Statusid	474
6.140.1.12 Thermaldata	474
6.140.1.13 Updatedat	474
6.141 ArandanoIRT_Backend.Domain.Entities.PlantDataEntity Class Reference	475
6.141.1 Detailed Description	476
6.141.2 Constructor & Destructor Documentation	476
6.141.2.1 PlantDataEntity()	476
6.141.3 Property Documentation	476
6.141.3.1 CropId	476
6.141.3.2 IdPlantState	477
6.141.3.3 NamePlant	477
6.141.3.4 RegisteredAt	477
6.141.3.5 StatusId	477
6.141.3.6 UpdatedAt	477
6.142 ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository Class Reference	478
6.142.1 Member Function Documentation	480
6.142.1.1 Create()	480
6.142.1.2 Delete()	480
6.142.1.3 GetAll()	480
6.142.1.4 GetById()	480
6.142.1.5 Update()	480
6.143 ArandanoIRT_Backend.Infrastructure.Data.Plantobservation Class Reference	481
6.143.1 Property Documentation	482
6.143.1.1 Additionalnotes	482
6.143.1.2 Createdat	482
6.143.1.3 Createdby	482
6.143.1.4 CreatedbyNavigation	482
6.143.1.5 Description	482
6.143.1.6 Hasaltereduniformity	482
6.143.1.7 Hasdecolorations	482
6.143.1.8 Idobservation	482
6.143.1.9 Lastsensordata	482

6.143.1.10 Lastsensordataid	483
6.143.1.11 Lastthermaldata	483
6.143.1.12 Lastthermaldataid	483
6.143.1.13 Leafstemnotes	483
6.143.1.14 Plant	483
6.143.1.15 Plantid	483
6.143.1.16 Status	483
6.143.1.17 Statusid	483
6.143.1.18 Subjectiverating	483
6.144 ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity Class Reference	484
6.144.1 Detailed Description	485
6.144.2 Constructor & Destructor Documentation	485
6.144.2.1 PlantObservationEntity()	485
6.144.3 Property Documentation	486
6.144.3.1 AdditionalNotes	486
6.144.3.2 CreatedAt	486
6.144.3.3 CreatedBy	486
6.144.3.4 Description	487
6.144.3.5 HasAlteredUniformity	487
6.144.3.6 HasDecolorations	487
6.144.3.7 IdObservation	487
6.144.3.8 LastSensorDataId	487
6.144.3.9 LastThermalDataId	487
6.144.3.10 LeafStemNotes	488
6.144.3.11 PlantId	488
6.144.3.12 StatusId	488
6.144.3.13 SubjectiveRating	488
6.145 ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository Class Reference	489
6.145.1 Member Function Documentation	491
6.145.1.1 Create()	491
6.145.1.2 Delete()	491
6.145.1.3 GetAll()	491
6.145.1.4 GetById()	491
6.145.1.5 Update()	492
6.146 ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory Class Reference	492
6.146.1 Property Documentation	493
6.146.1.1 Changedat	493
6.146.1.2 Changedby	493
6.146.1.3 ChangedbyNavigation	493

6.146.1.4 <code>Idplantdatahistory</code>	493
6.146.1.5 <code>Plant</code>	493
6.146.1.6 <code>Plantid</code>	493
6.146.1.7 <code>Status</code>	493
6.146.1.8 <code>Statusid</code>	493
6.147 <code>ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity</code> Class Reference	494
6.147.1 Detailed Description	495
6.147.2 Constructor & Destructor Documentation	495
6.147.2.1 <code>PlantStateHistoryEntity()</code>	495
6.147.3 Property Documentation	495
6.147.3.1 <code>ChangedAt</code>	495
6.147.3.2 <code>ChangedBy</code>	495
6.147.3.3 <code>IdPlantStateHistory</code>	496
6.147.3.4 <code>PlantId</code>	496
6.147.3.5 <code>StatusId</code>	496
6.148 <code>ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository</code> Class Reference	497
6.148.1 Member Function Documentation	499
6.148.1.1 <code>Create()</code>	499
6.148.1.2 <code>Delete()</code>	499
6.148.1.3 <code>GetAll()</code>	499
6.148.1.4 <code>GetById()</code>	499
6.148.1.5 <code>Update()</code>	500
6.149 <code>ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter</code> Class Reference	500
6.149.1 Detailed Description	501
6.149.2 Member Function Documentation	501
6.149.2.1 <code>Apply()</code>	501
6.150 <code>ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter</code> Class Reference	502
6.150.1 Detailed Description	503
6.150.2 Member Function Documentation	503
6.150.2.1 <code>Apply()</code>	503
6.151 <code>ArandanoIRT_Backend.Application.DTOs.Person.ProfileResponseDto</code> Class Reference	504
6.152 <code>ArandanoIRT_Backend.Infrastructure.Data.RefreshToken</code> Class Reference	505
6.152.1 Property Documentation	506
6.152.1.1 <code>Createdat</code>	506
6.152.1.2 <code>Deviceinfo</code>	506
6.152.1.3 <code>Expiresat</code>	506
6.152.1.4 <code>IdrefreshToken</code>	506
6.152.1.5 <code>Ipaddress</code>	506
6.152.1.6 <code>Person</code>	506

6.152.1.7 Personid	506
6.152.1.8 RefreshToken1	506
6.152.1.9 Replacedbytoken	506
6.152.1.10 Revokedat	507
6.152.1.11 Revokedbyip	507
6.152.1.12 Session	507
6.152.1.13 Useragent	507
6.153 ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity Class Reference	507
6.153.1 Detailed Description	509
6.153.2 Constructor & Destructor Documentation	509
6.153.2.1 RefreshTokenEntity()	509
6.153.3 Member Function Documentation	510
6.153.3.1 Replace()	510
6.153.3.2 Revoke()	510
6.153.4 Property Documentation	511
6.153.4.1 CreatedAt	511
6.153.4.2 DeviceInfo	511
6.153.4.3 ExpiresAt	511
6.153.4.4 IdRefreshToken	511
6.153.4.5 IpAddress	511
6.153.4.6 PersonId	512
6.153.4.7 ReplacedByToken	512
6.153.4.8 RevokedAt	512
6.153.4.9 RevokedByIp	512
6.153.4.10 Session	512
6.153.4.11 Token	512
6.153.4.12 UserAgent	513
6.154 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests Class Reference	513
6.154.1 Detailed Description	514
6.154.2 Member Function Documentation	514
6.154.2.1 Constructor_ShouldCreateInstance_WhenPersonIdIsNull()	514
6.154.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvidedIncludingNulls()	514
6.154.2.3 Constructor_ShouldThrowArgumentException_WhenDeviceInfosInvalid()	514
6.154.2.4 Constructor_ShouldThrowArgumentException_WhenIpAddressIsInvalid()	515
6.154.2.5 Constructor_ShouldThrowArgumentException_WhenTokenIsInvalid()	515
6.154.2.6 Constructor_ShouldThrowArgumentException_WhenUserAgentIsInvalid()	515
6.154.2.7 Replace_ShouldSetReplacedByToken()	516
6.154.2.8 Revoke_ShouldSetRevokedAtAndHandleNullRevokedByIp()	516
6.154.2.9 Revoke_ShouldSetRevokedAtAndRevokedByIp()	516

6.155 ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests Class Reference	516
6.155.1 Detailed Description	518
6.155.2 Constructor & Destructor Documentation	519
6.155.2.1 RefreshTokenRepositoryTests()	519
6.155.3 Member Function Documentation	519
6.155.3.1 Create_ShouldAddRefreshTokenToDatabase()	519
6.155.3.2 Dispose()	519
6.155.3.3 GetByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure()	519
6.155.3.4 GetByTokenAsync_WhenTokenExistsAndIsValid_ShouldReturnToken()	519
6.155.3.5 Update_ShouldModifyTokenInDatabase()	519
6.156 ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto Class Reference	520
6.156.1 Detailed Description	520
6.156.2 Property Documentation	520
6.156.2.1 AdminInfo	520
6.156.2.2 CaptchaToken	521
6.156.2.3 CropInfo	521
6.157 ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto Class Reference	521
6.157.1 Detailed Description	522
6.157.2 Property Documentation	522
6.157.2.1 CaptchaToken	522
6.157.2.2 UserInfo	522
6.158 ArandanoIRT_Backend.UI.Controllers.RegistrationController Class Reference	523
6.158.1 Detailed Description	525
6.158.2 Constructor & Destructor Documentation	525
6.158.2.1 RegistrationController()	525
6.158.3 Member Function Documentation	525
6.158.3.1 RegisterAdmin()	525
6.158.3.2 RegisterUser()	526
6.159 ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor Class Reference	526
6.159.1 Detailed Description	529
6.159.2 Constructor & Destructor Documentation	529
6.159.2.1 RequestContextAccessor()	529
6.159.3 Member Function Documentation	530
6.159.3.1 GetClientInfo()	530
6.159.3.2 GetCurrentCropId()	530
6.159.3.3 GetCurrentUserId()	530
6.159.3.4 GetFormattedDeviceInfo()	531
6.159.3.5 GetIpAddress()	531
6.159.3.6 GetUserAgent()	531

6.160 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests Class Reference	532
6.160.1 Detailed Description	533
6.160.2 Constructor & Destructor Documentation	533
6.160.2.1 RequestContextAccessorTests()	533
6.160.3 Member Function Documentation	533
6.160.3.1 GetClientInfo_ShouldCallUserAgentParserAndReturnResult()	533
6.160.3.2 GetClientInfo_WhenCalledMultipleTimes_ShouldReturnCachedResult()	533
6.160.3.3 GetCurrentCropId_WhenCropIdClaimExists_ShouldReturnCropId()	534
6.160.3.4 GetCurrentCropId_WhenCropIdClaimIsMissing_ShouldReturnNull()	534
6.160.3.5 GetCurrentCropId_WhenCropIdClaimIsNotInt_ShouldReturnNull()	534
6.160.3.6 GetCurrentCropId_WhenHttpContextIsNull_ShouldReturnNull()	534
6.160.3.7 GetCurrentCropId_WhenUserIsNotAuthenticated_ShouldReturnNull()	534
6.160.3.8 GetCurrentUserId_WhenHttpContextIsNull_ShouldReturnNull()	534
6.160.3.9 GetCurrentUserId_WhenNameIdentifierClaimIsMissing_ShouldReturnNull()	534
6.160.3.10 GetCurrentUserId_WhenNameIdentifierClaimIsNotInt_ShouldReturnNull()	534
6.160.3.11 GetCurrentUserId_WhenUserIsAuthenticatedWithNameIdentifier_ShouldReturnUserId()	535
6.160.3.12 GetCurrentUserId_WhenUserIsNotAuthenticated_ShouldReturnNull()	535
6.160.3.13 GetFormattedDeviceInfo_ShouldCallGetClientInfoAndFormat()	535
6.160.3.14 GetIpAddress_WhenHttpContextIsNull_ShouldReturnUnknown()	535
6.160.3.15 GetIpAddress_WhenNoXForwardedFor_ShouldReturnRemoteIpAddress()	535
6.160.3.16 GetIpAddress_WhenXForwardedForExists_ShouldReturnFirstIpFromHeader()	535
6.160.3.17 GetIpAddress_WhenXForwardedForHasMultipleIPs_ShouldReturnFirstIp()	535
6.160.3.18 GetUserAgent_WhenHeaderExists_ShouldReturnHeaderValue()	535
6.160.3.19 GetUserAgent_WhenHeaderMissing_ShouldReturnUnknown()	536
6.160.3.20 GetUserAgent_WhenHttpContextIsNull_ShouldReturnUnknown()	536
6.161 ArandanoIRT_Backend.Domain.ValueObjects.Result< T > Class Template Reference	536
6.161.1 Detailed Description	538
6.161.2 Constructor & Destructor Documentation	538
6.161.2.1 Result() [1/2]	538
6.161.2.2 Result() [2/2]	539
6.161.3 Member Function Documentation	539
6.161.3.1 Failure() [1/2]	539
6.161.3.2 Failure() [2/2]	540
6.161.3.3 Success() [1/2]	540
6.161.3.4 Success() [2/2]	540
6.161.4 Property Documentation	541
6.161.4.1 ErrorMessage	541
6.161.4.2 IsFailure	541

6.161.4.3 IsSuccess	541
6.161.4.4 Value	541
6.162 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests Class Reference	542
6.162.1 Detailed Description	542
6.162.2 Member Function Documentation	542
6.162.2.1 Failure_ShouldReturnFailedResult_WithErrorMessage()	542
6.162.2.2 Failure_ShouldThrowException_WhenErrorMessageIsNullOrEmpty()	543
6.162.2.3 Success_ShouldReturnSuccessfulResult()	543
6.163 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests Class Reference	543
6.163.1 Detailed Description	544
6.163.2 Member Function Documentation	545
6.163.2.1 Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultReferenceValue() . .	545
6.163.2.2 Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultValueTypeValue() .	545
6.163.2.3 FailureT_ShouldThrowException_WhenErrorMessageIsNullOrEmpty()	545
6.163.2.4 Success_ShouldReturnSuccessfulResult_WithReferenceTypeValue()	545
6.163.2.5 Success_ShouldReturnSuccessfulResult_WithValueTypeValue()	545
6.164 ArandanoIRT_Backend.Infrastructure.Utilities.RsaService Class Reference	546
6.164.1 Detailed Description	548
6.164.2 Constructor & Destructor Documentation	548
6.164.2.1 RsaService()	548
6.164.3 Member Function Documentation	548
6.164.3.1 Decrypt()	548
6.164.3.2 Encrypt()	549
6.164.3.3 GetPublicKeyPem()	549
6.165 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests Class Reference	550
6.165.1 Detailed Description	551
6.165.2 Member Function Documentation	551
6.165.2.1 Decrypt_WithInvalidBase64_ShouldThrowCryptoExceptionWithFormatException() .	551
6.165.2.2 Decrypt_WithInvalidRsaData_ShouldThrowCryptographicException()	551
6.165.2.3 Encrypt_Decrypt_RoundTrip_ShouldReturnOriginalText()	551
6.165.2.4 Encrypt_WithNullInput_ShouldThrowCryptoExceptionWithArgumentNullException() . .	551
6.165.2.5 GetPublicKeyPem_ShouldReturnNonEmptyPemString()	552
6.166 ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter Class Reference	552
6.166.1 Detailed Description	553
6.166.2 Constructor & Destructor Documentation	553
6.166.2.1 SanitizationActionFilter()	553
6.166.3 Member Function Documentation	554
6.166.3.1 OnActionExecutionAsync()	554
6.167 ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute Class Reference	554

6.167.1 Detailed Description	555
6.168 ArandanoIRT_Backend.Infrastructure.Services.SanitizerService Class Reference	556
6.168.1 Detailed Description	557
6.168.2 Constructor & Destructor Documentation	557
6.168.2.1 SanitizerService()	557
6.168.3 Member Function Documentation	557
6.168.3.1 Sanitize()	557
6.169 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests Class Reference	559
6.169.1 Detailed Description	561
6.169.2 Member Function Documentation	561
6.169.2.1 Sanitize_WithDangerousAttributes_ShouldRemoveAttributes()	561
6.169.2.2 Sanitize_WithEmptyInput_ShouldReturnEmptyString()	561
6.169.2.3 Sanitize_WithMalformedHtml_ShouldHandleGracefully()	562
6.169.2.4 Sanitize_WithMixedContentAndEntities_ShouldKeepSafeTagsAndDecodeEntities() .	562
6.169.2.5 Sanitize_WithNullInput_ShouldReturnEmptyString()	562
6.169.2.6 Sanitize_WithPlainText_ShouldReturnOriginalText()	562
6.169.2.7 Sanitize_WithSafeTagsAndAttributes_ShouldPreserveSafeTagsAndAttributes() .	562
6.169.2.8 Sanitize_WithScriptTags_ShouldRemoveScriptTagsAndContent()	563
6.169.2.9 Sanitize_WithSelfClosingTags_ShouldPreserveSafeTags()	563
6.169.2.10 Sanitize_WithSimpleSafeHtmlTags_ShouldPreserveTags()	563
6.169.2.11 Sanitize_WithWhitespaceInput_ShouldReturnWhitespaceString()	563
6.170 ArandanoIRT_Backend.Infrastructure.Data.Sensordata Class Reference	564
6.170.1 Property Documentation	565
6.170.1.1 Cityhumidity	565
6.170.1.2 Citytemperature	565
6.170.1.3 Crop	565
6.170.1.4 Cropid	565
6.170.1.5 Humidity	565
6.170.1.6 Idsensordata	565
6.170.1.7 Lightintensity	565
6.170.1.8 Plant	565
6.170.1.9 Plantid	565
6.170.1.10 Plantobservation	566
6.170.1.11 Recordedat	566
6.170.1.12 Temperature	566
6.171 ArandanoIRT_Backend.Domain.Entities.SensorDataEntity Class Reference	566
6.171.1 Detailed Description	567
6.171.2 Constructor & Destructor Documentation	567
6.171.2.1 SensorDataEntity()	567

6.171.3 Property Documentation	568
6.171.3.1 CityHumidity	568
6.171.3.2 CityTemperature	568
6.171.3.3 CropId	568
6.171.3.4 Humidity	569
6.171.3.5 IdSensorData	569
6.171.3.6 LightIntensity	569
6.171.3.7 PlantId	569
6.171.3.8 RecordedAt	569
6.171.3.9 Temperature	569
6.172 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests Class Reference	570
6.172.1 Detailed Description	571
6.172.2 Member Function Documentation	571
6.172.2.1 Constructor_ShouldCreateInstance_WhenOptionalValuesAreNull()	571
6.172.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()	571
6.172.2.3 Constructor_ShouldNotThrow_WhenHumidityIsExactly100()	571
6.172.2.4 Constructor_ShouldNotThrow_WhenHumidityIsExactlyZero()	572
6.172.2.5 Constructor_ShouldNotThrow_WhenLightIntensityIsExactlyZero()	572
6.172.2.6 Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsGreaterThan100()	572
6.172.2.7 Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsLessThanZero()	572
6.172.2.8 Constructor_ShouldThrowArgumentOutOfRangeException_WhenLightIntensityIsNegative()	572
6.173 ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository Class Reference	573
6.173.1 Member Function Documentation	575
6.173.1.1 Create()	575
6.173.1.2 Delete()	575
6.173.1.3 GetAll()	575
6.173.1.4 GetById()	575
6.173.1.5 Update()	575
6.174 ArandanoIRT_Backend.UI.Controllers.SessionController Class Reference	576
6.174.1 Detailed Description	578
6.174.2 Constructor & Destructor Documentation	578
6.174.2.1 SessionController()	578
6.174.3 Member Function Documentation	579
6.174.3.1 CheckAuth()	579
6.174.3.2 Login()	579
6.174.3.3 Logout()	579
6.174.3.4 LogoutEverywhere()	580
6.174.3.5 RefreshToken()	580
6.175 ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService Class Reference	581

6.175.1 Detailed Description	585
6.175.2 Constructor & Destructor Documentation	586
6.175.2.1 SmtpEmailService()	586
6.175.3 Member Function Documentation	586
6.175.3.1 GenerateAccountConfirmationBody()	586
6.175.3.2 GenerateHelpRequestBody()	586
6.175.3.3 GenerateInvitationBody()	587
6.175.3.4 GeneratePasswordChangeConfirmationBody()	587
6.175.3.5 GeneratePasswordResetBody()	588
6.175.3.6 GeneratePasswordResetTokenBody()	588
6.175.3.7 GenerateSuspiciousActivityBody()	588
6.175.3.8 GenerateWelcomeBody()	589
6.175.3.9 SendEmailAsync()	589
6.176 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests Class Reference	590
6.176.1 Detailed Description	592
6.176.2 Constructor & Destructor Documentation	592
6.176.2.1 SmtpEmailServiceTests()	592
6.176.3 Member Function Documentation	592
6.176.3.1 GenerateAccountConfirmationBody_ShouldContainUserNameAndLink()	592
6.176.3.2 GenerateHelpRequestBody_ShouldContainAllRequestDetails()	592
6.176.3.3 GenerateInvitationBody_ShouldContainInviterCropCodeAndExpiry()	592
6.176.3.4 GeneratePasswordChangeConfirmationBody_ShouldContainUserName()	593
6.176.3.5 GeneratePasswordResetBody_ShouldContainUserNameAndToken()	593
6.176.3.6 GeneratePasswordResetTokenBody_ShouldContainUserNameAndToken()	593
6.176.3.7 GenerateSuspiciousActivityBody_ShouldContainUserNameTimeAndIp()	593
6.176.3.8 GenerateWelcomeBody_ShouldContainUserName()	593
6.176.3.9 SendEmailAsync_WithInvalidEmail_ShouldReturnFailure()	593
6.176.3.10 SendEmailAsync_WithValidEmailAndConfig_ShouldReturnSuccess()	593
6.176.3.11 SendEmailAsync_WithValidEmailAndMissingConfig_ShouldThrowException()	594
6.177 ArandanoIRT_Backend.Infrastructure.Data.Status Class Reference	594
6.177.1 Property Documentation	595
6.177.1.1 Cropinvitation	595
6.177.1.2 Deviceactivation	595
6.177.1.3 Devicedata	595
6.177.1.4 Idstatus	595
6.177.1.5 Namestatus	595
6.177.1.6 Plantdata	595
6.177.1.7 Plantobservation	595
6.177.1.8 Plantstatehistory	595

6.177.1.9 Tablerelation	596
6.177.1.10 Tablerelationid	596
6.178 ArandanoIRT_Backend.Domain.Entities.StatusEntity Class Reference	596
6.178.1 Detailed Description	597
6.178.2 Constructor & Destructor Documentation	597
6.178.2.1 StatusEntity()	597
6.178.3 Property Documentation	598
6.178.3.1 IdStatus	598
6.178.3.2 NameStatus	598
6.178.3.3 TableRelation	598
6.178.3.4 TableRelationId	598
6.179 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests Class Reference	599
6.179.1 Detailed Description	599
6.179.2 Member Function Documentation	600
6.179.2.1 Constructor_ShouldCreateInstance_WhenTableRelationIdIsNull()	600
6.179.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()	600
6.179.2.3 Constructor_ShouldThrowArgumentException_WhenNameStatusIsEmpty()	600
6.179.2.4 Constructor_ShouldThrowArgumentException_WhenNameStatusIsNull()	600
6.179.2.5 Constructor_ShouldThrowArgumentException_WhenNameStatusIsWhitespace()	600
6.180 ArandanoIRT_Backend.Infrastructure.Data.Tablerelation Class Reference	601
6.180.1 Property Documentation	601
6.180.1.1 Idtablerelation	601
6.180.1.2 Status	601
6.180.1.3 Tablename	601
6.181 ArandanoIRT_Backend.Domain.Entities.TableRelationEntity Class Reference	602
6.181.1 Detailed Description	602
6.181.2 Constructor & Destructor Documentation	602
6.181.2.1 TableRelationEntity()	602
6.181.3 Property Documentation	603
6.181.3.1 IdTableRelation	603
6.181.3.2 TableName	603
6.182 ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository Class Reference	604
6.182.1 Member Function Documentation	606
6.182.1.1 Create()	606
6.182.1.2 Delete()	606
6.182.1.3 GetAll()	606
6.182.1.4 GetById()	606
6.182.1.5 Update()	607
6.183 ArandanoIRT_Backend.Infrastructure.Data.Thermaldata Class Reference	607

6.183.1 Property Documentation	608
6.183.1.1 Crop	608
6.183.1.2 Cropid	608
6.183.1.3 Idthermaldata	608
6.183.1.4 Plant	608
6.183.1.5 Plantid	608
6.183.1.6 Plantobservation	608
6.183.1.7 Recordedat	608
6.183.1.8 Rgbimagedata	608
6.183.1.9 Thermalimagedata	609
6.184 ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity Class Reference	609
6.184.1 Detailed Description	610
6.184.2 Constructor & Destructor Documentation	610
6.184.2.1 ThermalDataEntity()	610
6.184.3 Property Documentation	611
6.184.3.1 Cropld	611
6.184.3.2 IdThermalData	611
6.184.3.3 Plantld	611
6.184.3.4 RecordedAt	611
6.184.3.5 RgbImageData	611
6.184.3.6 ThermallImageData	612
6.185 ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository Class Reference	612
6.185.1 Member Function Documentation	614
6.185.1.1 Create()	614
6.185.1.2 Delete()	614
6.185.1.3 GetAll()	614
6.185.1.4 GetById()	614
6.185.1.5 Update()	615
6.186 ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto Class Reference	615
6.186.1 Detailed Description	615
6.186.2 Property Documentation	616
6.186.2.1 AccessToken	616
6.186.2.2 AccessTokenExpiration	616
6.186.2.3 RefreshToken	616
6.187 ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter Class Reference	616
6.187.1 Detailed Description	618
6.187.2 Constructor & Destructor Documentation	618
6.187.2.1 UAParserAdapter()	618
6.187.3 Member Function Documentation	618

6.187.3.1 Parse()	618
6.188 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests Class Reference	619
6.188.1 Detailed Description	621
6.188.2 Member Function Documentation	621
6.188.2.1 Parse_WithAndroidChromeUserAgent_ShouldReturnCorrectClientInfo()	621
6.188.2.2 Parse_WithBotUserAgent_ShouldReturnBotClientInfo()	621
6.188.2.3 Parse_WithChromeDesktopUserAgent_ShouldReturnCorrectClientInfo()	621
6.188.2.4 Parse_WithEmptyUserAgent_ShouldReturnDefaultClientInfo()	621
6.188.2.5 Parse_WithNullUserAgent_ShouldReturnDefaultClientInfo()	622
6.188.2.6 Parse_WithSafariMobileUserAgent_ShouldReturnCorrectClientInfo()	622
6.188.2.7 Parse_WithUncommonUserAgent_ShouldReturnPartialOrDefaultInfo()	622
6.188.2.8 Parse_WithWhitespaceUserAgent_ShouldReturnDefaultClientInfo()	622
6.189 ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter Class Reference	623
6.189.1 Detailed Description	624
6.189.2 Member Function Documentation	624
6.189.2.1 Apply()	624
6.190 ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto Class Reference	625
6.191 ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto Class Reference	625
6.191.1 Detailed Description	626
6.191.2 Property Documentation	626
6.191.2.1 AuthCode	626
6.191.2.2 Email	626
6.191.2.3 FirstName	626
6.191.2.4 LastName	627
6.191.2.5 Password	627
6.192 ArandanoIRT_Backend.Domain.Exceptions.ValidationException Class Reference	627
6.192.1 Detailed Description	628
6.192.2 Constructor & Destructor Documentation	628
6.192.2.1 ValidationException()	628
7 File Documentation	631
7.1 Application/DTOs/Auth/ChangePasswordRequestDto.cs File Reference	631
7.2 Application/DTOs/Auth/ForgotPasswordRequestDto.cs File Reference	631
7.3 Application/DTOs/Auth/HelpRequestDto.cs File Reference	632
7.4 Application/DTOs/Auth/LoginRequestDto.cs File Reference	632
7.5 Application/DTOs/Auth/LoginResponseDto.cs File Reference	632
7.6 Application/DTOs/Auth/LoginSuccessPayload.cs File Reference	633
7.7 Application/DTOs/Auth/RegisterAdminRequestDto.cs File Reference	633
7.8 Application/DTOs/Auth/RegisterUserRequestDto.cs File Reference	633

7.9 Application/DTOs/Auth/TokenResponseDto.cs File Reference	634
7.10 Application/DTOs/Objects/CropInfoDto.cs File Reference	634
7.11 Application/DTOs/Person/ProfileResponseDto.cs File Reference	634
7.12 Application/DTOs/Person/UpdatePasswordRequestDto.cs File Reference	635
7.13 Application/Interfaces/Auditing/AuditMetadata.cs File Reference	635
7.14 Application/Interfaces/Auditing/IAuditEntryGenerator.cs File Reference	635
7.15 Application/Interfaces/Auditing/IAuditHelperService.cs File Reference	636
7.16 Application/Interfaces/IServices/IAuthPasswordService.cs File Reference	636
7.17 Application/Interfaces/IServices/IAuthRegistrationService.cs File Reference	636
7.18 Application/Interfaces/IServices/IAuthSessionService.cs File Reference	637
7.19 Application/Interfaces/IServices/IHelpRequestService.cs File Reference	637
7.20 Application/Interfaces/Utilities/IAuthUtilities.cs File Reference	637
7.21 Application/Interfaces/Utilities/ICaptchaService.cs File Reference	638
7.22 Application/Interfaces/Utilities/IDateTimeProvider.cs File Reference	638
7.23 Application/Interfaces/Utilities/IEmailService.cs File Reference	638
7.24 Application/Interfaces/Utilities/IPasswordHasher.cs File Reference	639
7.25 Application/Interfaces/Utilities/IRequestContextAccessor.cs File Reference	639
7.26 Application/Interfaces/Utilities/IRsaService.cs File Reference	639
7.27 Application/Interfaces/Utilities/ISanitizerService.cs File Reference	640
7.28 Application/Interfaces/Utilities/IUserAgentParser.cs File Reference	640
7.29 Application/Mappings/AuthMapping.cs File Reference	640
7.30 Application/Services/AuthPasswordService.cs File Reference	641
7.31 Application/Services/AuthRegistrationService.cs File Reference	641
7.32 Application/Services/AuthSessionService.cs File Reference	641
7.33 Application/Services/HelpRequestService.cs File Reference	642
7.34 Application/Utilities/AuthUtilities.cs File Reference	642
7.35 Application/Utilities/EmailSenderUtility.cs File Reference	642
7.36 Application/Utilities/EmailValidatorUtility.cs File Reference	643
7.37 Domain/Entities/AuditCropEntity.cs File Reference	643
7.38 Domain/Entities/AuditDataTableEntity.cs File Reference	643
7.39 Domain/Entities/AuditDeviceEntity.cs File Reference	644
7.40 Domain/Entities/AuditPersonEntity.cs File Reference	644
7.41 Domain/Entities/AuditSensitiveDataEntity.cs File Reference	644
7.42 Domain/Entities/AuditSystemTableEntity.cs File Reference	645
7.43 Domain/Entities/ChangePasswordEntity.cs File Reference	645
7.44 Domain/Entities/CropEntity.cs File Reference	645
7.45 Domain/Entities/CropInvitationEntity.cs File Reference	646
7.46 Domain/Entities/DeviceActivationEntity.cs File Reference	646
7.47 Domain/Entities/DeviceDataEntity.cs File Reference	646

7.48 Domain/Entities/DeviceLogEntity.cs File Reference	647
7.49 Domain/Entities/DeviceTokenEntity.cs File Reference	647
7.50 Domain/Entities/FailedLoginAttemptEntity.cs File Reference	647
7.51 Domain/Entities/PersonEntity.cs File Reference	648
7.52 Domain/Entities/PlantDataEntity.cs File Reference	648
7.53 Domain/Entities/PlantObservationEntity.cs File Reference	648
7.54 Domain/Entities/PlantStateHistoryEntity.cs File Reference	649
7.55 Domain/Entities/RefreshTokenEntity.cs File Reference	649
7.56 Domain/Entities/SensorDataEntity.cs File Reference	649
7.57 Domain/Entities/StatusEntity.cs File Reference	650
7.58 Domain/Entities/TableRelationEntity.cs File Reference	650
7.59 Domain/Entities/ThermalDataEntity.cs File Reference	650
7.60 Domain/Exceptions/BusinessRuleViolationException.cs File Reference	651
7.61 Domain/Exceptions/DatabaseUpdateException.cs File Reference	651
7.62 Domain/Exceptions/NotFoundException.cs File Reference	651
7.63 Domain/Exceptions/ValidationException.cs File Reference	652
7.64 Domain/IRepositories/IAuditCropRepository.cs File Reference	652
7.65 Domain/IRepositories/IAuditDataTableRepository.cs File Reference	652
7.66 Domain/IRepositories/IAuditDeviceRepository.cs File Reference	653
7.67 Domain/IRepositories/IAuditPersonRepository.cs File Reference	653
7.68 Domain/IRepositories/IAuditSensitiveDataRepository.cs File Reference	653
7.69 Domain/IRepositories/IAuditSystemTableRepository.cs File Reference	654
7.70 Domain/IRepositories/IChangePasswordRepository.cs File Reference	654
7.71 Domain/IRepositories/ICropInvitationRepository.cs File Reference	654
7.72 Domain/IRepositories/ICropRepository.cs File Reference	655
7.73 Domain/IRepositories/IDeviceActivationRepository.cs File Reference	655
7.74 Domain/IRepositories/IDeviceDataRepository.cs File Reference	655
7.75 Domain/IRepositories/IDeviceLogRepository.cs File Reference	656
7.76 Domain/IRepositories/IDeviceTokenRepository.cs File Reference	656
7.77 Domain/IRepositories/IFailedLoginAttemptRepository.cs File Reference	656
7.78 Domain/IRepositories/IPersonRepository.cs File Reference	657
7.79 Domain/IRepositories/IPlantDataRepository.cs File Reference	657
7.80 Domain/IRepositories/IPlantObservationRepository.cs File Reference	657
7.81 Domain/IRepositories/IPlantStateHistoryRepository.cs File Reference	658
7.82 Domain/IRepositories/IRefreshTokenRepository.cs File Reference	658
7.83 Domain/IRepositories/IRepository.cs File Reference	658
7.84 Domain/IRepositories/ISensorDataRepository.cs File Reference	659
7.85 Domain/IRepositories/IStatusRepository.cs File Reference	659
7.86 Domain/IRepositories/ITableRelationRepository.cs File Reference	659

7.87 Domain/IRepositories/IThermalDataRepository.cs File Reference	660
7.88 Domain/ValueObjects/ClientInfo.cs File Reference	660
7.89 Domain/ValueObjects/ResultPattern.cs File Reference	660
7.90 Infrastructure/Attributes/SanitizeHtmlAttribute.cs File Reference	661
7.91 Infrastructure/Config/AuthConfig.cs File Reference	661
7.92 Infrastructure/Config/CORSConfig.cs File Reference	661
7.93 Infrastructure/Config/DependencyInjection.cs File Reference	662
7.94 Infrastructure/Config/JsonConfig.cs File Reference	662
7.95 Infrastructure/Config/LoggingConfig.cs File Reference	662
7.96 Infrastructure/Config/RateLimitConfig.cs File Reference	663
7.97 Infrastructure/Config/ResponseCompressionConfig.cs File Reference	663
7.98 Infrastructure/Config/SwaggerConfig.cs File Reference	663
7.99 Infrastructure/Data/ApplicationDbContext.cs File Reference	664
7.100 Infrastructure/Data/Auditcrop.cs File Reference	664
7.101 Infrastructure/Data/Auditdatatable.cs File Reference	664
7.102 Infrastructure/Data/Auditdevice.cs File Reference	664
7.103 Infrastructure/Data/Auditperson.cs File Reference	665
7.104 Infrastructure/Data/Auditsensitizeddata.cs File Reference	665
7.105 Infrastructure/Data/Auditsystemtable.cs File Reference	665
7.106 Infrastructure/Data/Changepassword.cs File Reference	666
7.107 Infrastructure/Data/Crop.cs File Reference	666
7.108 Infrastructure/Data/Cropinvitation.cs File Reference	666
7.109 Infrastructure/Data/Deviceactivation.cs File Reference	666
7.110 Infrastructure/Data/Devicedata.cs File Reference	667
7.111 Infrastructure/Data/Devicelog.cs File Reference	667
7.112 Infrastructure/Data/Devicetoken.cs File Reference	667
7.113 Infrastructure/Data/Failedloginattempt.cs File Reference	668
7.114 Infrastructure/Data/Person.cs File Reference	668
7.115 Infrastructure/Data/Plantdata.cs File Reference	668
7.116 Infrastructure/Data/Plantobservation.cs File Reference	668
7.117 Infrastructure/Data/Plantstatehistory.cs File Reference	669
7.118 Infrastructure/Data/Refreshtoken.cs File Reference	669
7.119 Infrastructure/Data/Sensordata.cs File Reference	669
7.120 Infrastructure/Data/Status.cs File Reference	670
7.121 Infrastructure/Data/Tablerelation.cs File Reference	670
7.122 Infrastructure/Data/Thermaldata.cs File Reference	670
7.123 Infrastructure/Filters/SanitizationActionFilter.cs File Reference	670
7.124 Infrastructure/Interfaces/IServices/ICacheService.cs File Reference	671
7.125 Infrastructure/Interfaces/IServices/ITokenService.cs File Reference	671

7.126 Infrastructure/Persistence/Auditing/AuditCropGenerator.cs File Reference	671
7.127 Infrastructure/Persistence/Auditing/AuditDataTableGenerator.cs File Reference	672
7.128 Infrastructure/Persistence/Auditing/AuditDeviceGenerator.cs File Reference	672
7.129 Infrastructure/Persistence/Auditing/AuditHelperService.cs File Reference	673
7.130 Infrastructure/Persistence/Auditing/AuditPersonGenerator.cs File Reference	673
7.131 Infrastructure/Persistence/Auditing/AuditSensitiveDataGenerator.cs File Reference	673
7.132 Infrastructure/Persistence/Auditing/AuditSystemTableGenerator.cs File Reference	674
7.133 Infrastructure/Persistence/Interceptors/AuditSaveChangesInterceptor.cs File Reference	674
7.134 Infrastructure/Repositories/AuditCropRepository.cs File Reference	674
7.135 Infrastructure/Repositories/AuditDataTableRepository.cs File Reference	675
7.136 Infrastructure/Repositories/AuditDeviceRepository.cs File Reference	675
7.137 Infrastructure/Repositories/AuditPersonRepository.cs File Reference	675
7.138 Infrastructure/Repositories/AuditSensitiveDataRepository.cs File Reference	675
7.139 Infrastructure/Repositories/AuditSystemTableRepository.cs File Reference	676
7.140 Infrastructure/Repositories/ChangePasswordRepository.cs File Reference	676
7.141 Infrastructure/Repositories/CropInvitationRepository.cs File Reference	676
7.142 Infrastructure/Repositories/CropRepository.cs File Reference	676
7.143 Infrastructure/Repositories/DeviceActivationRepository.cs File Reference	677
7.144 Infrastructure/Repositories/DeviceDataRepository.cs File Reference	677
7.145 Infrastructure/Repositories/DeviceLogRepository.cs File Reference	677
7.146 Infrastructure/Repositories/DeviceTokenRepository.cs File Reference	677
7.147 Infrastructure/Repositories/FailedLoginAttemptRepository.cs File Reference	678
7.148 Infrastructure/Repositories/PersonRepository.cs File Reference	678
7.149 Infrastructure/Repositories/PlantDataRepository.cs File Reference	678
7.150 Infrastructure/Repositories/PlantObservationRepository.cs File Reference	678
7.151 Infrastructure/Repositories/PlantStateHistoryRepository.cs File Reference	679
7.152 Infrastructure/Repositories/RefreshTokenRepository.cs File Reference	679
7.153 Infrastructure/Repositories/SensorDataRepository.cs File Reference	679
7.154 Infrastructure/Repositories/StatusRepository.cs File Reference	680
7.155 Infrastructure/Repositories/TableRelationRepository.cs File Reference	680
7.156 Infrastructure/Repositories/ThermalDataRepository.cs File Reference	680
7.157 Infrastructure/Services/CacheService.cs File Reference	681
7.158 Infrastructure/Services/CaptchaService.cs File Reference	681
7.159 Infrastructure/Services/CookiesService.cs File Reference	681
7.160 Infrastructure/Services/FiltersSwaggerService.cs File Reference	682
7.161 Infrastructure/Services/JwtTokenService.cs File Reference	682
7.162 Infrastructure/Services/LogCleanupService.cs File Reference	682
7.163 Infrastructure/Services/RequestContextAccessor.cs File Reference	683
7.164 Infrastructure/Services/SanitizerService.cs File Reference	683

7.165 Infrastructure/Utilities/BCryptPasswordHasher.cs File Reference	683
7.166 Infrastructure/Utilities/DateTimeProvider.cs File Reference	684
7.167 Infrastructure/Utilities/RsaService.cs File Reference	684
7.168 Infrastructure/Utilities/SmtpEmailService.cs File Reference	685
7.169 Infrastructure/Utilities/UAParserAdapter.cs File Reference	685
7.170 Tests/Integration/Common/DbContextTestHelper.cs File Reference	685
7.171 Tests/Integration/Repositories/ChangePasswordRepositoryTests.cs File Reference	686
7.172 Tests/Integration/Repositories/CropRepositoryTests.cs File Reference	686
7.173 Tests/Integration/Repositories/FailedLoginAttemptRepositoryTests.cs File Reference	686
7.174 Tests/Integration/Repositories/PersonRepositoryTests.cs File Reference	687
7.175 Tests/Integration/Repositories/RefreshTokenRepositoryTests.cs File Reference	687
7.176 Tests/Unit/Application/Services/AuthPasswordServiceTests.cs File Reference	687
7.177 Tests/Unit/Application/Services/AuthRegistrationServiceTests.cs File Reference	688
7.178 Tests/Unit/Application/Services/AuthSessionServiceTests.cs File Reference	688
7.179 Tests/Unit/Application/Services/HelpRequestServiceTests.cs File Reference	688
7.180 Tests/Unit/Application/Utilities/AuthUtilitiesTests.cs File Reference	689
7.181 Tests/Unit/Application/Utilities/EmailValidatorUtilityTests.cs File Reference	689
7.182 Tests/Unit/Domain/Entities/PersonEntityTests.cs File Reference	689
7.183 Tests/Unit/Domain/Entities/RefreshTokenEntityTests.cs File Reference	690
7.184 Tests/Unit/Domain/Entities/SensorDataEntityTests.cs File Reference	690
7.185 Tests/Unit/Domain/Entities/StatusEntityTests.cs File Reference	690
7.186 Tests/Unit/Domain/ValueObjects/ClientInfoTests.cs File Reference	691
7.187 Tests/Unit/Domain/ValueObjects/ResultTests.cs File Reference	691
7.188 Tests/Unit/Infrastructure/Persistence/AuditHelperServiceTests.cs File Reference	691
7.189 Tests/Unit/Infrastructure/Services/CacheServiceTests.cs File Reference	692
7.190 Tests/Unit/Infrastructure/Services/CloudflareTurnstileServiceTests.cs File Reference	692
7.191 Tests/Unit/Infrastructure/Services/CookiesServiceTests.cs File Reference	693
7.192 Tests/Unit/Infrastructure/Services/JwtTokenServiceTests.cs File Reference	693
7.193 Tests/Unit/Infrastructure/Services/RequestContextAccessorTests.cs File Reference	693
7.194 Tests/Unit/Infrastructure/Services/SanitizerServiceTests.cs File Reference	694
7.195 Tests/Unit/Infrastructure/Utilities/BCryptPasswordHasherTests.cs File Reference	694
7.196 Tests/Unit/Infrastructure/Utilities/DateTimeProviderTests.cs File Reference	695
7.197 Tests/Unit/Infrastructure/Utilities/RsaServiceTests.cs File Reference	695
7.198 Tests/Unit/Infrastructure/Utilities/SmtpEmailServiceTests.cs File Reference	695
7.199 Tests/Unit/Infrastructure/Utilities/UAParserAdapterTests.cs File Reference	696
7.200 UI/Controllers/BaseApiController.cs File Reference	696
7.201 UI/Controllers/CryptoController.cs File Reference	696
7.202 UI/Controllers/HelpRequestsController.cs File Reference	697
7.203 UI/Controllers>PasswordController.cs File Reference	697

7.204 UI/Controllers/RegistrationController.cs File Reference	697
7.205 UI/Controllers/SessionController.cs File Reference	698
7.206 UI/Middleware/ExceptionHandlingMiddleware.cs File Reference	698

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

ArandoIRT_Backend	23
ArandoIRT_Backend.Application	23
ArandoIRT_Backend.Application.DTOs	23
ArandoIRT_Backend.Application.DTOs.Auth	24
ArandoIRT_Backend.Application.DTOs.Objetcts	24
ArandoIRT_Backend.Application.DTOs.Person	24
ArandoIRT_Backend.Application.Interfaces	25
ArandoIRT_Backend.Application.Interfaces.Auditing	25
ArandoIRT_Backend.Application.Interfaces.IServices	26
ArandoIRT_Backend.Application.Interfaces.Utilities	26
ArandoIRT_Backend.Application.Mappings	26
ArandoIRT_Backend.Application.Services	27
ArandoIRT_Backend.Application.Utilities	27
ArandoIRT_Backend.Domain	27
ArandoIRT_Backend.Domain.Entities	27
ArandoIRT_Backend.Domain.Exceptions	29
ArandoIRT_Backend.Domain.IRepositories	29
ArandoIRT_Backend.Domain.ValueObjects	30
ArandoIRT_Backend.Infrastructure	31
ArandoIRT_Backend.Infrastructure.Attributes	32
ArandoIRT_Backend.Infrastructure.Config	32
ArandoIRT_Backend.Infrastructure.Data	32
ArandoIRT_Backend.Infrastructure.Filters	33
ArandoIRT_Backend.Infrastructure.Interfaces	33
ArandoIRT_Backend.Infrastructure.Interfaces.IServices	33
ArandoIRT_Backend.Infrastructure.Persistence	34
ArandoIRT_Backend.Infrastructure.Persistence.Auditing	34
ArandoIRT_Backend.Infrastructure.Persistence.Interceptors	34
ArandoIRT_Backend.Infrastructure.Repositories	35
ArandoIRT_Backend.Infrastructure.Services	36
ArandoIRT_Backend.Infrastructure.Utilities	37

ArandoNIRT_Backend.Tests	38
ArandoNIRT_Backend.Tests.Application	38
ArandoNIRT_Backend.Tests.Application.Services	38
ArandoNIRT_Backend.Tests.Application.Utilities	38
ArandoNIRT_Backend.Tests.Integration	39
ArandoNIRT_Backend.Tests.Integration.Common	39
ArandoNIRT_Backend.Tests.Integration.Repositories	39
ArandoNIRT_Backend.Tests.Unit	39
ArandoNIRT_Backend.Tests.Unit.Domain	40
ArandoNIRT_Backend.Tests.Unit.Domain.Entities	40
ArandoNIRT_Backend.Tests.Unit.Domain.ValueObjects	40
ArandoNIRT_Backend.Tests.Unit.Infrastructure	40
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Persistence	41
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Services	41
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Utilities	41
ArandoNIRT_Backend.UI	42
ArandoNIRT_Backend.UI.Controllers	42
ArandoNIRT_Backend.UI.Middleware	42

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto	43
Attribute	
ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute	554
ArandanoIRT_Backend.Infrastructure.Data.Auditcrop	51
ArandanoIRT_Backend.Domain.Entities.AuditCropEntity	53
ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable	64
ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity	66
ArandanoIRT_Backend.Infrastructure.Data.Auditdevice	78
ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity	80
ArandanoIRT_Backend.Infrastructure.Data.Auditperson	104
ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity	106
ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata	120
ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity	122
ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable	130
ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity	132
ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests	149
ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests	156
ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests	174
BackgroundService	
ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService	443
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests	182
ArandanoIRT_Backend.Infrastructure.Data.Changepassword	192
ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity	194
ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto	200
ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests	202
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests	208
ControllerBase	
ArandanoIRT_Backend.UI.Controllers.BaseApiController	177
ArandanoIRT_Backend.UI.Controllers.HelpRequestsController	298
ArandanoIRT_Backend.UI.Controllers.PasswordController	452
ArandanoIRT_Backend.UI.Controllers.RegistrationController	523

ArandanoIRT_Backend.UI.Controllers.SessionController	576
ArandanoIRT_Backend.UI.Controllers.CryptoController	237
ArandanoIRT_Backend.Infrastructure.Services.CookiesService	211
ArandanoIRT_Backend.Infrastructure.Data.Crop	220
ArandanoIRT_Backend.Domain.Entities.CropEntity	222
ArandanoIRT_Backend.Application.DTOs.Objetcts.CropInfoDto	226
ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation	227
ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity	230
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests	243
DbContext	
ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext	45
ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation	245
ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity	247
ArandanoIRT_Backend.Infrastructure.Data.Devicedata	253
ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity	256
ArandanoIRT_Backend.Infrastructure.Data.Devicelog	264
ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity	265
ArandanoIRT_Backend.Infrastructure.Data.Devicetoken	271
ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity	273
ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility	280
ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests	282
Exception	
ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException	185
ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException	240
ArandanoIRT_Backend.Domain.Exceptions.NotFoundException	450
ArandanoIRT_Backend.Domain.Exceptions.ValidationException	627
ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware	284
ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt	286
ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity	288
ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto	295
ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto	296
IAsyncActionFilter	
ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter	552
ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator	315
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator	57
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator	71
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator	85
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator	110
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator	137
ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService	317
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService	92
ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService	331
ArandanoIRT_Backend.Application.Services.AuthPasswordService	146
ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService	333
ArandanoIRT_Backend.Application.Services.AuthRegistrationService	153
ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService	336
ArandanoIRT_Backend.Application.Services.AuthSessionService	159
ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities	341
ArandanoIRT_Backend.Application.Utilities.AuthUtilities	171
ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService	343
ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService	346
ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService	205

ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider	356
ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider	241
IDisposable	
ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests	167
ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests	303
ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests	196
ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests	233
ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests	291
ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests	469
ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests	516
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests	98
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests	187
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests	214
ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService	369
ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService	581
ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService	379
ArandanoIRT_Backend.Application.Services.HelpRequestService	301
IOperationFilter	
ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter	144
ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter	500
ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter	502
ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter	623
ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher	381
ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher	179
ArandanoIRT_Backend.Domain.IRepositories IRepository< T >	398
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditCropEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditCropRepository	308
ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository	61
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditDataTableEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditDataTableRepository	310
ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository	75
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditDeviceEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditDeviceRepository	312
ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository	89
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditPersonEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditPersonRepository	322
ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository	114
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditSensitiveDataEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditSensitiveDataRepository	325
ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository	127
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditSystemTableEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IAuditSystemTableRepository	328
ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository	141
ArandanoIRT_Backend.Domain.IRepositories IRepository< ChangePasswordEntity >	398
ArandanoIRT_Backend.Domain.IRepositories IChangePasswordRepository	348
ArandanoIRT_Backend.Domain.IRepositories IRepository< CropEntity >	398
ArandanoIRT_Backend.Domain.IRepositories ICropRepository	354
ArandanoIRT_Backend.Domain.IRepositories IRepository< CropInvitationEntity >	398
ArandanoIRT_Backend.Domain.IRepositories ICropInvitationRepository	351
ArandanoIRT_Backend.Domain.IRepositories IRepository< DeviceActivationEntity >	398

ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository	358
ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository	250
ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceDataEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository	361
ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository	261
ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceLogEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository	364
ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository	268
ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceTokenEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository	366
ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository	277
ArandanoIRT_Backend.Domain.IRepositories.IRepository< FailedLoginAttemptEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository	376
ArandanoIRT_Backend.Domain.IRepositories.IRepository< PersonEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository	383
ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantDataEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository	386
ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository	478
ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantObservationEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository	389
ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository	489
ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantStateHistoryEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository	391
ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository	497
ArandanoIRT_Backend.Domain.IRepositories.IRepository< RefreshTokenEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository	394
ArandanoIRT_Backend.Domain.IRepositories.IRepository< SensorDataEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository	410
ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository	573
ArandanoIRT_Backend.Domain.IRepositories.IRepository< StatusEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository	412
ArandanoIRT_Backend.Domain.IRepositories.IRepository< TableRelationEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.ITableRelationRepository	414
ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository	604
ArandanoIRT_Backend.Domain.IRepositories.IRepository< ThermalDataEntity >	398
ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository	417
ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository	612
ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor	401
ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor	526
ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService	406
ArandanoIRT_Backend.Infrastructure.Utilities.RsaService	546
ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService	408
ArandanoIRT_Backend.Infrastructure.Services.SanitizerService	556
ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService	420
ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService	428
ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser	427
ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter	616
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests	436

ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto	445
ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto	447
ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload	449
ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest	455
ArandanoIRT_Backend.Infrastructure.Data.Person	457
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person	461
ArandanoIRT_Backend.Domain.Entities.PersonEntity	461
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests	466
ArandanoIRT_Backend.Infrastructure.Data.Plantdata	472
ArandanoIRT_Backend.Domain.Entities.PlantDataEntity	475
ArandanoIRT_Backend.Infrastructure.Data.Plantobservation	481
ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity	484
ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory	492
ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity	494
ArandanoIRT_Backend.Application.DTOs.Person.ProfileResponseDto	504
ArandanoIRT_Backend.Infrastructure.Data.RefreshToken	505
ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity	507
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests	513
ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto	520
ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto	521
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests	532
Result	
ArandanoIRT_Backend.Domain.ValueObjects.Result< T >	536
ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests	542
ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests	543
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests	550
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests	559
SaveChangesInterceptor	
ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor	117
ArandanoIRT_Backend.Infrastructure.Data.Sensordata	564
ArandanoIRT_Backend.Domain.Entities.SensorDataEntity	566
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests	570
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests	590
ArandanoIRT_Backend.Infrastructure.Data.Status	594
ArandanoIRT_Backend.Domain.Entities.StatusEntity	596
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests	599
ArandanoIRT_Backend.Infrastructure.Data.Tablerelation	601
ArandanoIRT_Backend.Domain.Entities.TableRelationEntity	602
ArandanoIRT_Backend.Infrastructure.Data.Thermaldata	607
ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity	609
ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto	615
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests	619
ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto	625
ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto	625

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ArandoalIRT_Backend.Application.DTOs.Auth.AdminInfoDto	Represents the personal and credential information for an administrator user	43
ArandoalIRT_Backend.Infrastructure.Data.ApplicationDbContext		45
ArandoalIRT_Backend.Infrastructure.Data.Auditcrop		51
ArandoalIRT_Backend.Domain.Entities.AuditCropEntity	Represents a single audit trail record tracking changes specifically related to crop data	53
ArandoalIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator	Implements IAuditEntryGenerator to create audit trail entries specifically for changes made to Crop entities	57
ArandoalIRT_Backend.Infrastructure.Repositories.AuditCropRepository		61
ArandoalIRT_Backend.Infrastructure.Data.Auditdatatable		64
ArandoalIRT_Backend.Domain.Entities.AuditDataTableEntity	Represents a generic audit trail record tracking changes made to specific columns within identified database tables, associated with a particular crop	66
ArandoalIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator	Implements IAuditEntryGenerator to create audit trail entries in the Auditdatatable for changes made to various data entities like Sensordata, Thermaldata, and Plantdata	71
ArandoalIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository		75
ArandoalIRT_Backend.Infrastructure.Data.Auditdevice		78
ArandoalIRT_Backend.Domain.Entities.AuditDeviceEntity	Represents a single audit trail record tracking changes specifically related to device entities or data, typically associated with a particular crop	80
ArandoalIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator	Implements IAuditEntryGenerator to create audit trail entries in the Auditdevice table for changes made to Devicedata entities	85
ArandoalIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository		89
ArandoalIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService	Implements IAuditHelperService providing utility methods commonly used during the generation of audit log entries, such as primary key retrieval, foreign key retrieval (CropId), and property value serialization	92
ArandoalIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests	Contains unit tests for the AuditHelperService class	98

ArandoNIRT_Backend.Infrastructure.Data.Auditperson	104
ArandoNIRT_Backend.Domain.Entities.AuditPersonEntity	
Represents a single audit trail record tracking changes specifically related to person (user) entities or data, associated with a particular crop	106
ArandoNIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator	
Implements IAuditEntryGenerator to create audit trail entries in the Auditperson table for changes made to Person entities	110
ArandoNIRT_Backend.Infrastructure.Repositories.AuditPersonRepository	114
ArandoNIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor	
An SaveChangesInterceptor that automatically generates audit log entries for entity changes detected by the DbContext's ChangeTracker before saving	117
ArandoNIRT_Backend.Infrastructure.Data.Auditsensitizeddata	120
ArandoNIRT_Backend.Domain.Entities.AuditSensitiveDataEntity	
Represents a single audit trail record specifically tracking actions involving access to, or operations on, sensitive data fields within specified database tables, associated with a crop	122
ArandoNIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository	127
ArandoNIRT_Backend.Infrastructure.Data.Auditsystemtable	130
ArandoNIRT_Backend.Domain.Entities.AuditSystemTableEntity	
Represents a generic audit trail record tracking changes made to specific columns within identified system-level or configuration database tables, potentially associated with a crop context	132
ArandoNIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator	
Implements IAuditEntryGenerator to create audit trail entries in the Auditsystemtable for changes made to specific "system" or related data entities like Plantstatehistory and Plantobservation	137
ArandoNIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository	141
ArandoNIRT_Backend.Infrastructure.Services.AuthOperationFilter	
An Swagger operation filter that adds a security requirement definition (referencing "cookieAuth") to operations protected by the AuthorizeAttribute	144
ArandoNIRT_Backend.Application.Services.AuthPasswordService	
Implements services for handling password reset requests and password changes using tokens	146
ArandoNIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests	
Contains unit tests for the AuthPasswordService class	149
ArandoNIRT_Backend.Application.Services.AuthRegistrationService	
Implements services for registering new administrators and regular users	153
ArandoNIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests	
Contains unit tests for the AuthRegistrationService class	156
ArandoNIRT_Backend.Application.Services.AuthSessionService	
Implements services for user session management including login, token refresh, and logout	159
ArandoNIRT_Backend.Tests.Application.Services.AuthSessionServiceTests	
Contains unit tests for the AuthSessionService class	167
ArandoNIRT_Backend.Application.Utilities.AuthUtilities	
Implements shared utility functions for authentication services	171
ArandoNIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests	
Contains unit tests for the AuthUtilities class	174
ArandoNIRT_Backend.UI.Controllers.BaseApiController	
Base controller for API endpoints providing shared functionality	177
ArandoNIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher	
Implements the IPasswordHasher interface using the BCrypt algorithm provided by the BCrypt.Net library	179
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests	
Contains unit tests for the BCryptPasswordHasher class	182
ArandoNIRT_Backend.Domain.Exceptions.BusinessRuleViolationException	
Represents an error that occurs when an operation violates a specific business rule or constraint	185

ArandoNIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests	Contains unit tests for the CacheService class	187
ArandoNIRT_Backend.Infrastructure.Data.Changepassword		192
ArandoNIRT_Backend.Domain.Entities.ChangePasswordEntity	Represents a record containing a temporary token generated to allow a user to securely reset their password	194
ArandoNIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests	Contains integration tests for the ChangePasswordRepository using an SQLite in-memory database	196
ArandoNIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto	Represents the request data for changing a user's password using a verification code	200
ArandoNIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests	Contains unit tests for the ClientInfo value object	202
ArandoNIRT_Backend.Infrastructure.Services.CloudflareTurnstileService	Implements the ICaptchaService interface using the Cloudflare Turnstile API for verifying CAPTCHA challenges	205
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests	Contains unit tests for the CloudflareTurnstileService class	208
ArandoNIRT_Backend.Infrastructure.Services.CookiesService	Provides methods for managing authentication-related HTTP cookies, such as setting, renewing, retrieving, and removing JWT, Refresh Token, and StayLoggedIn cookies	211
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests	Contains unit tests for the CookiesService class	214
ArandoNIRT_Backend.Infrastructure.Data.Crop		220
ArandoNIRT_Backend.Domain.Entities.CropEntity	Represents an agricultural crop entity, containing its identification, name, location, and administrative details	222
ArandoNIRT_Backend.Application.DTOs.Objetcts.CropInfoDto	Represents basic information about a specific crop	226
ArandoNIRT_Backend.Infrastructure.Data.Cropinvitation		227
ArandoNIRT_Backend.Domain.Entities.CropInvitationEntity	Represents an invitation code record used to grant access or link a new user to a specific crop	230
ArandoNIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests	Contains integration tests for the CropRepository using an SQLite in-memory database	233
ArandoNIRT_Backend.UI.Controllers.CryptoController	Handles cryptographic utility endpoints	237
ArandoNIRT_Backend.Domain.Exceptions.DatabaseUpdateException	Represents an error that occurs when a database update operation (e.g., insert, update, delete) fails unexpectedly or results in an inconsistent state	240
ArandoNIRT_Backend.Infrastructure.Utilities.DateTimeProvider	Implements the IDateTimeProvider interface	241
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests	Contains unit tests for the DateTimeProvider class	243
ArandoNIRT_Backend.Infrastructure.Data.Deviceactivation		245
ArandoNIRT_Backend.Domain.Entities.DeviceActivationEntity	Represents a record containing an activation code used to register, authorize, or link a physical device to the system	247
ArandoNIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository		250
ArandoNIRT_Backend.Infrastructure.Data.Devicedata		253
ArandoNIRT_Backend.Domain.Entities.DeviceDataEntity	Represents a data-collecting device, containing its identification, description, configuration settings, registration details, and associations (crop, plant)	256
ArandoNIRT_Backend.Infrastructure.Repositories.DeviceDataRepository		261
ArandoNIRT_Backend.Infrastructure.Data.Devicelog		264

ArandoIRT_Backend.Domain.Entities.DeviceLogEntity	
Represents a single log entry generated by, or related to, a specific device	265
ArandoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository	268
ArandoIRT_Backend.Infrastructure.Data.Devicetoken	271
ArandoIRT_Backend.Domain.Entities.DeviceTokenEntity	
Represents a pair of authentication tokens (access and refresh) issued specifically for a device, along with metadata about the token's lifecycle and the device context	273
ArandoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository	277
ArandoIRT_Backend.Application.Utilities.EmailSenderUtility	
Provides utility methods for sending emails using SMTP settings from configuration	280
ArandoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests	
Contains unit tests for the static EmailValidatorUtility class	282
ArandoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware	
ASP.NET Core middleware for centralized exception handling	284
ArandoIRT_Backend.Infrastructure.Data.Failedloginattempt	286
ArandoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity	
Represents a record logging a single failed login attempt against a specific user account	288
ArandoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests	
Contains integration tests for the FailedLoginAttemptRepository using an SQLite in-memory database	291
ArandoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto	
Represents the request data for initiating the password recovery process	295
ArandoIRT_Backend.Application.DTOs.Auth.HelpRequestDto	
Represents the data submitted through a help or support request form	296
ArandoIRT_Backend.UI.Controllers.HelpRequestsController	
Handles endpoints related to user help requests	298
ArandoIRT_Backend.Application.Services.HelpRequestService	
Implements services for handling user help requests	301
ArandoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests	
Contains unit tests for the HelpRequestService class	303
ArandoIRT_Backend.Domain.IRepositories.IAuditCropRepository	
Defines the repository contract specifically for managing AuditCropEntity instances	308
ArandoIRT_Backend.Domain.IRepositories.IAuditDataTableRepository	
Defines the repository contract specifically for managing AuditDataTableEntity instances	310
ArandoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository	
Defines the repository contract specifically for managing AuditDeviceEntity instances	312
ArandoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator	
Defines a contract for components responsible for generating audit log entries based on entity changes tracked by Entity Framework Core	315
ArandoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService	
Defines a contract for a helper service providing common utility functions used during the audit logging process, particularly with Entity Framework Core entities	317
ArandoIRT_Backend.Domain.IRepositories.IAuditPersonRepository	
Defines the repository contract specifically for managing AuditPersonEntity instances	322
ArandoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository	
Defines the repository contract specifically for managing AuditSensitiveDataEntity instances	325
ArandoIRT_Backend.Domain.IRepositories.IAuditSystemTableRepository	
Defines the repository contract specifically for managing AuditSystemTableEntity instances	328
ArandoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService	
Defines the contract for password management operations like reset requests and execution	331
ArandoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService	
Defines the contract for user registration operations	333

ArandoNoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService	Defines the contract for user session management operations, including login, token refresh, and logout	336
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities	Defines the contract for shared utility functions used across authentication-related services	341
ArandoNoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService	Defines a contract for a generic caching service	343
ArandoNoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService	Defines the contract for shared utility functions used across authentication-related services	346
ArandoNoIRT_Backend.Domain.IRepositories.IChangePasswordRepository	Defines the repository contract specifically for managing ChangePasswordEntity instances	348
ArandoNoIRT_Backend.Domain.IRepositories.ICropInvitationRepository	Defines the repository contract specifically for managing CropInvitationEntity instances	351
ArandoNoIRT_Backend.Domain.IRepositories.ICropRepository	Defines the repository contract specifically for managing CropEntity instances	354
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider	Provides the current time, abstracting away direct DateTime usage for testability	356
ArandoNoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository	Defines the repository contract specifically for managing DeviceActivationEntity instances	358
ArandoNoIRT_Backend.Domain.IRepositories.IDeviceDataRepository	Defines the repository contract specifically for managing DeviceDataEntity instances	361
ArandoNoIRT_Backend.Domain.IRepositories.IDeviceLogRepository	Defines the repository contract specifically for managing DeviceLogEntity instances	364
ArandoNoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository	Defines the repository contract specifically for managing DeviceTokenEntity instances	366
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IEmailService	Service interface for sending emails	369
ArandoNoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository	Defines the repository contract specifically for managing FailedLoginAttemptEntity instances	376
ArandoNoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService	Defines the contract for handling help request operations	379
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher	Service interface for hashing and verifying passwords	381
ArandoNoIRT_Backend.Domain.IRepositories.IPersonRepository	Defines the repository contract specifically for managing PersonEntity instances	383
ArandoNoIRT_Backend.Domain.IRepositories.IPlantDataRepository	Defines the repository contract specifically for managing PlantDataEntity instances	386
ArandoNoIRT_Backend.Domain.IRepositories.IPlantObservationRepository	Defines the repository contract specifically for managing PlantObservationEntity instances	389
ArandoNoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository	Defines the repository contract specifically for managing PlantStateHistoryEntity instances	391
ArandoNoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository	Defines the repository contract specifically for managing RefreshTokenEntity instances	394
ArandoNoIRT_Backend.Domain.IRepositories IRepository< T >	Defines a generic contract for repository operations for a specific entity type T	398
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor	Defines a contract for accessing information related to the current HTTP request context, such as client details and authenticated user information	401
ArandoNoIRT_Backend.Application.Interfaces.Utilities.IRsaService	Service interface for RSA encryption and decryption operations	406
ArandoNoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService	Defines the contract for a service that sanitizes input strings to remove potentially harmful content, such as HTML tags	408
ArandoNoIRT_Backend.Domain.IRepositories.ISensorDataRepository	Defines the repository contract specifically for managing SensorDataEntity instances	410

ArandoNIRT_Backend.Domain.IRepositories.IStatusRepository	Defines the repository contract specifically for managing StatusEntity instances	412
ArandoNIRT_Backend.Domain.IRepositories.ITableRelationRepository	Defines the repository contract specifically for managing TableRelationEntity instances	414
ArandoNIRT_Backend.Domain.IRepositories.IThermalDataRepository	Defines the repository contract specifically for managing ThermalDataEntity instances	417
ArandoNIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService	Defines a contract for services responsible for handling the lifecycle of authentication tokens, including the generation, storage, validation, refresh, and revocation of JWT access tokens and persistent refresh tokens	420
ArandoNIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser	Implements the ITokenService interface, providing services for generating, validating, refreshing, and revoking JWT access tokens and database-backed opaque refresh tokens	427
ArandoNIRT_Backend.Infrastructure.Services.JwtTokenService	Contains unit tests for the JwtTokenService class	428
ArandoNIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests	Contains unit tests for the JwtTokenService class	436
ArandoNIRT_Backend.Infrastructure.Services.LogCleanupService	A background service responsible for periodically deleting old log files from the configured log directory to manage disk space	443
ArandoNIRT_Backend.Application.DTOs.Auth.LoginRequestDto	Represents the request data for a user login attempt	445
ArandoNIRT_Backend.Application.DTOs.Auth.LoginResponseDto	Represents the data returned after a successful user authentication	447
ArandoNIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload	Represents the data returned after a successful login, including user details, access token, and the refresh token string	449
ArandoNIRT_Backend.Domain.Exceptions.NotFoundException	Represents an error that occurs when a requested resource, entity, or record could not be found in the system or data store	450
ArandoNIRT_Backend.UI.Controllers.PasswordController	Handles endpoints related to user password management, like reset requests	452
ArandoNIRT_Backend.Infrastructure.Utilities.PasswordRequest	Represents a simple request structure containing a password string	455
ArandoNIRT_Backend.Infrastructure.Data.Person	Contains integration tests for the PersonRepository using an SQLite in-memory database	457
ArandoNIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person	Contains unit tests for the PersonEntity class	461
ArandoNIRT_Backend.Domain.Entities.PersonEntity	Represents a user account in the system, containing personal details, credentials (typically hashed password), status flags, timestamps, and associations like the assigned crop	461
ArandoNIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests	Contains unit tests for the PersonEntity class	466
ArandoNIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests	Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop	469
ArandoNIRT_Backend.Infrastructure.Data.Plantdata	Contains integration tests for the PersonRepository using an SQLite in-memory database	472
ArandoNIRT_Backend.Domain.Entities.PlantDataEntity	Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop	475
ArandoNIRT_Backend.Infrastructure.Repositories.PlantDataRepository	Contains unit tests for the PersonEntity class	478
ArandoNIRT_Backend.Infrastructure.Data.Plantobservation	Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers	481
ArandoNIRT_Backend.Domain.Entities.PlantObservationEntity	Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers	484
ArandoNIRT_Backend.Infrastructure.Repositories.PlantObservationRepository	Contains unit tests for the PersonEntity class	489
ArandoNIRT_Backend.Infrastructure.Data.Plantstatehistory	Contains unit tests for the PersonEntity class	492

ArandoIRT_Backend.Domain.Entities.PlantStateHistoryEntity	
Represents a historical record documenting a change in a plant's status at a specific point in time, including who made the change	494
ArandoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository	497
ArandoIRT_Backend.Infrastructure.Services.ProducesJsonFilter	
An Swagger operation filter that attempts to determine the actual return type for 200 OK responses (handling Task and IActionResult with ProducesResponseType) and sets the corresponding JSON schema in the OpenAPI documentation	500
ArandoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter	
An Swagger operation filter that adds standard HTTP error response types (400, 404, 429, 500) to every operation in the OpenAPI documentation	502
ArandoIRT_Backend.Application.DTOs.Person.ProfileResponseDto	504
ArandoIRT_Backend.Infrastructure.Data.RefreshToken	505
ArandoIRT_Backend.Domain.Entities.RefreshTokenEntity	
Represents a refresh token used to obtain new access tokens, extending user sessions	507
ArandoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests	
Contains unit tests for the RefreshTokenEntity class	513
ArandoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests	
Contains integration tests for the RefreshTokenRepository using an SQLite in-memory database	516
ArandoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto	
Represents the request data for registering a new administrator account along with associated crop information	520
ArandoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto	
Represents the request data for registering a new standard user account	521
ArandoIRT_Backend.UI.Controllers.RegistrationController	
Handles endpoints related to user and administrator registration	523
ArandoIRT_Backend.Infrastructure.Services.RequestContextAccessor	
Implements IRequestContextAccessor to provide access to common data from the current HTTP request context, such as IP address, User Agent details, and authenticated user information (ID, Crop ID)	526
ArandoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests	
Contains unit tests for the RequestContextAccessor class	532
ArandoIRT_Backend.Domain.ValueObjects.Result< T >	
Represents the outcome of an operation, indicating either success or failure	536
ArandoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests	
Contains unit tests for the non-generic Result class	542
ArandoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests	
Contains unit tests for the generic Result< T > class	543
ArandoIRT_Backend.Infrastructure.Utilities.RsaService	
Implements the IRsaService interface, providing functionalities for RSA encryption (using public key) and decryption (using private key)	546
ArandoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests	
Contains unit tests for the RsaService class	550
ArandoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter	
An ASP.NET Core action filter that automatically sanitizes string properties marked with the SanitizeHtmlAttribute on action method arguments (typically DTOs)	552
ArandoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute	
Attribute used to mark string properties on DTOs or models that should have HTML tags removed by the SanitizationActionFilter	554
ArandoIRT_Backend.Infrastructure.Services.SanitizerService	
Implements the ISanitizerService	556
ArandoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests	
Contains unit tests for the SanitizerService class (using default HtmlSanitizer + HtmlDecode)	559
ArandoIRT_Backend.Infrastructure.Data.Sensordata	564

ArandanoIRT_Backend.Domain.Entities.SensorDataEntity	Represents a single snapshot of environmental sensor readings (temperature, humidity, light intensity) recorded at a specific time	566
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests	Contains unit tests for the SensorDataEntity class	570
ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository		573
ArandanoIRT_Backend.UI.Controllers.SessionController	Handles endpoints related to user session management, including login, token refresh, and logout operations	576
ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService	Implements the IEmailService interface using an underlying EmailSenderUtility to send emails via SMTP	581
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests	Contains unit tests for the SmtpEmailService class	590
ArandanoIRT_Backend.Infrastructure.Data.Status		594
ArandanoIRT_Backend.Domain.Entities.StatusEntity	Represents a defined status value (e.g., "Active", "Pending", "Used") that can be applied to various entities within the application	596
ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests	Contains unit tests for the StatusEntity class	599
ArandanoIRT_Backend.Infrastructure.Data.Tablerelation		601
ArandanoIRT_Backend.Domain.Entities.TableRelationEntity	Represents a mapping between a unique identifier and a database table name	602
ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository		604
ArandanoIRT_Backend.Infrastructure.Data.Thermaldata		607
ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity	Represents a single record containing thermal image data and corresponding RGB image data, captured at a specific time and potentially associated with a plant or crop	609
ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository		612
ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto	Represents the response containing authentication tokens (access and refresh)	615
ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter	Implements the IUserAgentParser interface by adapting the UAParser library (uap-csharp) to parse User-Agent strings into the application's Domain.ValueObjects.ClientInfo structure	616
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests	Contains unit tests for the UAParserAdapter class	619
ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter	An Swagger operation filter that adds the 401 Unauthorized response type to operations that are protected by an Authorize attribute (identified by name)	623
ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto		625
ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto	Represents the personal and credential information for a standard user during registration	625
ArandanoIRT_Backend.Domain.Exceptions.ValidationException	Represents a general error that occurs when input data or an object's state fails one or more validation checks	627

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Application/DTOs/Auth/ ChangePasswordRequestDto.cs	631
Application/DTOs/Auth/ ForgotPasswordRequestDto.cs	631
Application/DTOs/Auth/ HelpRequestDto.cs	632
Application/DTOs/Auth/ LoginRequestDto.cs	632
Application/DTOs/Auth/ LoginResponseDto.cs	632
Application/DTOs/Auth/ LoginSuccessPayload.cs	633
Application/DTOs/Auth/ RegisterAdminRequestDto.cs	633
Application/DTOs/Auth/ RegisterUserRequestDto.cs	633
Application/DTOs/Auth/ TokenResponseDto.cs	634
Application/DTOs/Objects/ CropInfoDto.cs	634
Application/DTOs/Person/ ProfileResponseDto.cs	634
Application/DTOs/Person/ UpdatePasswordRequestDto.cs	635
Application/Interfaces/Auditing/ AuditMetadata.cs	635
Application/Interfaces/Auditing/ IAuditEntryGenerator.cs	635
Application/Interfaces/Auditing/ IAuditHelperService.cs	636
Application/Interfaces/IServices/ IAuthPasswordService.cs	636
Application/Interfaces/IServices/ IAuthRegistrationService.cs	636
Application/Interfaces/IServices/ IAuthSessionService.cs	637
Application/Interfaces/IServices/ IHelpRequestService.cs	637
Application/Interfaces/Utilities/ IAuthUtilities.cs	637
Application/Interfaces/Utilities/ ICaptchaService.cs	638
Application/Interfaces/Utilities/ IDateTimeProvider.cs	638
Application/Interfaces/Utilities/ IEmailService.cs	638
Application/Interfaces/Utilities/ IPasswordHasher.cs	639
Application/Interfaces/Utilities/ IRequestContextAccessor.cs	639
Application/Interfaces/Utilities/ IRsaService.cs	639
Application/Interfaces/Utilities/ ISanitizerService.cs	640
Application/Interfaces/Utilities/ IUserAgentParser.cs	640
Application/Mappings/ AuthMapping.cs	640
Application/Services/ AuthPasswordService.cs	641
Application/Services/ AuthRegistrationService.cs	641

Application/Services/AuthSessionService.cs	641
Application/Services/HelpRequestService.cs	642
Application/Utilities/AuthUtilities.cs	642
Application/Utilities/EmailSenderUtility.cs	642
Application/Utilities/EmailValidatorUtility.cs	643
Domain/Entities/AuditCropEntity.cs	643
Domain/Entities/AuditDataTableEntity.cs	643
Domain/Entities/AuditDeviceEntity.cs	644
Domain/Entities/AuditPersonEntity.cs	644
Domain/Entities/AuditSensitiveDataEntity.cs	644
Domain/Entities/AuditSystemTableEntity.cs	645
Domain/Entities/ChangePasswordEntity.cs	645
Domain/Entities/CropEntity.cs	645
Domain/Entities/CropInvitationEntity.cs	646
Domain/Entities/DeviceActivationEntity.cs	646
Domain/Entities/DeviceDataEntity.cs	646
Domain/Entities/DeviceLogEntity.cs	647
Domain/Entities/DeviceTokenEntity.cs	647
Domain/Entities/FailedLoginAttemptEntity.cs	647
Domain/Entities/PersonEntity.cs	648
Domain/Entities/PlantDataEntity.cs	648
Domain/Entities/PlantObservationEntity.cs	648
Domain/Entities/PlantStateHistoryEntity.cs	649
Domain/Exceptions/RefreshTokenEntity.cs	649
Domain/Entities/SensorDataEntity.cs	649
Domain/Entities/StatusEntity.cs	650
Domain/Entities/TableRelationEntity.cs	650
Domain/Entities/ThermalDataEntity.cs	650
Domain/Exceptions/BusinessRuleViolationException.cs	651
Domain/Exceptions/DatabaseUpdateException.cs	651
Domain/Exceptions/NotFoundException.cs	651
Domain/Exceptions/ValidationException.cs	652
Domain/IRepositories/IAuditCropRepository.cs	652
Domain/IRepositories/IAuditDataTableRepository.cs	652
Domain/IRepositories/IAuditDeviceRepository.cs	653
Domain/IRepositories/IAuditPersonRepository.cs	653
Domain/IRepositories/IAuditSensitiveDataRepository.cs	653
Domain/IRepositories/IAuditSystemTableRepository.cs	654
Domain/IRepositories/IChangePasswordRepository.cs	654
Domain/IRepositories/ICropInvitationRepository.cs	654
Domain/IRepositories/ICropRepository.cs	655
Domain/IRepositories/IDeviceActivationRepository.cs	655
Domain/IRepositories/IDeviceDataRepository.cs	655
Domain/IRepositories/IDeviceLogRepository.cs	656
Domain/IRepositories/IDeviceTokenRepository.cs	656
Domain/IRepositories/IFailedLoginAttemptRepository.cs	656
Domain/IRepositories/IPersonRepository.cs	657
Domain/IRepositories/IPlantDataRepository.cs	657
Domain/IRepositories/IPlantObservationRepository.cs	657
Domain/IRepositories/IPlantStateHistoryRepository.cs	658
Domain/IRepositories/IRefreshTokenRepository.cs	658
Domain/IRepositories/ IRepository.cs	658
Domain/IRepositories/ISensorDataRepository.cs	659
Domain/IRepositories/IStatusRepository.cs	659

Domain/IRepositories/ITableRelationRepository.cs	659
Domain/IRepositories/IThermalDataRepository.cs	660
Domain/ValueObjects/ClientInfo.cs	660
Domain/ValueObjects/ResultPattern.cs	660
Infrastructure/Attributes/SanitizeHtmlAttribute.cs	661
Infrastructure/Config/AuthConfig.cs	661
Infrastructure/Config/CORSConfig.cs	661
Infrastructure/Config/DependencyInjection.cs	662
Infrastructure/Config/JsonConfig.cs	662
Infrastructure/Config/LoggingConfig.cs	662
Infrastructure/Config/RateLimitConfig.cs	663
Infrastructure/Config/ResponseCompressionConfig.cs	663
Infrastructure/Config/SwaggerConfig.cs	663
Infrastructure/Data/ApplicationDbContext.cs	664
Infrastructure/Data/Auditcrop.cs	664
Infrastructure/Data/Auditdatatable.cs	664
Infrastructure/Data/Auditdevice.cs	664
Infrastructure/Data/Auditperson.cs	665
Infrastructure/Data/Auditsensitizeddata.cs	665
Infrastructure/Data/Auditsystemtable.cs	665
Infrastructure/Data/Changepassword.cs	666
Infrastructure/Data/Crop.cs	666
Infrastructure/Data/Cropinvitation.cs	666
Infrastructure/Data/Deviceactivation.cs	666
Infrastructure/Data/Devicedata.cs	667
Infrastructure/Data/Devicelog.cs	667
Infrastructure/Data/Devicetoken.cs	667
Infrastructure/Data/Failedloginattempt.cs	668
Infrastructure/Data/Person.cs	668
Infrastructure/Data/Plantdata.cs	668
Infrastructure/Data/Plantobservation.cs	668
Infrastructure/Data/Plantstatehistory.cs	669
Infrastructure/Data/Refreshtoken.cs	669
Infrastructure/Data/Sensordata.cs	669
Infrastructure/Data>Status.cs	670
Infrastructure/Data/Tablerelation.cs	670
Infrastructure/Data/Thermaldata.cs	670
Infrastructure/Filters/SanitizationActionFilter.cs	670
Infrastructure/Interfaces/IServices/ICacheService.cs	671
Infrastructure/Interfaces/IServices/ITokenService.cs	671
Infrastructure/Persistence/Auditing/AuditCropGenerator.cs	671
Infrastructure/Persistence/Auditing/AuditDataTableGenerator.cs	672
Infrastructure/Persistence/Auditing/AuditDeviceGenerator.cs	672
Infrastructure/Persistence/Auditing/AuditHelperService.cs	673
Infrastructure/Persistence/Auditing/AuditPersonGenerator.cs	673
Infrastructure/Persistence/Auditing/AuditSensitiveDataGenerator.cs	673
Infrastructure/Persistence/Auditing/AuditSystemTableGenerator.cs	674
Infrastructure/Persistence/Interceptors/AuditSaveChangesInterceptor.cs	674
Infrastructure/Repositories/AuditCropRepository.cs	674
Infrastructure/Repositories/AuditDataTableRepository.cs	675
Infrastructure/Repositories/AuditDeviceRepository.cs	675
Infrastructure/Repositories/AuditPersonRepository.cs	675
Infrastructure/Repositories/AuditSensitiveDataRepository.cs	675
Infrastructure/Repositories/AuditSystemTableRepository.cs	676

Infrastructure/Repositories/ChangePasswordRepository.cs	676
Infrastructure/Repositories/CropInvitationRepository.cs	676
Infrastructure/Repositories/CropRepository.cs	676
Infrastructure/Repositories/DeviceActivationRepository.cs	677
Infrastructure/Repositories/DeviceDataRepository.cs	677
Infrastructure/Repositories/DeviceLogRepository.cs	677
Infrastructure/Repositories/DeviceTokenRepository.cs	677
Infrastructure/Repositories/FailedLoginAttemptRepository.cs	678
Infrastructure/Repositories/PersonRepository.cs	678
Infrastructure/Repositories/PlantDataRepository.cs	678
Infrastructure/Repositories/PlantObservationRepository.cs	678
Infrastructure/Repositories/PlantStateHistoryRepository.cs	679
Infrastructure/Repositories/RefreshTokenRepository.cs	679
Infrastructure/Repositories/SensorDataRepository.cs	679
Infrastructure/Repositories/StatusRepository.cs	680
Infrastructure/Repositories/TableRelationRepository.cs	680
Infrastructure/Repositories/ThermalDataRepository.cs	680
Infrastructure/Services/CacheService.cs	681
Infrastructure/Services/CaptchaService.cs	681
Infrastructure/Services/CookiesService.cs	681
Infrastructure/Services/FiltersSwaggerService.cs	682
Infrastructure/Services/JwtTokenService.cs	682
Infrastructure/Services/LogCleanupService.cs	682
Infrastructure/Services/RequestContextAccessor.cs	683
Infrastructure/Services/SanitizerService.cs	683
Infrastructure/Utilities/BCryptPasswordHasher.cs	683
Infrastructure/Utilities/DateTimeProvider.cs	684
Infrastructure/Utilities/RsaService.cs	684
Infrastructure/Utilities/SmtpEmailService.cs	685
Infrastructure/Utilities/UAParserAdapter.cs	685
Tests/Integration/Common/DbContextTestHelper.cs	685
Tests/Integration/Repositories/ChangePasswordRepositoryTests.cs	686
Tests/Integration/Repositories/CropRepositoryTests.cs	686
Tests/Integration/Repositories/FailedLoginAttemptRepositoryTests.cs	686
Tests/Integration/Repositories/PersonRepositoryTests.cs	687
Tests/Integration/Repositories/RefreshTokenRepositoryTests.cs	687
Tests/Unit/Application/Services/AuthPasswordServiceTests.cs	687
Tests/Unit/Application/Services/AuthRegistrationServiceTests.cs	688
Tests/Unit/Application/Services/AuthSessionServiceTests.cs	688
Tests/Unit/Application/Services/HelpRequestServiceTests.cs	688
Tests/Unit/Application/Utilities/AuthUtilitiesTests.cs	689
Tests/Unit/Application/Utilities/EmailValidatorUtilityTests.cs	689
Tests/Unit/Domain/Entities/PersonEntityTests.cs	689
Tests/Unit/Domain/Entities/RefreshTokenEntityTests.cs	690
Tests/Unit/Domain/Entities/SensorDataEntityTests.cs	690
Tests/Unit/Domain/Entities/StatusEntityTests.cs	690
Tests/Unit/Domain/ValueObjects/ClientInfoTests.cs	691
Tests/Unit/Domain/ValueObjects/ResultTests.cs	691
Tests/Unit/Infrastructure/Persistence/AuditHelperServiceTests.cs	691
Tests/Unit/Infrastructure/Services/CacheServiceTests.cs	692
Tests/Unit/Infrastructure/Services/CloudflareTurnstileServiceTests.cs	692
Tests/Unit/Infrastructure/Services/CookiesServiceTests.cs	693
Tests/Unit/Infrastructure/Services/JwtTokenServiceTests.cs	693
Tests/Unit/Infrastructure/Services/RequestContextAccessorTests.cs	693

Tests/Unit/Infrastructure/Services/ SanitizerServiceTests.cs	694
Tests/Unit/Infrastructure/Utilities/ BCryptPasswordHasherTests.cs	694
Tests/Unit/Infrastructure/Utilities/ DateTimeProviderTests.cs	695
Tests/Unit/Infrastructure/Utilities/ RsaServiceTests.cs	695
Tests/Unit/Infrastructure/Utilities/ SmtpEmailServiceTests.cs	695
Tests/Unit/Infrastructure/Utilities/ UAParserAdapterTests.cs	696
UI/Controllers/ BaseApiController.cs	696
UI/Controllers/ CryptoController.cs	696
UI/Controllers/ HelpRequestsController.cs	697
UI/Controllers/ PasswordController.cs	697
UI/Controllers/ RegistrationController.cs	697
UI/Controllers/ SessionController.cs	698
UI/Middleware/ ExceptionHandlingMiddleware.cs	698

Chapter 5

Namespace Documentation

5.1 ArandanoIRT_Backend Namespace Reference

Namespaces

- namespace [Application](#)
- namespace [Domain](#)
- namespace [Infrastructure](#)
- namespace [Tests](#)
- namespace [UI](#)

5.2 ArandanoIRT_Backend.Application Namespace Reference

Namespaces

- namespace [DTOs](#)
- namespace [Interfaces](#)
- namespace [Mappings](#)
- namespace [Services](#)
- namespace [Utilities](#)

5.3 ArandanoIRT_Backend.Application.DTOs Namespace Reference

Namespaces

- namespace [Auth](#)
- namespace [Objects](#)
- namespace [Person](#)

5.4 ArandanoIRT_Backend.Application.DTOs.Auth Namespace Reference

Classes

- class [AdminInfoDto](#)
Represents the personal and credential information for an administrator user.
- class [ChangePasswordRequestDto](#)
Represents the request data for changing a user's password using a verification code.
- class [ForgotPasswordRequestDto](#)
Represents the request data for initiating the password recovery process.
- class [HelpRequestDto](#)
Represents the data submitted through a help or support request form.
- class [LoginRequestDto](#)
Represents the request data for a user login attempt.
- class [LoginResponseDto](#)
Represents the data returned after a successful user authentication.
- class [LoginSuccessPayload](#)
Represents the data returned after a successful login, including user details, access token, and the refresh token string.
- class [RegisterAdminRequestDto](#)
Represents the request data for registering a new administrator account along with associated crop information.
- class [RegisterUserRequestDto](#)
Represents the request data for registering a new standard user account.
- class [TokenResponseDto](#)
Represents the response containing authentication tokens (access and refresh).
- class [UserInfoDto](#)
Represents the personal and credential information for a standard user during registration.

5.5 ArandanoIRT_Backend.Application.DTOs.Objects Namespace Reference

Classes

- class [CropInfoDto](#)
Represents basic information about a specific crop.

5.6 ArandanoIRT_Backend.Application.DTOs.Person Namespace Reference

Classes

- class [ProfileResponseDto](#)
- class [UpdatePasswordRequestDto](#)

5.7 ArandanoIRT_Backend.Application.Interfaces Namespace Reference

Namespaces

- namespace [Auditing](#)
- namespace [IServices](#)
- namespace [Utilities](#)

5.8 ArandanoIRT_Backend.Application.Interfaces.Auditing Namespace Reference

Classes

- interface [IAuditEntryGenerator](#)

Defines a contract for components responsible for generating audit log entries based on entity changes tracked by Entity Framework Core.

- interface [IAuditHelperService](#)

Defines a contract for a helper service providing common utility functions used during the audit logging process, particularly with Entity Framework Core entities.

Functions

- record [AuditMetadata](#) (int? UserId, string? IpAddress, string? UserAgent, DateTime PerformedAt)

Represents metadata associated with an audited event or action.

5.8.1 Function Documentation

5.8.1.1 AuditMetadata()

```
record ArandanoIRT_Backend.Application.Interfaces.Auditing.AuditMetadata (
    int? UserId,
    string? IpAddress,
    string? UserAgent,
    DateTime PerformedAt )
```

Represents metadata associated with an audited event or action.

Parameters

<i>UserId</i>	The unique identifier of the user who performed the action, if available.
<i>IpAddress</i>	The IP address from which the action originated, if available.
<i>UserAgent</i>	The user agent string of the client that performed the action, if available.
<i>PerformedAt</i>	The date and time (usually UTC) when the action was performed.

5.9 ArandanoIRT_Backend.Application.Interfaces.IServices Namespace Reference

Classes

- interface [IAuthPasswordService](#)
Defines the contract for password management operations like reset requests and execution.
- interface [IAuthRegistrationService](#)
Defines the contract for user registration operations.
- interface [IAuthSessionService](#)
Defines the contract for user session management operations, including login, token refresh, and logout.
- interface [IHelpRequestService](#)
Defines the contract for handling help request operations.

5.10 ArandanoIRT_Backend.Application.Interfaces.Utilities Namespace Reference

Classes

- interface [IAuthUtilities](#)
Defines the contract for shared utility functions used across authentication-related services.
- interface [ICaptchaService](#)
- interface [IDateTimeProvider](#)
Provides the current time, abstracting away direct DateTime usage for testability.
- interface [IEmailService](#)
- interface [IPasswordHasher](#)
Service interface for hashing and verifying passwords.
- interface [IRequestContextAccessor](#)
Defines a contract for accessing information related to the current HTTP request context, such as client details and authenticated user information.
- interface [IRsaService](#)
Service interface for RSA encryption and decryption operations.
- interface [ISanitizerService](#)
Defines the contract for a service that sanitizes input strings to remove potentially harmful content, such as HTML tags.
- interface [IUserAgentParser](#)

5.11 ArandanoIRT_Backend.Application.Mappings Namespace Reference

Classes

- class [AuthMapping](#)
Provides static mapping methods for authentication related DTOs and Entities.

5.12 ArandanoIRT_Backend.Application.Services Namespace Reference

Classes

- class [AuthPasswordService](#)
Implements services for handling password reset requests and password changes using tokens.
- class [AuthRegistrationService](#)
Implements services for registering new administrators and regular users.
- class [AuthSessionService](#)
Implements services for user session management including login, token refresh, and logout.
- class [HelpRequestService](#)
Implements services for handling user help requests.

5.13 ArandanoIRT_Backend.Application.Utilities Namespace Reference

Classes

- class [AuthUtilities](#)
Implements shared utility functions for authentication services.
- class [EmailSenderUtility](#)
Provides utility methods for sending emails using SMTP settings from configuration.
- class [EmailValidatorUtility](#)
Provides static utility methods for validating email addresses, including syntax checks and DNS record verification (MX/A).

5.14 ArandanoIRT_Backend.Domain Namespace Reference

Namespaces

- namespace [Entities](#)
- namespace [Exceptions](#)
- namespace [IRepositories](#)
- namespace [ValueObjects](#)

5.15 ArandanoIRT_Backend.Domain.Entities Namespace Reference

Classes

- class [AuditCropEntity](#)
Represents a single audit trail record tracking changes specifically related to crop data.
- class [AuditDataTableEntity](#)
Represents a generic audit trail record tracking changes made to specific columns within identified database tables, associated with a particular crop.
- class [AuditDeviceEntity](#)

Represents a single audit trail record tracking changes specifically related to device entities or data, typically associated with a particular crop.

- class [AuditPersonEntity](#)

Represents a single audit trail record tracking changes specifically related to person (user) entities or data, associated with a particular crop.

- class [AuditSensitiveDataEntity](#)

Represents a single audit trail record specifically tracking actions involving access to, or operations on, sensitive data fields within specified database tables, associated with a crop.

- class [AuditSystemTableEntity](#)

Represents a generic audit trail record tracking changes made to specific columns within identified system-level or configuration database tables, potentially associated with a crop context.

- class [ChangePasswordEntity](#)

Represents a record containing a temporary token generated to allow a user to securely reset their password.

- class [CropEntity](#)

Represents an agricultural crop entity, containing its identification, name, location, and administrative details.

- class [CropInvitationEntity](#)

Represents an invitation code record used to grant access or link a new user to a specific crop.

- class [DeviceActivationEntity](#)

Represents a record containing an activation code used to register, authorize, or link a physical device to the system.

- class [DeviceDataEntity](#)

Represents a data-collecting device, containing its identification, description, configuration settings, registration details, and associations (crop, plant).

- class [DeviceLogEntity](#)

Represents a single log entry generated by, or related to, a specific device.

- class [DeviceTokenEntity](#)

Represents a pair of authentication tokens (access and refresh) issued specifically for a device, along with metadata about the token's lifecycle and the device context.

- class [FailedLoginAttemptEntity](#)

Represents a record logging a single failed login attempt against a specific user account.

- class [PersonEntity](#)

Represents a user account in the system, containing personal details, credentials (typically hashed password), status flags, timestamps, and associations like the assigned crop.

- class [PlantDataEntity](#)

Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop.

- class [PlantObservationEntity](#)

Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers.

- class [PlantStateHistoryEntity](#)

Represents a historical record documenting a change in a plant's status at a specific point in time, including who made the change.

- class [RefreshTokenEntity](#)

Represents a refresh token used to obtain new access tokens, extending user sessions.

- class [SensorDataEntity](#)

Represents a single snapshot of environmental sensor readings (temperature, humidity, light intensity) recorded at a specific time.

- class [StatusEntity](#)

Represents a defined status value (e.g., "Active", "Pending", "Used") that can be applied to various entities within the application.

- class [TableRelationEntity](#)

Represents a mapping between a unique identifier and a database table name.

- class [ThermalDataEntity](#)

Represents a single record containing thermal image data and corresponding RGB image data, captured at a specific time and potentially associated with a plant or crop.

5.16 ArandanoIRT_Backend.Domain.Exceptions Namespace Reference

Classes

- class [BusinessRuleViolationException](#)

Represents an error that occurs when an operation violates a specific business rule or constraint.

- class [DatabaseUpdateException](#)

Represents an error that occurs when a database update operation (e.g., insert, update, delete) fails unexpectedly or results in an inconsistent state.

- class [NotFoundException](#)

Represents an error that occurs when a requested resource, entity, or record could not be found in the system or data store.

- class [ValidationException](#)

Represents a general error that occurs when input data or an object's state fails one or more validation checks.

5.17 ArandanoIRT_Backend.Domain.IRepositories Namespace Reference

Classes

- interface [IAuditCropRepository](#)

Defines the repository contract specifically for managing AuditCropEntity instances.

- interface [IAuditDataTableRepository](#)

Defines the repository contract specifically for managing AuditDataTableEntity instances.

- interface [IAuditDeviceRepository](#)

Defines the repository contract specifically for managing AuditDeviceEntity instances.

- interface [IAuditPersonRepository](#)

Defines the repository contract specifically for managing AuditPersonEntity instances.

- interface [IAuditSensitiveDataRepository](#)

Defines the repository contract specifically for managing AuditSensitiveDataEntity instances.

- interface [IAuditSystemTableRepository](#)

Defines the repository contract specifically for managing AuditSystemTableEntity instances.

- interface [IChangePasswordRepository](#)

Defines the repository contract specifically for managing ChangePasswordEntity instances.

- interface [ICropInvitationRepository](#)

Defines the repository contract specifically for managing CropInvitationEntity instances.

- interface [ICropRepository](#)

Defines the repository contract specifically for managing CropEntity instances.

- interface [IDeviceActivationRepository](#)

Defines the repository contract specifically for managing DeviceActivationEntity instances.

- interface [IDeviceDataRepository](#)

- **interface [IDeviceLogRepository](#)**
Defines the repository contract specifically for managing DeviceDataEntity instances.
- **interface [IDeviceTokenRepository](#)**
Defines the repository contract specifically for managing DeviceLogEntity instances.
- **interface [IFailedLoginAttemptRepository](#)**
Defines the repository contract specifically for managing DeviceTokenEntity instances.
- **interface [IPersonRepository](#)**
Defines the repository contract specifically for managing FailedLoginAttemptEntity instances.
- **interface [IPlantDataRepository](#)**
Defines the repository contract specifically for managing PersonEntity instances.
- **interface [IPlantObservationRepository](#)**
Defines the repository contract specifically for managing PlantDataEntity instances.
- **interface [IPlantStateHistoryRepository](#)**
Defines the repository contract specifically for managing PlantObservationEntity instances.
- **interface [IRefreshTokenRepository](#)**
Defines the repository contract specifically for managing PlantStateHistoryEntity instances.
- **interface [IRepository](#)**
Defines a generic contract for repository operations for a specific entity type T .
- **interface [ISensorDataRepository](#)**
Defines the repository contract specifically for managing SensorDataEntity instances.
- **interface [IStatusRepository](#)**
Defines the repository contract specifically for managing StatusEntity instances.
- **interface [ITableRelationRepository](#)**
Defines the repository contract specifically for managing TableRelationEntity instances.
- **interface [IThermalDataRepository](#)**
Defines the repository contract specifically for managing ThermalDataEntity instances.

5.18 ArandanoIRT_Backend.Domain.ValueObjects Namespace Reference

Classes

- **class [Result](#)**
Represents the outcome of an operation, indicating either success or failure.

Functions

- **record [ClientInfo](#) (string UserAgentFamily="Unknown", string UserAgentVersion="", string OSFamily="Unknown OS", string OSVersion="", string DeviceFamily="Unknown Device")**
Holds parsed information about the client (e.g., browser, OS, device) making a request, typically derived from the User-Agent string.

5.18.1 Function Documentation

5.18.1.1 ClientInfo()

```
record ArandanoIRT_Backend.Domain.ValueObjects.ClientInfo (
    string UserAgentFamily = "Unknown",
    string UserAgentVersion = "",
    string OSFamily = "Unknown OS",
    string OSVersion = "",
    string DeviceFamily = "Unknown Device" )
```

Holds parsed information about the client (e.g., browser, OS, device) making a request, typically derived from the User-Agent string.

This is an immutable record.

Parameters

<i>UserAgentFamily</i>	The family name of the user agent (e.g., "Chrome", "Firefox", "MyApp"). Defaults to "Unknown".
<i>UserAgentVersion</i>	The version string of the user agent (e.g., "123.0.0"). Defaults to empty string.
<i>OSFamily</i>	The family name of the operating system (e.g., "Windows", "Android", "iOS"). Defaults to "Unknown OS".
<i>OSVersion</i>	The version string of the operating system (e.g., "10", "13"). Defaults to empty string.
<i>DeviceFamily</i>	The family or model name of the device (e.g., "Generic PC", "iPhone", "Samsung SM-G998U"). Defaults to "Unknown Device".

Provides a user-friendly formatted string representation of the client information. Example: "Chrome 123.0.0 / Windows 10" or "MyApp / Android 13" or "iPhone".

Returns

A formatted string summarizing the client information, prioritizing User Agent and OS details, falling back to Device Family, and ultimately "Unknown Device" if no information is known.

5.19 ArandanoIRT_Backend.Infrastructure Namespace Reference

Namespaces

- namespace [Attributes](#)
- namespace [Config](#)
- namespace [Data](#)
- namespace [Filters](#)
- namespace [Interfaces](#)
- namespace [Persistence](#)
- namespace [Repositories](#)
- namespace [Services](#)
- namespace [Utilities](#)

5.20 ArandanoIRT_Backend.Infrastructure.Attributes Namespace Reference

Classes

- class [SanitizeHtmlAttribute](#)

Attribute used to mark string properties on DTOs or models that should have HTML tags removed by the SanitizationActionFilter.

5.21 ArandanoIRT_Backend.Infrastructure.Config Namespace Reference

Classes

- class **AuthConfig**

Provides extension methods for configuring authentication services for the application.

- class **CORSConfig**

Provides extension methods for configuring Cross-Origin Resource Sharing (CORS) policies for the application.

- class **DependencyInjection**

Provides extension methods for configuring dependency injection services for the Infrastructure and Application layers.

- class **JsonConfig**

Provides extension methods for configuring JSON serialization options for the application, specifically for controllers using System.Text.Json.

- class **LoggingConfig**

Provides extension methods for configuring application logging using Serilog.

- class **RateLimitConfig**

Provides extension methods for configuring request rate limiting policies for the application using Microsoft.AspNetCore.RateLimiting.

- class **ResponseCompressionConfig**

Provides extension methods for configuring response compression services for the ASP.NET Core application pipeline.

- class **SwaggerConfig**

Provides extension methods for configuring Swashbuckle Swagger/OpenAPI generation services.

5.22 ArandanoIRT_Backend.Infrastructure.Data Namespace Reference

Classes

- class [ApplicationDbContext](#)

- class [Auditcrop](#)

- class [Auditdatatable](#)

- class [Auditdevice](#)

- class [Auditperson](#)

- class [Auditsensitizeddata](#)

- class [Auditsystemtable](#)

- class [Changepassword](#)

- class [Crop](#)

- class [Cropinvitation](#)
- class [Deviceactivation](#)
- class [Devicedata](#)
- class [Devicelog](#)
- class [Devicetoken](#)
- class [Failedloginattempt](#)
- class [Person](#)
- class [Plantdata](#)
- class [Plantobservation](#)
- class [Plantstatehistory](#)
- class [Refreshtoken](#)
- class [Sensordata](#)
- class [Status](#)
- class [Tablerelation](#)
- class [Thermaldata](#)

5.23 ArandanoIRT_Backend.Infrastructure.Filters Namespace Reference

Classes

- class [SanitizationActionFilter](#)

An ASP.NET Core action filter that automatically sanitizes string properties marked with the SanitizeHtmlAttribute on action method arguments (typically DTOs).

5.24 ArandanoIRT_Backend.Infrastructure.Interfaces Namespace Reference

Namespaces

- namespace [IServices](#)

5.25 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices Namespace Reference

Classes

- interface [ICacheService](#)

Defines a contract for a generic caching service.

- interface [ITokenService](#)

Defines a contract for services responsible for handling the lifecycle of authentication tokens, including the generation, storage, validation, refresh, and revocation of JWT access tokens and persistent refresh tokens.

5.26 ArandanoIRT_Backend.Infrastructure.Persistence Namespace Reference

Namespaces

- namespace [Auditing](#)
- namespace [Interceptors](#)

5.27 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing Namespace Reference

Classes

- class [AuditCropGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries specifically for changes made to Crop entities.
- class [AuditDataTableGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditdatatable for changes made to various data entities like Sensordata, Thermaldata, and Plantdata.
- class [AuditDeviceGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditdevice table for changes made to Devicedata entities.
- class [AuditHelperService](#)
Implements IAuditHelperService providing utility methods commonly used during the generation of audit log entries, such as primary key retrieval, foreign key retrieval (CropId), and property value serialization.
- class [AuditPersonGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditperson table for changes made to Person entities.
- class [AuditSystemTableGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditsystemtable for changes made to specific "system" or related data entities like Plantstatehistory and Plantobservation.

5.28 ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors Namespace Reference

Classes

- class [AuditSaveChangesInterceptor](#)
An SaveChangesInterceptor that automatically generates audit log entries for entity changes detected by the DbContext's ChangeTracker before saving.

5.29 ArandanoIRT_Backend.Infrastructure.Repositories Namespace Reference

Classes

- class [AuditCropRepository](#)
- class [AuditDataTableRepository](#)
- class [AuditDeviceRepository](#)
- class [AuditPersonRepository](#)
- class [AuditSensitiveDataRepository](#)
- class [AuditSystemTableRepository](#)
- class [DeviceActivationRepository](#)
- class [DeviceDataRepository](#)
- class [DeviceLogRepository](#)
- class [DeviceTokenRepository](#)
- class [PlantDataRepository](#)
- class [PlantObservationRepository](#)
- class [PlantStateHistoryRepository](#)
- class [SensorDataRepository](#)
- class [TableRelationRepository](#)
- class [ThermalDataRepository](#)

Functions

- class [StatusRepository](#) (`ApplicationDbContext` context, `ICacheService` cacheService, `IDateTimeProvider` date←
TimeProvider, `ILogger< StatusRepository >` logger)

Implements the `IStatusRepository` interface, providing data access logic for status entities (`StatusEntity`) using Entity Framework Core.

5.29.1 Function Documentation

5.29.1.1 StatusRepository()

```
class ArandanoIRT_Backend.Infrastructure.Repositories.StatusRepository (
    ApplicationDbContext context,
    ICacheService cacheService,
    IDateTimeProvider dateTimeProvider,
    ILogger< StatusRepository > logger )
```

Implements the `IStatusRepository` interface, providing data access logic for status entities (`StatusEntity`) using Entity Framework Core.

Includes caching mechanisms to improve performance for frequently accessed status data. The database context used for data access.

The caching service used for storing and retrieving status data.

Provider for obtaining consistent UTC timestamps (dependency added based on field usage).

Logger instance for logging repository operations.

Cache key for storing the list of all statuses.

Format string for cache keys storing statuses related to a specific table name.

Default duration for caching status data.

Maps a database context Status entity to a domain `StatusEntity`. Also maps the nested `Data.Tablerelation` if included.

Parameters

<i>status</i>	The database entity instance.
---------------	-------------------------------

Returns

The mapped domain entity instance, or `null` if input is null (throws `NullReferenceException` due to `null!` usage).

Uses reflection to set the `TableRelation` navigation property on the domain entity.

Maps a domain `StatusEntity` to a database context `Status` entity. Updates existing instance if provided, otherwise creates a new one. Does not map navigation properties like `TableRelation`.

Parameters

<i>entity</i>	The domain entity instance.
<i>existingStatus</i>	Optional. The existing database entity to update.

Returns

The mapped or updated database entity instance.

Invalidates cache entries related to statuses. Always removes the "All Statuses" cache entry. Also removes the table-specific cache entry if the table name is provided.

Parameters

<i>tableName</i>	Optional. The specific table name whose status cache should be invalidated.
------------------	---

5.30 ArandanoIRT_Backend.Infrastructure.Services Namespace Reference

Classes

- class [AuthOperationFilter](#)

An Swagger operation filter that adds a security requirement definition (referencing "cookieAuth") to operations protected by the `AuthorizeAttribute`.
- class [CloudflareTurnstileService](#)

Implements the `ICaptchaService` interface using the Cloudflare Turnstile API for verifying CAPTCHA challenges.
- class [CookiesService](#)

Provides methods for managing authentication-related HTTP cookies, such as setting, renewing, retrieving, and removing JWT, Refresh Token, and StayLoggedIn cookies.
- class [JwtTokenService](#)

Implements the `ITokenService` interface, providing services for generating, validating, refreshing, and revoking JWT access tokens and database-backed opaque refresh tokens.

- class [LogCleanupService](#)
A background service responsible for periodically deleting old log files from the configured log directory to manage disk space.
- class [ProducesJsonFilter](#)
An Swagger operation filter that attempts to determine the actual return type for 200 OK responses (handling Task and IActionResult with ProducesResponseType) and sets the corresponding JSON schema in the OpenAPI documentation.
- class [ProducesResponseTypeFilter](#)
An Swagger operation filter that adds standard HTTP error response types (400, 404, 429, 500) to every operation in the OpenAPI documentation.
- class [RequestContextAccessor](#)
Implements IRequestContextAccessor to provide access to common data from the current HTTP request context, such as IP address, User Agent details, and authenticated user information (ID, Crop ID).
- class [SanitizerService](#)
Implements the ISanitizerService.
- class [UnauthorizedResponseFilter](#)
An Swagger operation filter that adds the 401 Unauthorized response type to operations that are protected by an Authorize attribute (identified by name).

5.31 ArandanoIRT_Backend.Infrastructure.Utilities Namespace Reference

Classes

- class [BCryptPasswordHasher](#)
Implements the IPasswordHasher interface using the BCrypt algorithm provided by the BCrypt.Net library.
- class [DateTimeProvider](#)
Implements the IDateTimeProvider interface.
- class [PasswordRequest](#)
Represents a simple request structure containing a password string.
- class [RSAExtensions](#)
Provides internal static extension methods related to RSA key formatting, primarily for PEM encoding.
- class [RsaKeys](#)
Internal container class to hold the RSA private and public keys as strings.
- class [RsaService](#)
Implements the IRsaService interface, providing functionalities for RSA encryption (using public key) and decryption (using private key).
- class [RSAUtility](#)
Internal helper class responsible for generating an RSA key pair and performing core encryption and decryption operations using that key pair.
- class [SmtpEmailService](#)
Implements the IEmailService interface using an underlying EmailSenderUtility to send emails via SMTP.
- class [UAParserAdapter](#)
Implements the IUserAgentParser interface by adapting the UAParser library (uap-csharp) to parse User-Agent strings into the application's Domain.ValueObjects.ClientInfo structure.

5.32 ArandanoIRT_Backend.Tests Namespace Reference

Namespaces

- namespace [Application](#)
- namespace [Integration](#)
- namespace [Unit](#)

5.33 ArandanoIRT_Backend.Tests.Application Namespace Reference

Namespaces

- namespace [Services](#)
- namespace [Utilities](#)

5.34 ArandanoIRT_Backend.Tests.Application.Services Namespace Reference

Classes

- class [AuthPasswordServiceTests](#)
Contains unit tests for the AuthPasswordService class.
- class [AuthRegistrationServiceTests](#)
Contains unit tests for the AuthRegistrationService class.
- class [AuthSessionServiceTests](#)
Contains unit tests for the AuthSessionService class.
- class [HelpRequestServiceTests](#)
Contains unit tests for the HelpRequestService class.

5.35 ArandanoIRT_Backend.Tests.Application.Utilities Namespace Reference

Classes

- class [AuthUtilitiesTests](#)
Contains unit tests for the AuthUtilities class.
- class [EmailValidatorUtilityTests](#)
Contains unit tests for the static EmailValidatorUtility class.

5.36 ArandanoIRT_Backend.Tests.Integration Namespace Reference

Namespaces

- namespace [Common](#)
- namespace [Repositories](#)

5.37 ArandanoIRT_Backend.Tests.Integration.Common Namespace Reference

Classes

- class [DbContextTestHelper](#)

Provides helper methods for creating DbContext instances for integration tests using an SQLite in-memory database.

5.38 ArandanoIRT_Backend.Tests.Integration.Repositories Namespace Reference

Classes

- class [ChangePasswordRepositoryTests](#)

Contains integration tests for the ChangePasswordRepository using an SQLite in-memory database.

- class [CropRepositoryTests](#)

Contains integration tests for the CropRepository using an SQLite in-memory database.

- class [FailedLoginAttemptRepositoryTests](#)

Contains integration tests for the FailedLoginAttemptRepository using an SQLite in-memory database.

- class [PersonRepositoryTests](#)

Contains integration tests for the PersonRepository using an SQLite in-memory database.

- class [RefreshTokenRepositoryTests](#)

Contains integration tests for the RefreshTokenRepository using an SQLite in-memory database.

5.39 ArandanoIRT_Backend.Tests.Unit Namespace Reference

Namespaces

- namespace [Domain](#)
- namespace [Infrastructure](#)

5.40 ArandanoIRT_Backend.Tests.Unit.Domain Namespace Reference

Namespaces

- namespace [Entities](#)
- namespace [ValueObjects](#)

5.41 ArandanoIRT_Backend.Tests.Unit.Domain.Entities Namespace Reference

Classes

- class [PersonEntityTests](#)
Contains unit tests for the PersonEntity class.
- class [RefreshTokenEntityTests](#)
Contains unit tests for the RefreshTokenEntity class.
- class [SensorDataEntityTests](#)
Contains unit tests for the SensorDataEntity class.
- class [StatusEntityTests](#)
Contains unit tests for the StatusEntity class.

5.42 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects Namespace Reference

Classes

- class [ClientInfoTests](#)
Contains unit tests for the ClientInfo value object.
- class [ResultTests](#)
Contains unit tests for the non-generic Result class.
- class [ResultTTests](#)
Contains unit tests for the generic Result<T> class.

5.43 ArandanoIRT_Backend.Tests.Unit.Infrastructure Namespace Reference

Namespaces

- namespace [Persistence](#)
- namespace [Services](#)
- namespace [Utilities](#)

5.44 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence Namespace Reference

Classes

- class [AuditHelperServiceTests](#)
Contains unit tests for the AuditHelperService class.

5.45 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services Namespace Reference

Classes

- class [CacheServiceTests](#)
Contains unit tests for the CacheService class.
- class [CloudflareTurnstileServiceTests](#)
Contains unit tests for the CloudflareTurnstileService class.
- class [CookiesServiceTests](#)
Contains unit tests for the CookiesService class.
- class [JwtTokenServiceTests](#)
Contains unit tests for the JwtTokenService class.
- class [RequestContextAccessorTests](#)
Contains unit tests for the RequestContextAccessor class.
- class [SanitizerServiceTests](#)
Contains unit tests for the SanitizerService class (using default HtmlSanitizer + HtmlDecode).

5.46 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities Namespace Reference

Classes

- class [BCryptPasswordHasherTests](#)
Contains unit tests for the BCryptPasswordHasher class.
- class [DateTimeProviderTests](#)
Contains unit tests for the DateTimeProvider class.
- class [RsaServiceTests](#)
Contains unit tests for the RsaService class.
- class [SmtpEmailServiceTests](#)
Contains unit tests for the SmtpEmailService class.
- class [UAParserAdapterTests](#)
Contains unit tests for the UAParserAdapter class.

5.47 ArandanoIRT_Backend.UI Namespace Reference

Namespaces

- namespace [Controllers](#)
- namespace [Middleware](#)

5.48 ArandanoIRT_Backend.UI.Controllers Namespace Reference

Classes

- class [BaseApiController](#)
Base controller for API endpoints providing shared functionality.
- class [CryptoController](#)
Handles cryptographic utility endpoints.
- class [HelpRequestsController](#)
Handles endpoints related to user help requests.
- class [PasswordController](#)
Handles endpoints related to user password management, like reset requests.
- class [RegistrationController](#)
Handles endpoints related to user and administrator registration.
- class [SessionController](#)
Handles endpoints related to user session management, including login, token refresh, and logout operations.

5.49 ArandanoIRT_Backend.UI.Middleware Namespace Reference

Classes

- class [ExceptionHandlingMiddleware](#)
ASP.NET Core middleware for centralized exception handling.
- class [ExceptionHandlingMiddlewareExtensions](#)
Provides an extension method for easily registering the ExceptionHandlingMiddleware in the ASP.NET Core application pipeline.

Chapter 6

Class Documentation

6.1 ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto Class Reference

Represents the personal and credential information for an administrator user.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto:

ArandanoIRT_Backend.Application. DTOs.Auth.AdminInfoDto
+ FirstName
+ LastName
+ Email
+ Password

Properties

- required string `FirstName` [get, set]
Gets or sets the administrator's first name.
- required string `LastName` [get, set]
Gets or sets the administrator's last name.
- required string `Email` [get, set]
Gets or sets the administrator's email address, typically used for login.
- required string `Password` [get, set]
Gets or sets the administrator's initial password for the account.

6.1.1 Detailed Description

Represents the personal and credential information for an administrator user.

6.1.2 Property Documentation

6.1.2.1 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto.Email [get], [set]
```

Gets or sets the administrator's email address, typically used for login.

6.1.2.2 FirstName

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto.FirstName [get], [set]
```

Gets or sets the administrator's first name.

string

6.1.2.3 LastName

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto.LastName [get], [set]
```

Gets or sets the administrator's last name.

string

6.1.2.4 Password

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto.Password [get], [set]
```

Gets or sets the administrator's initial password for the account.

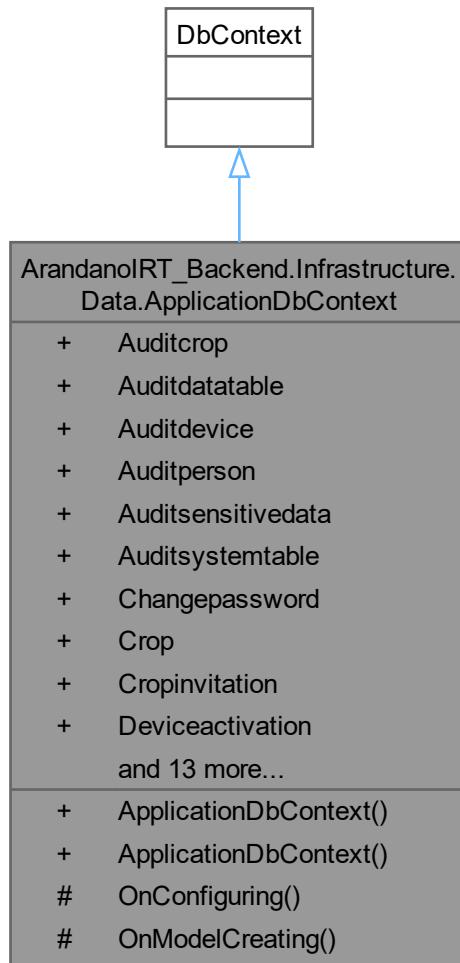
string

The documentation for this class was generated from the following file:

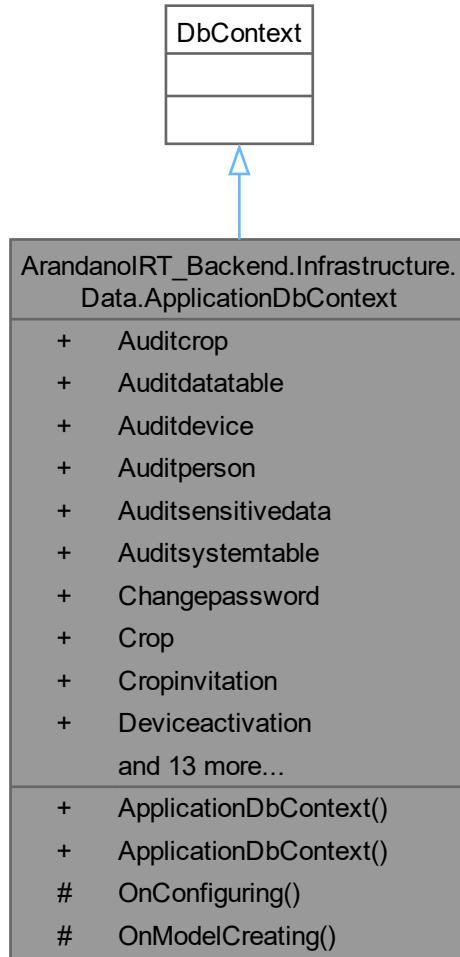
- Application/DTOs/Auth/[RegisterAdminRequestDto.cs](#)

6.2 ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext:



Public Member Functions

- [ApplicationDbContext \(\)](#)
- [ApplicationDbContext \(DbContextOptions< ApplicationDbContext > options\)](#)

Protected Member Functions

- override void [OnConfiguring \(DbContextOptionsBuilder optionsBuilder\)](#)
- override void [OnModelCreating \(ModelBuilder modelBuilder\)](#)

Properties

- virtual DbSet< Auditcrop > Auditcrop [get, set]
- virtual DbSet< Auditdatatable > Auditdatatable [get, set]
- virtual DbSet< Auditdevice > Auditdevice [get, set]
- virtual DbSet< Auditperson > Auditperson [get, set]
- virtual DbSet< Auditsensitizeddata > Auditsensitizeddata [get, set]
- virtual DbSet< Auditsystemtable > Auditsystemtable [get, set]
- virtual DbSet< Changepassword > Changepassword [get, set]
- virtual DbSet< Crop > Crop [get, set]
- virtual DbSet< Cropinvitation > Cropinvitation [get, set]
- virtual DbSet< Deviceactivation > Deviceactivation [get, set]
- virtual DbSet< Devicedata > Devicedata [get, set]
- virtual DbSet< Devicelog > Devicelog [get, set]
- virtual DbSet< Devicetoken > Devicetoken [get, set]
- virtual DbSet< Failedloginattempt > Failedloginattempt [get, set]
- virtual DbSet< Person > Person [get, set]
- virtual DbSet< Plantdata > Plantdata [get, set]
- virtual DbSet< Plantobservation > Plantobservation [get, set]
- virtual DbSet< Plantstatehistory > Plantstatehistory [get, set]
- virtual DbSet< RefreshToken > RefreshToken [get, set]
- virtual DbSet< Sensordata > Sensordata [get, set]
- virtual DbSet< Status > Status [get, set]
- virtual DbSet< Tablerelation > Tablerelation [get, set]
- virtual DbSet< Thermaldata > Thermaldata [get, set]

6.2.1 Constructor & Destructor Documentation

6.2.1.1 ApplicationDbContext() [1/2]

```
ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.ApplicationDbContext () [inline]
```

6.2.1.2 ApplicationDbContext() [2/2]

```
ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.ApplicationDbContext (
    DbContextOptions< ApplicationDbContext > options ) [inline]
```

6.2.2 Member Function Documentation

6.2.2.1 OnConfiguring()

```
override void ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.OnConfiguring (
    DbContextOptionsBuilder optionsBuilder ) [inline], [protected]
```

6.2.2.2 OnModelCreating()

```
override void ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.OnModelCreating (
    ModelBuilder modelBuilder ) [inline], [protected]
```

6.2.3 Property Documentation

6.2.3.1 Auditcrop

```
virtual DbSet<Auditcrop> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Auditcrop
[get], [set]
```

6.2.3.2 Auditdatatable

```
virtual DbSet<Auditdatatable> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←
Auditdatatable [get], [set]
```

6.2.3.3 Auditdevice

```
virtual DbSet<Auditdevice> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Auditdevice
[get], [set]
```

6.2.3.4 Auditperson

```
virtual DbSet<Auditperson> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Auditperson
[get], [set]
```

6.2.3.5 Auditsensitizeddata

```
virtual DbSet<Auditsensitizeddata> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←
Auditsensitizeddata [get], [set]
```

6.2.3.6 Auditstable

```
virtual DbSet<Auditstable> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←
Auditstable [get], [set]
```

6.2.3.7 Changepassword

```
virtual DbSet<Changepassword> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←
Changepassword [get], [set]
```

6.2.3.8 Crop

```
virtual DbSet<Crop> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Crop [get],  
[set]
```

6.2.3.9 Cropinvitation

```
virtual DbSet<Cropinvitation> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Cropinvitation [get], [set]
```

6.2.3.10 Deviceactivation

```
virtual DbSet<Deviceactivation> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Deviceactivation [get], [set]
```

6.2.3.11 Devicedata

```
virtual DbSet<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Devicedata  
[get], [set]
```

6.2.3.12 Devicelog

```
virtual DbSet<Devicelog> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Devicelog  
[get], [set]
```

6.2.3.13 Devicetoken

```
virtual DbSet<Devicetoken> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Devicetoken  
[get], [set]
```

6.2.3.14 Failedloginattempt

```
virtual DbSet<Failedloginattempt> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Failedloginattempt [get], [set]
```

6.2.3.15 Person

```
virtual DbSet<Person> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Person [get],  
[set]
```

6.2.3.16 Plantdata

```
virtual DbSet<Plantdata> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Plantdata  
[get], [set]
```

6.2.3.17 Plantobservation

```
virtual DbSet<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Plantobservation [get], [set]
```

6.2.3.18 Plantstatehistory

```
virtual DbSet<Plantstatehistory> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Plantstatehistory [get], [set]
```

6.2.3.19 RefreshToken

```
virtual DbSet<RefreshToken> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.RefreshToken  
[get], [set]
```

6.2.3.20 Sensordata

```
virtual DbSet<Sensordata> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Sensordata  
[get], [set]
```

6.2.3.21 Status

```
virtual DbSet<Status> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Status [get],  
[set]
```

6.2.3.22 Tablerelation

```
virtual DbSet<Tablerelation> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.←  
Tablerelation [get], [set]
```

6.2.3.23 Thermaldata

```
virtual DbSet<Thermaldata> ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext.Thermaldata  
[get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[ApplicationContext.cs](#)

6.3 ArandanoIRT_Backend.Infrastructure.Data.Auditcrop Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditcrop:

ArandanoIRT_Backend.Infrastructure. Data.Auditcrop	
+	Idauditcrop
+	Columnname
+	Recordid
+	Cropid
+	Action
+	Oldvalue
+	Newvalue
+	Performedat
+	Performedby
+	Performedbyip
+	Useragent

Properties

- int **Idauditcrop** [get, set]
- string **Columnname** = null! [get, set]
- int **Recordid** [get, set]
- int? **Cropid** [get, set]
- string **Action** = null! [get, set]
- string? **Oldvalue** [get, set]
- string? **Newvalue** [get, set]
- DateTime? **Performedat** [get, set]
- int? **Performedby** [get, set]
- string? **Performedbyip** [get, set]
- string? **Useragent** [get, set]

6.3.1 Property Documentation

6.3.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Action = null! [get], [set]
```

6.3.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Columnname = null! [get], [set]
```

6.3.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Cropid [get], [set]
```

6.3.1.4 Idauditcrop

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Idauditcrop [get], [set]
```

6.3.1.5 Newvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Newvalue [get], [set]
```

6.3.1.6 Oldvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Oldvalue [get], [set]
```

6.3.1.7 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Performedat [get], [set]
```

6.3.1.8 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Performedby [get], [set]
```

6.3.1.9 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Performedbyip [get], [set]
```

6.3.1.10 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Recordid [get], [set]
```

6.3.1.11 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditcrop.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditcrop.cs](#)

6.4 ArandanoIRT_Backend.Domain.Entities.AuditCropEntity Class Reference

Represents a single audit trail record tracking changes specifically related to crop data.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditCropEntity:

ArandanoIRT_Backend.Domain.Entities.AuditCropEntity
+ IdAuditCrop + ColumnName + RecordId + CropId + Action + OldValue + NewValue + PerformedAt + PerformedBy + PerformedByIp + UserAgent + AuditCropEntity()

Public Member Functions

- [AuditCropEntity](#) (int idAuditCrop, string columnName, int recordId, int cropId, string action, string? oldValue, string? newValue, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)

Initializes a new instance of the AuditCropEntity class with detailed audit information.

Properties

- int `IdAuditCrop` [get]
Gets the unique identifier for this audit log entry.
- string `ColumnName` [get]
Gets the name of the column or property that was changed.
- int `RecordId` [get]
Gets the identifier of the specific record (e.g., CropEntity ID) that was affected.
- int `CropId` [get]
Gets the identifier of the crop to which this audit entry pertains.
- string `Action` [get]
Gets the type of action performed (e.g., "INSERT", "UPDATE", "DELETE").
- string? `OldValue` [get]
Gets the value of the column/property before the change, if applicable (e.g., for UPDATE actions).
- string? `NewValue` [get]
Gets the value of the column/property after the change, if applicable (e.g., for INSERT or UPDATE actions).
- DateTime? `PerformedAt` [get]
Gets the date and time (usually UTC) when the action was performed.
- int? `PerformedBy` [get]
Gets the unique identifier of the user who performed the action, if available.
- string? `PerformedByIp` [get]
Gets the IP address from which the action was performed, if available.
- string? `UserAgent` [get]
Gets the user agent string of the client that performed the action, if available.

6.4.1 Detailed Description

Represents a single audit trail record tracking changes specifically related to crop data.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 AuditCropEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.AuditCropEntity (
    int idAuditCrop,
    string columnName,
    int recordId,
    int cropId,
    string action,
    string? oldValue,
    string? newValue,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditCropEntity class with detailed audit information.

Parameters

<i>idAuditCrop</i>	The unique ID for this audit record (often generated by the database).
<i>columnName</i>	The name of the column/property changed.
<i>recordId</i>	The ID of the affected crop-related record.
<i>cropId</i>	The ID of the crop this audit relates to.
<i>action</i>	The action performed (e.g., "UPDATE").
<i>oldValue</i>	The previous value (nullable).
<i>newValue</i>	The new value (nullable).
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedBy← Bylp</i>	The IP address of the user (nullable).
<i>userAgent</i>	The user agent of the client (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	---

6.4.3 Property Documentation

6.4.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.Action [get]
```

Gets the type of action performed (e.g., "INSERT", "UPDATE", "DELETE").

6.4.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.ColumnName [get]
```

Gets the name of the column or property that was changed.

6.4.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.CropId [get]
```

Gets the identifier of the crop to which this audit entry pertains.

6.4.3.4 **IdAuditCrop**

```
int ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.IdAuditCrop [get]
```

Gets the unique identifier for this audit log entry.

6.4.3.5 **NewValue**

```
string? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.NewValue [get]
```

Gets the value of the column/property after the change, if applicable (e.g., for INSERT or UPDATE actions).

Null otherwise.

6.4.3.6 **OldValue**

```
string? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.OldValue [get]
```

Gets the value of the column/property before the change, if applicable (e.g., for UPDATE actions).

Null otherwise.

6.4.3.7 **PerformedAt**

```
DateTime? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action was performed.

Nullable.

6.4.3.8 **PerformedBy**

```
int? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action, if available.

Nullable.

6.4.3.9 **PerformedByIp**

```
string? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.PerformedByIp [get]
```

Gets the IP address from which the action was performed, if available.

Nullable.

6.4.3.10 RecordId

```
int ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.RecordId [get]
```

Gets the identifier of the specific record (e.g., CropEntity ID) that was affected.

6.4.3.11 UserAgent

```
string? ArandanoIRT_Backend.Domain.Entities.AuditCropEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if available.

Nullable.

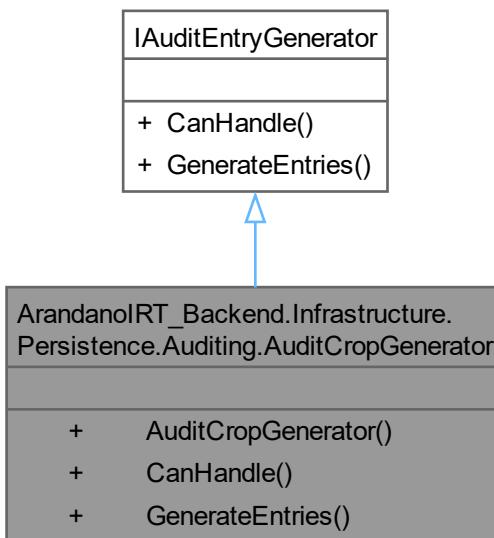
The documentation for this class was generated from the following file:

- Domain/Entities/[AuditCropEntity.cs](#)

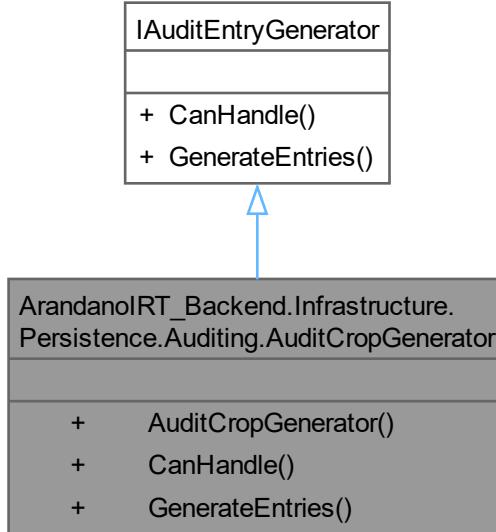
6.5 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator Class Reference

Implements IAuditEntryGenerator to create audit trail entries specifically for changes made to Crop entities.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator:



Public Member Functions

- **AuditCropGenerator** ([IAuditHelperService](#) auditHelper, [ILogger< AuditCropGenerator >](#) logger)

Initializes a new instance of the AuditCropGenerator class.

- **bool CanHandle** ([EntityEntry](#) entry)

Determines whether this generator can process the specified entity entry for auditing.

Parameters

entry	<i>The Entity Framework Core EntityEntry representing the tracked change.</i>
-------	---

Returns

true if this generator can handle the entry; otherwise, false.

- [IEnumerable< object >](#) **GenerateEntries** ([EntityEntry](#) entry, [AuditMetadata](#) metadata)

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

entry	<i>The EntityEntry containing the details of the entity change (e.g., original and current values).</i>
metadata	<i>The AuditMetadata containing context about the action being audited (user, time, etc.).</i>

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

6.5.1 Detailed Description

Implements `IAuditEntryGenerator` to create audit trail entries specifically for changes made to `Crop` entities.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 AuditCropGenerator()

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator.AuditCropGenerator (
    IAuditHelperService auditHelper,
    ILogger< AuditCropGenerator > logger ) [inline]
```

Initializes a new instance of the `AuditCropGenerator` class.

Parameters

<code>auditHelper</code>	The helper service for common auditing tasks like serialization and ID retrieval.
<code>logger</code>	The logger for recording information and warnings.

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>auditHelper</code> or <code>logger</code> is null.
------------------------------------	--

6.5.3 Member Function Documentation

6.5.3.1 CanHandle()

```
bool ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator.CanHandle (
    EntityEntry entry ) [inline]
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core <code>EntityEntry</code> representing the tracked change.
--------------------	---

Returns

true if this generator can handle the entry; otherwise, false.

This generator specifically handles entities of type `Crop`.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

6.5.3.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator.←
GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata ) [inline]
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	The EntityEntry containing the details of the entity change (e.g., original and current values).
<code>metadata</code>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Generates a single `AuditCrop` record for INSERT, UPDATE, or DELETE operations on a `Crop` entity. It currently performs row-level auditing by setting `ColumnName` to "ALL" and serializing the entire entity's original/current values using the `IAuditHelperService`. For INSERT operations, `RecordId` and `CropId` are set to placeholders (0/null) as they are not yet available.

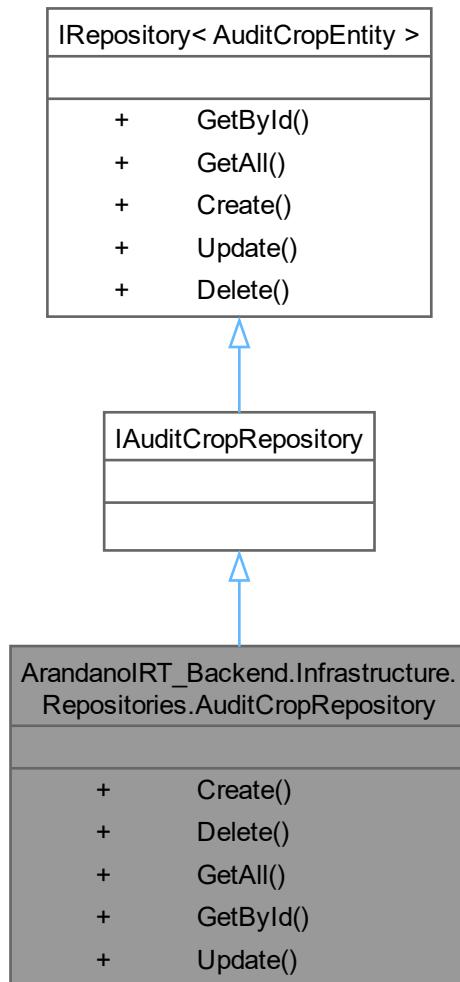
Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

The documentation for this class was generated from the following file:

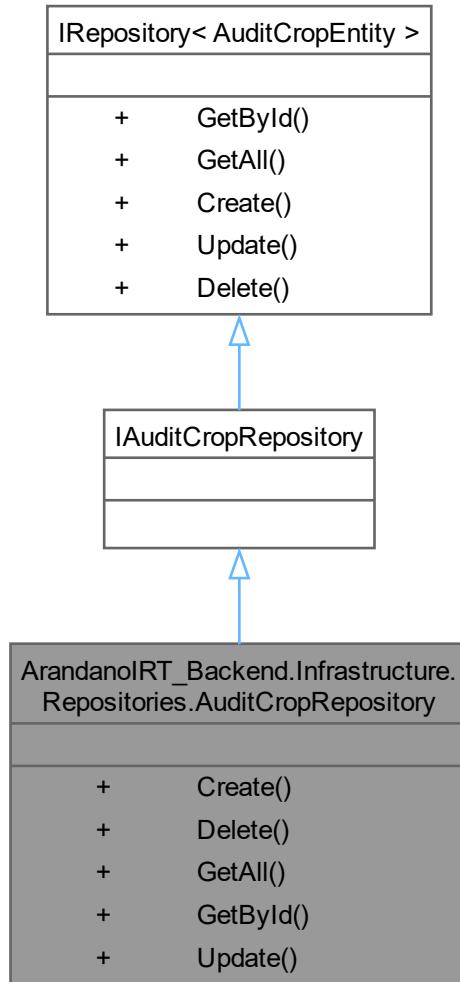
- Infrastructure/Persistence/Auditing/[AuditCropGenerator.cs](#)

6.6 ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository:



Public Member Functions

- Task< [Result< AuditCropEntity >](#) > [Create](#) (AuditCropEntity entity)
- Task< [Result< bool >](#) > [Delete](#) (int id)
- Task< [Result< IEnumerable< AuditCropEntity >](#) > > [GetAll](#) ()
- Task< [Result< AuditCropEntity >](#) > [GetById](#) (int id)
- Task< [Result< bool >](#) > [Update](#) (AuditCropEntity entity)

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditCropEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)

- Asynchronously retrieves an entity by its unique identifier.
Task< Result< IEnumerable< T > > > GetAll ()
- Asynchronously retrieves all entities of type *T*.
Task< Result< T > > Create (*T* entity)
- Asynchronously creates a new entity in the data store.
Task< Result< bool > > Update (*T* entity)
- Asynchronously updates an existing entity in the data store.
Task< Result< bool > > Delete (*int id*)
- Asynchronously deletes an entity from the data store by its unique identifier.

6.6.1 Member Function Documentation

6.6.1.1 Create()

```
Task< Result< AuditCropEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditCrop←
Repository.Create (
    AuditCropEntity entity) [inline]
```

6.6.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository.←
Delete (
    int id) [inline]
```

6.6.1.3 GetAll()

```
Task< Result< IEnumerable< AuditCropEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.←
AuditCropRepository.GetAll () [inline]
```

6.6.1.4 GetById()

```
Task< Result< AuditCropEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditCrop←
Repository.GetById (
    int id) [inline]
```

6.6.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository.←
Update (
    AuditCropEntity entity) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/[AuditCropRepository.cs](#)

6.7 ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable:

ArandanoIRT_Backend.Infrastructure. Data.Auditdatatable	
+	<code>Iauditdatatable</code>
+	<code>Tablename</code>
+	<code>Columnname</code>
+	<code>Recordid</code>
+	<code>Cropid</code>
+	<code>Action</code>
+	<code>Oldvalue</code>
+	<code>Newvalue</code>
+	<code>Performedat</code>
+	<code>Performedby</code>
+	<code>Performedbyip</code>
+	<code>Useragent</code>

Properties

- int `Iauditdatatable` [get, set]
- string `Tablename` = null! [get, set]
- string `Columnname` = null! [get, set]
- int `Recordid` [get, set]
- int? `Cropid` [get, set]
- string `Action` = null! [get, set]
- string? `Oldvalue` [get, set]
- string? `Newvalue` [get, set]
- DateTime? `Performedat` [get, set]
- int? `Performedby` [get, set]
- string? `Performedbyip` [get, set]
- string? `Useragent` [get, set]

6.7.1 Property Documentation

6.7.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Action = null! [get], [set]
```

6.7.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Columnname = null! [get], [set]
```

6.7.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Cropid [get], [set]
```

6.7.1.4 Idauditdatatable

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Idauditdatatable [get], [set]
```

6.7.1.5 Newvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Newvalue [get], [set]
```

6.7.1.6 Oldvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Oldvalue [get], [set]
```

6.7.1.7 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Performedat [get], [set]
```

6.7.1.8 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Performedby [get], [set]
```

6.7.1.9 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Performedbyip [get], [set]
```

6.7.1.10 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Recordid [get], [set]
```

6.7.1.11 Tablename

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Tablename = null! [get], [set]
```

6.7.1.12 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditdatatable.cs](#)

6.8 ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity Class Reference

Represents a generic audit trail record tracking changes made to specific columns within identified database tables, associated with a particular crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity:

ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity
+ IdAuditDataTable
+ TableName
+ ColumnName
+ RecordId
+ CropId
+ Action
+ OldValue
+ NewValue
+ PerformedAt
+ PerformedBy
+ PerformedByIp
+ UserAgent
+ AuditDataTableEntity()

Public Member Functions

- `AuditDataTableEntity` (int idAuditDataTable, string tableName, string columnName, int recordId, int cropId, string action, string? oldValue, string? newValue, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)

Initializes a new instance of the AuditDataTableEntity class with detailed audit information.

Properties

- int `IdAuditDataTable` [get]
Gets the unique identifier for this audit log entry.
- string `TableName` [get]
Gets the name of the database table where the change occurred.
- string `ColumnName` [get]
Gets the name of the column or property that was changed within the table.
- int `RecordId` [get]
Gets the identifier (primary key) of the specific record within the table that was affected.
- int `CropId` [get]
Gets the identifier of the crop to which this audited record pertains.
- string `Action` [get]
Gets the type of action performed (e.g., "INSERT", "UPDATE", "DELETE").
- string? `OldValue` [get]
Gets the value of the column before the change, if applicable (e.g., for UPDATE actions).
- string? `NewValue` [get]
Gets the value of the column after the change, if applicable (e.g., for INSERT or UPDATE actions).
- DateTime? `PerformedAt` [get]
Gets the date and time (usually UTC) when the action was performed.
- int? `PerformedBy` [get]
Gets the unique identifier of the user who performed the action, if available.
- string? `PerformedByIp` [get]
Gets the IP address from which the action was performed, if available.
- string? `UserAgent` [get]
Gets the user agent string of the client that performed the action, if available.

6.8.1 Detailed Description

Represents a generic audit trail record tracking changes made to specific columns within identified database tables, associated with a particular crop.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 AuditDataTableEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.AuditDataTableEntity (
    int idAuditDataTable,
    string tableName,
    string columnName,
    int recordId,
    int cropId,
    string action,
    string? oldValue,
    string? newValue,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditDataTableEntity class with detailed audit information.

Parameters

<i>idAuditDataTable</i>	The unique ID for this audit record (often generated by the database).
<i>tableName</i>	The name of the table where the change occurred.
<i>columnName</i>	The name of the column changed.
<i>recordId</i>	The ID of the affected record within the table.
<i>cropId</i>	The ID of the crop this audit relates to.
<i>action</i>	The action performed (e.g., "UPDATE").
<i>oldValue</i>	The previous value (nullable).
<i>newValue</i>	The new value (nullable).
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedByIp</i>	The IP address of the user (nullable).
<i>userAgent</i>	The user agent of the client (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>tableName</i> is null or whitespace (Spanish message: "El nombre de la tabla es requerido."). Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	--

6.8.3 Property Documentation

6.8.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.Action [get]
```

Gets the type of action performed (e.g., "INSERT", "UPDATE", "DELETE").

6.8.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.ColumnName [get]
```

Gets the name of the column or property that was changed within the table.

6.8.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.CropId [get]
```

Gets the identifier of the crop to which this audited record pertains.

6.8.3.4 IdAuditDataTable

```
int ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.IdAuditDataTable [get]
```

Gets the unique identifier for this audit log entry.

6.8.3.5 NewValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.NewValue [get]
```

Gets the value of the column after the change, if applicable (e.g., for INSERT or UPDATE actions).

Null otherwise.

6.8.3.6 OldValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.OldValue [get]
```

Gets the value of the column before the change, if applicable (e.g., for UPDATE actions).

Null otherwise.

6.8.3.7 PerformedAt

```
Datetime? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action was performed.

Nullable.

6.8.3.8 **PerformedBy**

```
int? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action, if available.

Nullable.

6.8.3.9 **PerformedByIp**

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.PerformedByIp [get]
```

Gets the IP address from which the action was performed, if available.

Nullable.

6.8.3.10 **RecordId**

```
int ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.RecordId [get]
```

Gets the identifier (primary key) of the specific record within the table that was affected.

6.8.3.11 **TableName**

```
string ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.TableName [get]
```

Gets the name of the database table where the change occurred.

6.8.3.12 **UserAgent**

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if available.

Nullable.

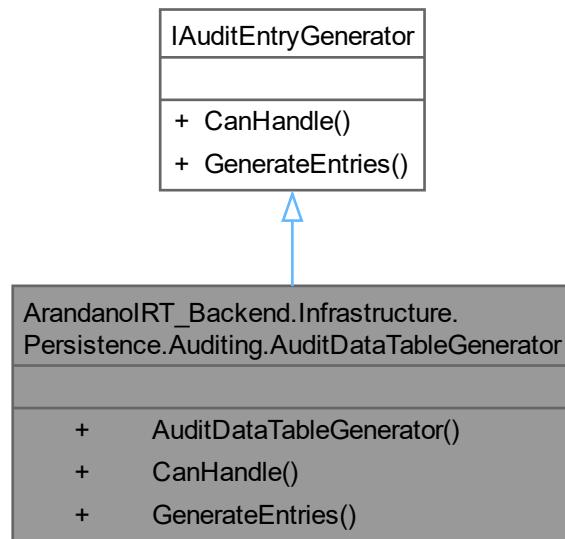
The documentation for this class was generated from the following file:

- Domain/Entities/[AuditDataTableEntity.cs](#)

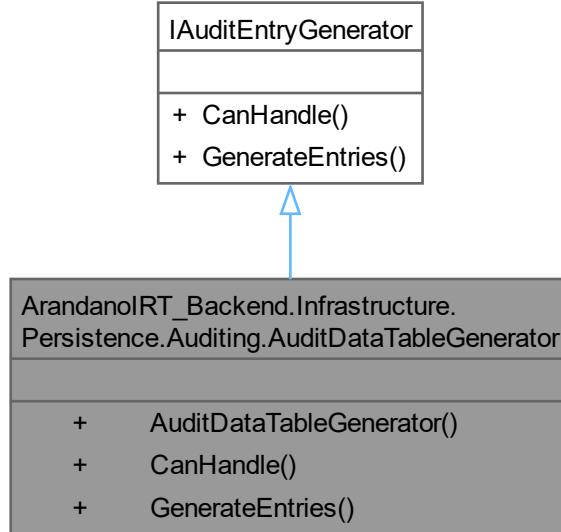
6.9 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator Class Reference

Implements IAuditEntryGenerator to create audit trail entries in the Auditdatatable for changes made to various data entities like Sensordata, Thermaldata, and Plantdata.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator:



Public Member Functions

- **AuditDataTableGenerator** ([IAuditHelperService](#) auditHelper, [ILogger< AuditDataTableGenerator >](#) logger)

Initializes a new instance of the AuditDataTableGenerator class.
- **bool CanHandle** ([EntityEntry](#) entry)

Determines whether this generator can process the specified entity entry for auditing.

Parameters

entry	<i>The Entity Framework Core EntityEntry representing the tracked change.</i>
-------	---

Returns

true if this generator can handle the entry; otherwise, false.

- **IEnumerable< object > GenerateEntries** ([EntityEntry](#) entry, [AuditMetadata](#) metadata)

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

entry	<i>The EntityEntry containing the details of the entity change (e.g., original and current values).</i>
metadata	<i>The AuditMetadata containing context about the action being audited (user, time, etc.).</i>

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

6.9.1 Detailed Description

Implements `IAuditEntryGenerator` to create audit trail entries in the `Auditdatatable` for changes made to various data entities like `Sensordata`, `Thermaldata`, and `Plantdata`.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 AuditDataTableGenerator()

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator.AuditDataTable←
Generator (
    IAuditHelperService auditHelper,
    ILogger< AuditDataTableGenerator > logger ) [inline]
```

Initializes a new instance of the `AuditDataTableGenerator` class.

Parameters

<code>auditHelper</code>	The helper service for common auditing tasks like serialization and ID retrieval.
<code>logger</code>	The logger for recording information and warnings.

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>auditHelper</code> or <code>logger</code> is null.
------------------------------------	--

6.9.3 Member Function Documentation

6.9.3.1 CanHandle()

```
bool ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator.CanHandle (
    EntityEntry entry ) [inline]
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core <code>EntityEntry</code> representing the tracked change.
--------------------	---

Returns

`true` if this generator can handle the entry; otherwise, `false`.

This generator handles entities of type `Sensordata`, `Thermaldata`, and `Plantdata`, directing their audit logs to the `Auditdatatable`.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

6.9.3.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTable←
Generator.GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata ) [inline]
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	The <code>EntityEntry</code> containing the details of the entity change (e.g., original and current values).
<code>metadata</code>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Generates a single `Auditdatatable` record for INSERT, UPDATE, or DELETE operations on supported entity types (`Sensordata`, `Thermaldata`, `Plantdata`). Performs row-level auditing ("ALL" column) using the `IAuditHelper` Service.

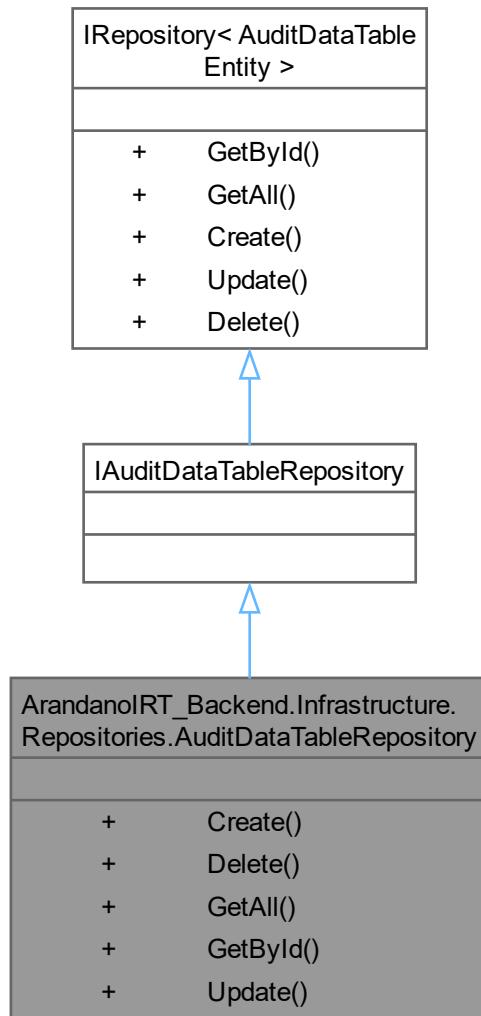
Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

The documentation for this class was generated from the following file:

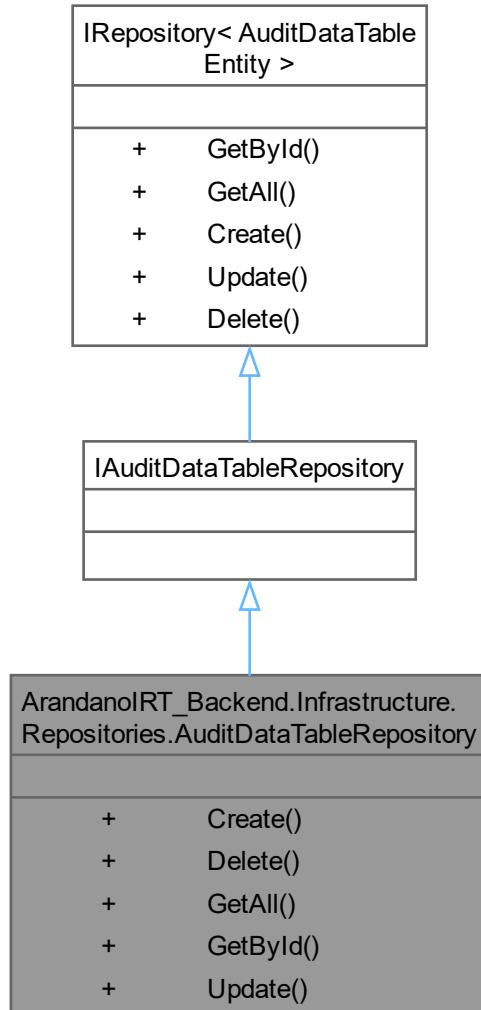
- Infrastructure/Persistence/Auditing/[AuditDataTableGenerator.cs](#)

6.10 ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository:



Public Member Functions

- Task< [Result< AuditDataTableEntity >](#) > [Create \(AuditDataTableEntity entity\)](#)
- Task< [Result< bool >](#) > [Delete \(int id\)](#)
- Task< [Result< IEnumerable< AuditDataTableEntity >](#) > > [GetAll \(\)](#)
- Task< [Result< AuditDataTableEntity >](#) > [GetById \(int id\)](#)
- Task< [Result< bool >](#) > [Update \(AuditDataTableEntity entity\)](#)

Public Member Functions inherited from ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditDataTableEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.10.1 Member Function Documentation

6.10.1.1 Create()

```
Task< Result< AuditDataTableEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.Audit←
DataTableRepository.Create (
    AuditDataTableEntity entity ) [inline]
```

6.10.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository.←
Delete (
    int id ) [inline]
```

6.10.1.3 GetAll()

```
Task< Result< IEnumerable< AuditDataTableEntity > > > ArandanoIRT_Backend.Infrastructure.←
Repositories.AuditDataTableRepository.GetAll ( ) [inline]
```

6.10.1.4 GetById()

```
Task< Result< AuditDataTableEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.Audit←
DataTableRepository.GetById (
    int id ) [inline]
```

6.10.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository.←  
Update (   
    AuditDataTableEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/AuditDataTableRepository.cs

6.11 ArandanoIRT_Backend.Infrastructure.Data.Auditdevice Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditdevice:

ArandanoIRT_Backend.Infrastructure. Data.Auditdevice	
+	Idauditdevice
+	Columnname
+	Recordid
+	Cropid
+	Action
+	Oldvalue
+	Newvalue
+	Performedat
+	Performedby
+	Performedbyip
+	Useragent

Properties

- int `Idauditdevice` [get, set]
- string `Columnname` = null! [get, set]
- int `Recordid` [get, set]
- int? `Cropid` [get, set]

- string **Action** = null! [get, set]
- string? **Oldvalue** [get, set]
- string? **Newvalue** [get, set]
- DateTime? **Performedat** [get, set]
- int? **Performedby** [get, set]
- string? **Performedbyip** [get, set]
- string? **Useragent** [get, set]

6.11.1 Property Documentation

6.11.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Action = null! [get], [set]
```

6.11.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Columnname = null! [get], [set]
```

6.11.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Cropid [get], [set]
```

6.11.1.4 Idauditdevice

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Idauditdevice [get], [set]
```

6.11.1.5 Newvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Newvalue [get], [set]
```

6.11.1.6 Oldvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Oldvalue [get], [set]
```

6.11.1.7 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Performedat [get], [set]
```

6.11.1.8 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Performedby [get], [set]
```

6.11.1.9 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Performedbyip [get], [set]
```

6.11.1.10 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Recordid [get], [set]
```

6.11.1.11 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditdevice.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditdevice.cs](#)

6.12 ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity Class Reference

Represents a single audit trail record tracking changes specifically related to device entities or data, typically associated with a particular crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity:

ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity
+ IdAuditDevice
+ ColumnName
+ RecordId
+ CropId
+ Action
+ OldValue
+ NewValue
+ PerformedAt
+ PerformedBy
+ PerformedByIp
+ UserAgent
+ AuditDeviceEntity()

Public Member Functions

- **AuditDeviceEntity** (int idAuditDevice, string columnName, int recordId, int cropId, string action, string? oldValue, string? newValue, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)

Initializes a new instance of the AuditDeviceEntity class with detailed audit information for a device-related change.

Properties

- int **IdAuditDevice** [get]
Gets the unique identifier for this device audit log entry.
- string **ColumnName** [get]
Gets the name of the column or property within the device entity that was changed.
- int **RecordId** [get]
Gets the identifier (primary key) of the specific device record that was affected.
- int **CropId** [get]
Gets the identifier of the crop to which the audited device belongs or is related.
- string **Action** [get]
Gets the type of action performed (e.g., "REGISTER", "UPDATE_CONFIG", "DELETE").
- string? **OldValue** [get]
Gets the value of the column/property before the change, if applicable (e.g., for UPDATE actions).

- string? **NewValue** [get]
Gets the value of the column/property after the change, if applicable (e.g., for creation or UPDATE actions).
- DateTime? **PerformedAt** [get]
Gets the date and time (usually UTC) when the action was performed.
- int? **PerformedBy** [get]
Gets the unique identifier of the user who performed the action, if available (e.g., linking a device).
- string? **PerformedByIp** [get]
Gets the IP address from which the action was performed or originated, if available.
- string? **UserAgent** [get]
Gets the user agent string of the client that performed the action, if applicable.

6.12.1 Detailed Description

Represents a single audit trail record tracking changes specifically related to device entities or data, typically associated with a particular crop.

6.12.2 Constructor & Destructor Documentation

6.12.2.1 AuditDeviceEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.AuditDeviceEntity (
    int idAuditDevice,
    string columnName,
    int recordId,
    int cropId,
    string action,
    string? oldValue,
    string? newValue,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditDeviceEntity class with detailed audit information for a device-related change.

Parameters

<i>idAuditDevice</i>	The unique ID for this audit record (often generated by the database).
<i>columnName</i>	The name of the column/property changed in the device entity.
<i>recordId</i>	The ID of the affected device record.
<i>cropId</i>	The ID of the crop this device relates to.
<i>action</i>	The action performed (e.g., "UPDATE_CONFIG").
<i>oldValue</i>	The previous value (nullable).
<i>newValue</i>	The new value (nullable).
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedByIp</i>	The IP address associated with the action (nullable).
<i>userAgent</i>	The user agent of the client, if applicable (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	---

6.12.3 Property Documentation

6.12.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.Action [get]
```

Gets the type of action performed (e.g., "REGISTER", "UPDATE_CONFIG", "DELETE").

6.12.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.ColumnName [get]
```

Gets the name of the column or property within the device entity that was changed.

6.12.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.CropId [get]
```

Gets the identifier of the crop to which the audited device belongs or is related.

6.12.3.4 IdAuditDevice

```
int ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.IdAuditDevice [get]
```

Gets the unique identifier for this device audit log entry.

6.12.3.5 NewValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.NewValue [get]
```

Gets the value of the column/property after the change, if applicable (e.g., for creation or UPDATE actions).

Null otherwise.

6.12.3.6 OldValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.OldValue [get]
```

Gets the value of the column/property before the change, if applicable (e.g., for UPDATE actions).

Nullable.

6.12.3.7 PerformedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action was performed.

Nullable.

6.12.3.8 PerformedBy

```
int? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action, if available (e.g., linking a device).

Nullable.

6.12.3.9 PerformedByIp

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.PerformedByIp [get]
```

Gets the IP address from which the action was performed or originated, if available.

Nullable.

6.12.3.10 RecordId

```
int ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.RecordId [get]
```

Gets the identifier (primary key) of the specific device record that was affected.

6.12.3.11 UserAgent

```
string? ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if applicable.

Nullable.

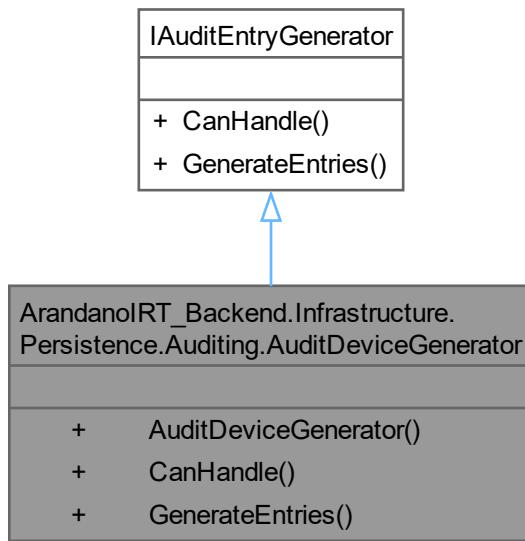
The documentation for this class was generated from the following file:

- Domain/Entities/[AuditDeviceEntity.cs](#)

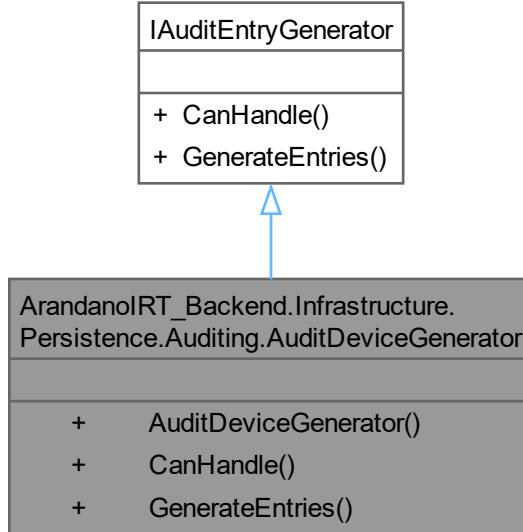
6.13 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator Class Reference

Implements IAuditEntryGenerator to create audit trail entries in the Auditdevice table for changes made to Devicedata entities.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator:



Public Member Functions

- **`AuditDeviceGenerator`** (`IAuditHelperService auditHelper, ILogger< AuditDeviceGenerator > logger)`
Initializes a new instance of the AuditDeviceGenerator class.
- **`bool CanHandle (EntityEntry entry)`**
Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	<i>The Entity Framework Core EntityEntry representing the tracked change.</i>
--------------------	---

Returns

true if this generator can handle the entry; otherwise, false.

- **`IEnumerable< object > GenerateEntries (EntityEntry entry, AuditMetadata metadata)`**
Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	<i>The EntityEntry containing the details of the entity change (e.g., original and current values).</i>
<code>metadata</code>	<i>The AuditMetadata containing context about the action being audited (user, time, etc.).</i>

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

6.13.1 Detailed Description

Implements `IAuditEntryGenerator` to create audit trail entries in the `Auditdevice` table for changes made to `Devicedata` entities.

6.13.2 Constructor & Destructor Documentation

6.13.2.1 AuditDeviceGenerator()

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator.AuditDeviceGenerator
(
    IAuditHelperService auditHelper,
    ILogger< AuditDeviceGenerator > logger ) [inline]
```

Initializes a new instance of the `AuditDeviceGenerator` class.

Parameters

<code>auditHelper</code>	The helper service for common auditing tasks.
<code>logger</code>	The logger for recording information and warnings.

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>auditHelper</code> or <code>logger</code> is null.
------------------------------------	--

6.13.3 Member Function Documentation

6.13.3.1 CanHandle()

```
bool ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator.CanHandle (
    EntityEntry entry ) [inline]
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core <code>EntityEntry</code> representing the tracked change.
--------------------	---

Returns

true if this generator can handle the entry; otherwise, false.

This generator specifically handles entities of type Devicedata.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

6.13.3.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator.←
GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata ) [inline]
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<i>entry</i>	The EntityEntry containing the details of the entity change (e.g., original and current values).
<i>metadata</i>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An IEnumerable<T> collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Generates a single Auditdevice record for INSERT, UPDATE, or DELETE operations on a Devicedata entity. Performs row-level auditing ("ALL" column) using the IAuditHelperService.

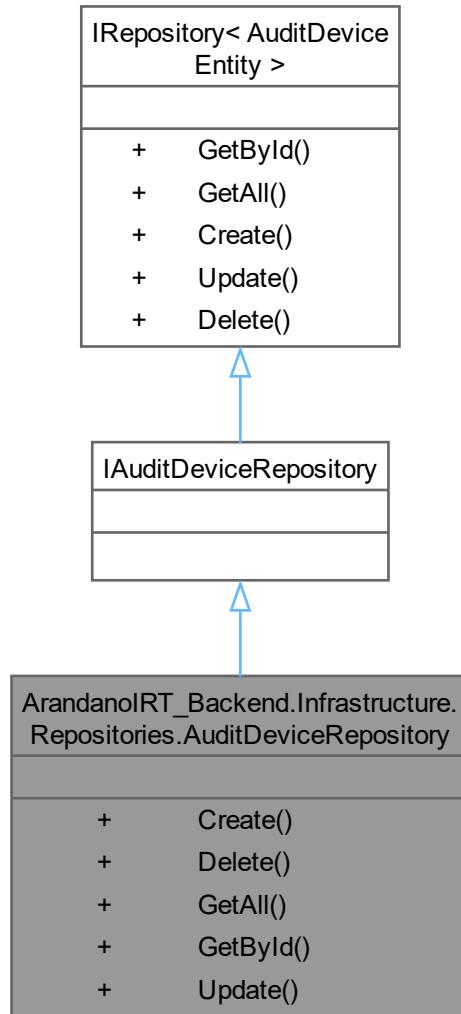
Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

The documentation for this class was generated from the following file:

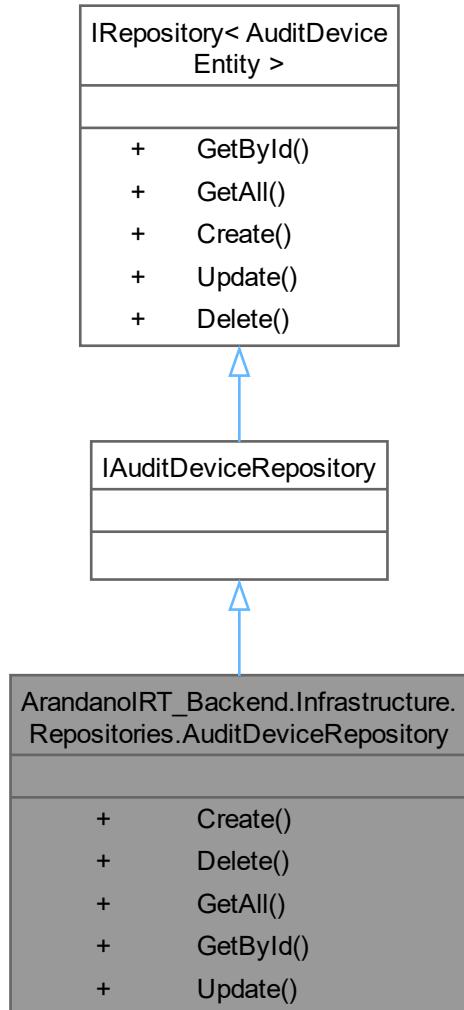
- Infrastructure/Persistence/Auditing/[AuditDeviceGenerator.cs](#)

6.14 ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository:



Public Member Functions

- `Task< Result< AuditDeviceEntity > > Create (AuditDeviceEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< AuditDeviceEntity > > > GetAll ()`
- `Task< Result< AuditDeviceEntity > > GetById (int id)`
- `Task< Result< bool > > Update (AuditDeviceEntity entity)`

Public Member Functions inherited from ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditDeviceEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.14.1 Member Function Documentation

6.14.1.1 Create()

```
Task< Result< AuditDeviceEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository.Create (
    AuditDeviceEntity entity ) [inline]
```

6.14.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository.Delete (
    int id ) [inline]
```

6.14.1.3 GetAll()

```
Task< Result< IEnumerable< AuditDeviceEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository.GetAll ( ) [inline]
```

6.14.1.4 GetById()

```
Task< Result< AuditDeviceEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository.GetById (
    int id ) [inline]
```

6.14.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository.←  
Update (   
    AuditDeviceEntity entity ) [inline]
```

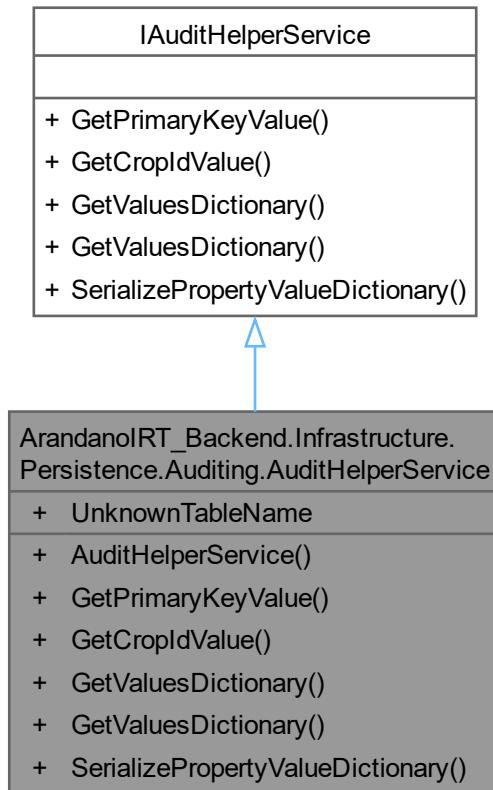
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/AuditDeviceRepository.cs

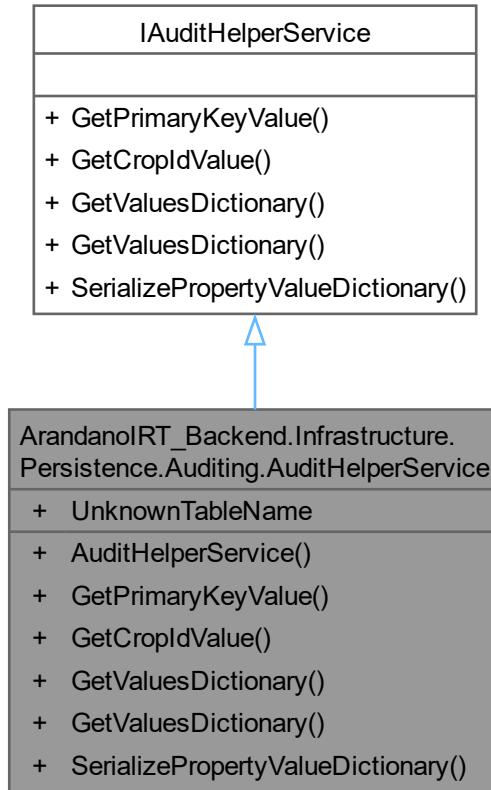
6.15 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService Class Reference

Implements IAuditHelperService providing utility methods commonly used during the generation of audit log entries, such as primary key retrieval, foreign key retrieval (CropId), and property value serialization.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService:



Classes

- class [Person](#)

Public Member Functions

- **[AuditHelperService](#)** (`ILogger< AuditHelperService > logger`)
Initializes a new instance of the AuditHelperService class.
- int? **[GetPrimaryKeyValue](#)** (`EntityEntry entry`)
Gets the primary key value of the entity represented by the specified entry, assuming the primary key is a single integer property.

Parameters

<code>entry</code>	<i>The EntityEntry for the tracked entity.</i>
--------------------	--

Returns

The integer primary key value of the entity if found and is a single integer key, otherwise null.

- int? [GetCropIdValue](#) (EntityEntry entry)

Gets the 'CropId' value (or equivalent identifier related to a crop) from the entity represented by the specified entry, if available.

Parameters

entry	<i>The EntityEntry for the tracked entity.</i>
-------	--

Returns

The integer 'CropId' value, or null if the property doesn't exist, is not an integer, or its value is null.

- IDictionary< string, object?>? [GetValuesDictionary](#) (PropertyValues? propertyValues)

Creates a dictionary mapping property names to their values from the provided PropertyValues object, excluding a predefined set of sensitive properties (like 'Password').

Parameters

propertyValues	<i>The PropertyValues object (e.g., OriginalValues, CurrentValues) containing the entity's property values. Can be null.</i>
----------------	--

Returns

A dictionary where keys are property names and values are the corresponding property values (as object?), excluding default sensitive properties. Returns null if propertyValues is null or the resulting dictionary is empty after exclusions.

- IDictionary< string, object?>? [GetValuesDictionary](#) (PropertyValues? propertyValues, IEnumerable< string >? additionalExcludedProperties)

Creates a dictionary mapping property names to their values from the provided PropertyValues object, excluding both a predefined set of sensitive properties and any additionally specified properties.

Parameters

propertyValues	<i>The PropertyValues object (e.g., OriginalValues, CurrentValues) containing the entity's property values. Can be null.</i>
additionalExcludedProperties	<i>An optional collection of additional property names to exclude from the resulting dictionary, beyond the default exclusions. Can be null or empty.</i>

Returns

A dictionary where keys are property names and values are the corresponding property values (as object?), excluding default and specified properties. Returns null if propertyValues is null or the resulting dictionary is empty after exclusions.

- string? [SerializePropertyValueDictionary](#) (IDictionary< string, object?>? values)

Serializes the provided dictionary of property values into a string representation, typically JSON format.

Parameters

values	<i>The dictionary containing property names and their values. Can be null.</i>
--------	--

Returns

A string representation of the dictionary (e.g., JSON formatted), or `null` if the input dictionary is `null` or empty.
Returns an error placeholder string on serialization failure.

Static Public Attributes

- const string `UnknownTableName` = "<Unknown>"

6.15.1 Detailed Description

Implements IAuditHelperService providing utility methods commonly used during the generation of audit log entries, such as primary key retrieval, foreign key retrieval (`CropId`), and property value serialization.

"Invalid email or password."

6.15.2 Constructor & Destructor Documentation**6.15.2.1 AuditHelperService()**

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.AuditHelperService ( 
    ILogger< AuditHelperService > logger ) [inline]
```

Initializes a new instance of the AuditHelperService class.

Parameters

<code>logger</code>	The logger for recording information and warnings related to helper operations.
---------------------	---

Exceptions

<code>ArgumentNullException</code>	Thrown if logger is null.
------------------------------------	---------------------------

6.15.3 Member Function Documentation**6.15.3.1 GetCropIdValue()**

```
int? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.GetCropIdValue ( 
    EntityEntry entry ) [inline]
```

Gets the 'CropId' value (or equivalent identifier related to a crop) from the entity represented by the specified entry, if available.

Parameters

<i>entry</i>	The EntityEntry for the tracked entity.
--------------	---

Returns

The integer 'CropId' value, or `null` if the property doesn't exist, is not an integer, or its value is null.

This implementation looks for a property named "CropId" (case-insensitive) on the entity. It prioritizes the `CurrentValue`, falls back to `OriginalValue` for `EntityState.Deleted` or if `CurrentValue` is null. Returns the integer value if found and convertible, otherwise returns `null`. Does not currently attempt to load navigation properties.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#).

6.15.3.2 GetPrimaryKeyValue()

```
int? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.GetPrimaryKeyValue (
    EntityEntry entry ) [inline]
```

Gets the primary key value of the entity represented by the specified entry, assuming the primary key is a single integer property.

Parameters

<i>entry</i>	The EntityEntry for the tracked entity.
--------------	---

Returns

The integer primary key value of the entity if found and is a single integer key, otherwise `null`.

This implementation attempts to find a single primary key property of type `int`. It prioritizes the `CurrentValue` but falls back to `OriginalValue` for `EntityState.Deleted` entities. It logs warnings and returns `null` if no PK is found, if the PK is composite, if the PK property is not an `int`, or if the PK value cannot be determined.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#).

6.15.3.3 GetValuesDictionary() [1/2]

```
IDictionary< string, object?>? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.GetValuesDictionary (
    PropertyValues? propertyValues ) [inline]
```

Creates a dictionary mapping property names to their values from the provided `PropertyValues` object, excluding a predefined set of sensitive properties (like 'Password').

Parameters

<i>propertyValues</i>	The <code>PropertyValues</code> object (e.g., <code>OriginalValues</code> , <code>CurrentValues</code>) containing the entity's property values. Can be <code>null</code> .
-----------------------	--

Returns

A dictionary where keys are property names and values are the corresponding property values (as `object?`), excluding default sensitive properties. Returns `null` if *propertyValues* is `null` or the resulting dictionary is empty after exclusions.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#).

6.15.3.4 GetValuesDictionary() [2/2]

```
IDictionary< string, object?>? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.GetValuesDictionary (
    PropertyValues? propertyValues,
    IEnumerable< string >? additionalExcludedProperties ) [inline]
```

Creates a dictionary mapping property names to their values from the provided `PropertyValues` object, excluding both a predefined set of sensitive properties and any additionally specified properties.

Parameters

<i>propertyValues</i>	The <code>PropertyValues</code> object (e.g., <code>OriginalValues</code> , <code>CurrentValues</code>) containing the entity's property values. Can be <code>null</code> .
<i>additionalExcludedProperties</i>	An optional collection of additional property names to exclude from the resulting dictionary, beyond the default exclusions. Can be <code>null</code> or empty.

//

Returns

A dictionary where keys are property names and values are the corresponding property values (as `object?`), excluding default and specified properties. Returns `null` if *propertyValues* is `null` or the resulting dictionary is empty after exclusions.

Creates a dictionary of property names and their values from the provided *propertyValues*. It excludes properties listed in the static `DefaultExcludedProperties` set (e.g., "Password") and any additional properties specified in the *additionalExcludedProperties* argument. Returns `null` if the input *propertyValues* is `null` or if the resulting dictionary is empty after exclusions.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#).

6.15.3.5 SerializePropertyValueDictionary()

```
string? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Serialize←  
PropertyValueDictionary<  
    IDictionary< string, object?>? values  
    [inline]
```

Serializes the provided dictionary of property values into a string representation, typically JSON format.

Parameters

values	The dictionary containing property names and their values. Can be null.
--------	---

Returns

A string representation of the dictionary (e.g., JSON formatted), or `null` if the input dictionary is `null` or empty.
Returns an error placeholder string on serialization failure.

Serializes the provided dictionary of property values into a JSON string using pre-configured, shared serializer options (`_jsonSerializerOptions`). Handles null or empty input dictionaries by returning `null`. If serialization fails, it logs the error and returns a specific error placeholder string.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#).

6.15.4 Member Data Documentation

6.15.4.1 UnknownTableName

```
const string ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Unknown←  
TableName = "<Unknown>" [static]
```

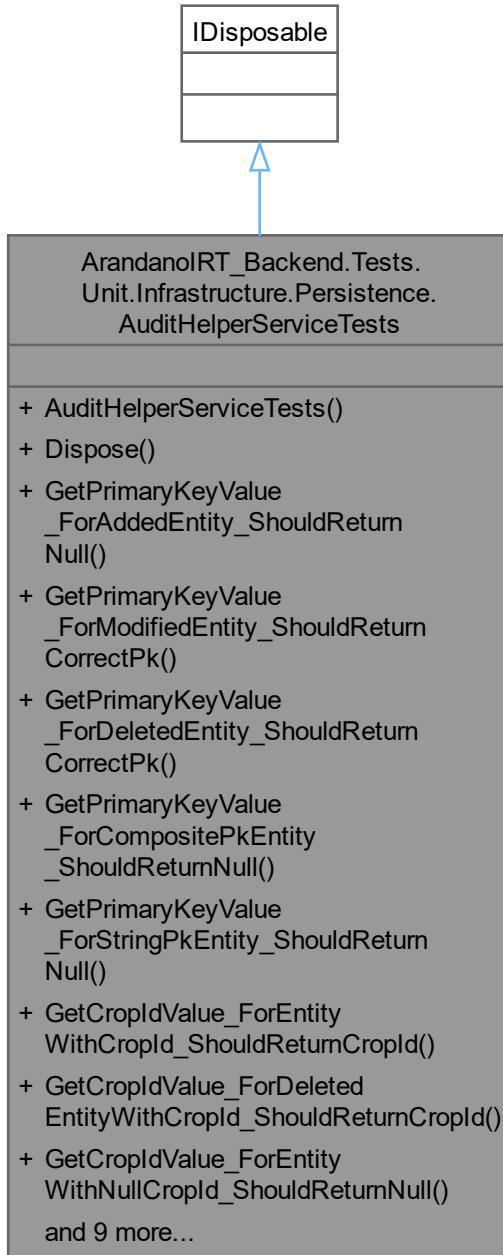
The documentation for this class was generated from the following file:

- Infrastructure/Persistence/Auditing/[AuditHelperService.cs](#)

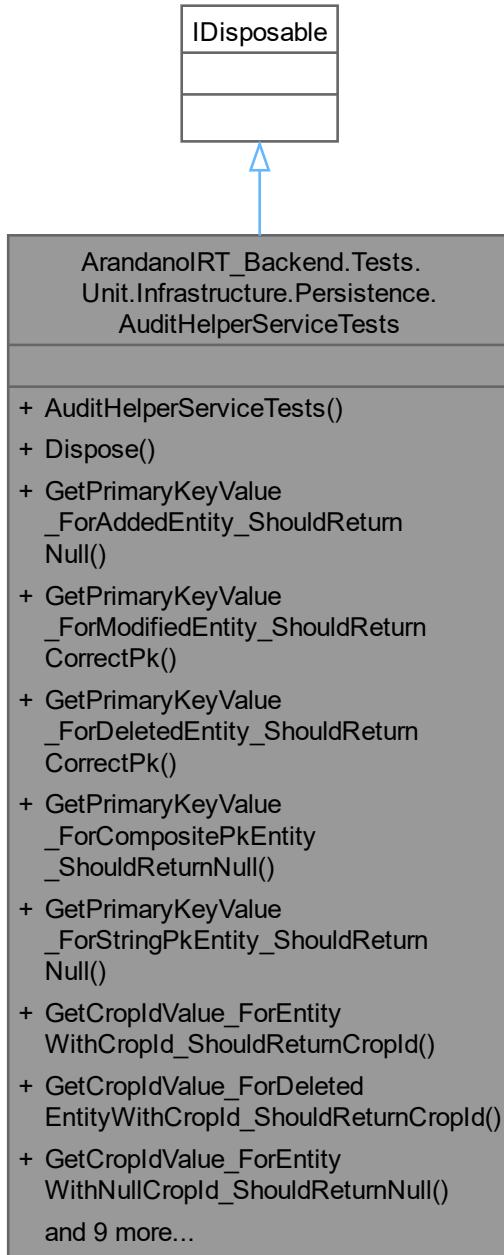
6.16 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests Class Reference

Contains unit tests for the `AuditHelperService` class.

Inheritance diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests:



Public Member Functions

- [AuditHelperServiceTests \(\)](#)

Initializes a new instance of the AuditHelperServiceTests class, setting up the InMemory database context and the service under test.

- void [Dispose \(\)](#)
Disposes the DbContext instance after each test run.
- void [GetPrimaryKeyValue_ForAddedEntity_ShouldReturnNull \(\)](#)
- void [GetPrimaryKeyValue_ForModifiedEntity_ShouldReturnCorrectPk \(\)](#)
- void [GetPrimaryKeyValue_ForDeletedEntity_ShouldReturnCorrectPk \(\)](#)
- void [GetPrimaryKeyValue_ForCompositePkEntity_ShouldReturnNull \(\)](#)
- void [GetPrimaryKeyValue_ForStringPkEntity_ShouldReturnNull \(\)](#)
- void [GetCropIdValue_ForEntityWithCropId_ShouldReturnCropId \(\)](#)
- void [GetCropIdValue_ForDeletedEntityWithCropId_ShouldReturnCropId \(\)](#)
- void [GetCropIdValue_ForEntityWithNullCropId_ShouldReturnNull \(\)](#)
- void [GetCropIdValue_ForEntityWithoutCropIdProperty_ShouldReturnNull \(\)](#)
- void [GetValuesDictionary_WithCurrentValues_ShouldReturnDictionaryWithoutExcluded \(\)](#)
- void [GetValuesDictionary_WithOriginalValues_ShouldReturnDictionaryWithoutExcluded \(\)](#)
- void [GetValuesDictionary_WithAdditionalExclusions_ShouldExcludeMoreProperties \(\)](#)
- void [GetValuesDictionary_WithNullPropertyValues_ShouldReturnNull \(\)](#)
- void [SerializePropertyValueDictionary_WithValidDictionary_ShouldReturnJsonString \(\)](#)
- void [SerializePropertyValueDictionary_WithNullDictionary_ShouldReturnNull \(\)](#)
- void [SerializePropertyValueDictionary_WithEmptyDictionary_ShouldReturnNull \(\)](#)
- void [SerializePropertyValueDictionary_WithError_ShouldHandleGracefully \(\)](#)

6.16.1 Detailed Description

Contains unit tests for the AuditHelperService class.

Uses an InMemory DbContext to simulate Entity Framework Core behavior.

6.16.2 Constructor & Destructor Documentation

6.16.2.1 AuditHelperServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.AuditHelperServiceTests ( ) [inline]
```

Initializes a new instance of the AuditHelperServiceTests class, setting up the InMemory database context and the service under test.

6.16.3 Member Function Documentation

6.16.3.1 Dispose()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.Dispose ( ) [inline]
```

Disposes the DbContext instance after each test run.

6.16.3.2 GetCropIdValue_ForDeletedEntityWithCropId_ShouldReturnCropId()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetCropIdValue_ForDeletedEntityWithCropId_ShouldReturnCropId () [inline]
```

6.16.3.3 GetCropIdValue_ForEntityWithCropId_ShouldReturnCropId()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetCropIdValue_ForEntityWithCropId_ShouldReturnCropId () [inline]
```

6.16.3.4 GetCropIdValue_ForEntityWithNullCropId_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetCropIdValue_ForEntityWithNullCropId_ShouldReturnNull () [inline]
```

6.16.3.5 GetCropIdValue_ForEntityWithoutCropIdProperty_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetCropIdValue_ForEntityWithoutCropIdProperty_ShouldReturnNull () [inline]
```

6.16.3.6 GetPrimaryKeyValue_ForAddedEntity_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetPrimaryKeyValue_ForAddedEntity_ShouldReturnNull () [inline]
```

6.16.3.7 GetPrimaryKeyValue_ForCompositePkEntity_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetPrimaryKeyValue_ForCompositePkEntity_ShouldReturnNull () [inline]
```

6.16.3.8 GetPrimaryKeyValue_ForDeletedEntity_ShouldReturnCorrectPk()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetPrimaryKeyValue_ForDeletedEntity_ShouldReturnCorrectPk () [inline]
```

6.16.3.9 GetPrimaryKeyValue_ForModifiedEntity_ShouldReturnCorrectPk()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetPrimaryKeyValue_ForModifiedEntity_ShouldReturnCorrectPk () [inline]
```

6.16.3.10 GetPrimaryKeyValue_ForStringPkEntity_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetPrimaryKeyValue_ForStringPkEntity_ShouldReturnNull () [inline]
```

6.16.3.11 GetValuesDictionary_WithAdditionalExclusions_ShouldExcludeMoreProperties()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetValuesDictionary_WithAdditionalExclusions_ShouldExcludeMoreProperties () [inline]
```

6.16.3.12 GetValuesDictionary_WithCurrentValue_ShouldReturnDictionaryWithoutExcluded()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetValuesDictionary_WithCurrentValue_ShouldReturnDictionaryWithoutExcluded () [inline]
```

6.16.3.13 GetValuesDictionary_WithNullPropertyValues_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetValuesDictionary_WithNullPropertyValues_ShouldReturnNull () [inline]
```

6.16.3.14 GetValuesDictionary_WithOriginalValues_ShouldReturnDictionaryWithoutExcluded()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.GetValuesDictionary_WithOriginalValues_ShouldReturnDictionaryWithoutExcluded () [inline]
```

6.16.3.15 SerializePropertyValueDictionary_WithEmptyDictionary_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.SerializePropertyValueDictionary_WithEmptyDictionary_ShouldReturnNull () [inline]
```

6.16.3.16 SerializePropertyValueDictionary_WithError_ShouldHandleGracefully()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.SerializePropertyValueDictionary_WithError_ShouldHandleGracefully () [inline]
```

6.16.3.17 SerializePropertyValueDictionary_WithNullDictionary_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.SerializePropertyValueDictionary_WithNullDictionary_ShouldReturnNull () [inline]
```

6.16.3.18 SerializePropertyValueDictionary_WithValidDictionary_ShouldReturnJsonString()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests.Serialize←
PropertyValueDictionary_WithValidDictionary_ShouldReturnJsonString ( ) [inline]
```

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Persistence/AuditHelperServiceTests.cs

6.17 ArandanoIRT_Backend.Infrastructure.Data.Auditperson Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditperson:

ArandanoIRT_Backend.Infrastructure. Data.Auditperson	
+	Idauditperson
+	Columnname
+	Recordid
+	Cropid
+	Action
+	Performedat
+	Performedby
+	Performedbyip
+	Useragent

Properties

- int [Idauditperson](#) [get, set]
- string [Columnname](#) = null! [get, set]
- int [Recordid](#) [get, set]
- int? [Cropid](#) [get, set]
- string [Action](#) = null! [get, set]
- DateTime? [Performedat](#) [get, set]
- int? [Performedby](#) [get, set]
- string? [Performedbyip](#) [get, set]
- string? [Useragent](#) [get, set]

6.17.1 Property Documentation

6.17.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Action = null! [get], [set]
```

6.17.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Columnname = null! [get], [set]
```

6.17.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Cropid [get], [set]
```

6.17.1.4 Idauditperson

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Idauditperson [get], [set]
```

6.17.1.5 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Performedat [get], [set]
```

6.17.1.6 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Performedby [get], [set]
```

6.17.1.7 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Performedbyip [get], [set]
```

6.17.1.8 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Recordid [get], [set]
```

6.17.1.9 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditperson.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditperson.cs](#)

6.18 ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity Class Reference

Represents a single audit trail record tracking changes specifically related to person (user) entities or data, associated with a particular crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity:

ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity	
+	IdAuditPerson
+	ColumnName
+	RecordId
+	CropId
+	Action
+	PerformedAt
+	PerformedBy
+	PerformedByIp
+	UserAgent
+	AuditPersonEntity()

Public Member Functions

- [AuditPersonEntity](#) (int idAuditPerson, string columnName, int recordId, int cropId, string action, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)

Initializes a new instance of the AuditPersonEntity class with detailed audit information for a person-related action.

Properties

- int [IdAuditPerson](#) [get]
Gets the unique identifier for this person audit log entry.
- string [ColumnName](#) [get]
Gets the name of the column or property within the person entity that was changed or involved in the action.
- int [RecordId](#) [get]
Gets the identifier (primary key) of the specific person record that was affected.
- int [CropId](#) [get]
Gets the identifier of the crop to which the audited person belongs or is related.

- string **Action** [get]
Gets the type of action performed (e.g., "LOGIN", "UPDATE_PROFILE", "PASSWORD_RESET").
- DateTime? **PerformedAt** [get]
Gets the date and time (usually UTC) when the action was performed.
- int? **PerformedBy** [get]
Gets the unique identifier of the user who performed the action (can be the user themselves or an admin).
- string? **PerformedByIp** [get]
Gets the IP address from which the action was performed, if available.
- string? **UserAgent** [get]
Gets the user agent string of the client that performed the action, if available.

6.18.1 Detailed Description

Represents a single audit trail record tracking changes specifically related to person (user) entities or data, associated with a particular crop.

Note: This specific audit entity does not track old/new values.

6.18.2 Constructor & Destructor Documentation

6.18.2.1 AuditPersonEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.AuditPersonEntity (
    int idAuditPerson,
    string columnName,
    int recordId,
    int cropId,
    string action,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditPersonEntity class with detailed audit information for a person-related action.

Parameters

<i>idAuditPerson</i>	The unique ID for this audit record (often generated by the database).
<i>columnName</i>	The name of the column/property involved in the action.
<i>recordId</i>	The ID of the affected person record.
<i>cropId</i>	The ID of the crop this person relates to.
<i>action</i>	The action performed (e.g., "LOGIN").
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedByIp</i>	The IP address associated with the action (nullable).
<i>userAgent</i>	The user agent of the client, if applicable (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	---

6.18.3 Property Documentation

6.18.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.Action [get]
```

Gets the type of action performed (e.g., "LOGIN", "UPDATE_PROFILE", "PASSWORD_RESET").

6.18.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.ColumnName [get]
```

Gets the name of the column or property within the person entity that was changed or involved in the action.

6.18.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.CropId [get]
```

Gets the identifier of the crop to which the audited person belongs or is related.

6.18.3.4 IdAuditPerson

```
int ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.IdAuditPerson [get]
```

Gets the unique identifier for this person audit log entry.

6.18.3.5 PerformedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action was performed.

Nullable.

6.18.3.6 PerformedBy

```
int? ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action (can be the user themselves or an admin).

Nullable.

6.18.3.7 PerformedByIp

```
string? ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.PerformedByIp [get]
```

Gets the IP address from which the action was performed, if available.

Nullable.

6.18.3.8 RecordId

```
int ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.RecordId [get]
```

Gets the identifier (primary key) of the specific person record that was affected.

6.18.3.9 UserAgent

```
string? ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if available.

Nullable.

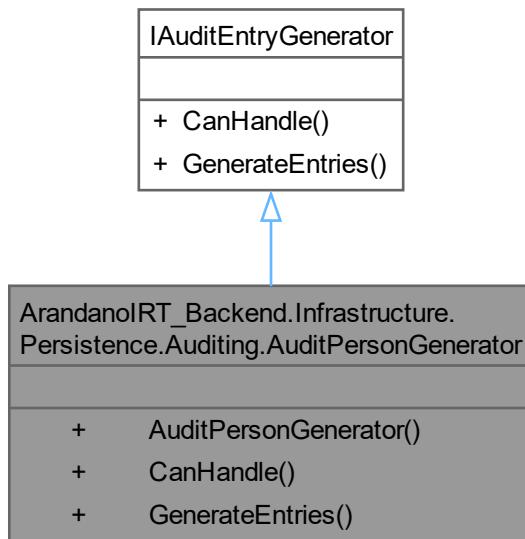
The documentation for this class was generated from the following file:

- Domain/Entities/[AuditPersonEntity.cs](#)

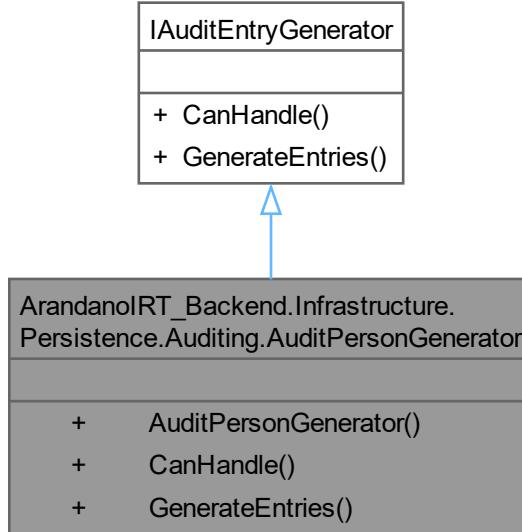
6.19 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator Class Reference

Implements IAuditEntryGenerator to create audit trail entries in the Auditperson table for changes made to Person entities.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator:



Public Member Functions

- **`AuditPersonGenerator`** (`IAuditHelperService auditHelper, ILogger< AuditPersonGenerator > logger`)
Initializes a new instance of the AuditPersonGenerator class.
- **`bool CanHandle`** (`EntityEntry entry`)
Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	<i>The Entity Framework Core EntityEntry representing the tracked change.</i>
--------------------	---

Returns

true if this generator can handle the entry; otherwise, false.

- **`IEnumerable< object > GenerateEntries`** (`EntityEntry entry, AuditMetadata metadata`)
Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	<i>The EntityEntry containing the details of the entity change (e.g., original and current values).</i>
<code>metadata</code>	<i>The AuditMetadata containing context about the action being audited (user, time, etc.).</i>

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

6.19.1 Detailed Description

Implements `IAuditEntryGenerator` to create audit trail entries in the `Auditperson` table for changes made to Person entities.

6.19.2 Constructor & Destructor Documentation

6.19.2.1 AuditPersonGenerator()

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator.AuditPersonGenerator
(
    IAuditHelperService auditHelper,
    ILogger< AuditPersonGenerator > logger ) [inline]
```

Initializes a new instance of the `AuditPersonGenerator` class.

Parameters

<code>auditHelper</code>	The helper service for common auditing tasks.
<code>logger</code>	The logger for recording information and warnings.

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>auditHelper</code> or <code>logger</code> is null.
------------------------------------	--

6.19.3 Member Function Documentation

6.19.3.1 CanHandle()

```
bool ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator.CanHandle (
    EntityEntry entry ) [inline]
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core <code>EntityEntry</code> representing the tracked change.
--------------------	---

Returns

`true` if this generator can handle the entry; otherwise, `false`.

This generator specifically handles entities of type `Person`.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

6.19.3.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator.←
GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata ) [inline]
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	The <code>EntityEntry</code> containing the details of the entity change (e.g., original and current values).
<code>metadata</code>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Generates `Auditperson` records based on the entity state:

- **Added:** Generates one entry with "ALL" as `ColumnName`.
- **Deleted:** Generates one entry with "ALL" as `ColumnName`.
- **Modified:** Generates one entry for *each* modified property (excluding "Password"). The `ColumnName` is set to the modified property's name. Old/New values are *not* recorded in this implementation. Uses [IAuditHelper](#) Service for retrieving IDs.

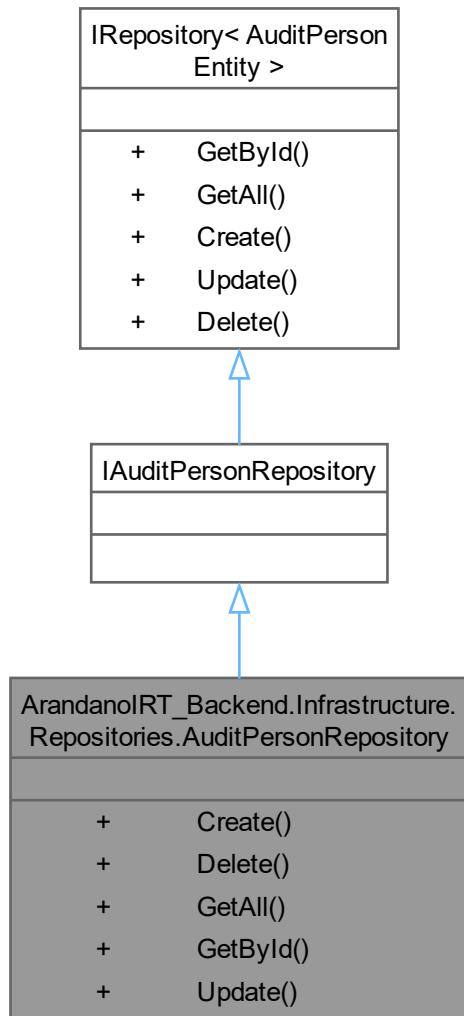
Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

The documentation for this class was generated from the following file:

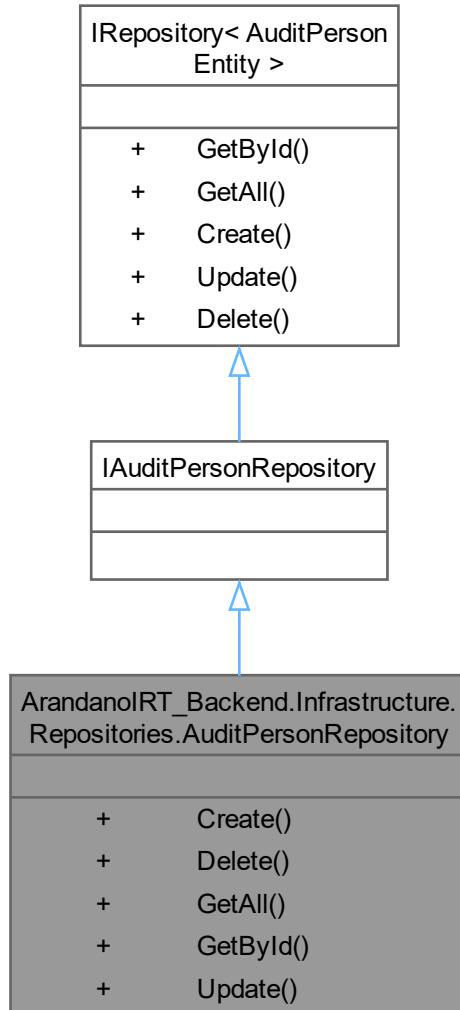
- Infrastructure/Persistence/Auditing/[AuditPersonGenerator.cs](#)

6.20 ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository:



Public Member Functions

- `Task< Result< AuditPersonEntity > > Create (AuditPersonEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< AuditPersonEntity > > > GetAll ()`
- `Task< Result< AuditPersonEntity > > GetById (int id)`
- `Task< Result< bool > > Update (AuditPersonEntity entity)`

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditPersonEntity >](#)

- Task< Result< T > > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.20.1 Member Function Documentation

6.20.1.1 Create()

```
Task< Result< AuditPersonEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository.Create (
    AuditPersonEntity entity ) [inline]
```

6.20.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository.Delete (
    int id ) [inline]
```

6.20.1.3 GetAll()

```
Task< Result< IEnumerable< AuditPersonEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository.GetAll ( ) [inline]
```

6.20.1.4 GetById()

```
Task< Result< AuditPersonEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository.GetById (
    int id ) [inline]
```

6.20.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository.←  
Update (←  
    AuditPersonEntity entity ) [inline]
```

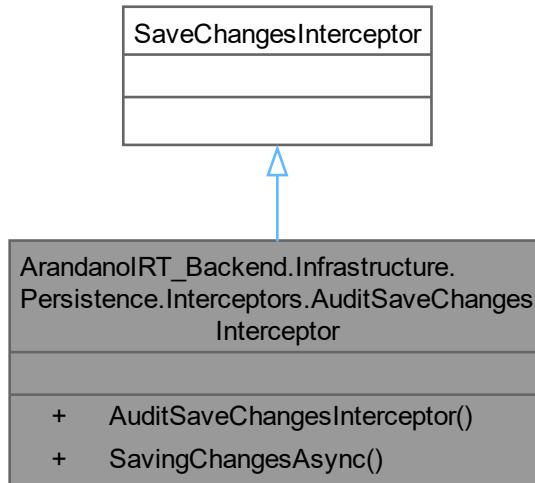
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/AuditPersonRepository.cs

6.21 ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor Class Reference

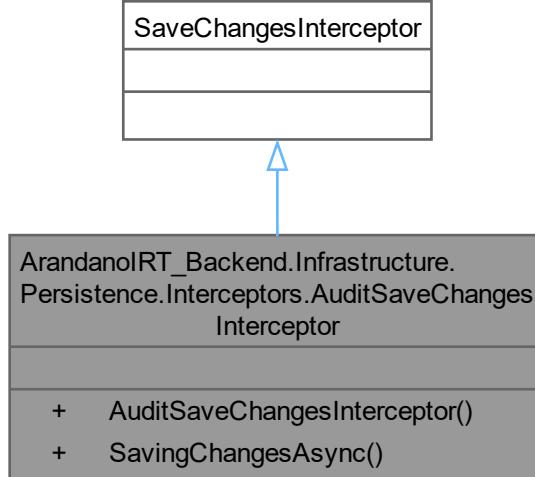
An SaveChangesInterceptor that automatically generates audit log entries for entity changes detected by the DbContext's ChangeTracker before saving.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChanges→

Interceptor:



Public Member Functions

- **AuditSaveChangesInterceptor** ([IRequestContextAccessor](#) requestContextAccessor, [IDateTimeProvider](#) date←
TimeProvider, [IEnumerable<IAuditEntryGenerator>](#) auditGenerators, [ILogger<AuditSaveChangesInterceptor>](#) logger)

Initializes a new instance of the AuditSaveChangesInterceptor class.

- override async ValueTask<[InterceptionResult<int>](#)> **SavingChangesAsync** ([DbContextEventData](#) eventData,
[InterceptionResult<int>](#) result, CancellationToken cancellationToken=default)

Intercepts the saving changes operation asynchronously before it executes.

6.21.1 Detailed Description

An `SaveChangesInterceptor` that automatically generates audit log entries for entity changes detected by the `DbContext`'s `ChangeTracker` before saving.

It uses registered `IAuditEntryGenerator` instances to create specific audit records.

6.21.2 Constructor & Destructor Documentation

6.21.2.1 AuditSaveChangesInterceptor()

`ArandanolIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor.AuditSaveChangesInterceptor (`

```
IRequestContextAccessor requestContextAccessor,
IDateTimeProvider dateTimeProvider,
IEnumerable< IAuditEntryGenerator > auditGenerators,
ILogger< AuditSaveChangesInterceptor > logger ) [inline]
```

Initializes a new instance of the AuditSaveChangesInterceptor class.

Parameters

<i>requestContextAccessor</i>	Accessor for retrieving current request context (user ID, IP, etc.).
<i>dateTimeProvider</i>	Provider for obtaining consistent UTC timestamps.
<i>auditGenerators</i>	An enumerable collection of registered audit entry generators.
<i>logger</i>	Logger for recording interceptor activity.

Exceptions

<i>ArgumentNullException</i>	Thrown if any injected service is null.
------------------------------	---

6.21.3 Member Function Documentation

6.21.3.1 SavingChangesAsync()

```
override async ValueTask< InterceptionResult< int > > ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor.SavingChangesAsync (
    DbContextEventData eventData,
    InterceptionResult< int > result,
    CancellationToken cancellationToken = default ) [inline]
```

Intercepts the saving changes operation asynchronously before it executes.

Generates audit entries for detected changes and adds them to the DbContext to be saved within the same transaction.

Parameters

<i>eventData</i>	Contextual information about the SaveChanges operation.
<i>result</i>	The current interception result.
<i>cancellationToken</i>	A token to cancel the operation.

Returns

A ValueTask<TResult> representing the asynchronous operation, containing the possibly modified interception result.

Refactored workflow:

1. Get DbContext.

2. Get common audit metadata via helper.
3. Generate all audit entries for tracked changes via helper.
4. Add generated entries to the DbContext if any exist.
5. Continue with the original SaveChanges operation.

The documentation for this class was generated from the following file:

- Infrastructure/Persistence/Interceptors/[AuditSaveChangesInterceptor.cs](#)

6.22 ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata:

ArandanoIRT_Backend.Infrastructure. Data.Auditsensitizeddata	
+	Idauditsensitizeddata
+	Tablename
+	Columnname
+	Recordid
+	Cropid
+	Action
+	Performedat
+	Performedby
+	Performedbyip
+	Useragent

Properties

- int [Idauditsensitizeddata](#) [get, set]
- string [Tablename](#) = null! [get, set]
- string [Columnname](#) = null! [get, set]
- int [Recordid](#) [get, set]

- int? **Cropid** [get, set]
- string **Action** = null! [get, set]
- DateTime? **Performedat** [get, set]
- int? **Performedby** [get, set]
- string? **Performedbyip** [get, set]
- string? **Useragent** [get, set]

6.22.1 Property Documentation

6.22.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Action = null! [get], [set]
```

6.22.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Columnname = null! [get], [set]
```

6.22.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Cropid [get], [set]
```

6.22.1.4 Idauditsensitizeddata

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Idauditsensitizeddata [get], [set]
```

6.22.1.5 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Performedat [get], [set]
```

6.22.1.6 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Performedby [get], [set]
```

6.22.1.7 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Performedbyip [get], [set]
```

6.22.1.8 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Recordid [get], [set]
```

6.22.1.9 Tablename

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Tablename = null! [get], [set]
```

6.22.1.10 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditsensitizeddata.cs](#)

6.23 ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity Class Reference

Represents a single audit trail record specifically tracking actions involving access to, or operations on, sensitive data fields within specified database tables, associated with a crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity:

ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity
+ IdAuditSensitiveData + TableName + ColumnName + RecordId + CropId + Action + PerformedAt + PerformedBy + PerformedByIp + UserAgent + AuditSensitiveDataEntity()

Public Member Functions

- [AuditSensitiveDataEntity](#) (int idAuditSensitiveData, string tableName, string columnName, int recordId, int cropId, string action, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)
Initializes a new instance of the AuditSensitiveDataEntity class with details about an action on sensitive data.

Properties

- int [IdAuditSensitiveData](#) [get]
Gets the unique identifier for this sensitive data audit log entry.
- string [TableName](#) [get]
Gets the name of the database table containing the sensitive data that was accessed or modified.
- string [ColumnName](#) [get]
Gets the name of the specific column containing the sensitive data involved in the action.
- int [RecordId](#) [get]
Gets the identifier (primary key) of the specific record within the table whose sensitive data was involved.
- int [CropId](#) [get]
Gets the identifier of the crop context relevant to the audited sensitive data action.
- string [Action](#) [get]
Gets the type of action performed on the sensitive data (e.g., "READ_ACCESS", "DECRYPTION_ATTEMPT", "EXPORTED").
- DateTime? [PerformedAt](#) [get]
Gets the date and time (usually UTC) when the action involving sensitive data was performed.
- int? [PerformedBy](#) [get]
Gets the unique identifier of the user who performed the action, if available.
- string? [PerformedByIp](#) [get]
Gets the IP address from which the action originated, if available.
- string? [UserAgent](#) [get]
Gets the user agent string of the client that performed the action, if available.

6.23.1 Detailed Description

Represents a single audit trail record specifically tracking actions involving access to, or operations on, sensitive data fields within specified database tables, associated with a crop.

Note: This entity primarily logs the action performed (e.g., access, decryption) rather than the data values themselves.

6.23.2 Constructor & Destructor Documentation

6.23.2.1 AuditSensitiveDataEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.AuditSensitiveDataEntity (
    int idAuditSensitiveData,
    string tableName,
    string columnName,
    int recordId,
    int cropId,
    string action,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditSensitiveDataEntity class with details about an action on sensitive data.

Parameters

<i>idAuditSensitiveData</i>	The unique ID for this audit record (often generated by the database).
<i>tableName</i>	The name of the table containing the sensitive data.
<i>columnName</i>	The name of the sensitive column involved.
<i>recordId</i>	The ID of the affected record within the table.
<i>cropId</i>	The ID of the crop context for this action.
<i>action</i>	The action performed (e.g., "READ_ACCESS").
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedByIp</i>	The IP address associated with the action (nullable).
<i>userAgent</i>	The user agent of the client, if applicable (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>tableName</i> is null or whitespace (Spanish message: "El nombre de la tabla es requerido."). Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	--

6.23.3 Property Documentation

6.23.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.Action [get]
```

Gets the type of action performed on the sensitive data (e.g., "READ_ACCESS", "DECRYPTION_ATTEMPT", "EXPORTED").

6.23.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.ColumnName [get]
```

Gets the name of the specific column containing the sensitive data involved in the action.

6.23.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.CropId [get]
```

Gets the identifier of the crop context relevant to the audited sensitive data action.

6.23.3.4 IdAuditSensitiveData

```
int ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.IdAuditSensitiveData [get]
```

Gets the unique identifier for this sensitive data audit log entry.

6.23.3.5 PerformedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action involving sensitive data was performed.

Nullable.

6.23.3.6 PerformedBy

```
int? ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action, if available.

Nullable.

6.23.3.7 PerformedByIp

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.PerformedByIp [get]
```

Gets the IP address from which the action originated, if available.

Nullable.

6.23.3.8 RecordId

```
int ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.RecordId [get]
```

Gets the identifier (primary key) of the specific record within the table whose sensitive data was involved.

6.23.3.9 TableName

```
string ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.TableName [get]
```

Gets the name of the database table containing the sensitive data that was accessed or modified.

6.23.3.10 UserAgent

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if available.

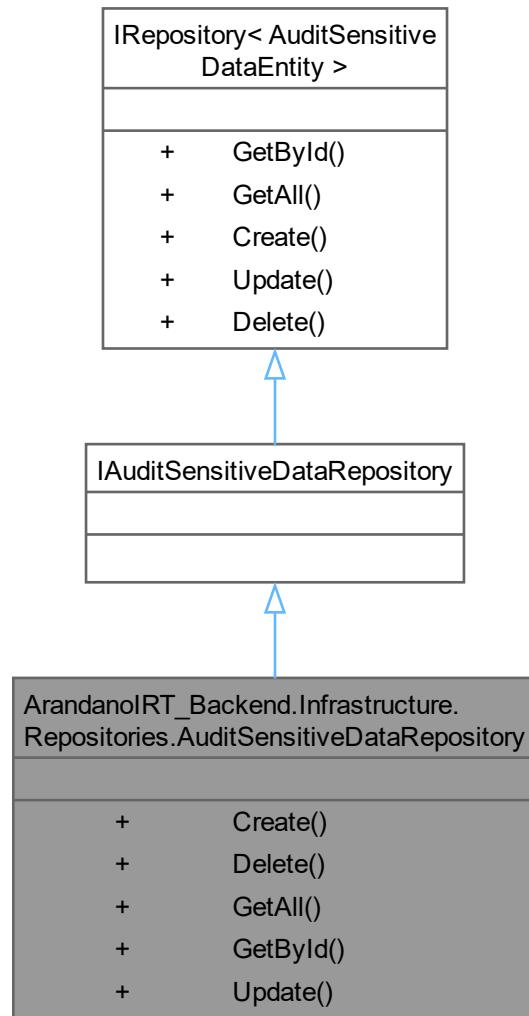
Nullable.

The documentation for this class was generated from the following file:

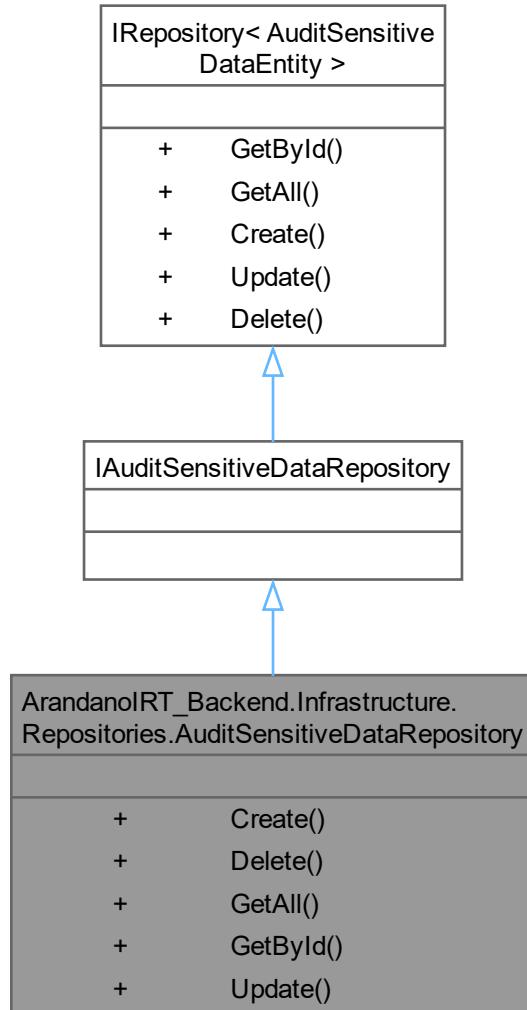
- Domain/Entities/[AuditSensitiveDataEntity.cs](#)

6.24 ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository:



Public Member Functions

- Task<`Result< AuditSensitiveDataEntity >`>> `Create (AuditSensitiveDataEntity entity)`
- Task<`Result< bool >`>> `Delete (int id)`
- Task<`Result< IEnumerable< AuditSensitiveDataEntity >`>>> `GetAll ()`
- Task<`Result< AuditSensitiveDataEntity >`>> `GetById (int id)`
- Task<`Result< bool >`>> `Update (AuditSensitiveDataEntity entity)`

Public Member Functions inherited from
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditSensitiveDataEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.24.1 Member Function Documentation

6.24.1.1 Create()

```
Task< Result< AuditSensitiveDataEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
AuditSensitiveDataRepository.Create (
    AuditSensitiveDataEntity entity) [inline]
```

6.24.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveData←
Repository.Delete (
    int id) [inline]
```

6.24.1.3 GetAll()

```
Task< Result< IEnumerable< AuditSensitiveDataEntity > > > ArandanoIRT_Backend.Infrastructure.←
Repositories.AuditSensitiveDataRepository.GetAll () [inline]
```

6.24.1.4 GetById()

```
Task< Result< AuditSensitiveDataEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
AuditSensitiveDataRepository.GetById (
    int id) [inline]
```

6.24.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveData←
Repository.Update (
    AuditSensitiveDataEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/AuditSensitiveDataRepository.cs

6.25 ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable:

ArandanoIRT_Backend.Infrastructure. Data.Auditsystemtable	
+	Idauditsystemtable
+	Tablename
+	Columnname
+	Recordid
+	Cropid
+	Action
+	Oldvalue
+	Newvalue
+	Performedat
+	Performedby
+	Performedbyip
+	Useragent

Properties

- int **Idauditsystemtable** [get, set]
- string **Tablename** = null! [get, set]
- string **Columnname** = null! [get, set]

- int `Recordid` [get, set]
- int? `Cropid` [get, set]
- string `Action` = null! [get, set]
- string? `Oldvalue` [get, set]
- string? `Newvalue` [get, set]
- DateTime? `Performedat` [get, set]
- int? `Performedby` [get, set]
- string? `Performedbyip` [get, set]
- string? `Useragent` [get, set]

6.25.1 Property Documentation

6.25.1.1 Action

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Action = null! [get], [set]
```

6.25.1.2 Columnname

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Columnname = null! [get], [set]
```

6.25.1.3 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Cropid [get], [set]
```

6.25.1.4 Idauditsystemtable

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Idauditsystemtable [get], [set]
```

6.25.1.5 Newvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Newvalue [get], [set]
```

6.25.1.6 Oldvalue

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Oldvalue [get], [set]
```

6.25.1.7 Performedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Performedat [get], [set]
```

6.25.1.8 Performedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Performedby [get], [set]
```

6.25.1.9 Performedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Performedbyip [get], [set]
```

6.25.1.10 Recordid

```
int ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Recordid [get], [set]
```

6.25.1.11 Tablename

```
string ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Tablename = null! [get], [set]
```

6.25.1.12 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Auditsystemtable.cs](#)

6.26 ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity Class Reference

Represents a generic audit trail record tracking changes made to specific columns within identified system-level or configuration database tables, potentially associated with a crop context.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity:

ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity
+ IdAuditSystemTable
+ TableName
+ ColumnName
+ RecordId
+ CropId
+ Action
+ OldValue
+ NewValue
+ PerformedAt
+ PerformedBy
+ PerformedByIp
+ UserAgent
+ AuditSystemTableEntity()

Public Member Functions

- [AuditSystemTableEntity](#) (int idAuditSystemTable, string tableName, string columnName, int recordId, int cropId, string action, string? oldValue, string? newValue, DateTime? performedAt, int? performedBy, string? performedByIp, string? userAgent)

Initializes a new instance of the AuditSystemTableEntity class with detailed audit information for a system table change.

Properties

- int [IdAuditSystemTable](#) [get]
Gets the unique identifier for this system table audit log entry.
- string [TableName](#) [get]
Gets the name of the system or configuration database table where the change occurred.
- string [ColumnName](#) [get]
Gets the name of the column or property within the system table that was changed.
- int [RecordId](#) [get]
Gets the identifier (primary key) of the specific record within the system table that was affected.
- int [CropId](#) [get]
Gets the identifier of the crop context relevant to this system table change, if applicable.

- string `Action` [get]
Gets the type of action performed (e.g., "UPDATE_CONFIG", "ENABLE_FEATURE", "DISABLE_FEATURE").
- string? `OldValue` [get]
Gets the value of the column before the change, if applicable (e.g., for UPDATE actions).
- string? `NewValue` [get]
Gets the value of the column after the change, if applicable (e.g., for creation or UPDATE actions).
- DateTime? `PerformedAt` [get]
Gets the date and time (usually UTC) when the action was performed.
- int? `PerformedBy` [get]
Gets the unique identifier of the user who performed the action, if available (e.g., an administrator).
- string? `PerformedByIp` [get]
Gets the IP address from which the action was performed, if available.
- string? `UserAgent` [get]
Gets the user agent string of the client that performed the action, if available.

6.26.1 Detailed Description

Represents a generic audit trail record tracking changes made to specific columns within identified system-level or configuration database tables, potentially associated with a crop context.

Structurally similar to AuditDataTableEntity, but potentially used for auditing system configuration tables.

6.26.2 Constructor & Destructor Documentation

6.26.2.1 AuditSystemTableEntity()

```
ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.AuditSystemTableEntity (
    int idAuditSystemTable,
    string tableName,
    string columnName,
    int recordId,
    int cropId,
    string action,
    string? oldValue,
    string? newValue,
    DateTime? performedAt,
    int? performedBy,
    string? performedByIp,
    string? userAgent ) [inline]
```

Initializes a new instance of the AuditSystemTableEntity class with detailed audit information for a system table change.

Parameters

<code>idAuditSystemTable</code>	The unique ID for this audit record (often generated by the database).
<code>tableName</code>	The name of the system table where the change occurred.
<code>columnName</code>	The name of the column changed.

Parameters

<i>recordId</i>	The ID of the affected record within the table.
<i>cropId</i>	The ID of the crop context for this action.
<i>action</i>	The action performed (e.g., "UPDATE_CONFIG").
<i>oldValue</i>	The previous value (nullable).
<i>newValue</i>	The new value (nullable).
<i>performedAt</i>	The timestamp of the action (nullable).
<i>performedBy</i>	The ID of the user performing the action (nullable).
<i>performedByIp</i>	The IP address associated with the action (nullable).
<i>userAgent</i>	The user agent of the client, if applicable (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>tableName</i> is null or whitespace (Spanish message: "El nombre de la tabla es requerido."). Thrown if <i>columnName</i> is null or whitespace (Spanish message: "El nombre de la columna es requerido."). Thrown if <i>action</i> is null or whitespace (Spanish message: "La acción es requerida.").
--------------------------	--

6.26.3 Property Documentation

6.26.3.1 Action

```
string ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.Action [get]
```

Gets the type of action performed (e.g., "UPDATE_CONFIG", "ENABLE_FEATURE", "DISABLE_FEATURE").

6.26.3.2 ColumnName

```
string ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.ColumnName [get]
```

Gets the name of the column or property within the system table that was changed.

6.26.3.3 CropId

```
int ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.CropId [get]
```

Gets the identifier of the crop context relevant to this system table change, if applicable.

6.26.3.4 IdAuditSystemTable

```
int ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.IdAuditSystemTable [get]
```

Gets the unique identifier for this system table audit log entry.

6.26.3.5 NewValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.NewValue [get]
```

Gets the value of the column after the change, if applicable (e.g., for creation or UPDATE actions).

Null otherwise.

6.26.3.6 OldValue

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.OldValue [get]
```

Gets the value of the column before the change, if applicable (e.g., for UPDATE actions).

Null otherwise.

6.26.3.7 PerformedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.PerformedAt [get]
```

Gets the date and time (usually UTC) when the action was performed.

Nullable.

6.26.3.8 PerformedBy

```
int? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.PerformedBy [get]
```

Gets the unique identifier of the user who performed the action, if available (e.g., an administrator).

Nullable.

6.26.3.9 PerformedByIp

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.PerformedByIp [get]
```

Gets the IP address from which the action was performed, if available.

Nullable.

6.26.3.10 RecordId

```
int ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.RecordId [get]
```

Gets the identifier (primary key) of the specific record within the system table that was affected.

6.26.3.11 TableName

```
string ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.TableName [get]
```

Gets the name of the system or configuration database table where the change occurred.

6.26.3.12 UserAgent

```
string? ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity.UserAgent [get]
```

Gets the user agent string of the client that performed the action, if available.

Nullable.

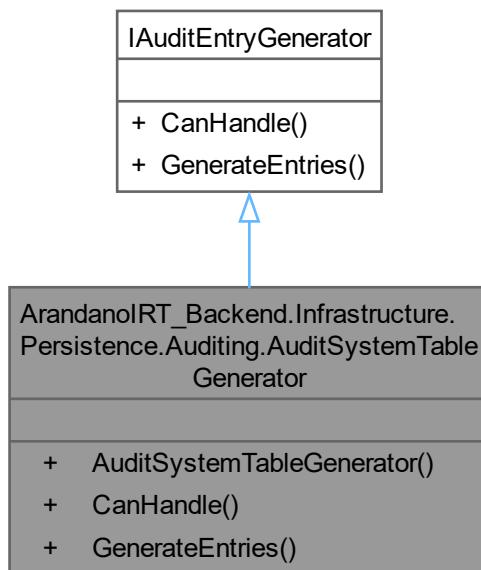
The documentation for this class was generated from the following file:

- Domain/Entities/[AuditSystemTableEntity.cs](#)

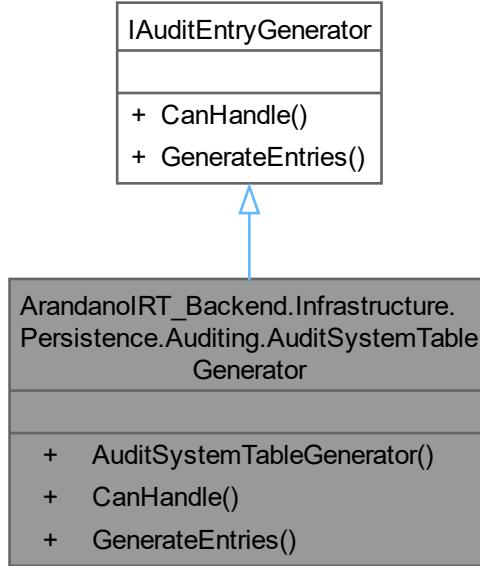
6.27 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator Class Reference

Implements IAuditEntryGenerator to create audit trail entries in the AuditSystemTable for changes made to specific "system" or related data entities like Plantstatehistory and Plantobservation.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator:



Public Member Functions

- **AuditSystemTableGenerator** ([IAuditHelperService auditHelper](#), [ILogger< AuditSystemTableGenerator > logger](#))
Initializes a new instance of the AuditSystemTableGenerator class.
- bool **CanHandle** ([EntityEntry entry](#))
Determines whether this generator can process the specified entity entry for auditing.

Parameters

entry	<i>The Entity Framework Core EntityEntry representing the tracked change.</i>
-----------------------	---

Returns

true if this generator can handle the entry; otherwise, false.

- [IEnumerable< object > GenerateEntries](#) ([EntityEntry entry](#), [AuditMetadata metadata](#))
Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

entry	<i>The EntityEntry containing the details of the entity change (e.g., original and current values).</i>
metadata	<i>The AuditMetadata containing context about the action being audited (user, time, etc.).</i>

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

6.27.1 Detailed Description

Implements `IAuditEntryGenerator` to create audit trail entries in the `AuditSystemTable` for changes made to specific "system" or related data entities like `Plantstatehistory` and `Plantobservation`.

6.27.2 Constructor & Destructor Documentation

6.27.2.1 AuditSystemTableGenerator()

```
ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator.AuditSystemTableGenerator ( [in]  
    IAuditHelperService auditHelper,  
    ILogger< AuditSystemTableGenerator > logger )
```

Initializes a new instance of the `AuditSystemTableGenerator` class.

Parameters

<code>auditHelper</code>	The helper service for common auditing tasks.
<code>logger</code>	The logger for recording information and warnings.

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>auditHelper</code> or <code>logger</code> is null.
------------------------------------	--

6.27.3 Member Function Documentation

6.27.3.1 CanHandle()

```
bool ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator.CanHandle ( EntityEntry entry ) [inline]
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core <code>EntityEntry</code> representing the tracked change.
--------------------	---

Returns

`true` if this generator can handle the entry; otherwise, `false`.

This generator handles only the specified entity types: `Plantstatehistory` and `Plantobservation`. It explicitly does NOT handle `Status`, `TableRelation`, `FailedLoginAttempt`, `DeviceLog`, etc.

Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

6.27.3.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTable←
Generator.GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata ) [inline]
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	The <code>EntityEntry</code> containing the details of the entity change (e.g., original and current values).
<code>metadata</code>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Generates a single `AuditSystemTable` record for `INSERT`, `UPDATE`, or `DELETE` operations on supported entity types (`Plantstatehistory`, `Plantobservation`). Performs row-level auditing ("ALL" column) using the `IAuditHelperService`. `CropId` is explicitly set to `null` for entries generated by this instance.

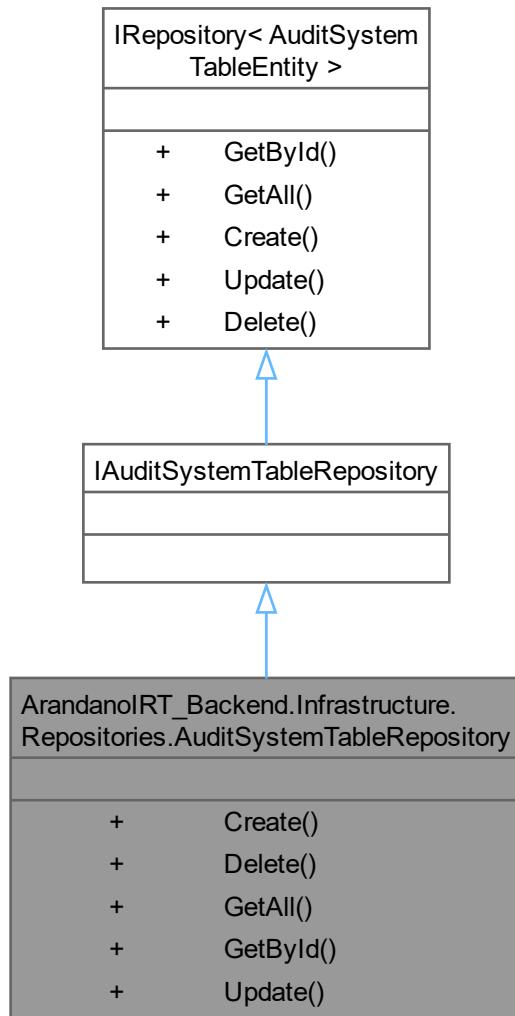
Implements [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#).

The documentation for this class was generated from the following file:

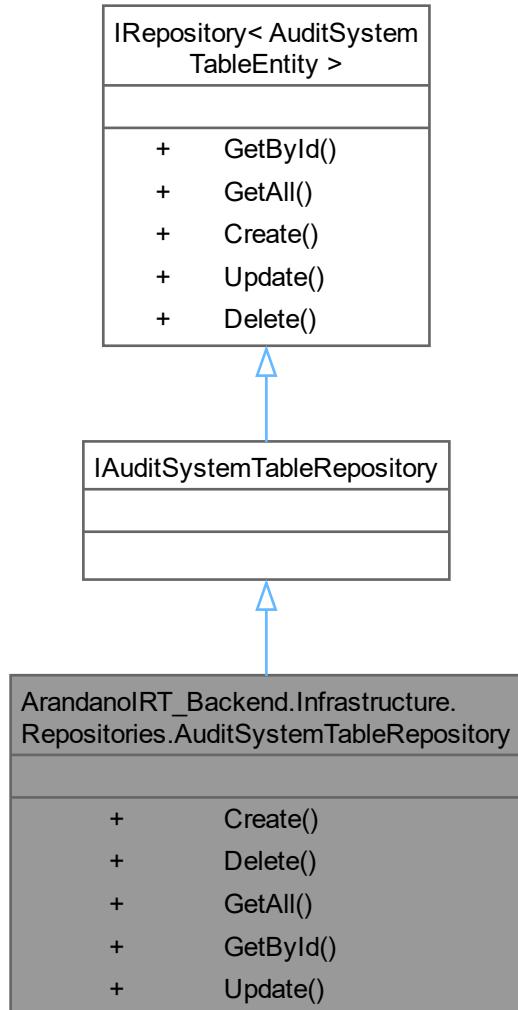
- Infrastructure/Persistence/Auditing/[AuditSystemTableGenerator.cs](#)

6.28 ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository:



Public Member Functions

- Task< [Result< AuditSystemTableEntity >](#) > > [Create \(AuditSystemTableEntity entity\)](#)
- Task< [Result< bool >](#) > > [Delete \(int id\)](#)
- Task< [Result< IEnumerable< AuditSystemTableEntity >](#) > > > [GetAll \(\)](#)
- Task< [Result< AuditSystemTableEntity >](#) > > [GetById \(int id\)](#)
- Task< [Result< bool >](#) > > [Update \(AuditSystemTableEntity entity\)](#)

Public Member Functions inherited from
ArandanoIRT_Backend.Domain.IRepositories IRepository< AuditSystemTableEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.28.1 Member Function Documentation

6.28.1.1 Create()

```
Task< Result< AuditSystemTableEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository.Create (
    AuditSystemTableEntity entity ) [inline]
```

6.28.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository.Delete (
    int id ) [inline]
```

6.28.1.3 GetAll()

```
Task< Result< IEnumerable< AuditSystemTableEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository.GetAll ( ) [inline]
```

6.28.1.4 GetById()

```
Task< Result< AuditSystemTableEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository.GetById (
    int id ) [inline]
```

6.28.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository.←  
Update (   
    AuditSystemTableEntity entity ) [inline]
```

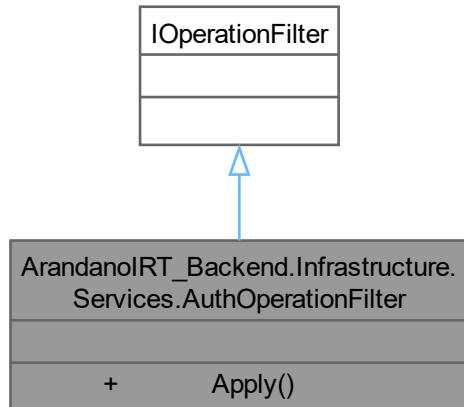
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/AuditSystemTableRepository.cs

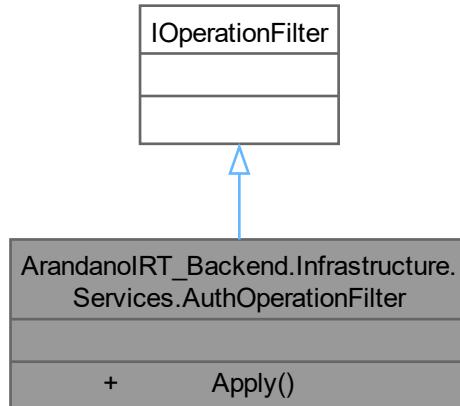
6.29 ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter Class Reference

An Swagger operation filter that adds a security requirement definition (referencing "cookieAuth") to operations protected by the AuthorizeAttribute.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter:



Public Member Functions

- void [Apply](#) (OpenApiOperation operation, OperationFilterContext context)
Applies the filter to the specified OpenAPI operation.

6.29.1 Detailed Description

An Swagger operation filter that adds a security requirement definition (referencing "cookieAuth") to operations protected by the AuthorizeAttribute.

6.29.2 Member Function Documentation

6.29.2.1 Apply()

```
void ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter.Apply (
    OpenApiOperation operation,
    OperationFilterContext context ) [inline]
```

Applies the filter to the specified OpenAPI operation.

Parameters

<i>operation</i>	The OpenAPI operation to modify.
<i>context</i>	The context providing information about the operation and method attributes.

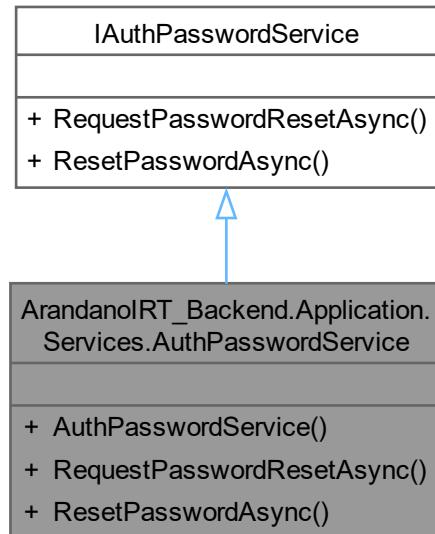
The documentation for this class was generated from the following file:

- Infrastructure/Services/[FiltersSwaggerService.cs](#)

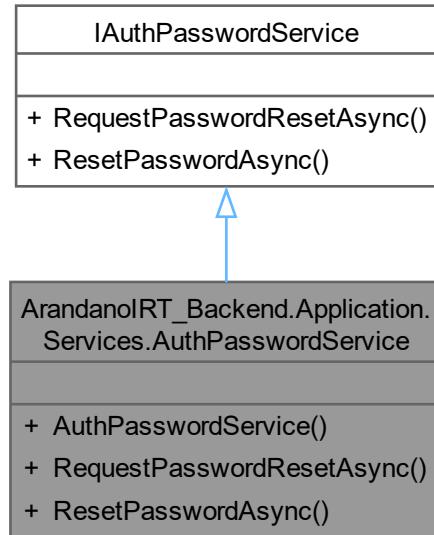
6.30 ArandanoIRT_Backend.Application.Services.AuthPasswordService Class Reference

Implements services for handling password reset requests and password changes using tokens.

Inheritance diagram for ArandanoIRT_Backend.Application.Services.AuthPasswordService:



Collaboration diagram for ArandanoIRT_Backend.Application.Services.AuthPasswordService:



Public Member Functions

- `AuthPasswordService (IPersonRepository personRepository, IChangePasswordRepository changePasswordRepository, IAuthUtilities authUtilities, IEmailService emailService, IDateTimeProvider dateServiceProvider, ILogger< AuthPasswordService > logger)`
- `async Task< Result > RequestPasswordResetAsync (ForgotPasswordRequestDto request)`

Initiates the password reset process for a given email address.

Generates a reset token, stores it, and typically sends it via email. Protects against email enumeration attacks.

Parameters

request	<i>DTO containing the user's email.</i>
---------	---

Returns

A Task representing the asynchronous operation. Always returns a success result to prevent email enumeration, even if the email doesn't exist or validation fails internally.

- `async Task< Result > ResetPasswordAsync (ChangePasswordRequestDto request)`

Resets the user's password using a valid reset token and a new password.

Parameters

request	<i>DTO containing the reset token, new password, and confirmation.</i>
---------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure of the password reset.

6.30.1 Detailed Description

Implements services for handling password reset requests and password changes using tokens.

6.30.2 Constructor & Destructor Documentation

6.30.2.1 AuthPasswordService()

```
ArandanoIRT_Backend.Application.Services.AuthPasswordService.AuthPasswordService (
    IPersonRepository personRepository,
    IChangePasswordRepository changePasswordRepository,
    IAuthUtilities authUtilities,
    IEmailService emailService,
    IDateTimeProvider dateTimeProvider,
    ILogger< AuthPasswordService > logger ) [inline]
```

6.30.3 Member Function Documentation

6.30.3.1 RequestPasswordResetAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthPasswordService.RequestPasswordResetAsync (
    ForgotPasswordRequestDto request ) [inline]
```

Initiates the password reset process for a given email address.

Generates a reset token, stores it, and typically sends it via email. Protects against email enumeration attacks.

Parameters

<code>request</code>	DTO containing the user's email.
----------------------	----------------------------------

Returns

A Task representing the asynchronous operation. Always returns a success result to prevent email enumeration, even if the email doesn't exist or validation fails internally.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService](#).

6.30.3.2 ResetPasswordAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthPasswordService.ResetPasswordAsync ( ChangePasswordRequestDto request ) [inline]
```

Resets the user's password using a valid reset token and a new password.

Parameters

<i>request</i>	DTO containing the reset token, new password, and confirmation.
----------------	---

Returns

A Task representing the asynchronous operation, with a result indicating success or failure of the password reset.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService](#).

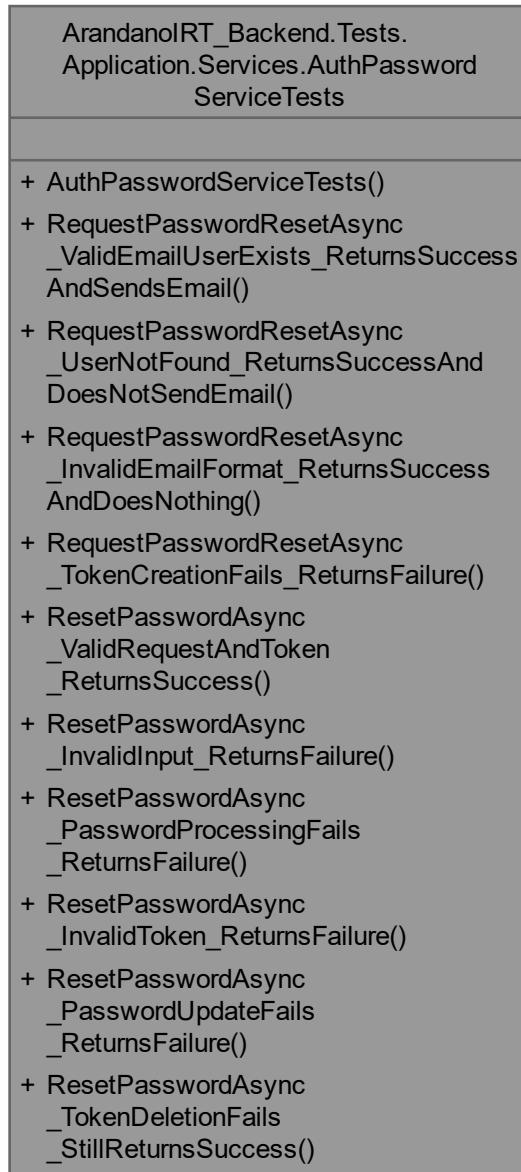
The documentation for this class was generated from the following file:

- Application/Services/[AuthPasswordService.cs](#)

6.31 ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests Class Reference

Contains unit tests for the AuthPasswordService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests:



Public Member Functions

- [AuthPasswordServiceTests \(\)](#)
Initializes a new instance of the AuthPasswordServiceTests class, setting up all mock dependencies and the service under test.
- [async Task RequestPasswordResetAsync _ValidEmailUserExists_ReturnsSuccessAndSendsEmail \(\)](#)

- async Task [RequestPasswordResetAsync_UserNotFound_ReturnsSuccessAndDoesNotSendEmail \(\)](#)
- async Task [RequestPasswordResetAsync_InvalidEmailFormat_ReturnsSuccessAndDoesNothing \(\)](#)
- async Task [RequestPasswordResetAsync_TokenCreationFails_ReturnsFailure \(\)](#)
- async Task [ResetPasswordAsync_ValidRequestAndToken_ReturnsSuccess \(\)](#)
- async Task [ResetPasswordAsync_InvalidInput_ReturnsFailure \(string? token, string? newPass, string? confirmPass\)](#)
- async Task [ResetPasswordAsync_PasswordProcessingFails_ReturnsFailure \(\)](#)
- async Task [ResetPasswordAsync_InvalidToken_ReturnsFailure \(\)](#)
- async Task [ResetPasswordAsync_PasswordUpdateFails_ReturnsFailure \(\)](#)
- async Task [ResetPasswordAsync_TokenDeletionFails_StillReturnsSuccess \(\)](#)

6.31.1 Detailed Description

Contains unit tests for the AuthPasswordService class.

6.31.2 Constructor & Destructor Documentation

6.31.2.1 AuthPasswordServiceTests()

```
ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.AuthPasswordServiceTests ( ) [inline]
```

Initializes a new instance of the AuthPasswordServiceTests class, setting up all mock dependencies and the service under test.

6.31.3 Member Function Documentation

6.31.3.1 RequestPasswordResetAsync_InvalidEmailFormat_ReturnsSuccessAndDoesNothing()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.RequestPasswordResetAsync_InvalidEmailFormat_ReturnsSuccessAndDoesNothing ( ) [inline]
```

6.31.3.2 RequestPasswordResetAsync_TokenCreationFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.RequestPasswordResetAsync_TokenCreationFails_ReturnsFailure ( ) [inline]
```

6.31.3.3 RequestPasswordResetAsync_UserNotFound_ReturnsSuccessAndDoesNotSendEmail()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.RequestPasswordResetAsync_UserNotFound_ReturnsSuccessAndDoesNotSendEmail ( ) [inline]
```

6.31.3.4 RequestPasswordResetAsync_ValidEmailUserExists_ReturnsSuccessAndSendsEmail()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.RequestPasswordResetAsync_ValidEmailUserExists_ReturnsSuccessAndSendsEmail ( ) [inline]
```

6.31.3.5 ResetPasswordAsync_InvalidInput_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_InvalidInput_ReturnsFailure (
    string? token,
    string? newPass,
    string? confirmPass ) [inline]
```

6.31.3.6 ResetPasswordAsync_InvalidToken_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_InvalidToken_ReturnsFailure ( ) [inline]
```

6.31.3.7 ResetPasswordAsync_PasswordProcessingFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_PasswordProcessingFails_ReturnsFailure ( ) [inline]
```

6.31.3.8 ResetPasswordAsync_PasswordUpdateFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_PasswordUpdateFails_ReturnsFailure ( ) [inline]
```

6.31.3.9 ResetPasswordAsync_TokenDeletionFails_StillReturnsSuccess()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_TokenDeletionFails_StillReturnsSuccess ( ) [inline]
```

6.31.3.10 ResetPasswordAsync_ValidRequestAndToken_ReturnsSuccess()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests.ResetPasswordAsync_ValidRequestAndToken_ReturnsSuccess ( ) [inline]
```

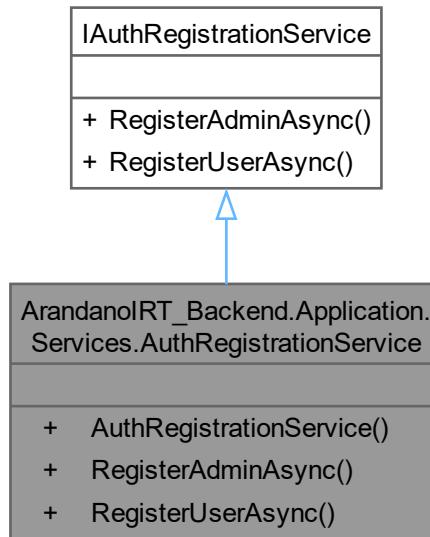
The documentation for this class was generated from the following file:

- Tests/Unit/Application/Services/[AuthPasswordServiceTests.cs](#)

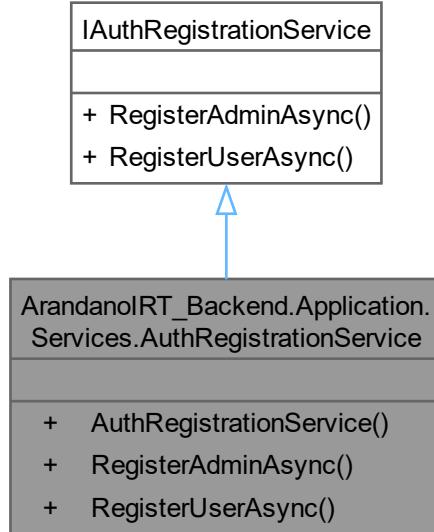
6.32 ArandanoIRT_Backend.Application.Services.AuthRegistrationService Class Reference

Implements services for registering new administrators and regular users.

Inheritance diagram for ArandanoIRT_Backend.Application.Services.AuthRegistrationService:



Collaboration diagram for ArandanoIRT_Backend.Application.Services.AuthRegistrationService:



Public Member Functions

- `AuthRegistrationService (IPersonRepository personRepository, ICropRepository cropRepository, ICropInvitationRepository cropInvitationRepository, IAuthUtilities authUtilities, IEmailService emailService, IDateTimeProvider dateServiceProvider, ILogger< AuthRegistrationService > logger)`
- `async Task< Result > RegisterAdminAsync (RegisterAdminRequestDto request)`

Registers a new administrator user along with their initial crop.

Parameters

<code>request</code>	<i>DTO containing admin and crop information.</i>
----------------------	---

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

- `async Task< Result > RegisterUserAsync (RegisterUserRequestDto request)`

Registers a new regular user using an invitation code.

Parameters

<code>request</code>	<i>DTO containing user information and the invitation code.</i>
----------------------	---

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

6.32.1 Detailed Description

Implements services for registering new administrators and regular users.

6.32.2 Constructor & Destructor Documentation

6.32.2.1 AuthRegistrationService()

```
ArandanoIRT_Backend.Application.Services.AuthRegistrationService.AuthRegistrationService (
    IPersonRepository personRepository,
    ICropRepository cropRepository,
    ICropInvitationRepository cropInvitationRepository,
    IAuthUtilities authUtilities,
    IEmailService emailService,
    IDateTimeProvider dateTimeProvider,
    ILogger< AuthRegistrationService > logger ) [inline]
```

6.32.3 Member Function Documentation

6.32.3.1 RegisterAdminAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthRegistrationService.Register←
AdminAsync (
    RegisterAdminRequestDto request ) [inline]
```

Registers a new administrator user along with their initial crop.

Parameters

<code>request</code>	DTO containing admin and crop information.
----------------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService](#).

6.32.3.2 RegisterUserAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthRegistrationService.Register←
UserAsync (
    RegisterUserRequestDto request ) [inline]
```

Registers a new regular user using an invitation code.

Parameters

<i>request</i>	DTO containing user information and the invitation code.
----------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService](#).

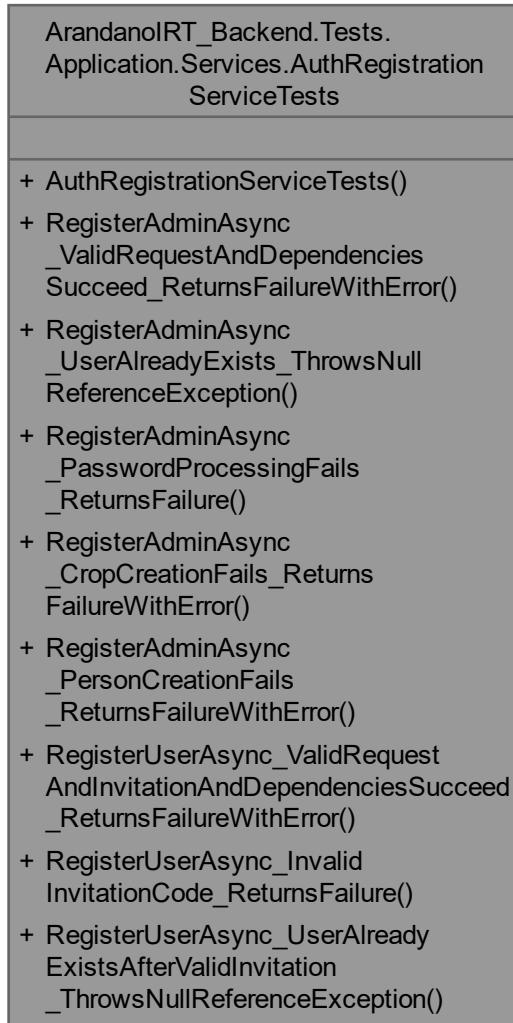
The documentation for this class was generated from the following file:

- Application/Services/[AuthRegistrationService.cs](#)

6.33 ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests Class Reference

Contains unit tests for the AuthRegistrationService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests:



Public Member Functions

- [AuthRegistrationServiceTests \(\)](#)
Initializes a new instance of the AuthRegistrationServiceTests class, setting up all mock dependencies and the service under test.
- [async Task RegisterAdminAsync_ValidRequestAndDependenciesSucceed_ReturnsFailureWithException \(\)](#)
- [async Task RegisterAdminAsync_UserAlreadyExists_ThrowsNullReferenceException \(\)](#)
- [async Task RegisterAdminAsync_PasswordProcessingFails_ReturnsFailure \(\)](#)
- [async Task RegisterAdminAsync_CropCreationFails_ReturnsFailureWithException \(\)](#)
- [async Task RegisterAdminAsync_PersonCreationFails_ReturnsFailureWithException \(\)](#)

- async Task [RegisterUserAsync_ValidRequestAndInvitationAndDependenciesSucceed_ReturnsFailureWithError\(\)](#)
- async Task [RegisterUserAsync_InvalidInvitationCode_ReturnsFailure\(\)](#)
- async Task [RegisterUserAsync_UserAlreadyExistsAfterValidInvitation_ThrowsNullReferenceException\(\)](#)

6.33.1 Detailed Description

Contains unit tests for the AuthRegistrationService class.

6.33.2 Constructor & Destructor Documentation

6.33.2.1 AuthRegistrationServiceTests()

```
ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.AuthRegistration←  
ServiceTests ( ) [inline]
```

Initializes a new instance of the AuthRegistrationServiceTests class, setting up all mock dependencies and the service under test.

6.33.3 Member Function Documentation

6.33.3.1 RegisterAdminAsync_CropCreationFails_ReturnsFailureWithError()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
AdminAsync_CropCreationFails_ReturnsFailureWithError ( ) [inline]
```

6.33.3.2 RegisterAdminAsync_PasswordProcessingFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
AdminAsync_PasswordProcessingFails_ReturnsFailure ( ) [inline]
```

6.33.3.3 RegisterAdminAsync_PersonCreationFails_ReturnsFailureWithError()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
AdminAsync_PersonCreationFails_ReturnsFailureWithError ( ) [inline]
```

6.33.3.4 RegisterAdminAsync_UserAlreadyExists_ThrowsNullReferenceException()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
AdminAsync_UserAlreadyExists_ThrowsNullReferenceException ( ) [inline]
```

6.33.3.5 RegisterAdminAsync_ValidRequestAndDependenciesSucceed_ReturnsFailureWithError()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
AdminAsync_ValidRequestAndDependenciesSucceed_ReturnsFailureWithError () [inline]
```

6.33.3.6 RegisterUserAsync_InvalidInvitationCode_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
UserAsync_InvalidInvitationCode_ReturnsFailure () [inline]
```

6.33.3.7 RegisterUserAsync_UserAlreadyExistsAfterValidInvitation_ThrowsNullReferenceException()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
UserAsync_UserAlreadyExistsAfterValidInvitation_ThrowsNullReferenceException () [inline]
```

6.33.3.8 RegisterUserAsync_ValidRequestAndInvitationAndDependenciesSucceed_ReturnsFailureWithError()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests.Register←  
UserAsync_ValidRequestAndInvitationAndDependenciesSucceed_ReturnsFailureWithError () [inline]
```

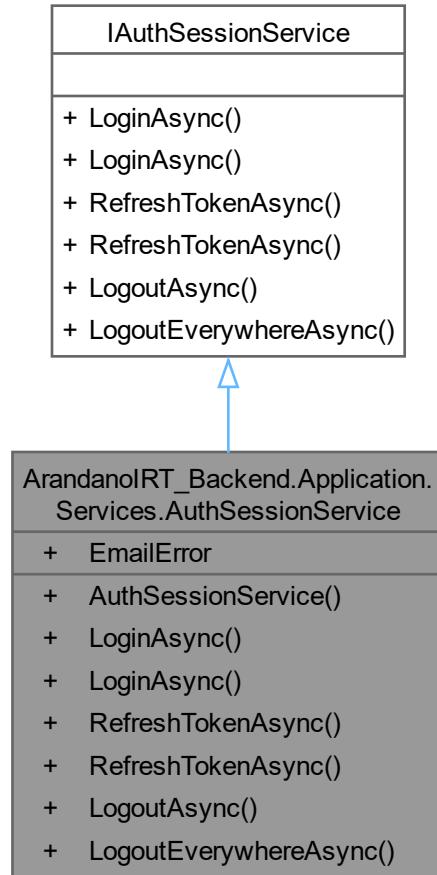
The documentation for this class was generated from the following file:

- Tests/Unit/Application/Services/[AuthRegistrationServiceTests.cs](#)

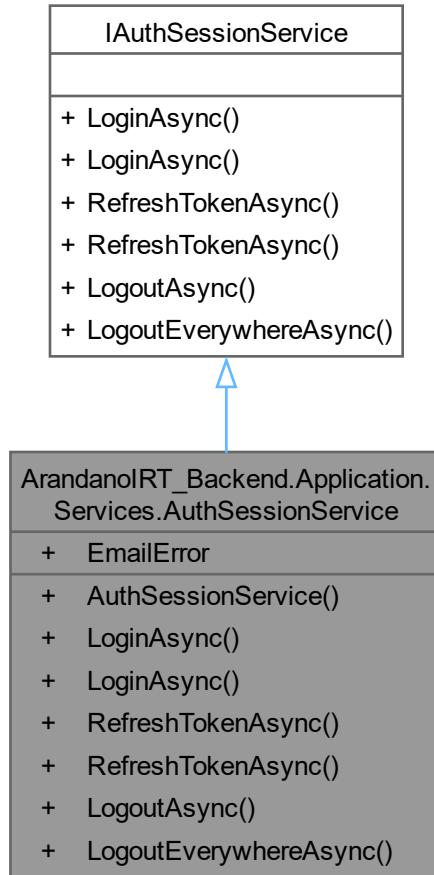
6.34 ArandanoIRT_Backend.Application.Services.AuthSessionService Class Reference

Implements services for user session management including login, token refresh, and logout.

Inheritance diagram for ArandanoIRT_Backend.Application.Services.AuthSessionService:



Collaboration diagram for ArandanoIRT_Backend.Application.Services.AuthSessionService:



Public Member Functions

- `AuthSessionService (IAuthUtilities authUtilities, IPersonRepository personRepository, IRefreshTokenRepository refreshTokenRepository, IFailedLoginAttemptRepository failedLoginAttemptRepository, IRsaService rsaService, IPasswordHasher passwordHasher, ITokenService tokenService, IEmailService emailService, IDateTimeProvider dateServiceProvider, ILogger<AuthSessionService> logger)`
- `async Task<Result<LoginSuccessPayload>> LoginAsync (LoginRequestDto request, string ipAddress, string userAgent)`

Authenticates a user based on email and password without device information.

Parameters

request	DTO containing login credentials.
ipAddress	The IP address of the requesting client.
userAgent	The user agent string of the requesting client.

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

- `async Task< Result< LoginSuccessPayload > > LoginAsync (LoginRequestDto request, string ipAddress, string userAgent, string? deviceInfo)`

Authenticates a user based on email and password, including device information.

Parameters

request	<i>DTO containing login credentials.</i>
ipAddress	<i>The IP address of the requesting client.</i>
userAgent	<i>The user agent string of the requesting client.</i>
deviceInfo	<i>Specific device information (nullable).</i>

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

- `async Task< Result< TokenResponseDto > > RefreshTokenAsync (string refreshTokenValue, string ipAddress, string userAgent)`

Refreshes the access token using a valid refresh token without device information.

Implements refresh token rotation via the underlying token service.

Parameters

refreshTokenValue	<i>The opaque refresh token string.</i>
ipAddress	<i>The IP address of the requesting client.</i>
userAgent	<i>The user agent string of the requesting client.</i>

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

- `async Task< Result< TokenResponseDto > > RefreshTokenAsync (string refreshTokenValue, string ipAddress, string userAgent, string? deviceInfo)`

Refreshes the access token using a valid refresh token, including device information.

Implements refresh token rotation via the underlying token service.

Parameters

refreshTokenValue	<i>The opaque refresh token string.</i>
ipAddress	<i>The IP address of the requesting client.</i>
userAgent	<i>The user agent string of the requesting client.</i>
deviceInfo	<i>Specific device information (nullable).</i>

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

- `async Task< Result > LogoutAsync (string refreshTokenValue, string ipAddress)`

Revokes the specific refresh token used for the current session (logout).

Parameters

refreshTokenValue	<i>The opaque refresh token string to revoke.</i>
ipAddress	<i>The IP address performing the logout.</i>

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

- `async Task< Result > LogoutEverywhereAsync (long sessionId, string ipAddress)`

Revokes all active refresh tokens associated with a user's session ID (logout everywhere).

Parameters

<code>session← Id</code>	<i>The session ID (typically from the JWT 'sid' claim) to revoke tokens for.</i>
<code>ipAddress</code>	<i>The IP address performing the logout.</i>

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Static Public Attributes

- `const string EmailError = "Invalid email or password."`

6.34.1 Detailed Description

Implements services for user session management including login, token refresh, and logout.

6.34.2 Constructor & Destructor Documentation**6.34.2.1 AuthSessionService()**

```
ArandanoIRT_Backend.Application.Services.AuthSessionService.AuthSessionService (
    IAuthUtilities authUtilities,
    IPersonRepository personRepository,
    IRefreshTokenRepository refreshTokenRepository,
    IFailedLoginAttemptRepository failedLoginAttemptRepository,
    IRsaService rsaService,
    IPasswordHasher passwordHasher,
    ITokenService tokenService,
    IEmailService emailService,
    IDateTimeProvider dateTimeProvider,
    ILogger< AuthSessionService > logger ) [inline]
```

6.34.3 Member Function Documentation**6.34.3.1 LoginAsync() [1/2]**

```
async Task< Result< LoginSuccessPayload > > ArandanoIRT_Backend.Application.Services.Auth←
SessionService.LoginAsync (
    LoginRequestDto request,
    string ipAddress,
    string userAgent ) [inline]
```

Authenticates a user based on email and password without device information.

Parameters

<i>request</i>	DTO containing login credentials.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.3.2 LoginAsync() [2/2]

```
async Task< Result< LoginSuccessPayload > > ArandanoIRT_Backend.Application.Services.AuthSessionService.LoginAsync (
    LoginRequestDto request,
    string ipAddress,
    string userAgent,
    string? deviceInfo ) [inline]
```

Authenticates a user based on email and password, including device information.

Parameters

<i>request</i>	DTO containing login credentials.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.
<i>deviceInfo</i>	Specific device information (nullable).

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.3.3 LogoutAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthSessionService.LogoutAsync (
    string refreshTokenValue,
    string ipAddress ) [inline]
```

Revokes the specific refresh token used for the current session (logout).

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string to revoke.
<i>ipAddress</i>	The IP address performing the logout.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.3.4 LogoutEverywhereAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.AuthSessionService.LogoutEverywhereAsync (
    long sessionId,
    string ipAddress ) [inline]
```

Revokes all active refresh tokens associated with a user's session ID (logout everywhere).

Parameters

<i>sessionId</i>	The session ID (typically from the JWT 'sid' claim) to revoke tokens for.
<i>ipAddress</i>	The IP address performing the logout.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.3.5 RefreshTokenAsync() [1/2]

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Application.Services.AuthSessionService.RefreshTokenAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent ) [inline]
```

Refreshes the access token using a valid refresh token without device information.

Implements refresh token rotation via the underlying token service.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.3.6 RefreshTokenAsync() [2/2]

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Application.Services.AuthSessionService.RefreshTokenAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent,
    string? deviceInfo ) [inline]
```

Refreshes the access token using a valid refresh token, including device information.

Implements refresh token rotation via the underlying token service.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.
<i>deviceInfo</i>	Specific device information (nullable).

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#).

6.34.4 Member Data Documentation

6.34.4.1 EmailError

```
const string ArandanoIRT_Backend.Application.Services.AuthSessionService.EmailError = "Invalid
email or password." [static]
```

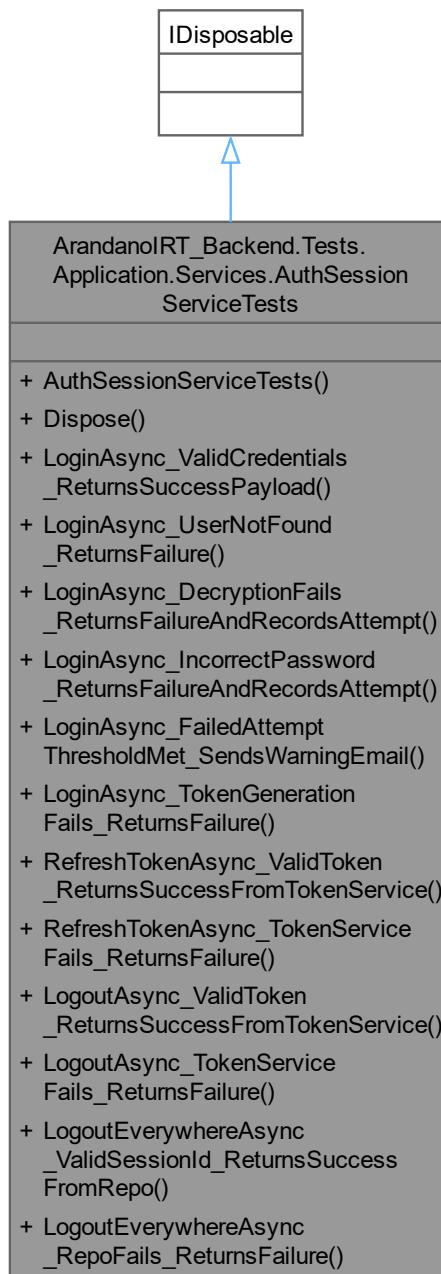
The documentation for this class was generated from the following file:

- Application/Services/[AuthSessionService.cs](#)

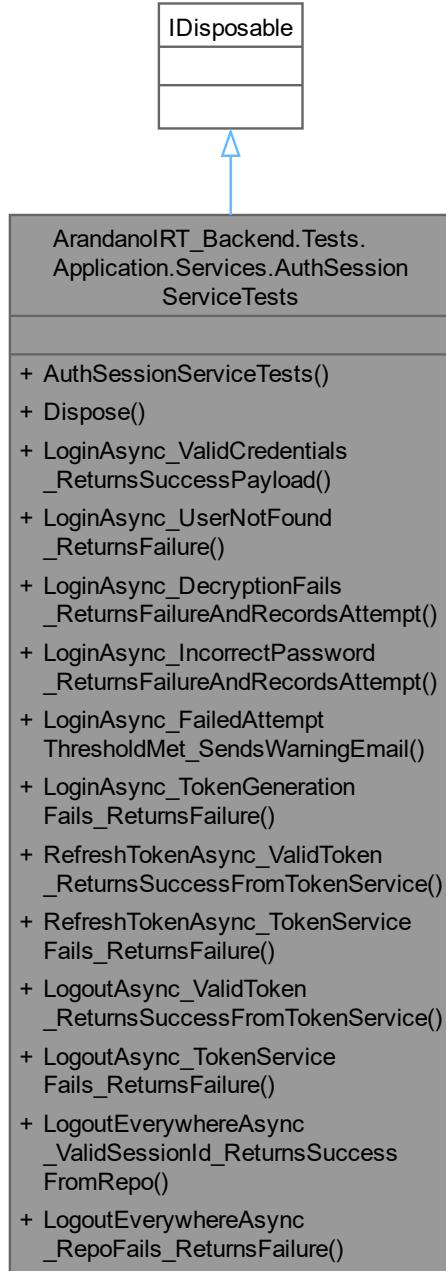
6.35 ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests Class Reference

Contains unit tests for the AuthSessionService class.

Inheritance diagram for ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests:



Public Member Functions

- [AuthSessionServiceTests \(\)](#)

Initializes a new instance of the AuthSessionServiceTests class, setting up all mock dependencies and the service under test.

- void [Dispose \(\)](#)
Disposes resources used by the test context, if any.
- async Task [LoginAsync_ValidCredentials_ReturnsSuccessPayload \(\)](#)
- async Task [LoginAsync_UserNotFound_ReturnsFailure \(\)](#)
- async Task [LoginAsync_DecryptionFails_ReturnsFailureAndRecordsAttempt \(\)](#)
- async Task [LoginAsync_IncorrectPassword_ReturnsFailureAndRecordsAttempt \(\)](#)
- async Task [LoginAsync_FailedAttemptThresholdMet_SendsWarningEmail \(\)](#)
- async Task [LoginAsync_TokenGenerationFails_ReturnsFailure \(\)](#)
- async Task [RefreshTokenAsync_ValidToken_ReturnsSuccessFromTokenService \(\)](#)
- async Task [RefreshTokenAsync_TokenServiceFails_ReturnsFailure \(\)](#)
- async Task [LogoutAsync_ValidToken_ReturnsSuccessFromTokenService \(\)](#)
- async Task [LogoutAsync_TokenServiceFails_ReturnsFailure \(\)](#)
- async Task [LogoutEverywhereAsync_ValidSessionId_ReturnsSuccessFromRepo \(\)](#)
- async Task [LogoutEverywhereAsync_RepoFails_ReturnsFailure \(\)](#)

6.35.1 Detailed Description

Contains unit tests for the AuthSessionService class.

6.35.2 Constructor & Destructor Documentation

6.35.2.1 AuthSessionServiceTests()

```
ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.AuthSessionServiceTests ( )  
[inline]
```

Initializes a new instance of the AuthSessionServiceTests class, setting up all mock dependencies and the service under test.

6.35.3 Member Function Documentation

6.35.3.1 Dispose()

```
void ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.Dispose ( ) [inline]
```

Disposes resources used by the test context, if any.

Currently relies on GC but could implement IDisposable pattern if needed.

6.35.3.2 LoginAsync_DecryptionFails_ReturnsFailureAndRecordsAttempt()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_<-  
DecryptionFails_ReturnsFailureAndRecordsAttempt ( ) [inline]
```

6.35.3.3 **LoginAsync_FailedAttemptThresholdMet_SendsWarningEmail()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_←  
FailedAttemptThresholdMet_SendsWarningEmail ( ) [inline]
```

6.35.3.4 **LoginAsync_IncorrectPassword_ReturnsFailureAndRecordsAttempt()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_←  
IncorrectPassword_ReturnsFailureAndRecordsAttempt ( ) [inline]
```

6.35.3.5 **LoginAsync_TokenGenerationFails_ReturnsFailure()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_←  
TokenGenerationFails_ReturnsFailure ( ) [inline]
```

6.35.3.6 **LoginAsync_UserNotFound_ReturnsFailure()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_←  
UserNotFound_ReturnsFailure ( ) [inline]
```

6.35.3.7 **LoginAsync_ValidCredentials_ReturnsSuccessPayload()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LoginAsync_←  
ValidCredentials_ReturnsSuccessPayload ( ) [inline]
```

6.35.3.8 **LogoutAsync_TokenServiceFails_ReturnsFailure()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LogoutAsync_←  
TokenServiceFails_ReturnsFailure ( ) [inline]
```

6.35.3.9 **LogoutAsync_ValidToken_ReturnsSuccessFromTokenService()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LogoutAsync_←  
ValidToken_ReturnsSuccessFromTokenService ( ) [inline]
```

6.35.3.10 **LogoutEverywhereAsync_RepoFails_ReturnsFailure()**

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LogoutEverywhere_←  
Async_RepoFails_ReturnsFailure ( ) [inline]
```

6.35.3.11 LogoutEverywhereAsync_ValidSessionId_ReturnsSuccessFromRepo()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.LogoutEverywhere←
Async_ValidSessionId_ReturnsSuccessFromRepo ( ) [inline]
```

6.35.3.12 RefreshTokenAsync_TokenServiceFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.RefreshToken←
Async_TokenServiceFails_ReturnsFailure ( ) [inline]
```

6.35.3.13 RefreshTokenAsync_ValidToken_ReturnsSuccessFromTokenService()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests.RefreshToken←
Async_ValidToken_ReturnsSuccessFromTokenService ( ) [inline]
```

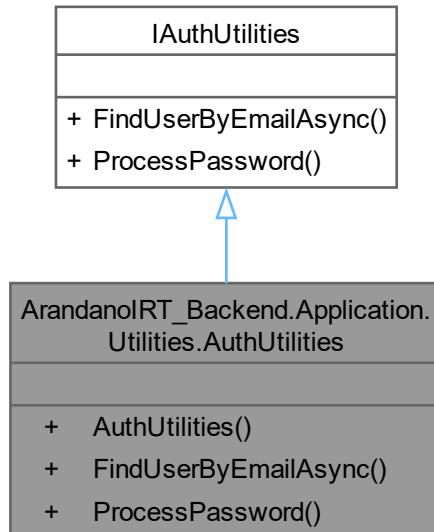
The documentation for this class was generated from the following file:

- Tests/Unit/Application/Services/AuthSessionServiceTests.cs

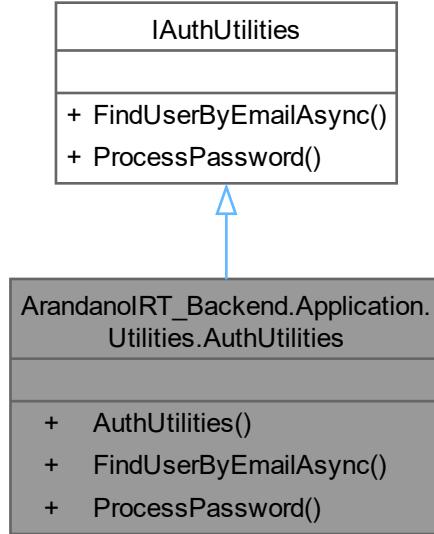
6.36 ArandanoIRT_Backend.Application.Utilities.AuthUtilities Class Reference

Implements shared utility functions for authentication services.

Inheritance diagram for ArandanoIRT_Backend.Application.Utilities.AuthUtilities:



Collaboration diagram for ArandanoIRT_Backend.Application.Utilities.AuthUtilities:



Public Member Functions

- `AuthUtilities (IPersonRepository personRepository, IRsaService rsaService, IPasswordHasher passwordHasher, ILogger<AuthUtilities> logger)`
- `async Task<Result<PersonEntity>> FindUserByEmailAsync (string email)`

Finds a user by email in the repository.

Parameters

email	The email address to search for.
-------	----------------------------------

Returns

A Task resulting in a Result containing the PersonEntity on success, or Failure if not found.

- `Result<string> ProcessPassword (string encryptedPassword, string operationContextForLog)`

Decrypts, validates policy, and hashes a password.

Parameters

encryptedPassword	The encrypted password string.
operationContextForLog	A string indicating the context (e.g., "AdminRegistration", "PasswordReset") for logging purposes.

Returns

Result containing the hashed password on success, or Failure.

6.36.1 Detailed Description

Implements shared utility functions for authentication services.

6.36.2 Constructor & Destructor Documentation

6.36.2.1 AuthUtilities()

```
ArandanoIRT_Backend.Application.Utilities.AuthUtilities.AuthUtilities (
    IPersonRepository personRepository,
    IRsaService rsaService,
    IPasswordHasher passwordHasher,
    ILogger< AuthUtilities > logger ) [inline]
```

6.36.3 Member Function Documentation

6.36.3.1 FindUserByEmailAsync()

```
async Task< Result< PersonEntity > > ArandanoIRT_Backend.Application.Utilities.AuthUtilities.←
FindUserByEmailAsync (
    string email ) [inline]
```

Finds a user by email in the repository.

Parameters

<i>email</i>	The email address to search for.
--------------	----------------------------------

Returns

A Task resulting in a Result containing the PersonEntity on success, or Failure if not found.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities](#).

6.36.3.2 ProcessPassword()

```
Result< string > ArandanoIRT_Backend.Application.Utilities.AuthUtilities.ProcessPassword (
    string encryptedPassword,
    string operationContextForLog ) [inline]
```

Decrypts, validates policy, and hashes a password.

Parameters

<i>encryptedPassword</i>	The encrypted password string.
<i>operationContextForLog</i>	A string indicating the context (e.g., "AdminRegistration", "PasswordReset") for logging purposes.

Returns

Result containing the hashed password on success, or Failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities](#).

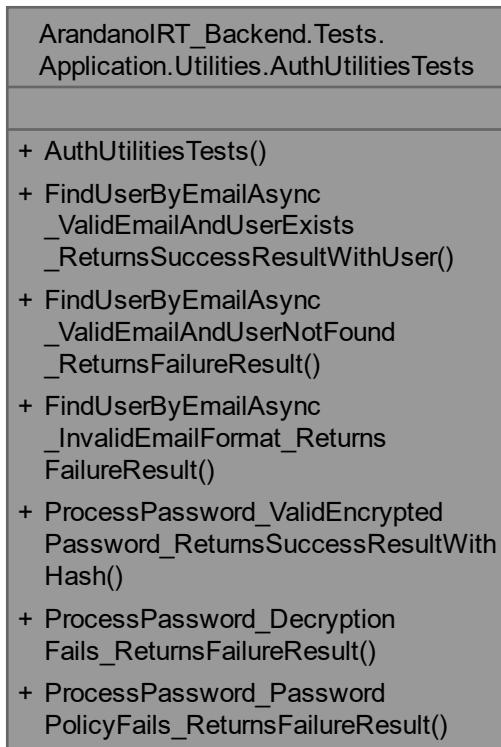
The documentation for this class was generated from the following file:

- Application/Utilities/AuthUtilities.cs

6.37 ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests Class Reference

Contains unit tests for the AuthUtilities class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests:



Public Member Functions

- [AuthUtilitiesTests \(\)](#)
Initializes a new instance of the AuthUtilitiesTests class, setting up mocks and the system under test.
- [async Task FindUserByEmailAsync_ValidateEmailAndUserExists_ReturnsSuccessResultWithUser \(\)](#)
Tests that FindUserByEmailAsync returns a successful result with the correct user when the email is valid and the user exists in the repository.
- [async Task FindUserByEmailAsync_ValidEmailAndUserNotFound_ReturnsFailureResult \(\)](#)
Tests that FindUserByEmailAsync returns a failure result when the email is valid but the user does not exist in the repository.
- [async Task FindUserByEmailAsync_InvalidEmailFormat_ReturnsFailureResult \(string? invalidEmail\)](#)
Tests that FindUserByEmailAsync returns a failure result when the email format is invalid, bypassing the repository call.
- [void ProcessPassword_VerifyEncryptedPassword_ReturnsSuccessResultWithHash \(\)](#)
Tests that ProcessPassword returns a successful result with the correctly hashed password when decryption and policy validation succeed.
- [void ProcessPassword_DecryptionFails_ReturnsFailureResult \(\)](#)
Tests that ProcessPassword returns a failure result when the RSA service fails to decrypt the password.
- [void ProcessPassword_PasswordPolicyFails_ReturnsFailureResult \(\)](#)
Tests that ProcessPassword returns a failure result when decryption succeeds but the password fails the length policy check.

6.37.1 Detailed Description

Contains unit tests for the AuthUtilities class.

6.37.2 Constructor & Destructor Documentation

6.37.2.1 AuthUtilitiesTests()

```
ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.AuthUtilitiesTests ( ) [inline]
```

Initializes a new instance of the AuthUtilitiesTests class, setting up mocks and the system under test.

6.37.3 Member Function Documentation

6.37.3.1 FindUserByEmailAsync_InvalidEmailFormat_ReturnsFailureResult()

```
async Task ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.FindUserByEmailAsync_InvalidEmailFormat_ReturnsFailureResult (
    string? invalidEmail) [inline]
```

Tests that FindUserByEmailAsync returns a failure result when the email format is invalid, bypassing the repository call.

Note: This test relies on the behavior of the static EmailValidatorUtility.

6.37.3.2 `FindUserByEmailAsync_ValidEmailAndUserExists_ReturnsSuccessResultWithUser()`

```
async Task ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.FindUserByEmail←  
Async_ValidEmailAndUserExists_ReturnsSuccessResultWithUser ( ) [inline]
```

Tests that `FindUserByEmailAsync` returns a successful result with the correct user when the email is valid and the user exists in the repository.

6.37.3.3 `FindUserByEmailAsync_ValidEmailAndUserNotFound_ReturnsFailureResult()`

```
async Task ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.FindUserByEmail←  
Async_ValidEmailAndUserNotFound_ReturnsFailureResult ( ) [inline]
```

Tests that `FindUserByEmailAsync` returns a failure result when the email is valid but the user does not exist in the repository.

6.37.3.4 `ProcessPassword_DecryptionFails_ReturnsFailureResult()`

```
void ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.ProcessPassword_Decryption←  
Fails_ReturnsFailureResult ( ) [inline]
```

Tests that `ProcessPassword` returns a failure result when the RSA service fails to decrypt the password.

6.37.3.5 `ProcessPassword_PasswordPolicyFails_ReturnsFailureResult()`

```
void ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.ProcessPassword_Password←  
PolicyFails_ReturnsFailureResult ( ) [inline]
```

Tests that `ProcessPassword` returns a failure result when decryption succeeds but the password fails the length policy check.

6.37.3.6 `ProcessPassword_ValidEncryptedPassword_ReturnsSuccessResultWithHash()`

```
void ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests.ProcessPassword_Valid←  
EncryptedPassword_ReturnsSuccessResultWithHash ( ) [inline]
```

Tests that `ProcessPassword` returns a successful result with the correctly hashed password when decryption and policy validation succeed.

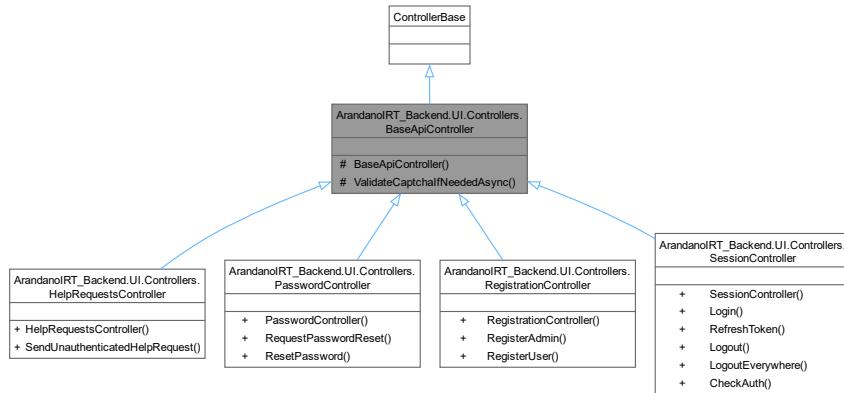
The documentation for this class was generated from the following file:

- Tests/Application/Utilities/[AuthUtilitiesTests.cs](#)

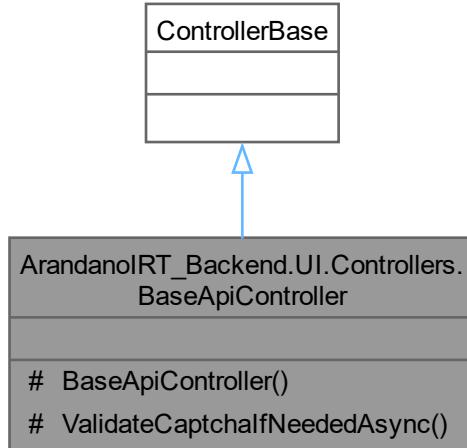
6.38 ArandanoIRT_Backend.UI.Controllers.BaseApiController Class Reference

Base controller for API endpoints providing shared functionality.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.BaseApiController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.BaseApiController:



Protected Member Functions

- `BaseApiController (ICaptchaService captchaService, IWebHostEnvironment environment)`

- Initializes a new instance of the BaseApiController class.*
- `async Task< IActionResult?> ValidateCaptchaIfNeededAsync (string? captchaToken, ILogger logger)`
Validates the CAPTCHA token if the application is not running in the Development environment.

6.38.1 Detailed Description

Base controller for API endpoints providing shared functionality.

6.38.2 Constructor & Destructor Documentation

6.38.2.1 BaseApiController()

```
ArandanoIRT_Backend.UI.Controllers.BaseApiController.BaseApiController (
    ICaptchaService captchaService,
    IWebHostEnvironment environment ) [inline], [protected]
```

Initializes a new instance of the BaseApiController class.

Parameters

<code>captchaService</code>	Service for CAPTCHA verification.
<code>environment</code>	Provides information about the web hosting environment.

Exceptions

<code>ArgumentNullException</code>	Thrown if captchaService or environment is null.
------------------------------------	--

6.38.3 Member Function Documentation

6.38.3.1 ValidateCaptchaIfNeededAsync()

```
async Task< IActionResult?> ArandanoIRT_Backend.UI.Controllers.BaseApiController.ValidateCaptchaIfNeededAsync (
    string? captchaToken,
    ILogger logger ) [inline], [protected]
```

Validates the CAPTCHA token if the application is not running in the Development environment.

Parameters

<code>captchaToken</code>	The CAPTCHA token received from the client.
<code>logger</code>	The specific logger instance from the derived controller for contextual logging.

Returns

A Task resulting in an IActionResult (specifically BadRequestObjectResult) if CAPTCHA validation fails or is required but missing. Returns null if validation is successful or skipped (in Development environment).

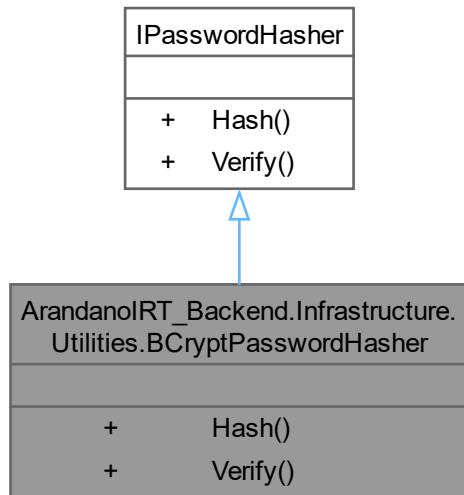
The documentation for this class was generated from the following file:

- UI/Controllers/[BaseApiController.cs](#)

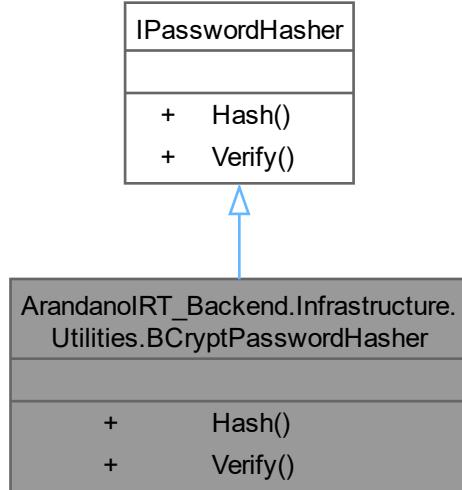
6.39 ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher Class Reference

Implements the IPasswordHasher interface using the BCrypt algorithm provided by the BCrypt.Net library.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher:



Public Member Functions

- string [Hash](#) (string password)

Creates a hash from a plain text password using BCrypt.
- bool [Verify](#) (string hash, string providedPassword)

Verifies a plain text password against a stored BCrypt hash.

6.39.1 Detailed Description

Implements the IPasswordHasher interface using the BCrypt algorithm provided by the BCrypt.Net library.

6.39.2 Member Function Documentation

6.39.2.1 Hash()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher.Hash (
    string password ) [inline]
```

Creates a hash from a plain text password using BCrypt.

Parameters

<i>password</i>	The plain text password to hash.
-----------------	----------------------------------

Returns

The generated BCrypt password hash string.

Exceptions

<i>ArgumentException</i>	Thrown if the input password is null, empty, or whitespace.
--------------------------	---

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher](#).

6.39.2.2 Verify()

```
bool ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher.Verify (
    string hash,
    string providedPassword ) [inline]
```

Verifies a plain text password against a stored BCrypt hash.

Parameters

<i>hash</i>	The stored BCrypt password hash.
<i>providedPassword</i>	The plain text password provided by the user for verification.

Returns

`true` if the provided password matches the hash; otherwise, `false`.

Returns `false` immediately if either the hash or the provided password is null or empty. Also returns `false` and logs a warning if the provided hash string is not a valid BCrypt format.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher](#).

The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[BCryptPasswordHasher.cs](#)

6.40 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCrypt PasswordHasherTests Class Reference

Contains unit tests for the BCryptPasswordHasher class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests:

ArandanoIRT_Backend.Tests. Unit.Infrastructure.Utilities. BCryptPasswordHasherTests
+ Hash_WithValidPassword _ShouldReturnNonEmptyHash() + Hash_WithNullPassword _ShouldThrowArgumentException() + Hash_WithEmptyPassword _ShouldThrowArgumentException() + Hash_WithWhitespacePassword _ShouldThrowArgumentException() + Verify_WithCorrectPassword _ShouldReturnTrue() + Verify_WithIncorrectPassword _ShouldReturnFalse() + Verify_WithNullHash _ShouldReturnFalse() + Verify_WithNullProvidedPassword _ShouldReturnFalse() + Verify_WithEmptyHash _ShouldReturnFalse() + Verify_WithEmptyProvided Password_ShouldReturnFalse() + Verify_WithInvalidHashFormat _ShouldReturnFalse()

Public Member Functions

- void [Hash_WithValidPassword_ShouldReturnNonEmptyHash \(\)](#)
Tests that the Hash method returns a non-empty string (the hash) when provided with a valid, non-empty password.
- void [Hash_WithNullPassword_ShouldThrowArgumentException \(\)](#)

Tests that the Hash method throws an ArgumentException when the provided password is null.

- void [Hash_WithEmptyPassword_ShouldThrowArgumentException \(\)](#)

Tests that the Hash method throws an ArgumentException when the provided password is an empty string.

- void [Hash_WithWhitespacePassword_ShouldThrowArgumentException \(\)](#)

Tests that the Hash method throws an ArgumentException when the provided password consists only of whitespace.

- void [Verify_WithCorrectPassword_ShouldReturnTrue \(\)](#)

Tests that the Verify method returns true when comparing a correct password against its corresponding BCrypt hash generated by the Hash method.

- void [Verify_WithIncorrectPassword_ShouldReturnFalse \(\)](#)

Tests that the Verify method returns false when comparing an incorrect password against a valid BCrypt hash.

- void [Verify_WithNullHash_ShouldReturnFalse \(\)](#)

Tests that the Verify method returns false when the provided hash string is null.

- void [Verify_WithNullProvidedPassword_ShouldReturnFalse \(\)](#)

Tests that the Verify method returns false when the provided password to check is null.

- void [Verify_WithEmptyHash_ShouldReturnFalse \(\)](#)

Tests that the Verify method returns false when the provided hash string is empty.

- void [Verify_WithEmptyProvidedPassword_ShouldReturnFalse \(\)](#)

Tests that the Verify method returns false when the provided password to check is empty.

- void [Verify_WithInvalidHashFormat_ShouldReturnFalse \(string invalidHash\)](#)

Tests that the Verify method returns false when provided with a hash string that does not conform to the expected BCrypt format.

6.40.1 Detailed Description

Contains unit tests for the BCryptPasswordHasher class.

6.40.2 Member Function Documentation

6.40.2.1 Hash_WithEmptyPassword_ShouldThrowArgumentException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Hash_WithEmptyPassword_ShouldThrowArgumentException ( ) [inline]
```

Tests that the Hash method throws an ArgumentException when the provided password is an empty string.

6.40.2.2 Hash_WithNullPassword_ShouldThrowArgumentException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Hash_WithNullPassword_ShouldThrowArgumentException ( ) [inline]
```

Tests that the Hash method throws an ArgumentException when the provided password is null.

6.40.2.3 Hash_WithValidPassword_ShouldReturnNonEmptyHash()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Hash_With←
ValidPassword_ShouldReturnNonEmptyHash ( ) [inline]
```

Tests that the Hash method returns a non-empty string (the hash) when provided with a valid, non-empty password.

6.40.2.4 Hash_WithWhitespacePassword_ShouldThrowArgumentException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Hash_With←
WhitespacePassword_ShouldThrowArgumentException ( ) [inline]
```

Tests that the Hash method throws an ArgumentException when the provided password consists only of whitespace.

6.40.2.5 Verify_WithCorrectPassword_ShouldReturnTrue()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←
WithCorrectPassword_ShouldReturnTrue ( ) [inline]
```

Tests that the Verify method returns true when comparing a correct password against its corresponding BCrypt hash generated by the Hash method.

6.40.2.6 Verify_WithEmptyHash_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←
WithEmptyHash_ShouldReturnFalse ( ) [inline]
```

Tests that the Verify method returns false when the provided hash string is empty.

6.40.2.7 Verify_WithEmptyProvidedPassword_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←
WithEmptyProvidedPassword_ShouldReturnFalse ( ) [inline]
```

Tests that the Verify method returns false when the provided password to check is empty.

6.40.2.8 Verify_WithIncorrectPassword_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←
WithIncorrectPassword_ShouldReturnFalse ( ) [inline]
```

Tests that the Verify method returns false when comparing an incorrect password against a valid BCrypt hash.

6.40.2.9 Verify_WithInvalidHashFormat_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←
WithInvalidHashFormat_ShouldReturnFalse (
    string invalidHash ) [inline]
```

Tests that the Verify method returns false when provided with a hash string that does not conform to the expected BCrypt format.

It also ensures no unexpected exception is thrown directly from the Verify method itself.

Parameters

<i>invalidHash</i>	An invalid or improperly formatted hash string.
--------------------	---

6.40.2.10 Verify_WithNullHash_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←  
WithNullHash_ShouldReturnFalse ( ) [inline]
```

Tests that the Verify method returns false when the provided hash string is null.

6.40.2.11 Verify_WithNullProvidedPassword_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests.Verify_←  
WithNullProvidedPassword_ShouldReturnFalse ( ) [inline]
```

Tests that the Verify method returns false when the provided password to check is null.

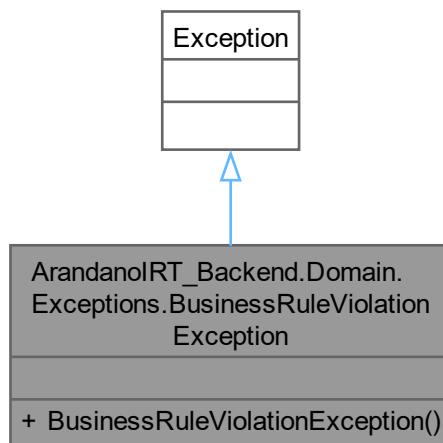
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Utilities/[BCryptPasswordHasherTests.cs](#)

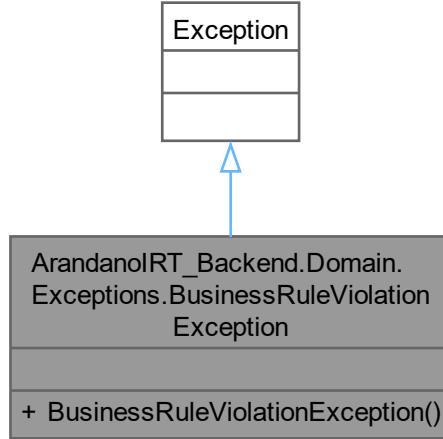
6.41 ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException Class Reference

Represents an error that occurs when an operation violates a specific business rule or constraint.

Inheritance diagram for ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException:



Collaboration diagram for ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException:



Public Member Functions

- [BusinessRuleViolationException](#) (string message)

Initializes a new instance of the BusinessRuleViolationException class with a specified error message that describes the business rule violation.

6.41.1 Detailed Description

Represents an error that occurs when an operation violates a specific business rule or constraint.

This exception inherits from the base `Exception` class.

6.41.2 Constructor & Destructor Documentation

6.41.2.1 BusinessRuleViolationException()

```
ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException.BusinessRuleViolationException<br/>(<br/>    string message )  [inline]
```

Initializes a new instance of the `BusinessRuleViolationException` class with a specified error message that describes the business rule violation.

Parameters

<i>message</i>	The message that describes the error and the business rule violated.
----------------	--

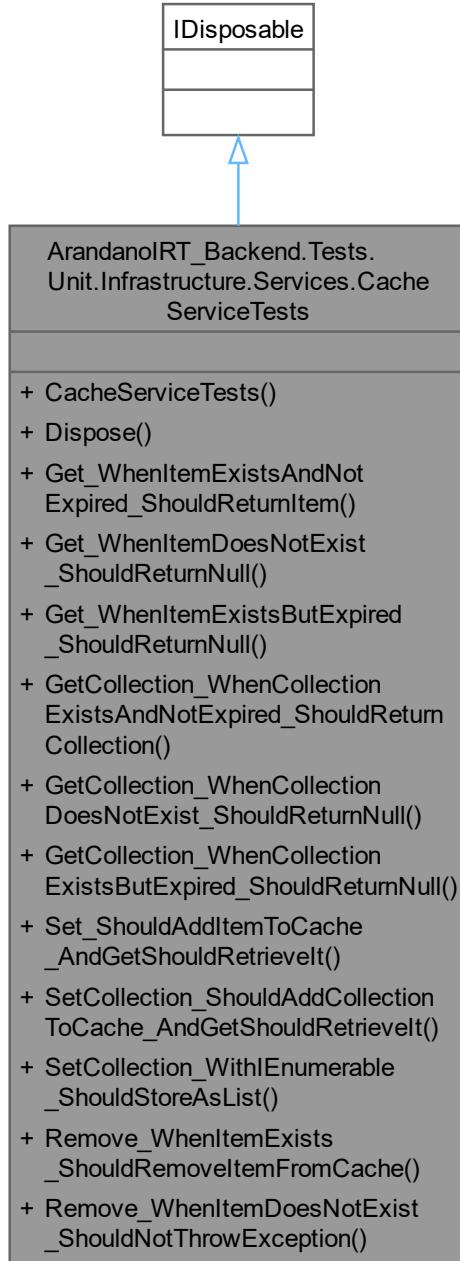
The documentation for this class was generated from the following file:

- Domain/Exceptions/[BusinessRuleViolationException.cs](#)

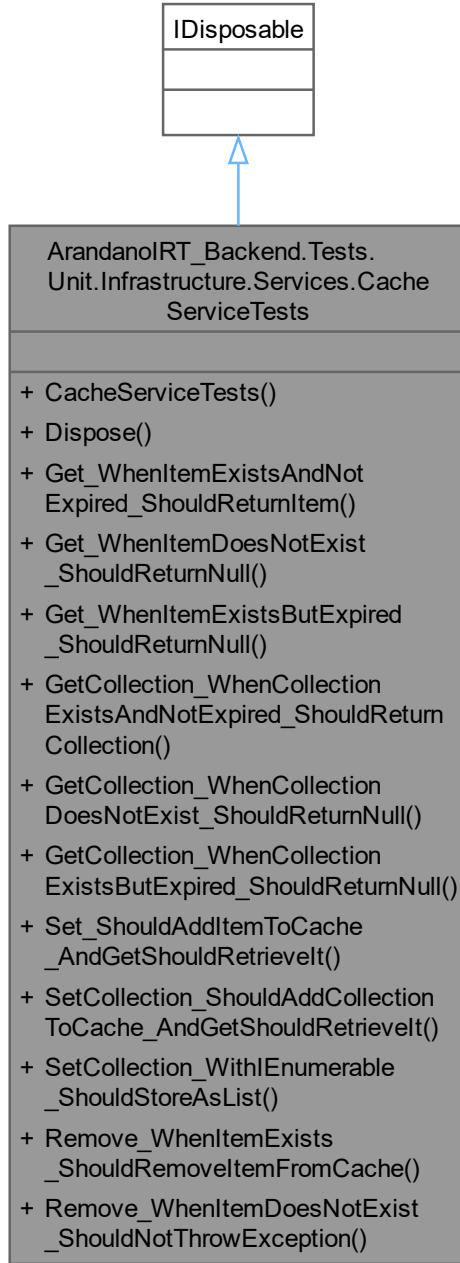
6.42 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests Class Reference

Contains unit tests for the CacheService class.

Inheritance diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests:



Public Member Functions

- [CacheServiceTests \(\)](#)

- Initializes a new instance of the CacheServiceTests class.*
- void [Dispose\(\)](#)
Cleans up the MemoryCache instance after each test run by calling its Dispose method.
 - void [Get_WhenItemExistsAndNotExpired_ShouldReturnItem\(\)](#)
 - void [Get_WhenItemDoesNotExist_ShouldReturnNull\(\)](#)
 - async Task [Get_WhenItemExistsButExpired_ShouldReturnNull\(\)](#)
 - void [GetCollection_WhenCollectionExistsAndNotExpired_ShouldReturnCollection\(\)](#)
 - void [GetCollection_WhenCollectionDoesNotExist_ShouldReturnNull\(\)](#)
 - async Task [GetCollection_WhenCollectionExistsButExpired_ShouldReturnNull\(\)](#)
 - void [Set_ShouldAddItemToCache_AndGetShouldRetrievelt\(\)](#)
 - void [SetCollection_ShouldAddCollectionToCache_AndGetShouldRetrievelt\(\)](#)
 - void [SetCollection_WithIEnumerable_ShouldStoreAsList\(\)](#)
 - void [Remove_WhenItemExists_ShouldRemoveItemFromCache\(\)](#)
 - void [Remove_WhenItemDoesNotExist_ShouldNotThrowException\(\)](#)

6.42.1 Detailed Description

Contains unit tests for the CacheService class.

Uses a real MemoryCache instance for testing cache interactions. Implements IDisposable to ensure the cache is disposed after tests.

6.42.2 Constructor & Destructor Documentation

6.42.2.1 CacheServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.CacheServiceTests ( )  
[inline]
```

Initializes a new instance of the CacheServiceTests class.

Sets up the real MemoryCache instance and the CacheService.

6.42.3 Member Function Documentation

6.42.3.1 Dispose()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Dispose ( ) [inline]
```

Cleans up the MemoryCache instance after each test run by calling its Dispose method.

6.42.3.2 Get_WhenItemDoesNotExist_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Get_WhenItemDoesNotExist_ShouldReturnNull ( ) [inline]
```

6.42.3.3 Get_WhenItemExistsAndNotExpired_ShouldReturnItem()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Get_WhenItemExistsAndNotExpired_ShouldReturnItem () [inline]
```

6.42.3.4 Get_WhenItemExistsButExpired_ShouldReturnNull()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Get_WhenItemExistsButExpired_ShouldReturnNull () [inline]
```

6.42.3.5 GetCollection_WhenCollectionDoesNotExist_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.GetCollection_WhenCollectionDoesNotExist_ShouldReturnNull () [inline]
```

6.42.3.6 GetCollection_WhenCollectionExistsAndNotExpired_ShouldReturnCollection()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.GetCollection_WhenCollectionExistsAndNotExpired_ShouldReturnCollection () [inline]
```

6.42.3.7 GetCollection_WhenCollectionExistsButExpired_ShouldReturnNull()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.GetCollection_WhenCollectionExistsButExpired_ShouldReturnNull () [inline]
```

6.42.3.8 Remove_WhenItemDoesNotExist_ShouldNotThrowException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Remove_WhenItemDoesNotExist_ShouldNotThrowException () [inline]
```

6.42.3.9 Remove_WhenItemExists_ShouldRemoveItemFromCache()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Remove_WhenItemExists_ShouldRemoveItemFromCache () [inline]
```

6.42.3.10 Set_ShouldAddItemToCache_AndGetShouldRetrieveIt()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.Set_ShouldAddItemToCache_AndGetShouldRetrieveIt () [inline]
```

6.42.3.11 SetCollection_ShouldAddCollectionToCache_AndGetShouldRetrieveIt()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.SetCollection_←
ShouldAddCollectionToCache_AndGetShouldRetrieveIt ( ) [inline]
```

6.42.3.12 SetCollection_WithIEnumerable_ShouldStoreAsList()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests.SetCollection_With←
IEnumerable_ShouldStoreAsList ( ) [inline]
```

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[CacheServiceTests.cs](#)

6.43 ArandanoIRT_Backend.Infrastructure.Data.Changepassword Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Changepassword:

ArandanoIRT_Backend.Infrastructure. Data.Changepassword
+ Idchangepassword + Passwordresettoken + Resettokenexpiresat + Tokencreatedat + Personid + Person

Properties

- int **Idchangepassword** [get, set]
- string **Passwordresettoken** = null! [get, set]
- DateTime **Resettokenexpiresat** [get, set]
- DateTime **Tokencreatedat** [get, set]
- int? **Personid** [get, set]
- virtual ? Person **Person** [get, set]

6.43.1 Property Documentation

6.43.1.1 Idchangepassword

```
int ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Idchangepassword [get], [set]
```

6.43.1.2 Passwordresettoken

```
string ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Passwordresettoken = null! [get], [set]
```

6.43.1.3 Person

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Person [get], [set]
```

6.43.1.4 Personid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Personid [get], [set]
```

6.43.1.5 Resettokenexpiresat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Resettokenexpiresat [get], [set]
```

6.43.1.6 Tokencreatedat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Changepassword.Tokencreatedat [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Changepassword.cs](#)

6.44 ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity Class Reference

Represents a record containing a temporary token generated to allow a user to securely reset their password.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity:

ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity
<ul style="list-style-type: none"> + IdChangePassword + PasswordResetToken + ResetTokenExpiresAt + TokenCreatedAt + PersonId + ChangePasswordEntity()

Public Member Functions

- `ChangePasswordEntity` (int idChangePassword, string passwordResetToken, DateTime resetTokenExpiresAt, DateTime tokenCreatedAt, int? personId)

Initializes a new instance of the ChangePasswordEntity class with specific token details.

Properties

- int `IdChangePassword` [get]
Gets the unique identifier for this password change request record.
- string `PasswordResetToken` [get]
Gets the secure, unique token string sent to the user for password reset verification.
- DateTime `ResetTokenExpiresAt` [get]
Gets the date and time (usually UTC) when this password reset token expires and becomes invalid.
- DateTime `TokenCreatedAt` [get]
Gets the date and time (usually UTC) when this password reset token was originally created.
- int? `PersonId` [get]
Gets the unique identifier of the person (user) to whom this password reset token belongs.

6.44.1 Detailed Description

Represents a record containing a temporary token generated to allow a user to securely reset their password.

6.44.2 Constructor & Destructor Documentation

6.44.2.1 ChangePasswordEntity()

```
ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.ChangePasswordEntity (
    int idChangePassword,
    string passwordResetToken,
    DateTime resetTokenExpiresAt,
    DateTime tokenCreatedAt,
    int? personId ) [inline]
```

Initializes a new instance of the ChangePasswordEntity class with specific token details.

Parameters

<i>idChangePassword</i>	The unique ID for this record (often generated by the database).
<i>passwordResetToken</i>	The secure token string.
<i>resetTokenExpiresAt</i>	The expiration timestamp for the token.
<i>tokenCreatedAt</i>	The creation timestamp for the token.
<i>personId</i>	The ID of the user associated with this token.

6.44.3 Property Documentation

6.44.3.1 IdChangePassword

```
int ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.IdChangePassword [get]
```

Gets the unique identifier for this password change request record.

6.44.3.2 PasswordResetToken

```
string ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.PasswordResetToken [get]
```

Gets the secure, unique token string sent to the user for password reset verification.

6.44.3.3 PersonId

```
int? ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.PersonId [get]
```

Gets the unique identifier of the person (user) to whom this password reset token belongs.

6.44.3.4 ResetTokenExpiresAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.ResetTokenExpiresAt [get]
```

Gets the date and time (usually UTC) when this password reset token expires and becomes invalid.

6.44.3.5 TokenCreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity.TokenCreatedAt [get]
```

Gets the date and time (usually UTC) when this password reset token was originally created.

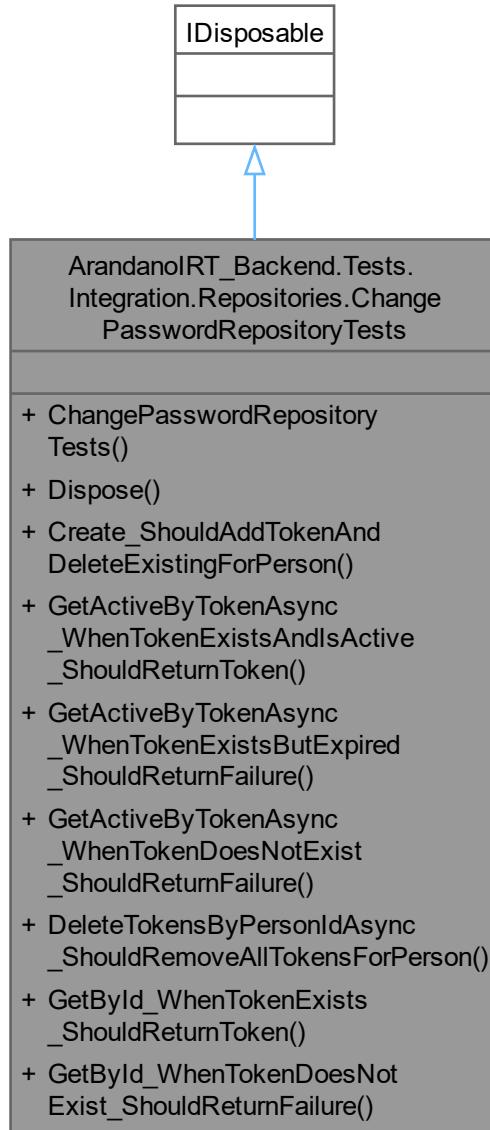
The documentation for this class was generated from the following file:

- Domain/Entities/[ChangePasswordEntity.cs](#)

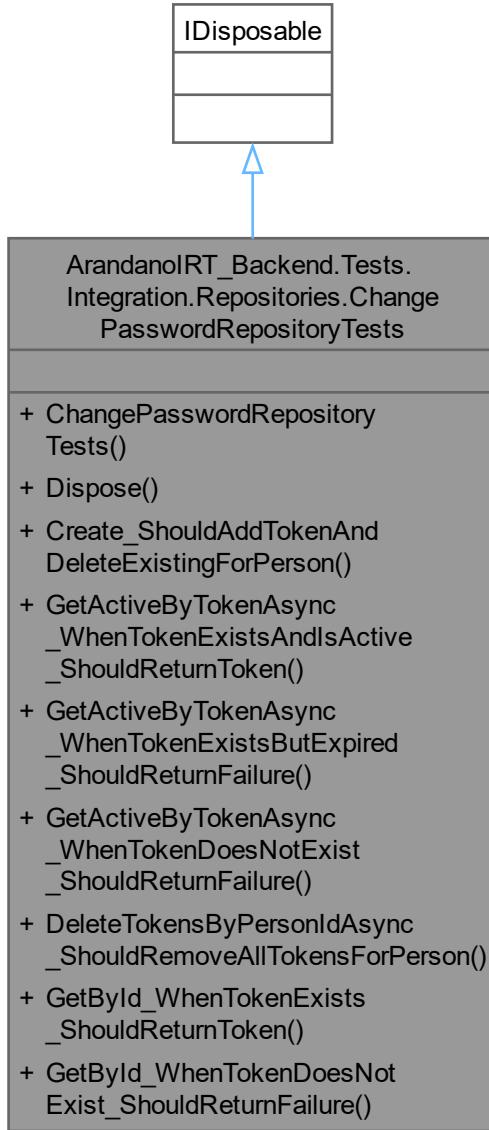
6.45 ArandanoIRT_Backend.Tests.Integration.Repositories.Change>PasswordRepositoryTests Class Reference

Contains integration tests for the ChangePasswordRepository using an SQLite in-memory database.

Inheritance diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests:



Public Member Functions

- [ChangePasswordRepositoryTests \(\)](#)

Initializes a new instance of the ChangePasswordRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

- [void Dispose \(\)](#)

Disposes the database context and connection used by the test instance.

- async Task [Create_ShouldAddTokenAndDeleteExistingForPerson \(\)](#)
- async Task [GetActiveByTokenAsync_WhenTokenExistsAndIsActive_ShouldReturnToken \(\)](#)
- async Task [GetActiveByTokenAsync_WhenTokenExistsButExpired_ShouldReturnFailure \(\)](#)
- async Task [GetActiveByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure \(\)](#)
- async Task [DeleteTokensByPersonIdAsync_ShouldRemoveAllTokensForPerson \(\)](#)
- async Task [GetById_WhenTokenExists_ShouldReturnToken \(\)](#)
- async Task [GetById_WhenTokenDoesNotExist_ShouldReturnFailure \(\)](#)

6.45.1 Detailed Description

Contains integration tests for the ChangePasswordRepository using an SQLite in-memory database.

6.45.2 Constructor & Destructor Documentation

6.45.2.1 ChangePasswordRepositoryTests()

```
ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.ChangePassword←  
RepositoryTests ( ) [inline]
```

Initializes a new instance of the ChangePasswordRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

6.45.3 Member Function Documentation

6.45.3.1 Create_ShouldAddTokenAndDeleteExistingForPerson()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Create←  
_ShouldAddTokenAndDeleteExistingForPerson ( ) [inline]
```

6.45.3.2 DeleteTokensByPersonIdAsync_ShouldRemoveAllTokensForPerson()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Delete←  
TokensByPersonIdAsync_ShouldRemoveAllTokensForPerson ( ) [inline]
```

6.45.3.3 Dispose()

```
void ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Dispose ( )  
[inline]
```

Disposes the database context and connection used by the test instance.

6.45.3.4 GetActiveByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Get←
ActiveByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure ( ) [inline]
```

6.45.3.5 GetActiveByTokenAsync_WhenTokenExistsAndIsActive_ShouldReturnToken()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Get←
ActiveByTokenAsync_WhenTokenExistsAndIsActive_ShouldReturnToken ( ) [inline]
```

6.45.3.6 GetActiveByTokenAsync_WhenTokenExistsButExpired_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Get←
ActiveByTokenAsync_WhenTokenExistsButExpired_ShouldReturnFailure ( ) [inline]
```

6.45.3.7 GetById_WhenTokenDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Get←
ById_WhenTokenDoesNotExist_ShouldReturnFailure ( ) [inline]
```

6.45.3.8 GetById_WhenTokenExists_ShouldReturnToken()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests.Get←
ById_WhenTokenExists_ShouldReturnToken ( ) [inline]
```

The documentation for this class was generated from the following file:

- Tests/Integration/Repositories/[ChangePasswordRepositoryTests.cs](#)

6.46 ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto Class Reference

Represents the request data for changing a user's password using a verification code.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto:

ArandanoIRT_Backend.Application. DTOs.Auth.ChangePasswordRequestDto
<ul style="list-style-type: none"> + ChangeCode + NewPassword + ConfirmPassword

Properties

- required string [ChangeCode](#) [get, set]
Gets or sets the unique code required to authorize the password change.
- required string [NewPassword](#) [get, set]
Gets or sets the desired new password for the user account.
- required string [ConfirmPassword](#) [get, set]
Gets or sets the confirmation of the new password.

6.46.1 Detailed Description

Represents the request data for changing a user's password using a verification code.

6.46.2 Property Documentation

6.46.2.1 ChangeCode

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto.ChangeCode  
[get], [set]
```

Gets or sets the unique code required to authorize the password change.

This code is typically sent to the user via email or another secure channel.

6.46.2.2 ConfirmPassword

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto.ConfirmPassword  
[get], [set]
```

Gets or sets the confirmation of the new password.

This field is used to ensure the user typed the new password correctly.

string

6.46.2.3 NewPassword

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto.NewPassword  
[get], [set]
```

Gets or sets the desired new password for the user account.

string

The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[ChangePasswordRequestDto.cs](#)

6.47 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests Class Reference

Contains unit tests for the [ClientInfo](#) value object.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests:

ArandanoIRT_Backend.Tests. Unit.Domain.ValueObjects.Client InfoTests
+ ToFormattedString_Should ReturnAgentAndOSWithVersions AndDevice_WhenAllProvided() + ToFormattedString_Should ReturnAgentAndDevice_WhenOSIs Unknown() + ToFormattedString_Should ReturnOSAndDevice_WhenAgentIs Unknown() + ToFormattedString_Should ReturnAgentFamilyAndDevice _WhenAgentVersionAndOSAreMissing() + ToFormattedString_Should ReturnOSFamilyAndDevice _WhenOSVersionAndAgentAreMissing() + ToFormattedString_Should ReturnDeviceFamily_WhenAgent AndOSAreUnknown() + ToFormattedString_Should ReturnUnknownDevice_WhenAllInfo IsUnknown() + ToFormattedString_Should TrimWhitespaceAndIncludeDevice _WhenPresent() + ToFormattedString_Should OmitDeviceFamilyOther _WhenAgentOrOSIsKnown()

Public Member Functions

- void [ToFormattedString_ShouldReturnAgentAndOSWithVersionsAndDevice_WhenAllProvided \(\)](#)

Tests the ToFormattedString method returns the expected combined string when all properties (User Agent, OS with versions, and a significant Device) are provided.

- void [ToFormattedString_ShouldReturnAgentAndDevice_WhenOSIsUnknown \(\)](#)
Tests the ToFormattedString method returns User Agent (Family and Version) and Device Family when OS details are unknown (default).
- void [ToFormattedString_ShouldReturnOSAndDevice_WhenAgentIsUnknown \(\)](#)
Tests the ToFormattedString method returns OS (Family and Version) and Device Family when User Agent details are unknown (default).
- void [ToFormattedString_ShouldReturnAgentFamilyAndDevice_WhenAgentVersionAndOSAreMissing \(\)](#)
Tests the ToFormattedString method returns the User Agent Family and Device Family when the agent version is missing and OS details are unknown.
- void [ToFormattedString_ShouldReturnOSFamilyAndDevice_WhenOSVersionAndAgentAreMissing \(\)](#)
Tests the ToFormattedString method returns the OS Family and Device Family when the OS version is missing and User Agent details are unknown.
- void [ToFormattedString_ShouldReturnDeviceFamily_WhenAgentAndOSAreUnknown \(\)](#)
Tests the ToFormattedString method returns only the Device Family when both User Agent and OS details are unknown or default.
- void [ToFormattedString_ShouldReturnUnknownDevice_WhenAllInfoIsUnknown \(\)](#)
Tests the ToFormattedString method returns the ultimate fallback "Unknown Device" when all User Agent, OS, and Device Family details are unknown or default.
- void [ToFormattedString_ShouldTrimWhitespaceAndIncludeDevice_WhenPresent \(\)](#)
Tests that the ToFormattedString method handles whitespace in version strings correctly by trimming them, and includes the significant Device Family.
- void [ToFormattedString_ShouldOmitDeviceFamilyOther_WhenAgentOrOSIsKnown \(\)](#)
Tests that the ToFormattedString method omits the Device Family when it is "Other" and either the User Agent or OS information is known (not default/unknown).

6.47.1 Detailed Description

Contains unit tests for the [ClientInfo](#) value object.

Focuses on the logic of the ToFormattedString method.

6.47.2 Member Function Documentation

6.47.2.1 ToFormattedString_ShouldOmitDeviceFamilyOther_WhenAgentOrOSIsKnown()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←
OmitDeviceFamilyOther_WhenAgentOrOSIsKnown ( ) [inline]
```

Tests that the ToFormattedString method omits the Device Family when it is "Other" and either the User Agent or OS information is known (not default/unknown).

6.47.2.2 ToFormattedString_ShouldReturnAgentAndDevice_WhenOSIsUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←
ReturnAgentAndDevice_WhenOSIsUnknown ( ) [inline]
```

Tests the ToFormattedString method returns User Agent (Family and Version) and Device Family when OS details are unknown (default).

6.47.2.3 ToFormattedString_ShouldReturnAgentAndOSWithVersionsAndDevice_WhenAllProvided()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnAgentAndOSWithVersionsAndDevice_WhenAllProvided ( ) [inline]
```

Tests the ToFormattedString method returns the expected combined string when all properties (User Agent, OS with versions, and a significant Device) are provided.

6.47.2.4 ToFormattedString_ShouldReturnAgentFamilyAndDevice_WhenAgentVersionAndOSAreMissing()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnAgentFamilyAndDevice_WhenAgentVersionAndOSAreMissing ( ) [inline]
```

Tests the ToFormattedString method returns the User Agent Family and Device Family when the agent version is missing and OS details are unknown.

6.47.2.5 ToFormattedString_ShouldReturnDeviceFamily_WhenAgentAndOSAreUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnDeviceFamily_WhenAgentAndOSAreUnknown ( ) [inline]
```

Tests the ToFormattedString method returns only the Device Family when both User Agent and OS details are unknown or default.

6.47.2.6 ToFormattedString_ShouldReturnOSAndDevice_WhenAgentIsUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnOSAndDevice_WhenAgentIsUnknown ( ) [inline]
```

Tests the ToFormattedString method returns OS (Family and Version) and Device Family when User Agent details are unknown (default).

6.47.2.7 ToFormattedString_ShouldReturnOSFamilyAndDevice_WhenOSVersionAndAgentAreMissing()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnOSFamilyAndDevice_WhenOSVersionAndAgentAreMissing ( ) [inline]
```

Tests the ToFormattedString method returns the OS Family and Device Family when the OS version is missing and User Agent details are unknown.

6.47.2.8 ToFormattedString_ShouldReturnUnknownDevice_WhenAllInfoIsUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_Should←  
ReturnUnknownDevice_WhenAllInfoIsUnknown ( ) [inline]
```

Tests the ToFormattedString method returns the ultimate fallback "Unknown Device" when all User Agent, OS, and Device Family details are unknown or default.

6.47.2.9 ToFormattedString_ShouldTrimWhitespaceAndIncludeDevice_WhenPresent()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests.ToFormattedString_ShouldTrimWhitespaceAndIncludeDevice_WhenPresent ( ) [inline]
```

Tests that the ToFormattedString method handles whitespace in version strings correctly by trimming them, and includes the significant Device Family.

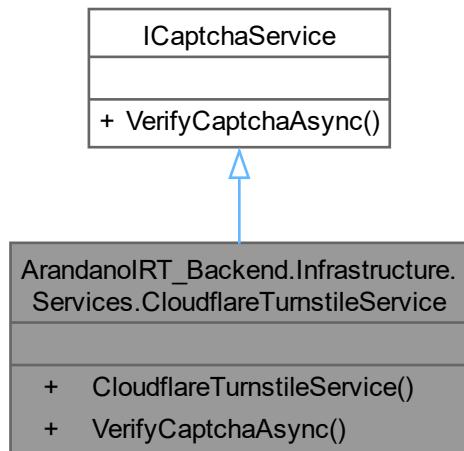
The documentation for this class was generated from the following file:

- Tests/Unit/Domain/ValueObjects/[ClientInfoTests.cs](#)

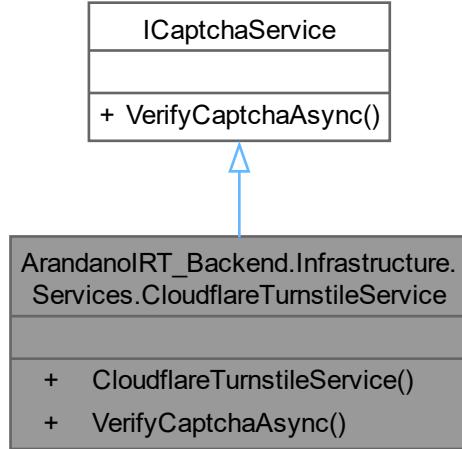
6.48 ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstile Service Class Reference

Implements the ICaptchaService interface using the Cloudflare Turnstile API for verifying CAPTCHA challenges.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService:



Public Member Functions

- [CloudflareTurnstileService \(IHttpClientFactory httpClientFactory, IConfiguration configuration, ILogger<CloudflareTurnstileService> logger\)](#)
Initializes a new instance of the CloudflareTurnstileService class.
- [async Task< Result > VerifyCaptchaAsync \(string token\)](#)
Verifies the CAPTCHA token received from the client.

Parameters

token	The CAPTCHA response token from the client.
-------	---

Returns

Result indicating success (verified) or failure.

6.48.1 Detailed Description

Implements the ICaptchaService interface using the Cloudflare Turnstile API for verifying CAPTCHA challenges.

6.48.2 Constructor & Destructor Documentation

6.48.2.1 CloudflareTurnstileService()

ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService.CloudflareTurnstileService
(

```
IHttpClientFactory httpClientFactory,
 IConfiguration configuration,
 ILogger< CloudflareTurnstileService > logger ) [inline]
```

Initializes a new instance of the CloudflareTurnstileService class.

Parameters

<i>httpClientFactory</i>	The factory for creating HttpClient instances.
<i>configuration</i>	The application configuration provider.
<i>logger</i>	The logger instance for this service.

Exceptions

<i>ArgumentNullException</i>	Thrown if httpClientFactory, configuration, or logger is null.
<i>InvalidOperationException</i>	Thrown if the 'Captcha:TurnstileSecretKey' configuration value is null or empty.

6.48.3 Member Function Documentation

6.48.3.1 VerifyCaptchaAsync()

```
async Task< Result > ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService.←
VerifyCaptchaAsync (
    string token ) [inline]
```

Verifies the CAPTCHA token received from the client.

Parameters

<i>token</i>	The CAPTCHA response token from the client.
--------------	---

Returns

Result indicating success (verified) or failure.

This method communicates with the Cloudflare Turnstile API endpoint (VerifyEndpoint). Its success depends on network connectivity, the availability of the Cloudflare service, and the correct configuration of the `Captcha:TurnstileSecretKey` setting. It uses cached JsonSerializerOptions for efficient response deserialization.

Implements [ArandanoIRT_Backend.Application.Interfaces.ICaptchaService](#).

The documentation for this class was generated from the following file:

- Infrastructure/Services/[CaptchaService.cs](#)

6.49 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests Class Reference

Contains unit tests for the CloudflareTurnstileService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests:

ArandanoIRT_Backend.Tests. Unit.Infrastructure.Services.Cloudflare TurnstileServiceTests
+ CloudflareTurnstileService Tests() + VerifyCaptchaAsync _WithValidTokenAndSuccessful Response_ShouldReturnSuccess() + VerifyCaptchaAsync _WithValidTokenAndFailureResponse _ShouldReturnFailureWithErrorCodes() + VerifyCaptchaAsync _WithHttpError_ShouldReturn Failure() + VerifyCaptchaAsync _WithNetworkError_ShouldReturn Failure() + VerifyCaptchaAsync _WithInvalidJsonResponse _ShouldReturnFailure() + VerifyCaptchaAsync _WithNullToken_ShouldReturn Failure() + VerifyCaptchaAsync _WithEmptyToken_ShouldReturn Failure() + VerifyCaptchaAsync _WithWhitespaceToken _ShouldReturnFailure()

Public Member Functions

- [CloudflareTurnstileServiceTests \(\)](#)

Initializes a new instance of the CloudflareTurnstileServiceTests class, setting up the necessary mocks for configuration, HTTP client factory, and message handler.

- `async Task VerifyCaptchaAsync_WithValidTokenAndSuccessfulResponse_ShouldReturnSuccess ()`
Tests that VerifyCaptchaAsync returns success when a valid token is provided and the mocked Cloudflare response indicates success.
- `async Task VerifyCaptchaAsync_WithValidTokenAndFailureResponse_ShouldReturnFailureWithErrorCodes ()`
Tests that VerifyCaptchaAsync returns failure when the Cloudflare response indicates failure (success: false) and includes error codes.
- `async Task VerifyCaptchaAsync_WithHttpError_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure when the HTTP request to Cloudflare results in an unsuccessful status code (e.g., 500 Internal Server Error).
- `async Task VerifyCaptchaAsync_WithNetworkError_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure when a network error (simulated by an HttpRequestException) occurs during the HTTP request.
- `async Task VerifyCaptchaAsync_WithInvalidJsonResponse_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure when the response body from Cloudflare is not valid JSON, preventing successful deserialization.
- `async Task VerifyCaptchaAsync_WithNullToken_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure immediately if the provided token is null, without making an HTTP call.
- `async Task VerifyCaptchaAsync_WithEmptyToken_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure immediately if the provided token is empty, without making an HTTP call.
- `async Task VerifyCaptchaAsync_WithWhitespaceToken_ShouldReturnFailure ()`
Tests that VerifyCaptchaAsync returns failure immediately if the provided token is whitespace, without making an HTTP call.

6.49.1 Detailed Description

Contains unit tests for the CloudflareTurnstileService class.

6.49.2 Constructor & Destructor Documentation

6.49.2.1 CloudflareTurnstileServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests.CloudflareTurnstileServiceTests ( ) [inline]
```

Initializes a new instance of the CloudflareTurnstileServiceTests class, setting up the necessary mocks for configuration, HTTP client factory, and message handler.

6.49.3 Member Function Documentation

6.49.3.1 VerifyCaptchaAsync_WithEmptyToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests.VerifyCaptchaAsync_WithEmptyToken_ShouldReturnFailure ( ) [inline]
```

Tests that VerifyCaptchaAsync returns failure immediately if the provided token is empty, without making an HTTP call.

6.49.3.2 VerifyCaptchaAsync_WithHttpError_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithHttpError_ShouldReturnFailure () [inline]
```

Tests that VerifyCaptchaAsync returns failure when the HTTP request to Cloudflare results in an unsuccessful status code (e.g., 500 Internal Server Error).

6.49.3.3 VerifyCaptchaAsync_WithInvalidJsonResponse_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithInvalidJsonResponse_ShouldReturnFailure () [inline]
```

Tests that VerifyCaptchaAsync returns failure when the response body from Cloudflare is not valid JSON, preventing successful deserialization.

6.49.3.4 VerifyCaptchaAsync_WithNetworkError_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithNetworkError_ShouldReturnFailure () [inline]
```

Tests that VerifyCaptchaAsync returns failure when a network error (simulated by an HttpRequestException) occurs during the HTTP request.

6.49.3.5 VerifyCaptchaAsync_WithNullToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithNullToken_ShouldReturnFailure () [inline]
```

Tests that VerifyCaptchaAsync returns failure immediately if the provided token is null, without making an HTTP call.

6.49.3.6 VerifyCaptchaAsync_WithValidTokenAndFailureResponse_ShouldReturnFailureWithErrorCodes()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithValidTokenAndFailureResponse_ShouldReturnFailureWithErrorCodes () [inline]
```

Tests that VerifyCaptchaAsync returns failure when the Cloudflare response indicates failure (success: false) and includes error codes.

6.49.3.7 VerifyCaptchaAsync_WithValidTokenAndSuccessfulResponse_ShouldReturnSuccess()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithValidTokenAndSuccessfulResponse_ShouldReturnSuccess () [inline]
```

Tests that VerifyCaptchaAsync returns success when a valid token is provided and the mocked Cloudflare response indicates success.

6.49.3.8 VerifyCaptchaAsync_WithWhitespaceToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileService<-->
Tests.VerifyCaptchaAsync_WithWhitespaceToken_ShouldReturnFailure ( ) [inline]
```

Tests that VerifyCaptchaAsync returns failure immediately if the provided token is whitespace, without making an HTTP call.

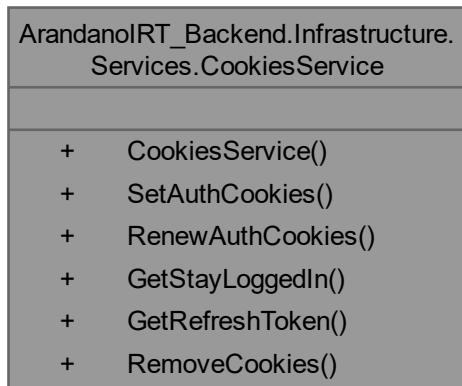
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[CloudflareTurnstileService Tests.cs](#)

6.50 ArandanoIRT_Backend.Infrastructure.Services.CookiesService Class Reference

Provides methods for managing authentication-related HTTP cookies, such as setting, renewing, retrieving, and removing JWT, Refresh Token, and StayLoggedIn cookies.

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.CookiesService:



Public Member Functions

- [CookiesService](#) (IConfiguration configuration)
Initializes a new instance of the CookiesService class.
- void [SetAuthCookies](#) (HttpContext context, string jwtToken, string refreshToken, bool stayLoggedIn)
Sets the authentication cookies (jwt, refreshToken, stayLoggedIn) in the HTTP response.
- void [RenewAuthCookies](#) (HttpContext context, string jwtToken, string refreshToken)

- *Renews the JWT and Refresh Token cookies using the existing 'stayLoggedIn' preference from the request cookies.*
- bool [GetStayLoggedIn](#) (HttpContext context)
 - Reads the 'stayLoggedIn' cookie from the request and parses its boolean value.*
- string? [GetRefreshToken](#) (HttpContext context)
 - Reads the 'refreshToken' cookie value from the HTTP request.*
- void [RemoveCookies](#) (HttpContext context)
 - Removes the authentication cookies (jwt, refreshToken, stayLoggedIn) from the HTTP response by setting their value to empty and their expiration date to the past.*

6.50.1 Detailed Description

Provides methods for managing authentication-related HTTP cookies, such as setting, renewing, retrieving, and removing JWT, Refresh Token, and StayLoggedIn cookies.

6.50.2 Constructor & Destructor Documentation

6.50.2.1 CookiesService()

```
ArandanoIRT_Backend.Infrastructure.Services.CookiesService.CookiesService (
    IConfiguration configuration ) [inline]
```

Initializes a new instance of the CookiesService class.

Reads the cookie domain configuration setting.

Parameters

<i>configuration</i>	The application configuration provider.
----------------------	---

6.50.3 Member Function Documentation

6.50.3.1 GetRefreshToken()

```
string? ArandanoIRT_Backend.Infrastructure.Services.CookiesService.GetRefreshToken (
    HttpContext context ) [inline]
```

Reads the 'refreshToken' cookie value from the HTTP request.

Parameters

<i>context</i>	The current HttpContext.
----------------	--------------------------

Returns

The refresh token string if the cookie exists; otherwise, null.

6.50.3.2 GetStayLoggedIn()

```
bool ArandanoIRT_Backend.Infrastructure.Services.CookiesService.GetStayLoggedIn (
    HttpContext context ) [inline]
```

Reads the 'stayLoggedIn' cookie from the request and parses its boolean value.

Parameters

<i>context</i>	The current HttpContext.
----------------	--------------------------

Returns

true if the 'stayLoggedIn' cookie exists and its value parses to true (case-insensitive); otherwise, false.

6.50.3.3 RemoveCookies()

```
void ArandanoIRT_Backend.Infrastructure.Services.CookiesService.RemoveCookies (
    HttpContext context ) [inline]
```

Removes the authentication cookies (jwt, refreshToken, stayLoggedIn) from the HTTP response by setting their value to empty and their expiration date to the past.

Parameters

<i>context</i>	The current HttpContext.
----------------	--------------------------

6.50.3.4 RenewAuthCookies()

```
void ArandanoIRT_Backend.Infrastructure.Services.CookiesService.RenewAuthCookies (
    HttpContext context,
    string jwtToken,
    string refreshToken ) [inline]
```

Renews the JWT and Refresh Token cookies using the existing 'stayLoggedIn' preference from the request cookies.

Parameters

<i>context</i>	The current HttpContext.
<i>jwtToken</i>	The new JWT access token string.
<i>refreshToken</i>	The new refresh token string.

6.50.3.5 SetAuthCookies()

```
void ArandanoIRT_Backend.Infrastructure.Services.CookiesService.SetAuthCookies (
    HttpContext context,
    string jwtToken,
    string refreshToken,
    bool stayLoggedIn ) [inline]
```

Sets the authentication cookies (jwt, refreshToken, stayLoggedIn) in the HTTP response.

Parameters

<i>context</i>	The current HttpContext.
<i>jwtToken</i>	The JWT access token string.
<i>refreshToken</i>	The refresh token string.
<i>stayLoggedIn</i>	Flag indicating if the session should be persistent (sets cookie expiry).

Cookies are configured as HttpOnly, Secure, and SameSite=None. SameSite=None requires Secure=true and implies cross-site usage scenarios. Expiry is set only if *stayLoggedIn* is true, otherwise they are session cookies.

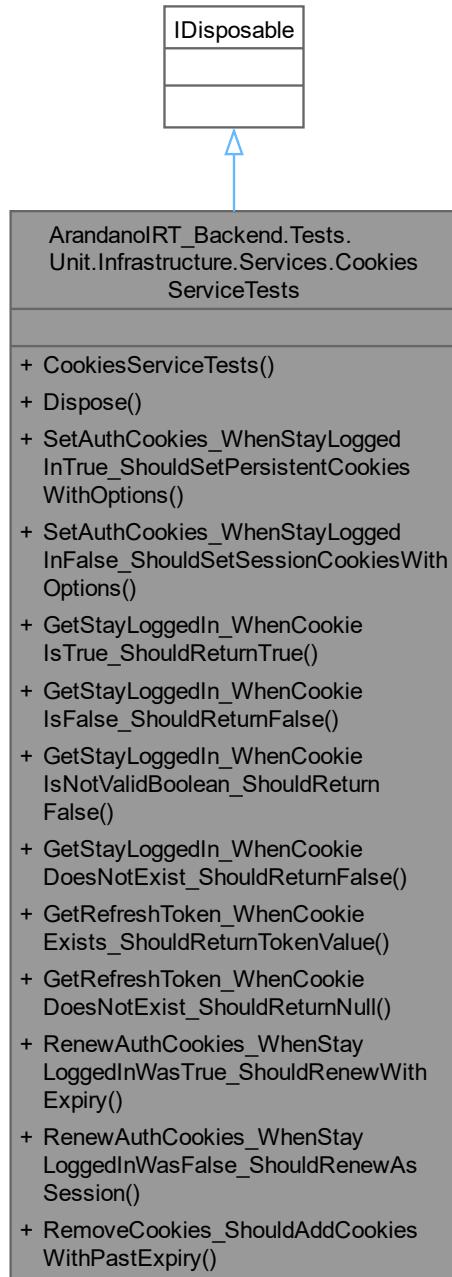
The documentation for this class was generated from the following file:

- Infrastructure/Services/[CookiesService.cs](#)

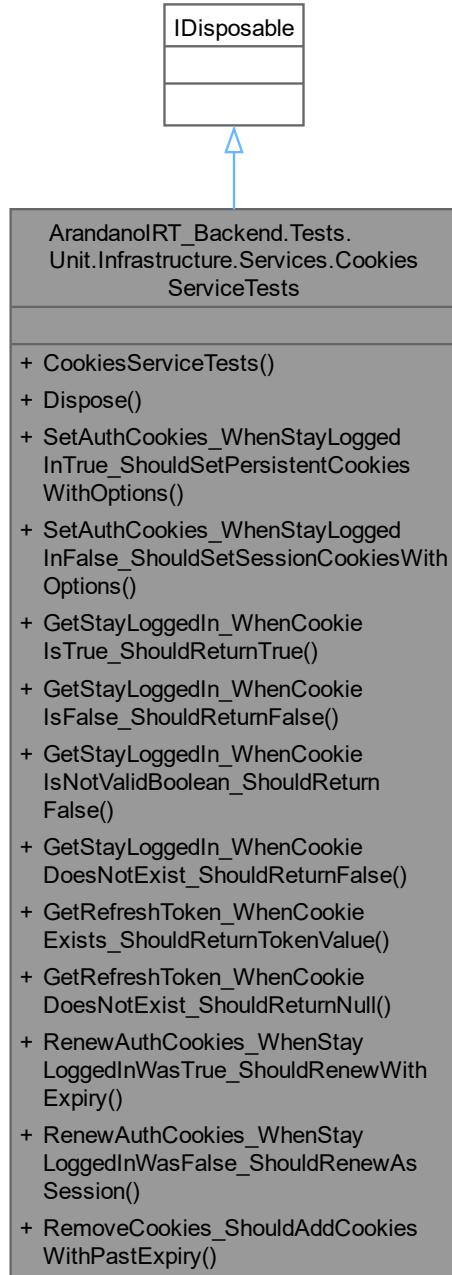
6.51 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests Class Reference

Contains unit tests for the CookiesService class.

Inheritance diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests:



Public Member Functions

- [CookiesServiceTests \(\)](#)

Initializes a new instance of the CookiesServiceTests class.

- void [Dispose\(\)](#)
Disposes resources if necessary.
- void [SetAuthCookies_WhenStayLoggedInTrue_ShouldSetPersistentCookiesWithOptions\(\)](#)
- void [SetAuthCookies_WhenStayLoggedInFalse_ShouldSetSessionCookiesWithOptions\(\)](#)
- void [GetStayLoggedIn_WhenCookieIsTrue_ShouldReturnTrue\(\)](#)
- void [GetStayLoggedIn_WhenCookieIsFalse_ShouldReturnFalse\(\)](#)
- void [GetStayLoggedIn_WhenCookieIsNotValidBoolean_ShouldReturnFalse\(string cookieValue\)](#)
- void [GetStayLoggedIn_WhenCookieDoesNotExist_ShouldReturnFalse\(\)](#)
- void [GetRefreshToken_WhenCookieExists_ShouldReturnTokenValue\(\)](#)
- void [GetRefreshToken_WhenCookieDoesNotExist_ShouldReturnNull\(\)](#)
- void [RenewAuthCookies_WhenStayLoggedInWasTrue_ShouldRenewWithExpiry\(\)](#)
- void [RenewAuthCookies_WhenStayLoggedInWasFalse_ShouldRenewAsSession\(\)](#)
- void [RemoveCookies_ShouldAddCookiesWithPastExpiry\(\)](#)

6.51.1 Detailed Description

Contains unit tests for the CookiesService class.

Uses DefaultHttpContext and mocks IConfiguration and IRequestCookieCollection for controlled testing. Implements IDisposable although no unmanaged resources are directly used here.

6.51.2 Constructor & Destructor Documentation

6.51.2.1 CookiesServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.CookiesServiceTests ( )  
[inline]
```

Initializes a new instance of the CookiesServiceTests class.

Sets up mocks for dependencies and creates a fresh HttpContext for each test.

6.51.3 Member Function Documentation

6.51.3.1 Dispose()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.Dispose ( ) [inline]
```

Disposes resources if necessary.

Currently no explicit unmanaged resources.

6.51.3.2 GetRefreshToken_WhenCookieDoesNotExist_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetRefreshToken_←  
WhenCookieDoesNotExist_ShouldReturnNull ( ) [inline]
```

6.51.3.3 GetRefreshToken_WhenCookieExists_ShouldReturnTokenValue()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetRefreshToken_←  
WhenCookieExists_ShouldReturnTokenValue ( ) [inline]
```

6.51.3.4 GetStayLoggedIn_WhenCookieDoesNotExist_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetStayLoggedIn_←  
WhenCookieDoesNotExist_ShouldReturnFalse ( ) [inline]
```

6.51.3.5 GetStayLoggedIn_WhenCookieIsFalse_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetStayLoggedIn_←  
WhenCookieIsFalse_ShouldReturnFalse ( ) [inline]
```

6.51.3.6 GetStayLoggedIn_WhenCookieIsNotValidBoolean_ShouldReturnFalse()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetStayLoggedIn_←  
WhenCookieIsNotValidBoolean_ShouldReturnFalse (   
    string cookieValue ) [inline]
```

6.51.3.7 GetStayLoggedIn_WhenCookieIsTrue_ShouldReturnTrue()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.GetStayLoggedIn_←  
WhenCookieIsTrue_ShouldReturnTrue ( ) [inline]
```

6.51.3.8 RemoveCookies_ShouldAddCookiesWithPastExpiry()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.RemoveCookies_←  
ShouldAddCookiesWithPastExpiry ( ) [inline]
```

6.51.3.9 RenewAuthCookies_WhenStayLoggedInWasFalse_ShouldRenewAsSession()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.RenewAuthCookies_←  
_WhenStayLoggedInWasFalse_ShouldRenewAsSession ( ) [inline]
```

6.51.3.10 RenewAuthCookies_WhenStayLoggedInWasTrue_ShouldRenewWithExpiry()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.RenewAuthCookies_←  
_WhenStayLoggedInWasTrue_ShouldRenewWithExpiry ( ) [inline]
```

6.51.3.11 SetAuthCookies_WhenStayLoggedInFalse_ShouldSetSessionCookiesWithOptions()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.SetAuthCookies_←  
WhenStayLoggedInFalse_ShouldSetSessionCookiesWithOptions ( ) [inline]
```

6.51.3.12 SetAuthCookies_WhenStayLoggedInTrue_ShouldSetPersistentCookiesWithOptions()

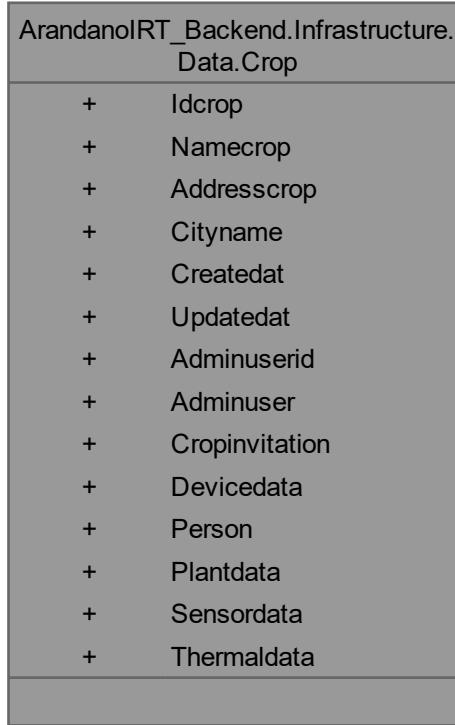
```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests.SetAuthCookies_←  
WhenStayLoggedInTrue_ShouldSetPersistentCookiesWithOptions ( ) [inline]
```

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[CookiesServiceTests.cs](#)

6.52 ArandanoIRT_Backend.Infrastructure.Data.Crop Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Crop:



Properties

- int **Idcrop** [get, set]
- string **Namecrop** = null! [get, set]
- string **Addresscrop** = null! [get, set]
- string **Cityname** = null! [get, set]
- DateTime **Createdat** [get, set]
- DateTime? **Updatedat** [get, set]
- int? **Adminuserid** [get, set]
- virtual ? **Person Adminuser** [get, set]
- virtual ICollection< **Cropinvitation** > **Cropinvitation** = new List< **Cropinvitation** >() [get, set]
- virtual ICollection< **Devicedata** > **Devicedata** = new List< **Devicedata** >() [get, set]
- virtual ICollection< **Person** > **Person** = new List< **Person** >() [get, set]
- virtual ICollection< **Plantdata** > **Plantdata** = new List< **Plantdata** >() [get, set]
- virtual ICollection< **Sensordata** > **Sensordata** = new List< **Sensordata** >() [get, set]
- virtual ICollection< **Thermaldata** > **Thermaldata** = new List< **Thermaldata** >() [get, set]

6.52.1 Property Documentation

6.52.1.1 Addresscrop

```
string ArandanoIRT_Backend.Infrastructure.Data.Crop.Addresscrop = null! [get], [set]
```

6.52.1.2 Adminuser

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Crop.Adminuser [get], [set]
```

6.52.1.3 Adminuserid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Crop.Adminuserid [get], [set]
```

6.52.1.4 Cityname

```
string ArandanoIRT_Backend.Infrastructure.Data.Crop.Cityname = null! [get], [set]
```

6.52.1.5 Createdat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Crop.Createdat [get], [set]
```

6.52.1.6 Cropinvitation

```
virtual ICollection<Cropinvitation> ArandanoIRT_Backend.Infrastructure.Data.Crop.Cropinvitation =  
new List<Cropinvitation>() [get], [set]
```

6.52.1.7 Devicedata

```
virtual ICollection<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.Crop.Devicedata = new  
List<Devicedata>() [get], [set]
```

6.52.1.8 Idcrop

```
int ArandanoIRT_Backend.Infrastructure.Data.Crop.Idcrop [get], [set]
```

6.52.1.9 Namecrop

```
string ArandanoIRT_Backend.Infrastructure.Data.Crop.Namecrop = null! [get], [set]
```

6.52.1.10 Person

```
virtual ICollection<Person> ArandanoIRT_Backend.Infrastructure.Data.Crop.Person = new List<Person>()
[get], [set]
```

6.52.1.11 Plantdata

```
virtual ICollection<Plantdata> ArandanoIRT_Backend.Infrastructure.Data.Crop.Plantdata = new List<Plantdata>()
[get], [set]
```

6.52.1.12 Sensordata

```
virtual ICollection<Sensordata> ArandanoIRT_Backend.Infrastructure.Data.Crop.Sensordata = new
List<Sensordata>() [get], [set]
```

6.52.1.13 Thermaldata

```
virtual ICollection<Thermaldata> ArandanoIRT_Backend.Infrastructure.Data.Crop.Thermaldata = new
List<Thermaldata>() [get], [set]
```

6.52.1.14 Updatedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Crop.Updatedat [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Crop.cs](#)

6.53 ArandanoIRT_Backend.Domain.Entities.CropEntity Class Reference

Represents an agricultural crop entity, containing its identification, name, location, and administrative details.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.CropEntity:

ArandanoIRT_Backend.Domain.Entities.CropEntity
+ IdCrop + NameCrop + AddressCrop + CityName + CreatedAt + UpdatedAt + AdminUserId + CropEntity()

Public Member Functions

- `CropEntity` (int idCrop, string nameCrop, string addressCrop, string cityName, DateTime createdAt, int? adminUserId=null)
Initializes a new instance of the CropEntity class.

Properties

- int `IdCrop` [get]
Gets the unique identifier for the crop.
- string `NameCrop` [get]
Gets the name assigned to the crop.
- string `AddressCrop` [get]
Gets the physical address associated with the crop's location.
- string `CityName` [get]
Gets the name of the city where the crop is located.
- DateTime `CreatedAt` [get]
Gets the date and time (usually UTC) when the crop entity was created.
- DateTime? `UpdatedAt` [get]
Gets the date and time (usually UTC) when the crop entity was last updated.
- int? `AdminUserId` [get]
Gets the unique identifier of the administrator user primarily associated with this crop.

6.53.1 Detailed Description

Represents an agricultural crop entity, containing its identification, name, location, and administrative details.

6.53.2 Constructor & Destructor Documentation

6.53.2.1 CropEntity()

```
ArandanoIRT_Backend.Domain.Entities.CropEntity.CropEntity (
    int idCrop,
    string nameCrop,
    string addressCrop,
    string cityName,
    DateTime createdAt,
    int? adminUserId = null ) [inline]
```

Initializes a new instance of the CropEntity class.

Parameters

<i>idCrop</i>	The unique ID for the crop (often generated by the database).
<i>nameCrop</i>	The name of the crop.
<i>addressCrop</i>	The physical address of the crop.
<i>cityName</i>	The city where the crop is located.
<i>createdAt</i>	The creation timestamp.
<i>adminUserId</i>	The ID of the administrator user associated with the crop (optional, defaults to null).

Exceptions

<i>ArgumentException</i>	Thrown if <i>nameCrop</i> is null or whitespace (Spanish message: "El nombre del cultivo es requerido."). Thrown if <i>addressCrop</i> is null or whitespace (Spanish message: "La dirección es requerida."). Thrown if <i>cityName</i> is null or whitespace (Spanish message: "El nombre de la ciudad es requerido.").
--------------------------	--

6.53.3 Property Documentation

6.53.3.1 AddressCrop

```
string ArandanoIRT_Backend.Domain.Entities.CropEntity.AddressCrop [get]
```

Gets the physical address associated with the crop's location.

Note: The property name contains the typo 'Addres'.

6.53.3.2 AdminUserId

```
int? ArandanoIRT_Backend.Domain.Entities.CropEntity.AdminUserId [get]
```

Gets the unique identifier of the administrator user primarily associated with this crop.

Nullable.

6.53.3.3 CityName

```
string ArandanoIRT_Backend.Domain.Entities.CropEntity.CityName [get]
```

Gets the name of the city where the crop is located.

6.53.3.4 CreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.CropEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when the crop entity was created.

6.53.3.5 IdCrop

```
int ArandanoIRT_Backend.Domain.Entities.CropEntity.IdCrop [get]
```

Gets the unique identifier for the crop.

6.53.3.6 NameCrop

```
string ArandanoIRT_Backend.Domain.Entities.CropEntity.NameCrop [get]
```

Gets the name assigned to the crop.

6.53.3.7 UpdatedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.CropEntity.UpdatedAt [get]
```

Gets the date and time (usually UTC) when the crop entity was last updated.

Nullable if never updated.

The documentation for this class was generated from the following file:

- Domain/Entities/[CropEntity.cs](#)

6.54 ArandanoIRT_Backend.Application.DTOs.Objetcts.CropInfoDto Class Reference

Represents basic information about a specific crop.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Objetcts.CropInfoDto:

ArandanoIRT_Backend.Application. DTOs.Objetcts.CropInfoDto	
+	NameCrop
+	AddresCrop
+	UbicationCrop

Properties

- required string `NameCrop` [get, set]
Gets or sets the name assigned to the crop.
- required string `AddresCrop` [get, set]
Gets or sets the address associated with the crop's physical location.
- required string `UbicationCrop` [get, set]
Gets or sets the geographic location (e.g., coordinates, region) of the crop.

6.54.1 Detailed Description

Represents basic information about a specific crop.

6.54.2 Property Documentation

6.54.2.1 AddresCrop

```
required string ArandanoIRT_Backend.Application.DTOs.Objetcts.CropInfoDto.AddresCrop [get], [set]
```

Gets or sets the address associated with the crop's physical location.

6.54.2.2 NameCrop

```
required string ArandanoIRT_Backend.Application.DTOs.Objectcts.CropInfoDto.NameCrop [get], [set]
```

Gets or sets the name assigned to the crop.

string

6.54.2.3 UbicationCrop

```
required string ArandanoIRT_Backend.Application.DTOs.Objectcts.CropInfoDto.UbicationCrop [get], [set]
```

Gets or sets the geographic location (e.g., coordinates, region) of the crop.

The documentation for this class was generated from the following file:

- Application/DTOs/Objects/[CropInfoDto.cs](#)

6.55 ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation:

ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation	
+	Idcropinvitation
+	Accescode
+	Createdat
+	Expiresat
+	Statusid
+	Createdby
+	Usedby
+	Cropid
+	CreatedbyNavigation
+	Crop
+	Status
+	UsedbyNavigation

Properties

- int `Idcropinvitation` [get, set]
- string `Accescode` = null! [get, set]
- DateTime `Createdat` [get, set]
- DateTime `Expiresat` [get, set]
- int? `Statusid` [get, set]
- int? `Createdby` [get, set]
- int? `Usedby` [get, set]
- int? `Cropid` [get, set]
- virtual ? Person `CreatedbyNavigation` [get, set]
- virtual ? Crop `Crop` [get, set]
- virtual ? Status `Status` [get, set]
- virtual ? Person `UsedbyNavigation` [get, set]

6.55.1 Property Documentation

6.55.1.1 `Accescode`

```
string ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Accescode = null! [get], [set]
```

6.55.1.2 `Createdat`

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Createdat [get], [set]
```

6.55.1.3 `Createdby`

```
int? ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Createdby [get], [set]
```

6.55.1.4 `CreatedbyNavigation`

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.CreatedbyNavigation [get], [set]
```

6.55.1.5 `Crop`

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Crop [get], [set]
```

6.55.1.6 `Cropid`

```
int? ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Cropid [get], [set]
```

6.55.1.7 Expiresat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Expiresat [get], [set]
```

6.55.1.8 Idcropinvitation

```
int ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Idcropinvitation [get], [set]
```

6.55.1.9 Status

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Status [get], [set]
```

6.55.1.10 Statusid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Statusid [get], [set]
```

6.55.1.11 Usedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.Usedby [get], [set]
```

6.55.1.12 UsedbyNavigation

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation.UsedbyNavigation [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Cropinvitation.cs](#)

6.56 ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity Class Reference

Represents an invitation code record used to grant access or link a new user to a specific crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity:

ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity
<ul style="list-style-type: none"> + IdCropInvitation + AccessCode + CreatedAt + ExpiresAt + StatusId + CreatedBy + UsedBy + CropId + CropInvitationEntity()

Public Member Functions

- [CropInvitationEntity](#) (int idCropInvitation, string accessCode, DateTime createdAt, DateTime expiresAt, int? statusId, int? createdBy, int? usedBy, int cropId)

Initializes a new instance of the CropInvitationEntity class.

Properties

- int [IdCropInvitation](#) [get]
Gets the unique identifier for this crop invitation record.
- string [AccessCode](#) [get]
Gets the unique access code string associated with this invitation.
- DateTime [CreatedAt](#) [get]
Gets the date and time (usually UTC) when this invitation code was created.
- DateTime [ExpiresAt](#) [get]
Gets the date and time (usually UTC) when this invitation code expires and becomes invalid.
- int? [StatusId](#) [get]
Gets the identifier representing the current status of the invitation (e.g., Pending, Used, Expired).

- int? [CreatedBy](#) [get]
Gets the unique identifier of the user who created this invitation.
- int? [UsedBy](#) [get]
Gets the unique identifier of the user who used or accepted this invitation.
- int [CropId](#) [get]
Gets the unique identifier of the crop to which this invitation grants access.

6.56.1 Detailed Description

Represents an invitation code record used to grant access or link a new user to a specific crop.

6.56.2 Constructor & Destructor Documentation

6.56.2.1 CropInvitationEntity()

```
ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.CropInvitationEntity (
    int idCropInvitation,
    string accessCode,
    DateTime createdAt,
    DateTime expiresAt,
    int? statusId,
    int? createdBy,
    int? usedBy,
    int cropId ) [inline]
```

Initializes a new instance of the CropInvitationEntity class.

Parameters

<i>idCropInvitation</i>	The unique ID for this invitation record (often generated by the database).
<i>accessCode</i>	The unique access code string.
<i>createdAt</i>	The creation timestamp.
<i>expiresAt</i>	The expiration timestamp.
<i>statusId</i>	The initial status ID (nullable).
<i>createdBy</i>	The ID of the user creating the invitation (nullable).
<i>usedBy</i>	The ID of the user who used the invitation (nullable).
<i>cropId</i>	The ID of the crop this invitation relates to.

Exceptions

<i>ArgumentException</i>	Thrown if <i>accessCode</i> is null or whitespace (Spanish message: "El código de acceso es requerido.").
--------------------------	---

6.56.3 Property Documentation

6.56.3.1 AccessCode

```
string ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.AccessCode [get]
```

Gets the unique access code string associated with this invitation.

Note: The property name contains the typo 'Acces'.

6.56.3.2 CreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when this invitation code was created.

6.56.3.3 CreatedBy

```
int? ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.CreatedBy [get]
```

Gets the unique identifier of the user who created this invitation.

Nullable.

6.56.3.4 CropId

```
int ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.CropId [get]
```

Gets the unique identifier of the crop to which this invitation grants access.

6.56.3.5 ExpiresAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.ExpiresAt [get]
```

Gets the date and time (usually UTC) when this invitation code expires and becomes invalid.

6.56.3.6 IdCropInvitation

```
int ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.IdCropInvitation [get]
```

Gets the unique identifier for this crop invitation record.

6.56.3.7 StatusId

```
int? ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.StatusId [get]
```

Gets the identifier representing the current status of the invitation (e.g., Pending, Used, Expired).

Nullable.

6.56.3.8 UsedBy

```
int? ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity.UsedBy [get]
```

Gets the unique identifier of the user who used or accepted this invitation.

Nullable if unused.

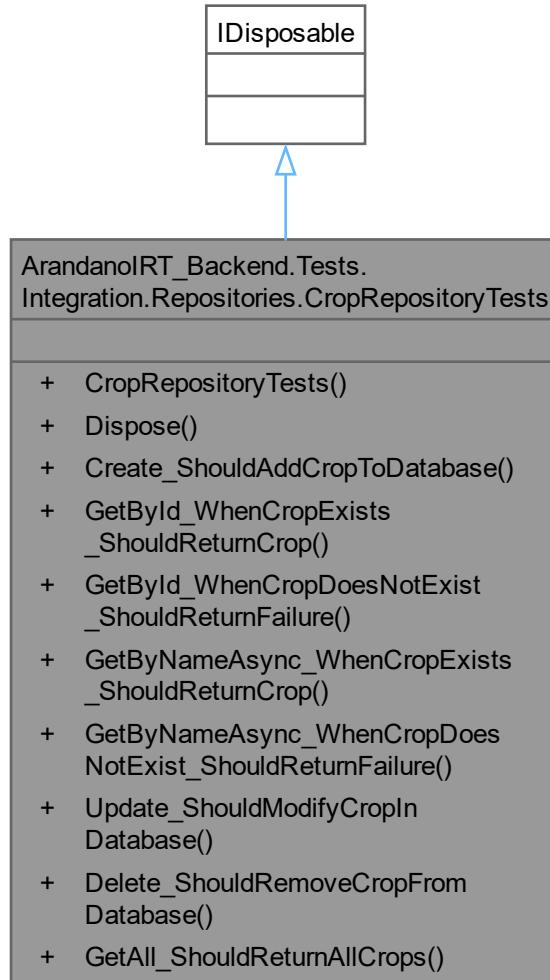
The documentation for this class was generated from the following file:

- Domain/Entities/[CropInvitationEntity.cs](#)

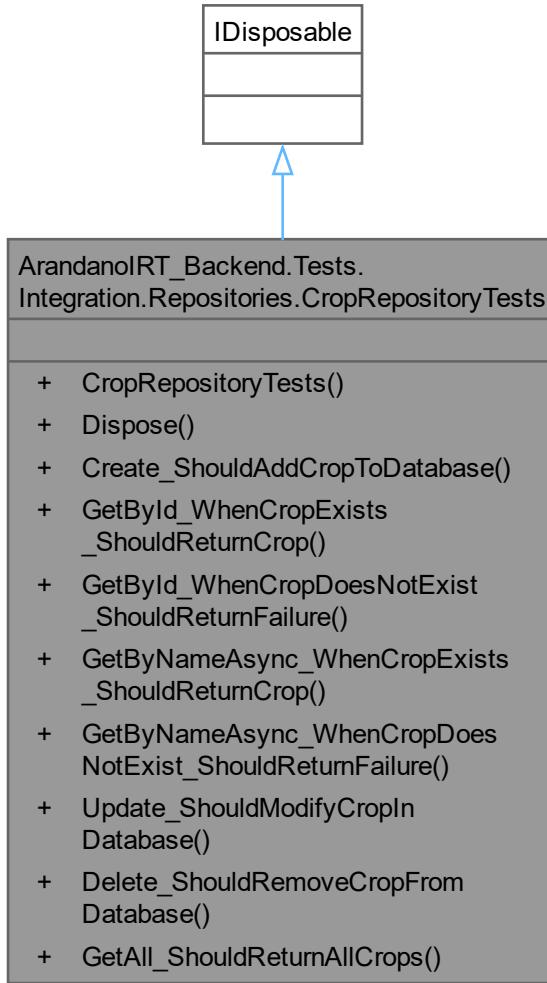
6.57 ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests Class Reference

Contains integration tests for the CropRepository using an SQLite in-memory database.

Inheritance diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests:



Public Member Functions

- [CropRepositoryTests \(\)](#)
Initializes a new instance of the `CropRepositoryTests` class, setting up the in-memory database, mocks, and repository under test.
- [void Dispose \(\)](#)
Disposes the database context and connection used by the test instance.
- [async Task Create_ShouldAddCropToDatabase \(\)](#)
- [async Task GetById_WhenCropExists_ShouldReturnCrop \(\)](#)
- [async Task GetById_WhenCropDoesNotExist_ShouldReturnFailure \(\)](#)
- [async Task GetByNameAsync_WhenCropExists_ShouldReturnCrop \(\)](#)

- async Task [GetByNameAsync_WhenCropDoesNotExist_ShouldReturnFailure\(\)](#)
- async Task [Update_ShouldModifyCropInDatabase\(\)](#)
- async Task [Delete_ShouldRemoveCropFromDatabase\(\)](#)
- async Task [GetAll_ShouldReturnAllCrops\(\)](#)

6.57.1 Detailed Description

Contains integration tests for the CropRepository using an SQLite in-memory database.

6.57.2 Constructor & Destructor Documentation

6.57.2.1 [CropRepositoryTests\(\)](#)

```
ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.CropRepositoryTests ( )  
[inline]
```

Initializes a new instance of the CropRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

6.57.3 Member Function Documentation

6.57.3.1 [Create_ShouldAddCropToDatabase\(\)](#)

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.Create_Should←  
AddCropToDatabase ( ) [inline]
```

6.57.3.2 [Delete_ShouldRemoveCropFromDatabase\(\)](#)

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.Delete_Should←  
RemoveCropFromDatabase ( ) [inline]
```

6.57.3.3 [Dispose\(\)](#)

```
void ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.Dispose ( ) [inline]
```

Disposes the database context and connection used by the test instance.

6.57.3.4 [GetAll_ShouldReturnAllCrops\(\)](#)

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.GetAll_Should←  
ReturnAllCrops ( ) [inline]
```

6.57.3.5 GetById_WhenCropDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.GetById_When←  
CropDoesNotExist_ShouldReturnFailure () [inline]
```

6.57.3.6 GetById_WhenCropExists_ShouldReturnCrop()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.GetById_When←  
CropExists_ShouldReturnCrop () [inline]
```

6.57.3.7 GetByNameAsync_WhenCropDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.GetByNameAsync←  
_WhenCropDoesNotExist_ShouldReturnFailure () [inline]
```

6.57.3.8 GetByNameAsync_WhenCropExists_ShouldReturnCrop()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.GetByNameAsync←  
_WhenCropExists_ShouldReturnCrop () [inline]
```

6.57.3.9 Update_ShouldModifyCropInDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests.Update_Should←  
ModifyCropInDatabase () [inline]
```

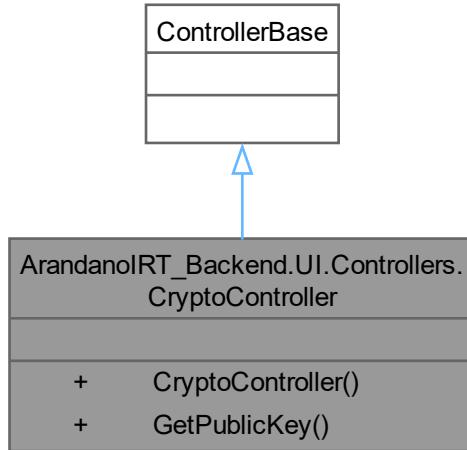
The documentation for this class was generated from the following file:

- Tests/Integration/Repositories/[CropRepositoryTests.cs](#)

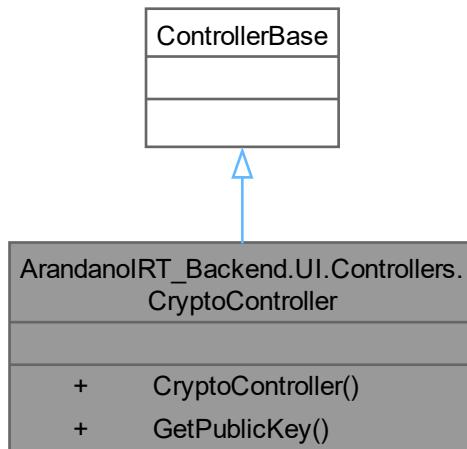
6.58 ArandanoIRT_Backend.UI.Controllers.CryptoController Class Reference

Handles cryptographic utility endpoints.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.CryptoController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.CryptoController:



Public Member Functions

- `CryptoController (IRsaService rsaService, ILogger<CryptoController> logger)`

Initializes a new instance of the CryptoController class.

- `IActionResult GetPublicKey()`

Retrieves the server's public RSA key in PEM format.

6.58.1 Detailed Description

Handles cryptographic utility endpoints.

6.58.2 Constructor & Destructor Documentation

6.58.2.1 CryptoController()

```
ArandanoIRT_Backend.UI.Controllers.CryptoController.CryptoController (
    IRsaService rsaService,
    ILogger< CryptoController > logger ) [inline]
```

Initializes a new instance of the CryptoController class.

Parameters

<code>rsaService</code>	The service for RSA cryptographic operations.
<code>logger</code>	The logger for this controller.

6.58.3 Member Function Documentation

6.58.3.1 GetPublicKey()

```
IActionResult ArandanoIRT_Backend.UI.Controllers.CryptoController.GetPublicKey ( ) [inline]
```

Retrieves the server's public RSA key in PEM format.

This key is used by clients (e.g., frontend) to encrypt sensitive data like passwords before sending it.

Returns

The public key as a PEM formatted string.

```
<response code="200">Public key retrieved successfully. Returns the PEM string.</response> <response code="500">Internal server error if the key cannot be retrieved.</response>
```

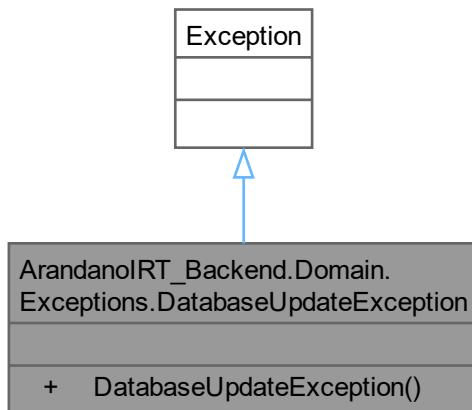
The documentation for this class was generated from the following file:

- UI/Controllers/[CryptoController.cs](#)

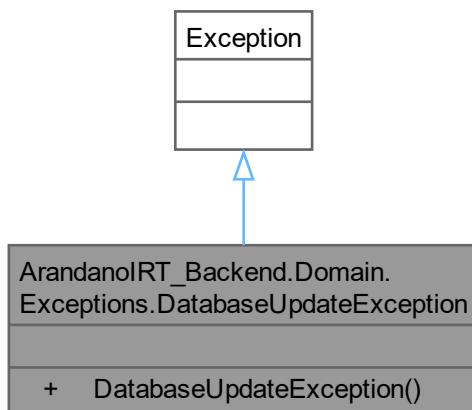
6.59 ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException Class Reference

Represents an error that occurs when a database update operation (e.g., insert, update, delete) fails unexpectedly or results in an inconsistent state.

Inheritance diagram for ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException:



Collaboration diagram for ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException:



Public Member Functions

- [DatabaseUpdateException \(string message\)](#)

Initializes a new instance of the DatabaseUpdateException class with a specified error message that describes the database update failure.

6.59.1 Detailed Description

Represents an error that occurs when a database update operation (e.g., insert, update, delete) fails unexpectedly or results in an inconsistent state.

This exception inherits from the base Exception class. Consider adding inner exception details in more complex scenarios if needed.

6.59.2 Constructor & Destructor Documentation

6.59.2.1 DatabaseUpdateException()

```
ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException.DatabaseUpdateException (
    string message ) [inline]
```

Initializes a new instance of the DatabaseUpdateException class with a specified error message that describes the database update failure.

Parameters

<i>message</i>	The message that describes the error encountered during the database update.
----------------	--

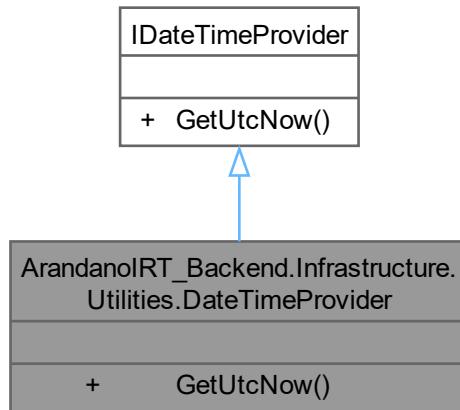
The documentation for this class was generated from the following file:

- Domain/Exceptions/[DatabaseUpdateException.cs](#)

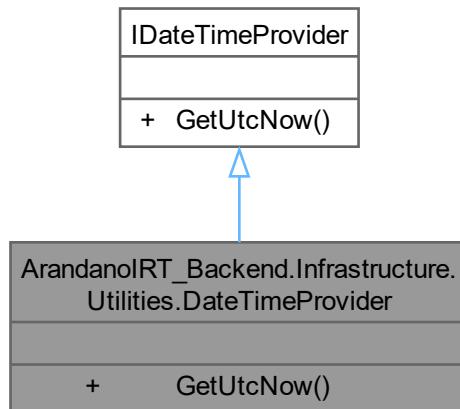
6.60 ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider Class Reference

Implements the `IDateTimeProvider` interface.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider:



Public Member Functions

- DateTime [GetUtcNow \(\)](#)
Gets the current UTC DateTime.

6.60.1 Detailed Description

Implements the `IDateTimeProvider` interface.

Provides a mechanism to retrieve the current Coordinated Universal Time (UTC), primarily facilitating testability for components dependent on the current time.

6.60.2 Member Function Documentation

6.60.2.1 GetUtcNow()

```
DateTime ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider.GetUtcNow () [inline]
```

Gets the current UTC DateTime.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider](#).

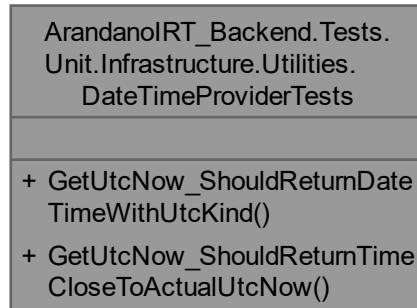
The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[DateTimeProvider.cs](#)

6.61 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests Class Reference

Contains unit tests for the `DateTimeProvider` class.

Collaboration diagram for `ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests`:



Public Member Functions

- void [GetUtcNow_ShouldReturnDateTimeWithUtcKind \(\)](#)
Tests that GetUtcNow returns a DateTime object whose Kind property is set to Utc.
- void [GetUtcNow_ShouldReturnTimeCloseToActualUtcNow \(\)](#)
Tests that GetUtcNow returns a DateTime value that is close to the actual current UTC time, allowing for minor execution delays.

6.61.1 Detailed Description

Contains unit tests for the DateTimeProvider class.

6.61.2 Member Function Documentation

6.61.2.1 [GetUtcNow_ShouldReturnDateTimeWithUtcKind\(\)](#)

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests.GetUtcNow_←  
ShouldReturnDateTimeWithUtcKind ( ) [inline]
```

Tests that GetUtcNow returns a DateTime object whose Kind property is set to Utc.

6.61.2.2 [GetUtcNow_ShouldReturnTimeCloseToActualUtcNow\(\)](#)

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests.GetUtcNow_←  
ShouldReturnTimeCloseToActualUtcNow ( ) [inline]
```

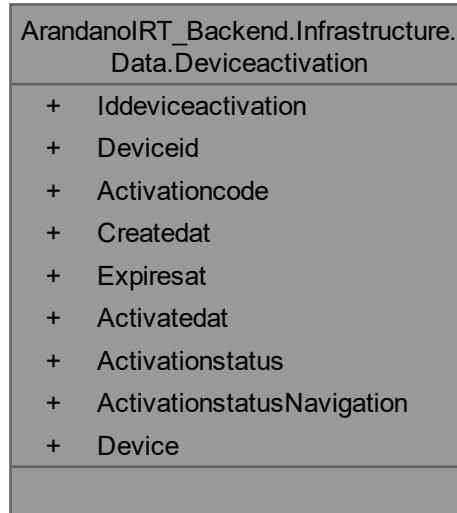
Tests that GetUtcNow returns a DateTime value that is close to the actual current UTC time, allowing for minor execution delays.

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Utilities/[DateTimeProviderTests.cs](#)

6.62 ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation:



Properties

- int `Iddeviceactivation` [get, set]
- int? `Deviceid` [get, set]
- string `Activationcode` = null! [get, set]
- DateTime `Createdat` [get, set]
- DateTime `Expiresat` [get, set]
- DateTime? `Activatedat` [get, set]
- int? `Activationstatus` [get, set]
- virtual ? Status `ActivationstatusNavigation` [get, set]
- virtual ? Devicedata `Device` [get, set]

6.62.1 Property Documentation

6.62.1.1 Activatedat

```
DateTime? ArandanoIRT_Backend::Infrastructure::Data::Deviceactivation::Activatedat [get], [set]
```

6.62.1.2 Activationcode

```
string ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Activationcode = null! [get],  
[set]
```

6.62.1.3 Activationstatus

```
int? ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Activationstatus [get], [set]
```

6.62.1.4 ActivationstatusNavigation

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Activationstatus←  
Navigation [get], [set]
```

6.62.1.5 Createdat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Createdat [get], [set]
```

6.62.1.6 Device

```
virtual ? Devicedata ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Device [get],  
[set]
```

6.62.1.7 Deviceid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Deviceid [get], [set]
```

6.62.1.8 Expiresat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Expiresat [get], [set]
```

6.62.1.9 Iddeviceactivation

```
int ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation.Iddeviceactivation [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Deviceactivation.cs](#)

6.63 ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity Class Reference

Represents a record containing an activation code used to register, authorize, or link a physical device to the system.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity:

ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity
<ul style="list-style-type: none"> + IdDeviceActivation + DeviceId + ActivationCode + CreatedAt + ExpiresAt + ActivatedAt + ActivationStatus + DeviceActivationEntity()

Public Member Functions

- [DeviceActivationEntity](#) (int idDeviceActivation, int? deviceId, string activationCode, DateTime createdAt, DateTime expiresAt, int? activationStatus=null)

Initializes a new instance of the DeviceActivationEntity class, typically used when generating a new activation code.

Properties

- int [IdDeviceActivation](#) [get]
Gets the unique identifier for this device activation record.
- int? [DeviceId](#) [get]
Gets the unique identifier of the device associated with this activation code, once activated.
- string [ActivationCode](#) [get]
Gets the unique code required to activate the device.
- DateTime [CreatedAt](#) [get]
Gets the date and time (usually UTC) when this activation code was created.
- DateTime [ExpiresAt](#) [get]
Gets the date and time (usually UTC) when this activation code expires and becomes invalid.
- DateTime? [ActivatedAt](#) [get, set]
Gets the date and time (usually UTC) when the device was successfully activated using this code.
- int? [ActivationStatus](#) [get]
Gets the identifier representing the current status of the activation code (e.g., Pending, Active, Expired, Used).

6.63.1 Detailed Description

Represents a record containing an activation code used to register, authorize, or link a physical device to the system.

6.63.2 Constructor & Destructor Documentation

6.63.2.1 DeviceActivationEntity()

```
ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.DeviceActivationEntity (
    int idDeviceActivation,
    int? deviceId,
    string activationCode,
    DateTime createdAt,
    DateTime expiresAt,
    int? activationStatus = null ) [inline]
```

Initializes a new instance of the DeviceActivationEntity class, typically used when generating a new activation code.

Parameters

<i>idDeviceActivation</i>	The unique ID for this record (often generated by the database).
<i>deviceId</i>	The ID of the device this code might be linked to (nullable initially).
<i>activationCode</i>	The activation code string.
<i>createdAt</i>	The creation timestamp.
<i>expiresAt</i>	The expiration timestamp.
<i>activationStatus</i>	The initial activation status ID (optional, defaults to null).

Exceptions

<i>ArgumentNullException</i>	Thrown if <i>activationCode</i> is null.
------------------------------	--

Note that the ActivatedAt property is not set by this constructor and remains null initially.

6.63.3 Property Documentation

6.63.3.1 ActivatedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.ActivatedAt [get], [set]
```

Gets the date and time (usually UTC) when the device was successfully activated using this code.

Nullable if the code has not been used yet.

6.63.3.2 ActivationCode

```
string ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.ActivationCode [get]
```

Gets the unique code required to activate the device.

6.63.3.3 ActivationStatus

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.ActivationStatus [get]
```

Gets the identifier representing the current status of the activation code (e.g., Pending, Active, Expired, Used).

Nullable.

6.63.3.4 CreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when this activation code was created.

6.63.3.5 DeviceId

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.DeviceId [get]
```

Gets the unique identifier of the device associated with this activation code, once activated.

Nullable if the code is pending activation or not linked yet.

6.63.3.6 ExpiresAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.ExpiresAt [get]
```

Gets the date and time (usually UTC) when this activation code expires and becomes invalid.

6.63.3.7 IdDeviceActivation

```
int ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity.IdDeviceActivation [get]
```

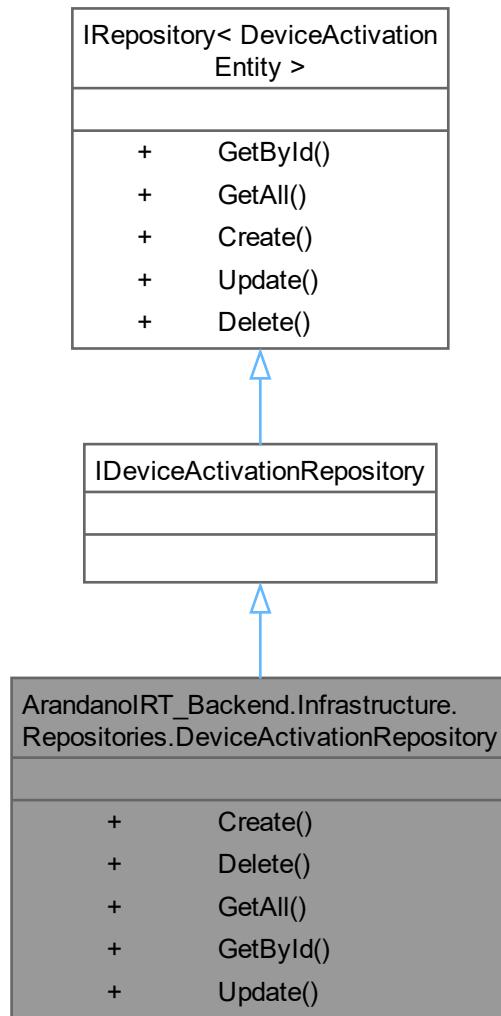
Gets the unique identifier for this device activation record.

The documentation for this class was generated from the following file:

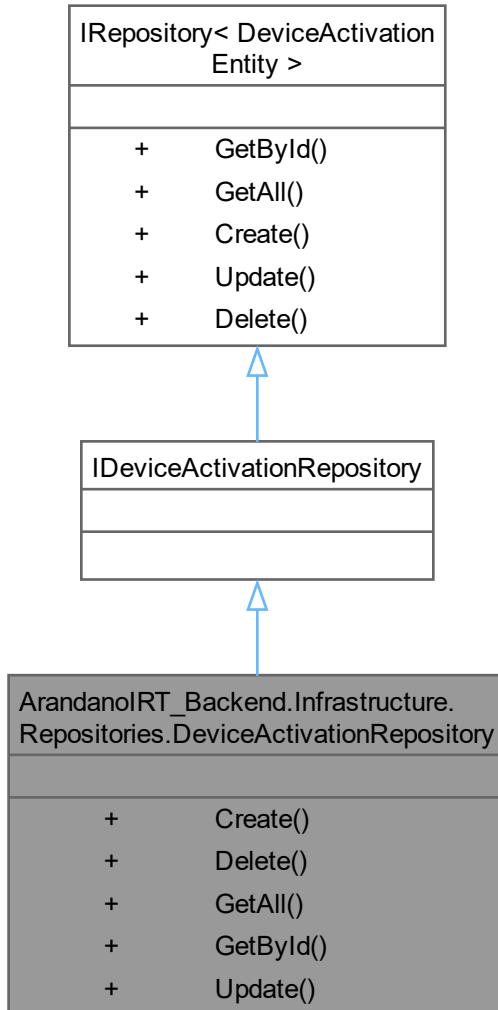
- Domain/Entities/[DeviceActivationEntity.cs](#)

6.64 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivation Repository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository:



Public Member Functions

- `Task< Result< DeviceActivationEntity > > Create (DeviceActivationEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< DeviceActivationEntity > > > GetAll ()`
- `Task< Result< DeviceActivationEntity > > GetById (int id)`
- `Task< Result< bool > > Update (DeviceActivationEntity entity)`

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories IRepository< DeviceActivationEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T >](#) > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.64.1 Member Function Documentation

6.64.1.1 Create()

```
Task< Result< DeviceActivationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
DeviceActivationRepository.Create (
    DeviceActivationEntity entity ) [inline]
```

6.64.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository.←
Delete (
    int id ) [inline]
```

6.64.1.3 GetAll()

```
Task< Result< IEnumerable< DeviceActivationEntity > > > ArandanoIRT_Backend.Infrastructure.←
Repositories.DeviceActivationRepository.GetAll ( ) [inline]
```

6.64.1.4 GetById()

```
Task< Result< DeviceActivationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
DeviceActivationRepository.GetById (
    int id ) [inline]
```

6.64.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository.←  
Update (   
    DeviceActivationEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/DeviceActivationRepository.cs

6.65 ArandanoIRT_Backend.Infrastructure.Data.Devicedata Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Devicedata:

ArandanoIRT_Backend.Infrastructure. Data.Devicedata	
+	Iddevicedata
+	Namedevice
+	Descriptiondevice
+	Datacollectiontime
+	Statusid
+	Registeredat
+	Registeredby
+	Updatedat
+	Updatedby
+	Cropid
and 9 more...	

Properties

- int Iddevicedata [get, set]
- string? Namedevice [get, set]
- string? Descriptiondevice [get, set]
- short Datacollectiontime [get, set]

- int? `Statusid` [get, set]
- DateTime `Registeredat` [get, set]
- int? `Registeredby` [get, set]
- DateTime? `Updatedat` [get, set]
- int? `Updatedby` [get, set]
- int? `Cropid` [get, set]
- int? `Plantid` [get, set]
- virtual ? `Crop Crop` [get, set]
- virtual ICollection<`Deviceactivation`> `Deviceactivation` = new List<`Deviceactivation`>() [get, set]
- virtual ICollection<`Devicelog`> `Devicelog` = new List<`Devicelog`>() [get, set]
- virtual ICollection<`Devicetoken`> `Devicetoken` = new List<`Devicetoken`>() [get, set]
- virtual ? `Plantdata Plant` [get, set]
- virtual ? `Person RegisteredbyNavigation` [get, set]
- virtual ? `Status Status` [get, set]
- virtual ? `Person UpdatedbyNavigation` [get, set]

6.65.1 Property Documentation

6.65.1.1 `Crop`

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Crop [get], [set]
```

6.65.1.2 `Cropid`

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Cropid [get], [set]
```

6.65.1.3 `Datacollectiontime`

```
short ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Datacollectiontime [get], [set]
```

6.65.1.4 `Descriptiondevice`

```
string? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Descriptiondevice [get], [set]
```

6.65.1.5 `Deviceactivation`

```
virtual ICollection<Deviceactivation> ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Deviceactivation = new List<Deviceactivation>() [get], [set]
```

6.65.1.6 `Devicelog`

```
virtual ICollection<Devicelog> ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Devicelog = new List<Devicelog>() [get], [set]
```

6.65.1.7 Devicetoken

```
virtual ICollection<Devicetoken> ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Devicetoken =  
new List<Devicetoken>() [get], [set]
```

6.65.1.8 Iddevicedata

```
int ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Iddevicedata [get], [set]
```

6.65.1.9 Namedevice

```
string? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Namedevice [get], [set]
```

6.65.1.10 Plant

```
virtual ? Plantdata ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Plant [get], [set]
```

6.65.1.11 Plantid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Plantid [get], [set]
```

6.65.1.12 Registeredat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Registeredat [get], [set]
```

6.65.1.13 Registeredby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Registeredby [get], [set]
```

6.65.1.14 Registeredbynavigation

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Registeredbynavigation [get],  
[set]
```

6.65.1.15 Status

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Status [get], [set]
```

6.65.1.16 Statusid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Statusid [get], [set]
```

6.65.1.17 Updatedat

```
Datetime? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Updatedat [get], [set]
```

6.65.1.18 Updatedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicedata.Updatedby [get], [set]
```

6.65.1.19 UpdatedbyNavigation

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Devicedata.UpdatedbyNavigation [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Devicedata.cs](#)

6.66 ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity Class Reference

Represents a data-collecting device, containing its identification, description, configuration settings, registration details, and associations (crop, plant).

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity:

ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity
<ul style="list-style-type: none"> + IdDeviceData + NameDevice + DescriptionDevice + DataCollectionTime + StatusId + RegisteredAt + RegisteredBy + UpdatedAt + UpdatedBy + Cropld + PlantId + DeviceDataEntity()

Public Member Functions

- `DeviceDataEntity` (int idDeviceData, string nameDevice, string descriptionDevice, short dataCollectionTime, DateTime registeredAt, int? registeredBy, int? statusId, int? cropld, int? plantId)
- Initializes a new instance of the DeviceDataEntity class, typically upon device registration.*

Properties

- int `IdDeviceData` [get]
Gets the unique identifier for the device data record.
- string `NameDevice` [get]
Gets the name assigned to the device.
- string `DescriptionDevice` [get]
Gets a description of the device.
- short `DataCollectionTime` [get]
Gets the configured time interval for data collection (e.g., in minutes or seconds).
- int? `StatusId` [get]
Gets the identifier representing the current status of the device (e.g., Active, Inactive, Error).
- DateTime `RegisteredAt` [get]
Gets the date and time (usually UTC) when the device was registered in the system.

- int? [RegisteredBy](#) [get]
Gets the unique identifier of the user who registered the device.
- DateTime? [UpdatedAt](#) [get, set]
Gets the date and time (usually UTC) when the device entity was last updated.
- int? [UpdatedBy](#) [get, set]
Gets the unique identifier of the user who last updated the device entity.
- int? [CropId](#) [get]
Gets the unique identifier of the crop to which this device is assigned.
- int? [PlantId](#) [get]
Gets the unique identifier of the specific plant this device is monitoring, if applicable.

6.66.1 Detailed Description

Represents a data-collecting device, containing its identification, description, configuration settings, registration details, and associations (crop, plant).

6.66.2 Constructor & Destructor Documentation

6.66.2.1 DeviceDataEntity()

```
ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.DeviceDataEntity (
    int idDeviceData,
    string nameDevice,
    string descriptionDevice,
    short dataCollectionTime,
    DateTime registeredAt,
    int? registeredBy,
    int? statusId,
    int? cropId,
    int? plantId ) [inline]
```

Initializes a new instance of the DeviceDataEntity class, typically upon device registration.

Parameters

<i>idDeviceData</i>	The unique ID for this device record (often generated by the database).
<i>nameDevice</i>	The name of the device.
<i>descriptionDevice</i>	The description of the device.
<i>dataCollectionTime</i>	The data collection interval.
<i>registeredAt</i>	The registration timestamp.
<i>registeredBy</i>	The ID of the user registering the device (nullable).
<i>statusId</i>	The initial status ID (nullable).
<i>cropId</i>	The ID of the crop the device is assigned to (nullable).
<i>plantId</i>	The ID of the plant the device is assigned to (nullable).

Exceptions

<code>ArgumentOutOfRangeException</code>	Thrown if <code>dataCollectionTime</code> is negative (Spanish message: "El tiempo de recolección no puede ser negativo.").
--	---

Note that `UpdatedAt` and `UpdatedBy` properties are not set by this constructor and remain null initially.

6.66.3 Property Documentation

6.66.3.1 CropId

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.CropId [get]
```

Gets the unique identifier of the crop to which this device is assigned.

Nullable.

6.66.3.2 DataCollectionTime

```
short ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.DataCollectionTime [get]
```

Gets the configured time interval for data collection (e.g., in minutes or seconds).

6.66.3.3 DescriptionDevice

```
string ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.DescriptionDevice [get]
```

Gets a description of the device.

6.66.3.4 IdDeviceData

```
int ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.IdDeviceData [get]
```

Gets the unique identifier for the device data record.

6.66.3.5 NameDevice

```
string ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.NameDevice [get]
```

Gets the name assigned to the device.

6.66.3.6 PlantId

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.PlantId [get]
```

Gets the unique identifier of the specific plant this device is monitoring, if applicable.

Nullable.

6.66.3.7 RegisteredAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.RegisteredAt [get]
```

Gets the date and time (usually UTC) when the device was registered in the system.

6.66.3.8 RegisteredBy

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.RegisteredBy [get]
```

Gets the unique identifier of the user who registered the device.

Nullable.

6.66.3.9 StatusId

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.StatusId [get]
```

Gets the identifier representing the current status of the device (e.g., Active, Inactive, Error).

Nullable.

6.66.3.10 UpdatedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.UpdatedAt [get], [set]
```

Gets the date and time (usually UTC) when the device entity was last updated.

Nullable if never updated.

6.66.3.11 UpdatedBy

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity.UpdatedBy [get], [set]
```

Gets the unique identifier of the user who last updated the device entity.

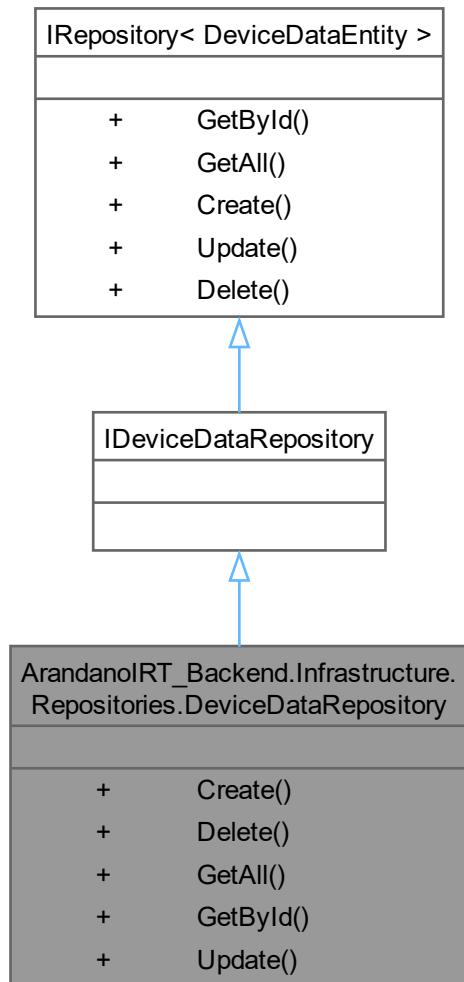
Nullable if never updated.

The documentation for this class was generated from the following file:

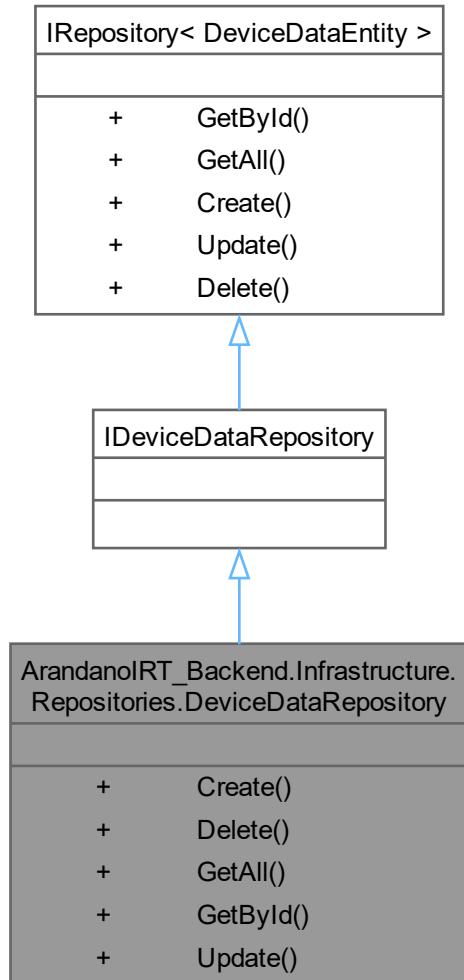
- Domain/Entities/[DeviceDataEntity.cs](#)

6.67 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository:



Public Member Functions

- `Task< Result< DeviceDataEntity > > Create (DeviceDataEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< DeviceDataEntity > > > GetAll ()`
- `Task< Result< DeviceDataEntity > > GetById (int id)`
- `Task< Result< bool > > Update (DeviceDataEntity entity)`

Public Member Functions inherited from

`ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceDataEntity >`

- `Task< Result< T > > GetById (int id)`

- Task<`Result< IEnumerable< T > >`> `GetAll()`
Asynchronously retrieves all entities of type T.
- Task<`Result< T >`> `Create(T entity)`
Asynchronously creates a new entity in the data store.
- Task<`Result< bool >`> `Update(T entity)`
Asynchronously updates an existing entity in the data store.
- Task<`Result< bool >`> `Delete(int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.67.1 Member Function Documentation

6.67.1.1 Create()

```
Task< Result< DeviceDataEntity >> ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository.Create (
    DeviceDataEntity entity ) [inline]
```

6.67.1.2 Delete()

```
Task< Result< bool >> ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository.Delete (
    int id ) [inline]
```

6.67.1.3 GetAll()

```
Task< Result< IEnumerable< DeviceDataEntity > >> ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository.GetAll () [inline]
```

6.67.1.4 GetById()

```
Task< Result< DeviceDataEntity >> ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository.GetById (
    int id ) [inline]
```

6.67.1.5 Update()

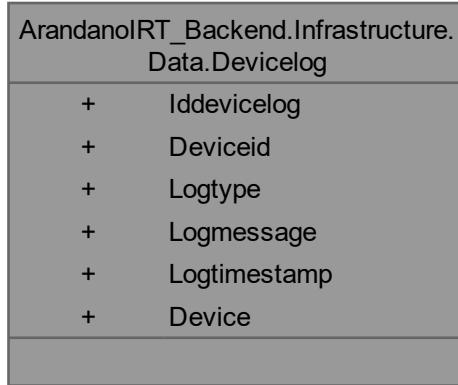
```
Task< Result< bool >> ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository.Update (
    DeviceDataEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/[DeviceDataRepository.cs](#)

6.68 ArandanoIRT_Backend.Infrastructure.Data.Devicelog Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Devicelog:



Properties

- int `Iddevicelog` [get, set]
- int `Deviceid` [get, set]
- string `Logtype` = null! [get, set]
- string `Logmessage` = null! [get, set]
- DateTime `Logtimestamp` [get, set]
- virtual `Devicedata Device` = null! [get, set]

6.68.1 Property Documentation

6.68.1.1 Device

```
virtual Devicedata ArandanoIRT_Backend::Infrastructure::Data::Devicelog::Device = null! [get], [set]
```

6.68.1.2 Deviceid

```
int ArandanoIRT_Backend::Infrastructure::Data::Devicelog::Deviceid [get], [set]
```

6.68.1.3 Iddevicelog

```
int ArandanoIRT_Backend::Infrastructure::Data::Devicelog::Iddevicelog [get], [set]
```

6.68.1.4 Logmessage

```
string ArandanoIRT_Backend.Infrastructure.Data.Devicelog.Logmessage = null! [get], [set]
```

6.68.1.5 Logtimestamp

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Devicelog.Logtimestamp [get], [set]
```

6.68.1.6 Logtype

```
string ArandanoIRT_Backend.Infrastructure.Data.Devicelog.Logtype = null! [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Devicelog.cs](#)

6.69 ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity Class Reference

Represents a single log entry generated by, or related to, a specific device.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity:

ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity
+ IdDeviceLog
+ DeviceId
+ LogType
+ LogMessage
+ LogTimestamp
+ DeviceLogEntity()

Public Member Functions

- [DeviceLogEntity](#) (int idDeviceLog, int deviceId, string logType, string logMessage, DateTime logTimestamp)
Initializes a new instance of the DeviceLogEntity class.

Properties

- int `IdDeviceLog` [get]
Gets the unique identifier for this device log record.
- int `DeviceId` [get]
Gets the unique identifier of the device that generated or is related to this log entry.
- string `LogType` [get]
Gets the type or category of the log entry (e.g., "INFO", "WARNING", "ERROR", "CONNECTION").
- string `LogMessage` [get]
Gets the detailed message content of the log entry.
- DateTime `LogTimestamp` [get]
Gets the date and time (usually UTC) when the event logged occurred or was recorded.

6.69.1 Detailed Description

Represents a single log entry generated by, or related to, a specific device.

Used for recording events, errors, or informational messages from devices.

6.69.2 Constructor & Destructor Documentation

6.69.2.1 DeviceLogEntity()

```
ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.DeviceLogEntity (
    int idDeviceLog,
    int deviceId,
    string logType,
    string logMessage,
    DateTime logTimestamp ) [inline]
```

Initializes a new instance of the DeviceLogEntity class.

Parameters

<code>idDeviceLog</code>	The unique ID for this log record (often generated by the database).
<code>deviceId</code>	The ID of the related device.
<code>logType</code>	The type or category of the log.
<code>logMessage</code>	The message content of the log.
<code>logTimestamp</code>	The timestamp when the logged event occurred.

Exceptions

<code>ArgumentException</code>	Thrown if <code>logType</code> is null or whitespace (Spanish message: "El tipo de log es requerido."). Thrown if <code>logMessage</code> is null or whitespace (Spanish message: "El mensaje de log es requerido.").
--------------------------------	--

6.69.3 Property Documentation

6.69.3.1 DeviceId

```
int ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.DeviceId [get]
```

Gets the unique identifier of the device that generated or is related to this log entry.

6.69.3.2 IdDeviceLog

```
int ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.IdDeviceLog [get]
```

Gets the unique identifier for this device log record.

6.69.3.3 LogMessage

```
string ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.LogMessage [get]
```

Gets the detailed message content of the log entry.

6.69.3.4 LogTimestamp

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.LogTimestamp [get]
```

Gets the date and time (usually UTC) when the event logged occurred or was recorded.

6.69.3.5 LogType

```
string ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity.LogType [get]
```

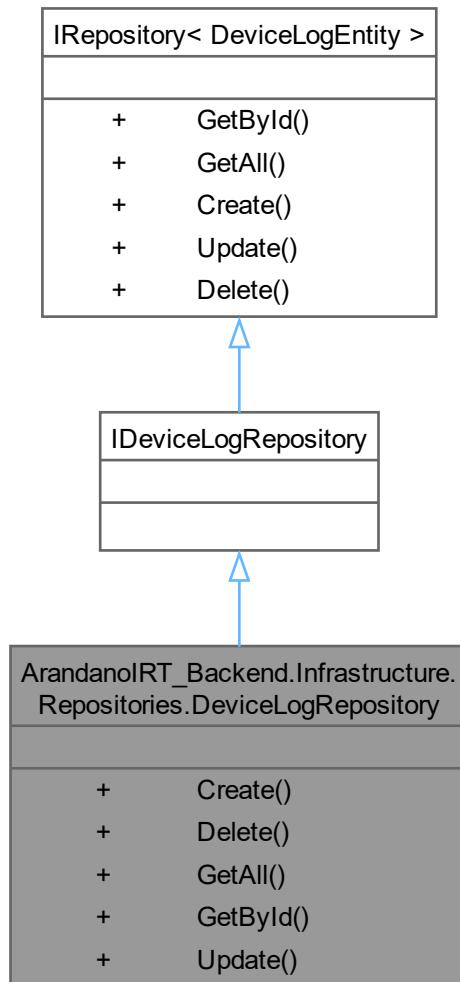
Gets the type or category of the log entry (e.g., "INFO", "WARNING", "ERROR", "CONNECTION").

The documentation for this class was generated from the following file:

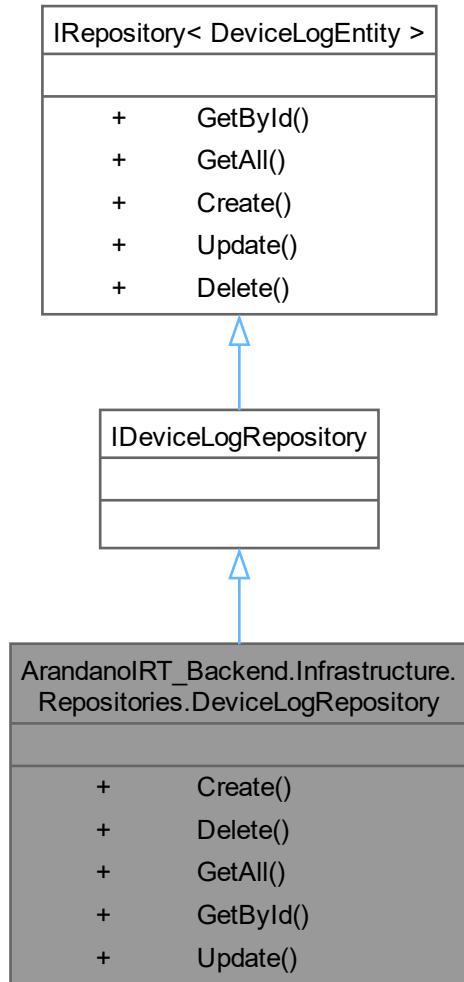
- Domain/Entities/[DeviceLogEntity.cs](#)

6.70 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository:



Public Member Functions

- `Task< Result< DeviceLogEntity > > Create (DeviceLogEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< DeviceLogEntity > > > GetAll ()`
- `Task< Result< DeviceLogEntity > > GetById (int id)`
- `Task< Result< bool > > Update (DeviceLogEntity entity)`

Public Member Functions inherited from

`ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceLogEntity >`

- `Task< Result< T > > GetById (int id)`

- Task<[Result](#)< I`Enumerable`< T > > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task<[Result](#)< T > > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task<[Result](#)< bool > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task<[Result](#)< bool > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.70.1 Member Function Documentation

6.70.1.1 Create()

```
Task< Result< DeviceLogEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLog←
Repository.Create (
    DeviceLogEntity entity ) [inline]
```

6.70.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository.←
Delete (
    int id ) [inline]
```

6.70.1.3 GetAll()

```
Task< Result< IEnumerable< DeviceLogEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.←
DeviceLogRepository.GetAll ( ) [inline]
```

6.70.1.4 GetById()

```
Task< Result< DeviceLogEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLog←
Repository.GetById (
    int id ) [inline]
```

6.70.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository.←
Update (
    DeviceLogEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/[DeviceLogRepository.cs](#)

6.71 ArandanoIRT_Backend.Infrastructure.Data.Devicetoken Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Devicetoken:

ArandanoIRT_Backend.Infrastructure.Data.Devicetoken	
+	Iddevicetoken
+	Deviceid
+	Token
+	Refresh token
+	Createdat
+	Expiresat
+	Revokedat
+	Revokedbyip
+	Deviceinfo
+	Useragent
+	Device

Properties

- int `Iddevicetoken` [get, set]
- int? `Deviceid` [get, set]
- string `Token` = null! [get, set]
- string `Refresh token` = null! [get, set]
- DateTime `Createdat` [get, set]
- DateTime `Expiresat` [get, set]
- DateTime? `Revokedat` [get, set]
- string? `Revokedbyip` [get, set]
- string? `Deviceinfo` [get, set]
- string? `Useragent` [get, set]
- virtual ? Devicedata `Device` [get, set]

6.71.1 Property Documentation

6.71.1.1 `Createdat`

DateTime `ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Createdat` [get], [set]

6.71.1.2 Device

```
virtual ? Devicedata ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Device [get], [set]
```

6.71.1.3 Deviceid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Deviceid [get], [set]
```

6.71.1.4 Deviceinfo

```
string? ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Deviceinfo [get], [set]
```

6.71.1.5 Expiresat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Expiresat [get], [set]
```

6.71.1.6 Iddevicetoken

```
int ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Iddevicetoken [get], [set]
```

6.71.1.7 RefreshToken

```
string ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.RefreshToken = null! [get], [set]
```

6.71.1.8 Revokedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Revokedat [get], [set]
```

6.71.1.9 Revokedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Revokedbyip [get], [set]
```

6.71.1.10 Token

```
string ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Token = null! [get], [set]
```

6.71.1.11 Useragent

```
string? ArandanoIRT_Backend.Infrastructure.Data.Devicetoken.Useragent [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/Devcetoken.cs

6.72 ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity Class Reference

Represents a pair of authentication tokens (access and refresh) issued specifically for a device, along with metadata about the token's lifecycle and the device context.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity:

ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity
+ IdDeviceToken
+ Deviceld
+ Token
+ RefreshToken
+ CreatedAt
+ ExpiresAt
+ RevokedAt
+ RevokedByIp
+ DeviceInfo
+ UserAgent
+ DeviceTokenEntity()

Public Member Functions

- [DeviceTokenEntity](#) (int idDeviceToken, int? deviceld, string token, string refreshToken, DateTime createdAt, DateTime expiresAt, DateTime? revokedAt, string revokedByIp, string deviceInfo, string userAgent)
Initializes a new instance of the DeviceTokenEntity class.

Properties

- int `IdDeviceToken` [get]
Gets the unique identifier for this device token record.
- int? `DeviceId` [get]
Gets the unique identifier of the device associated with these tokens.
- string `Token` [get]
Gets the access token string (e.g., JWT) used for short-term authentication.
- string `RefreshToken` [get]
Gets the refresh token string used to obtain new access tokens.
- DateTime `CreatedAt` [get]
Gets the date and time (usually UTC) when these tokens were created.
- DateTime `ExpiresAt` [get]
Gets the date and time (usually UTC) when the refresh token expires.
- DateTime? `RevokedAt` [get]
Gets the date and time (usually UTC) when the refresh token was revoked.
- string `RevokedByIp` [get]
Gets the IP address from which the revocation request originated, if applicable.
- string `DeviceInfo` [get]
Gets descriptive information about the device associated with these tokens at the time of issuance or refresh.
- string `UserAgent` [get]
Gets the user agent string of the client associated with these tokens at the time of issuance or refresh.

6.72.1 Detailed Description

Represents a pair of authentication tokens (access and refresh) issued specifically for a device, along with metadata about the token's lifecycle and the device context.

6.72.2 Constructor & Destructor Documentation

6.72.2.1 DeviceTokenEntity()

```
ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.DeviceTokenEntity (
    int idDeviceToken,
    int? deviceId,
    string token,
    string refreshToken,
    DateTime createdAt,
    DateTime expiresAt,
    DateTime? revokedAt,
    string revokedByIp,
    string deviceInfo,
    string userAgent ) [inline]
```

Initializes a new instance of the DeviceTokenEntity class.

Parameters

<i>idDeviceToken</i>	The unique ID for this token record (often generated by the database).
<i>deviceid</i>	The ID of the associated device (nullable).
<i>token</i>	The access token string.
<i>refreshToken</i>	The refresh token string.
<i>createdAt</i>	The creation timestamp.
<i>expiresAt</i>	The expiration timestamp for the refresh token.
<i>revokedAt</i>	The revocation timestamp (nullable if active).
<i>revokedByIp</i>	The IP address causing revocation (if applicable).
<i>deviceInfo</i>	Information about the associated device.
<i>userAgent</i>	The user agent string of the client.

Exceptions

<i>ArgumentException</i>	Thrown if <i>token</i> is null or whitespace (Spanish message: "El token es requerido."). Thrown if <i>refreshToken</i> is null or whitespace (Spanish message: "El refresh token es requerido.").
--------------------------	--

6.72.3 Property Documentation

6.72.3.1 CreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when these tokens were created.

6.72.3.2 DeviceId

```
int? ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.DeviceId [get]
```

Gets the unique identifier of the device associated with these tokens.

Nullable if not directly linked.

6.72.3.3 DeviceInfo

```
string ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.DeviceInfo [get]
```

Gets descriptive information about the device associated with these tokens at the time of issuance or refresh.

6.72.3.4 ExpiresAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.ExpiresAt [get]
```

Gets the date and time (usually UTC) when the refresh token expires.

6.72.3.5 **IdDeviceToken**

```
int ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.IdDeviceToken [get]
```

Gets the unique identifier for this device token record.

6.72.3.6 **RefreshToken**

```
string ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.RefreshToken [get]
```

Gets the refresh token string used to obtain new access tokens.

6.72.3.7 **RevokedAt**

```
DateTime? ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.RevokedAt [get]
```

Gets the date and time (usually UTC) when the refresh token was revoked.

Nullable if still active.

6.72.3.8 **RevokedByIp**

```
string ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.RevokedByIp [get]
```

Gets the IP address from which the revocation request originated, if applicable.

6.72.3.9 **Token**

```
string ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.Token [get]
```

Gets the access token string (e.g., JWT) used for short-term authentication.

6.72.3.10 **UserAgent**

```
string ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity.UserAgent [get]
```

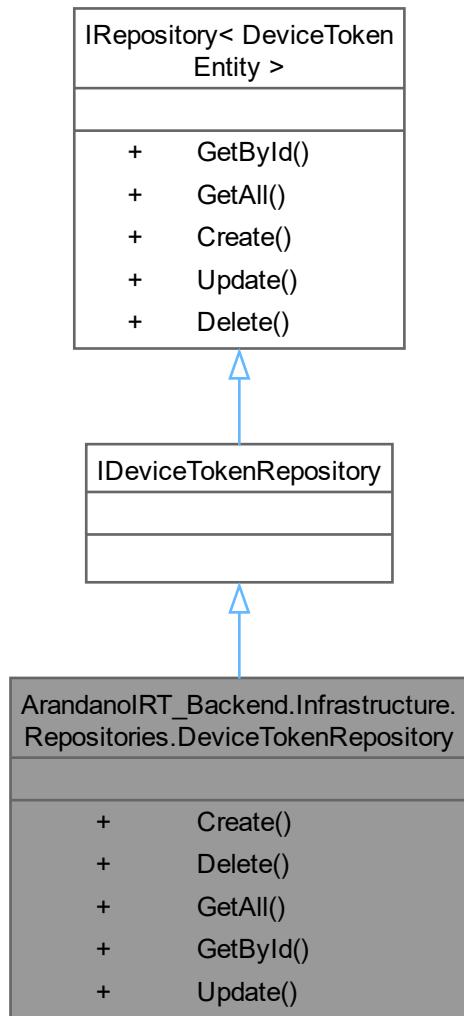
Gets the user agent string of the client associated with these tokens at the time of issuance or refresh.

The documentation for this class was generated from the following file:

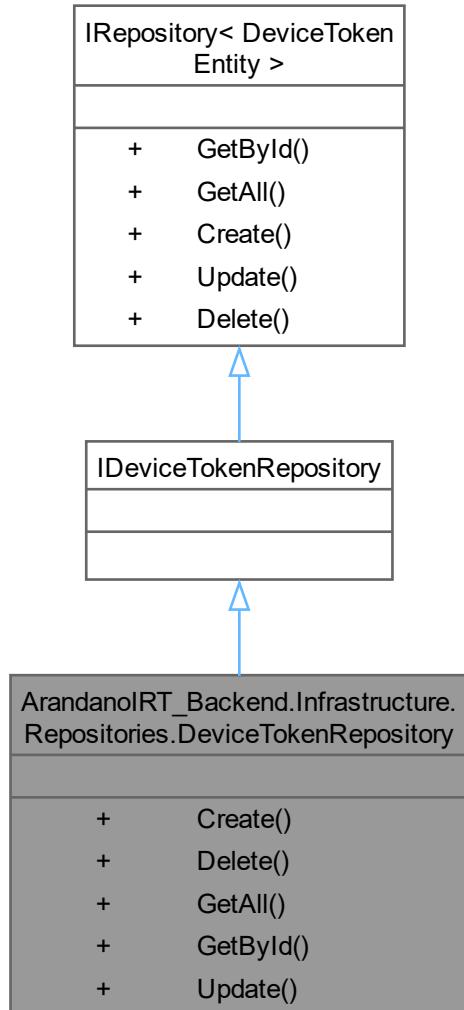
- Domain/Entities/[DeviceTokenEntity.cs](#)

6.73 ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository:



Public Member Functions

- Task< [Result< DeviceTokenEntity >](#) > [Create](#) (`DeviceTokenEntity` entity)
- Task< [Result< bool >](#) > [Delete](#) (int id)
- Task< [Result< IEnumerable< DeviceTokenEntity >](#) > > [GetAll](#) ()
- Task< [Result< DeviceTokenEntity >](#) > > [GetById](#) (int id)
- Task< [Result< bool >](#) > > [Update](#) (`DeviceTokenEntity` entity)

Public Member Functions inherited from**ArandanoIRT_Backend.Domain.IRepositories IRepository< DeviceTokenEntity >**

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.73.1 Member Function Documentation

6.73.1.1 Create()

```
Task< Result< DeviceTokenEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository.Create (
    DeviceTokenEntity entity) [inline]
```

6.73.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository.Delete (
    int id) [inline]
```

6.73.1.3 GetAll()

```
Task< Result< IEnumerable< DeviceTokenEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository.GetAll () [inline]
```

6.73.1.4 GetById()

```
Task< Result< DeviceTokenEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository.GetById (
    int id) [inline]
```

6.73.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository.←
Update (
    DeviceTokenEntity entity ) [inline]
```

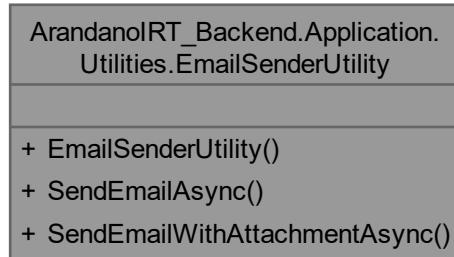
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/DeviceTokenRepository.cs

6.74 ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility Class Reference

Provides utility methods for sending emails using SMTP settings from configuration.

Collaboration diagram for ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility:



Public Member Functions

- [EmailSenderUtility \(IConfiguration configuration\)](#)
Initializes a new instance of the EmailSenderUtility class, configuring the SmtpClient from application settings.
- [async Task SendEmailAsync \(string destiny, string affair, string message\)](#)
Asynchronously sends a simple email message with HTML body content.
- [async Task SendEmailWithAttachmentAsync \(string recipient, string subject, string message, string fileBase64, string fileName\)](#)
Asynchronously sends an email message with an HTML body and a single PDF attachment provided as a Base64 string.

6.74.1 Detailed Description

Provides utility methods for sending emails using SMTP settings from configuration.

Includes asynchronous methods for sending emails.

6.74.2 Constructor & Destructor Documentation

6.74.2.1 EmailSenderUtility()

```
ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility.EmailSenderUtility (
    IConfiguration configuration ) [inline]
```

Initializes a new instance of the EmailSenderUtility class, configuring the SmtpClient from application settings.

Parameters

<i>configuration</i>	The application configuration containing SMTP settings.
----------------------	---

Exceptions

<i>ArgumentNullException</i>	Thrown if configuration is null.
<i>InvalidOperationException</i>	Thrown if required SMTP configuration keys are missing or the Port value is not a valid integer format or range.

6.74.3 Member Function Documentation

6.74.3.1 SendEmailAsync()

```
async Task ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility.SendEmailAsync (
    string destiny,
    string affair,
    string message ) [inline]
```

Asynchronously sends a simple email message with HTML body content.

Parameters

<i>destiny</i>	The recipient's email address.
<i>affair</i>	The subject line of the email.
<i>message</i>	The HTML body content of the email.

Returns

A Task representing the asynchronous operation.

This method sends the email asynchronously using SendMailAsync for better performance and non-blocking I/O. Exceptions during sending are caught and logged using Serilog, but not propagated. The sender display name is currently hardcoded as "Biokudi". Consider making it configurable. This method signature is a BREAKING CHANGE from any previous synchronous version. Callers must now use 'await'.

6.74.3.2 SendEmailWithAttachmentAsync()

```
async Task ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility.SendEmailWithAttachment<-
Async (
    string recipient,
    string subject,
    string message,
    string fileBase64,
    string fileName ) [inline]
```

Asynchronously sends an email message with an HTML body and a single PDF attachment provided as a Base64 string.

Parameters

<i>recipient</i>	The recipient's email address.
<i>subject</i>	The subject line of the email.
<i>message</i>	The HTML body content of the email.
<i>fileBase64</i>	The Base64 encoded string representation of the attachment file content.
<i>fileName</i>	The desired name for the attached file (e.g., "report.pdf"). Should include extension.

Returns

A Task representing the asynchronous operation.

This method sends the email asynchronously using `SendMailAsync`. It includes an asynchronous validation check on the recipient's email format. It assumes the provided Base64 string represents a PDF file (`MediaTypesNames`. \leftarrow `Application.Pdf`). Consider making content type dynamic or a parameter. Exceptions during sending are caught and logged using Serilog, but not propagated. This method signature is a BREAKING CHANGE from any previous synchronous version. Callers must now use 'await'.

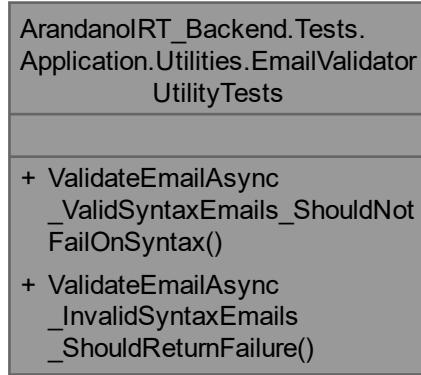
The documentation for this class was generated from the following file:

- Application/Utilities/[EmailSenderUtility.cs](#)

6.75 ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests Class Reference

Contains unit tests for the static `EmailValidatorUtility` class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests:



Public Member Functions

- async Task [ValidateEmailAsync_ValidSyntaxEmails_ShouldNotFailOnSyntax](#) (string validEmail)
Tests that ValidateEmailAsync does not fail due to syntax errors for various valid email formats.
- async Task [ValidateEmailAsync_InvalidSyntaxEmails_ShouldReturnFailure](#) (string? invalidEmail)
Tests that ValidateEmailAsync returns a failure result for various invalid email formats, primarily due to syntax errors or emptiness.

6.75.1 Detailed Description

Contains unit tests for the static EmailValidatorUtility class.

Focuses primarily on syntax validation which is self-contained.

6.75.2 Member Function Documentation

6.75.2.1 ValidateEmailAsync_InvalidSyntaxEmails_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests.Validate<-->
EmailAsync_InvalidSyntaxEmails_ShouldReturnFailure (
    string? invalidEmail ) [inline]
```

Tests that ValidateEmailAsync returns a failure result for various invalid email formats, primarily due to syntax errors or emptiness.

Parameters

<i>invalidEmail</i>	An invalid email address string (or null).
---------------------	--

6.75.2.2 ValidateEmailAsync_ValidSyntaxEmails_ShouldNotFailOnSyntax()

```
async Task ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests.Validate←
EmailAsync_ValidSyntaxEmails_ShouldNotFailOnSyntax (
    string validEmail ) [inline]
```

Tests that ValidateEmailAsync does not fail due to syntax errors for various valid email formats.

Note: This test implicitly assumes DNS checks would pass for these syntactically valid emails, as DNS mocking is not feasible for the static method as implemented.

Parameters

<i>validEmail</i>	A syntactically valid email address string.
-------------------	---

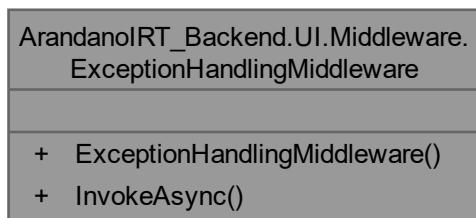
The documentation for this class was generated from the following file:

- Tests/Unit/Application/Utilities/[EmailValidatorUtilityTests.cs](#)

6.76 ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware Class Reference

ASP.NET Core middleware for centralized exception handling.

Collaboration diagram for ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware:



Public Member Functions

- `ExceptionHandlingMiddleware` (`RequestDelegate` next, `ILogger< ExceptionHandlingMiddleware >` logger)
Initializes a new instance of the ExceptionHandlingMiddleware class.
- `async Task InvokeAsync` (`HttpContext` context)
Processes an HTTP request within the middleware pipeline.

6.76.1 Detailed Description

ASP.NET Core middleware for centralized exception handling.

Catches unhandled exceptions, logs them, and converts them into standardized JSON error responses with appropriate HTTP status codes, avoiding leakage of internal details.

6.76.2 Constructor & Destructor Documentation

6.76.2.1 `ExceptionHandlingMiddleware()`

```
ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware.ExceptionHandlingMiddleware ( 
    RequestDelegate next,
    ILogger< ExceptionHandlingMiddleware > logger ) [inline]
```

Initializes a new instance of the `ExceptionHandlingMiddleware` class.

Parameters

<code>next</code>	The next middleware delegate in the pipeline.
<code>logger</code>	The logger instance.

Exceptions

<code>ArgumentNullException</code>	Thrown if next or logger is null.
------------------------------------	-----------------------------------

6.76.3 Member Function Documentation

6.76.3.1 `InvokeAsync()`

```
async Task ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware.InvokeAsync ( 
    HttpContext context ) [inline]
```

Processes an HTTP request within the middleware pipeline.

Executes the next middleware and handles any unhandled exceptions that occur by logging them and generating a standardized JSON error response.

Parameters

<code>context</code>	The HttpContext for the current request.
----------------------	--

Returns

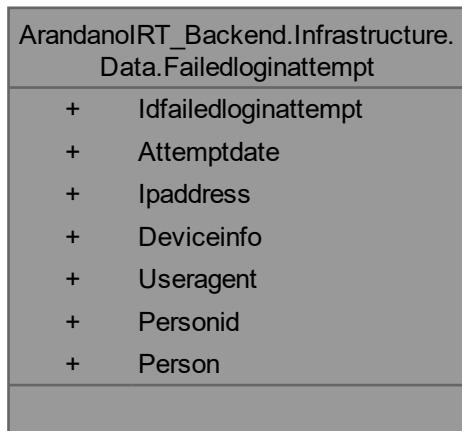
A Task representing the processing of the request.

The documentation for this class was generated from the following file:

- UI/Middleware/[ExceptionHandlingMiddleware.cs](#)

6.77 ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt:

**Properties**

- int `Idfailedloginattempt` [get, set]
- DateTime `Attemptdate` [get, set]
- string `Ipadress` = null! [get, set]
- string `Deviceinfo` = null! [get, set]
- string `Useragent` = null! [get, set]
- int? `Personid` [get, set]
- virtual ? Person `Person` [get, set]

6.77.1 Property Documentation

6.77.1.1 Attemptdate

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Attemptdate [get], [set]
```

6.77.1.2 Deviceinfo

```
string ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Deviceinfo = null! [get], [set]
```

6.77.1.3 Idfailedloginattempt

```
int ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Idfailedloginattempt [get], [set]
```

6.77.1.4 Ipaddress

```
string ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Ipaddress = null! [get], [set]
```

6.77.1.5 Person

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Person [get], [set]
```

6.77.1.6 Personid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Personid [get], [set]
```

6.77.1.7 Useragent

```
string ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt.Useragent = null! [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Failedloginattempt.cs](#)

6.78 ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity Class Reference

Represents a record logging a single failed login attempt against a specific user account.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity:

ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity
<ul style="list-style-type: none"> + IdFailedLoginAttempt + AttemptDate + IpAddress + DeviceInfo + UserAgent + PersonId + FailedLoginAttemptEntity()

Public Member Functions

- [FailedLoginAttemptEntity](#) (int idFailedLoginAttempt, DateTime attemptDate, string ipAddress, string deviceInfo, string userAgent, int personId)

Initializes a new instance of the FailedLoginAttemptEntity class.

Properties

- int [IdFailedLoginAttempt](#) [get]
Gets the unique identifier for this failed login attempt record.
- DateTime [AttemptDate](#) [get]
Gets the date and time (usually UTC) when the failed login attempt occurred.
- string [IpAddress](#) [get]
Gets the IP address from which the failed login attempt originated.
- string [DeviceInfo](#) [get]
Gets descriptive information about the device used for the failed login attempt.
- string [UserAgent](#) [get]
Gets the user agent string of the client used for the failed login attempt.
- int [PersonId](#) [get]
Gets the unique identifier of the person (user) account that was targeted by the failed login attempt.

6.78.1 Detailed Description

Represents a record logging a single failed login attempt against a specific user account.

Used for security monitoring and potentially implementing lockout mechanisms.

6.78.2 Constructor & Destructor Documentation

6.78.2.1 FailedLoginAttemptEntity()

```
ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.FailedLoginAttemptEntity (
    int idFailedLoginAttempt,
    DateTime attemptDate,
    string ipAddress,
    string deviceInfo,
    string userAgent,
    int personId ) [inline]
```

Initializes a new instance of the FailedLoginAttemptEntity class.

Parameters

<i>idFailedLoginAttempt</i>	The unique ID for this record (often generated by the database).
<i>attemptDate</i>	The timestamp of the failed attempt.
<i>ipAddress</i>	The source IP address.
<i>deviceInfo</i>	Information about the device used.
<i>userAgent</i>	The user agent string of the client.
<i>personId</i>	The ID of the user account targeted.

Exceptions

<i>ArgumentException</i>	Thrown if <i>ipAddress</i> is null or whitespace (Spanish message: "La dirección IP es requerida."). Thrown if <i>deviceInfo</i> is null or whitespace (Spanish message: "La información del dispositivo es requerida."). Thrown if <i>userAgent</i> is null or whitespace (Spanish message: "El user agent es requerido.").
--------------------------	--

6.78.3 Property Documentation

6.78.3.1 AttemptDate

```
DateTime ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.AttemptDate [get]
```

Gets the date and time (usually UTC) when the failed login attempt occurred.

6.78.3.2 DeviceInfo

```
string ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.DeviceInfo [get]
```

Gets descriptive information about the device used for the failed login attempt.

6.78.3.3 IdFailedLoginAttempt

```
int ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.IdFailedLoginAttempt [get]
```

Gets the unique identifier for this failed login attempt record.

6.78.3.4 IpAddress

```
string ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.IpAddress [get]
```

Gets the IP address from which the failed login attempt originated.

6.78.3.5 PersonId

```
int ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.PersonId [get]
```

Gets the unique identifier of the person (user) account that was targeted by the failed login attempt.

6.78.3.6 UserAgent

```
string ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity.UserAgent [get]
```

Gets the user agent string of the client used for the failed login attempt.

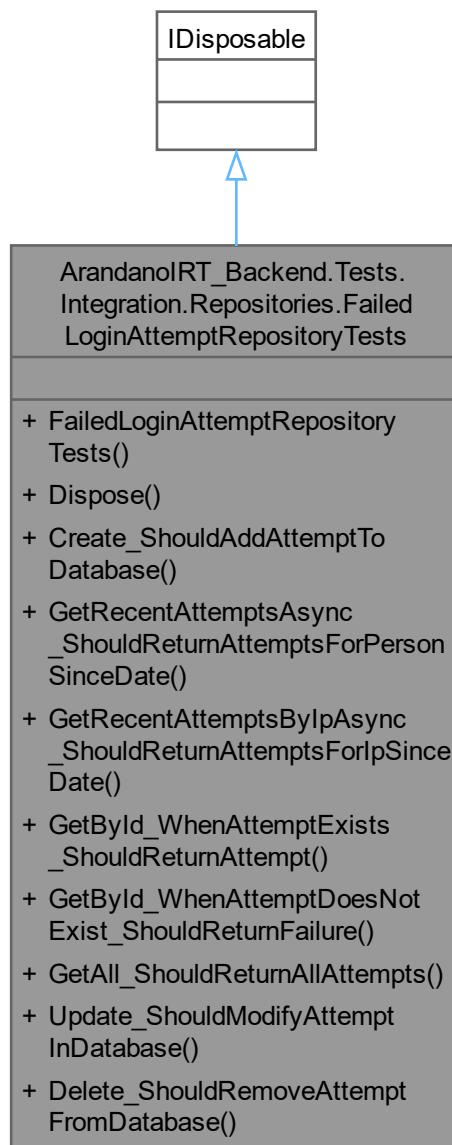
The documentation for this class was generated from the following file:

- Domain/Entities/[FailedLoginAttemptEntity.cs](#)

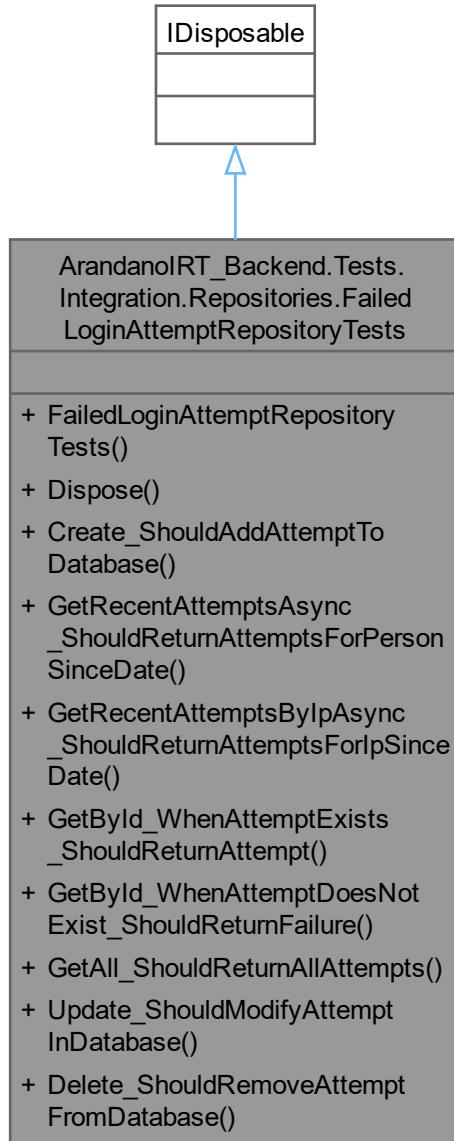
6.79 ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests Class Reference

Contains integration tests for the FailedLoginAttemptRepository using an SQLite in-memory database.

Inheritance diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests:



Public Member Functions

- [FailedLoginAttemptRepositoryTests \(\)](#)

Initializes a new instance of the FailedLoginAttemptRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

- [void Dispose \(\)](#)

Disposes the database context and connection used by the test instance.

- `async Task Create_ShouldAddAttemptToDatabase ()`
- `async Task GetRecentAttemptsAsync_ShouldReturnAttemptsForPersonSinceDate ()`
- `async Task GetRecentAttemptsByIpAsync_ShouldReturnAttemptsForIpSinceDate ()`
- `async Task GetById_WhenAttemptExists_ShouldReturnAttempt ()`
- `async Task GetById_WhenAttemptDoesNotExist_ShouldReturnFailure ()`
- `async Task GetAll_ShouldReturnAllAttempts ()`
- `async Task Update_ShouldModifyAttemptInDatabase ()`
- `async Task Delete_ShouldRemoveAttemptFromDatabase ()`

6.79.1 Detailed Description

Contains integration tests for the FailedLoginAttemptRepository using an SQLite in-memory database.

6.79.2 Constructor & Destructor Documentation

6.79.2.1 FailedLoginAttemptRepositoryTests()

```
ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.FailedLoginAttemptRepositoryTests ( ) [inline]
```

Initializes a new instance of the FailedLoginAttemptRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

6.79.3 Member Function Documentation

6.79.3.1 Create_ShouldAddAttemptToDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.Create_ShouldAddAttemptToDatabase ( ) [inline]
```

6.79.3.2 Delete_ShouldRemoveAttemptFromDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.Delete_ShouldRemoveAttemptFromDatabase ( ) [inline]
```

6.79.3.3 Dispose()

```
void ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.Dispose ( ) [inline]
```

Disposes the database context and connection used by the test instance.

6.79.3.4 GetAll_ShouldReturnAllAttempts()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
GetAll_ShouldReturnAllAttempts () [inline]
```

6.79.3.5 GetById_WhenAttemptDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
GetById_WhenAttemptDoesNotExist_ShouldReturnFailure () [inline]
```

6.79.3.6 GetById_WhenAttemptExists_ShouldReturnAttempt()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
GetById_WhenAttemptExists_ShouldReturnAttempt () [inline]
```

6.79.3.7 GetRecentAttemptsAsync_ShouldReturnAttemptsForPersonSinceDate()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
GetRecentAttemptsAsync_ShouldReturnAttemptsForPersonSinceDate () [inline]
```

6.79.3.8 GetRecentAttemptsByIpAsync_ShouldReturnAttemptsForIpSinceDate()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
GetRecentAttemptsByIpAsync_ShouldReturnAttemptsForIpSinceDate () [inline]
```

6.79.3.9 Update_ShouldModifyAttemptInDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests.←  
Update_ShouldModifyAttemptInDatabase () [inline]
```

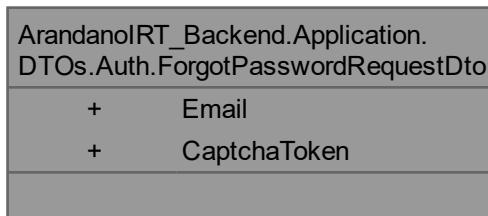
The documentation for this class was generated from the following file:

- Tests/Integration/Repositories/[FailedLoginAttemptRepositoryTests.cs](#)

6.80 ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto Class Reference

Represents the request data for initiating the password recovery process.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto:



Properties

- required string [Email](#) [get, set]
Gets or sets the email address of the user requesting the password reset.
- string? [CaptchaToken](#) [get, set]
Gets or sets the optional CAPTCHA verification token provided by the user.

6.80.1 Detailed Description

Represents the request data for initiating the password recovery process.

6.80.2 Property Documentation

6.80.2.1 CaptchaToken

```
string? ArandanoIRT_Backend_ApplicationDTOsAuthForgotPasswordRequestDto.CaptchaToken [get],  
[set]
```

Gets or sets the optional CAPTCHA verification token provided by the user.

This may be required depending on security configurations.

6.80.2.2 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto.Email [get, set]
```

Gets or sets the email address of the user requesting the password reset.

The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[ForgotPasswordRequestDto.cs](#)

6.81 ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto Class Reference

Represents the data submitted through a help or support request form.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto:

ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto
+ Name + Email + Subject + CropName + Message + CaptchaToken

Properties

- required string [Name](#) [get, set]

Gets or sets the name of the person submitting the help request.
- required string [Email](#) [get, set]

Gets or sets the email address of the person submitting the help request.
- required string [Subject](#) [get, set]

Gets or sets the subject line of the help request.
- required string [CropName](#) [get, set]

Gets or sets the name of the crop related to the help request.
- required string [Message](#) [get, set]

Gets or sets the detailed message or description of the issue or question provided by the user.
- string? [CaptchaToken](#) [get, set]

Gets or sets the optional CAPTCHA verification token provided by the user.

6.81.1 Detailed Description

Represents the data submitted through a help or support request form.

6.81.2 Property Documentation

6.81.2.1 CaptchaToken

```
string? ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.CaptchaToken [get], [set]
```

Gets or sets the optional CAPTCHA verification token provided by the user.

This may be required depending on security configurations or the nature of the request.

6.81.2.2 CropName

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.CropName [get], [set]
```

Gets or sets the name of the crop related to the help request.

6.81.2.3 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.Email [get], [set]
```

Gets or sets the email address of the person submitting the help request.

This is typically used for follow-up communication.

6.81.2.4 Message

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.Message [get], [set]
```

Gets or sets the detailed message or description of the issue or question provided by the user.

6.81.2.5 Name

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.Name [get], [set]
```

Gets or sets the name of the person submitting the help request.

string

//

6.81.2.6 Subject

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto.Subject [get], [set]
```

Gets or sets the subject line of the help request.

string

//

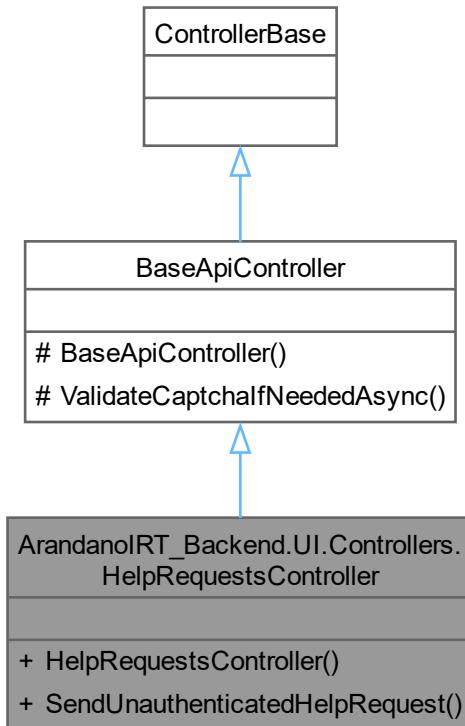
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[HelpRequestDto.cs](#)

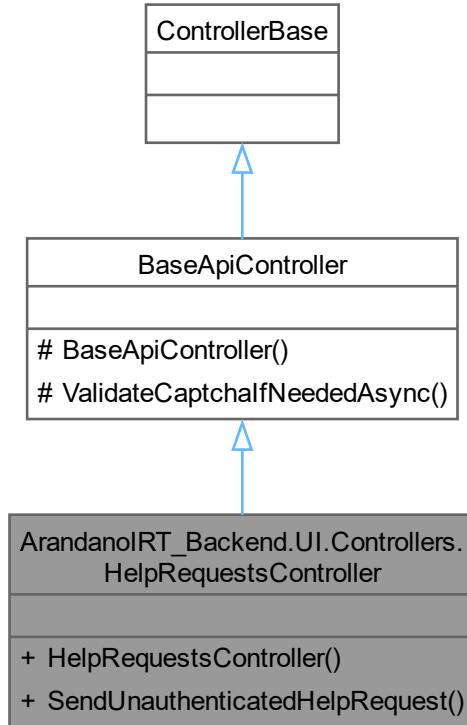
6.82 ArandanoIRT_Backend.UI.Controllers.HelpRequestsController Class Reference

Handles endpoints related to user help requests.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.HelpRequestsController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.HelpRequestsController:



Public Member Functions

- **HelpRequestsController** (`IHelpRequestService` helpRequestService, `ICaptchaService` captchaService, `IWebHostEnvironment` environment, `ILogger<HelpRequestsController>` logger)

Initializes a new instance of the HelpRequestsController class.

- `async Task< IActionResult > SendUnauthenticatedHelpRequest ([FromBody] HelpRequestDto request)`

Submits a help request from an unauthenticated user.

Additional Inherited Members

Protected Member Functions inherited from ArandanoIRT_Backend.UI.Controllers.BaseApiController

- **BaseApiController** (`ICaptchaService` captchaService, `IWebHostEnvironment` environment)

Initializes a new instance of the BaseApiController class.

- `async Task< IActionResult?> ValidateCaptchaNeededAsync (string? captchaToken, ILogger logger)`

Validates the CAPTCHA token if the application is not running in the Development environment.

6.82.1 Detailed Description

Handles endpoints related to user help requests.

6.82.2 Constructor & Destructor Documentation

6.82.2.1 HelpRequestsController()

```
ArandanoIRT_Backend.UI.Controllers.HelpRequestsController.HelpRequestsController (
    IHelpRequestService helpRequestService,
    ICaptchaService captchaService,
    IWebHostEnvironment environment,
    ILogger< HelpRequestsController > logger ) [inline]
```

Initializes a new instance of the HelpRequestsController class.

Parameters

<i>helpRequestService</i>	The application service for help request logic.
<i>captchaService</i>	The service for CAPTCHA validation (passed to base).
<i>environment</i>	The web hosting environment (passed to base).
<i>logger</i>	The logger for this controller.

6.82.3 Member Function Documentation

6.82.3.1 SendUnauthenticatedHelpRequest()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.HelpRequestsController.SendUnauthenticatedHelpRequest (
    [FromBody] HelpRequestDto request ) [inline]
```

Submits a help request from an unauthenticated user.

Validates the specified crop name exists and sends an email to the crop's administrator. Requires CAPTCHA validation outside of Development environment.

Parameters

<i>request</i>	Data for the help request (name, email, subject, crop name, message, optional captcha).
----------------	---

Returns

An IActionResult indicating success (204 No Content) or failure (400 Bad Request).

<response code="204">Help request sent successfully.</response> <response code="400">Invalid request (e.g., missing data, crop not found, invalid CAPTCHA, email send failed).</response>

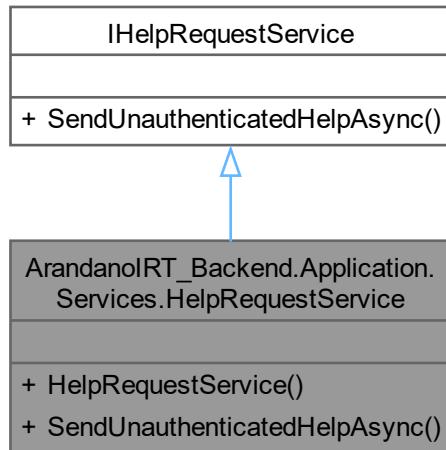
The documentation for this class was generated from the following file:

- UI/Controllers/[HelpRequestsController.cs](#)

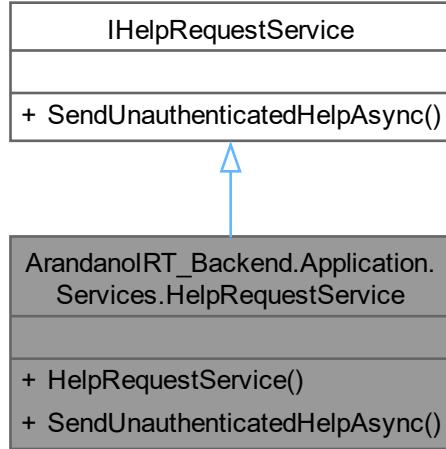
6.83 ArandanoIRT_Backend.Application.Services.HelpRequestService Class Reference

Implements services for handling user help requests.

Inheritance diagram for ArandanoIRT_Backend.Application.Services.HelpRequestService:



Collaboration diagram for ArandanoIRT_Backend.Application.Services.HelpRequestService:



Public Member Functions

- `HelpRequestService (ICropRepository cropRepository, IPersonRepository personRepository, IEmailService emailService, ILogger< HelpRequestService > logger)`
- `async Task< Result > SendUnauthenticatedHelpAsync (HelpRequestDto request)`

Processes a help request submitted by an unauthenticated user.

Validates the associated crop name and sends an email notification to the crop's administrator.

Parameters

<code>request</code>	<i>The help request data transfer object containing user details, message, and crop name.</i>
----------------------	---

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

6.83.1 Detailed Description

Implements services for handling user help requests.

6.83.2 Constructor & Destructor Documentation

6.83.2.1 HelpRequestService()

```
ArandanoIRT_Backend.Application.Services.HelpRequestService.HelpRequestService (
    ICropRepository cropRepository,
```

```
IPersonRepository personRepository,  
IEmailService emailService,  
ILogger< HelpRequestService > logger ) [inline]
```

6.83.3 Member Function Documentation

6.83.3.1 SendUnauthenticatedHelpAsync()

```
async Task< Result > ArandanoIRT_Backend.Application.Services.HelpRequestService.SendUnauthenticatedHelpAsync (   
    HelpRequestDto request ) [inline]
```

Processes a help request submitted by an unauthenticated user.

Validates the associated crop name and sends an email notification to the crop's administrator.

Parameters

<code>request</code>	The help request data transfer object containing user details, message, and crop name.
----------------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implements [ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService](#).

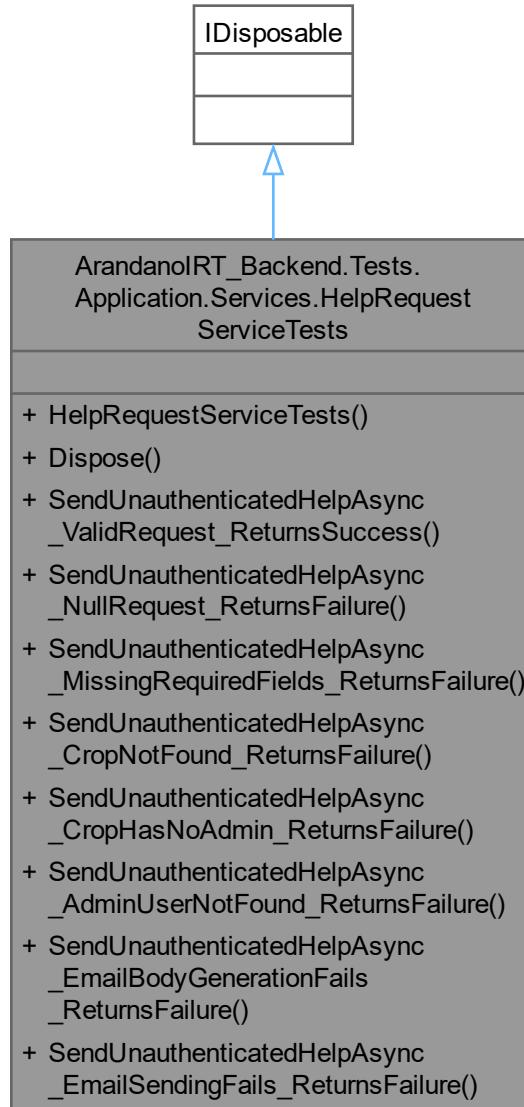
The documentation for this class was generated from the following file:

- Application/Services/[HelpRequestService.cs](#)

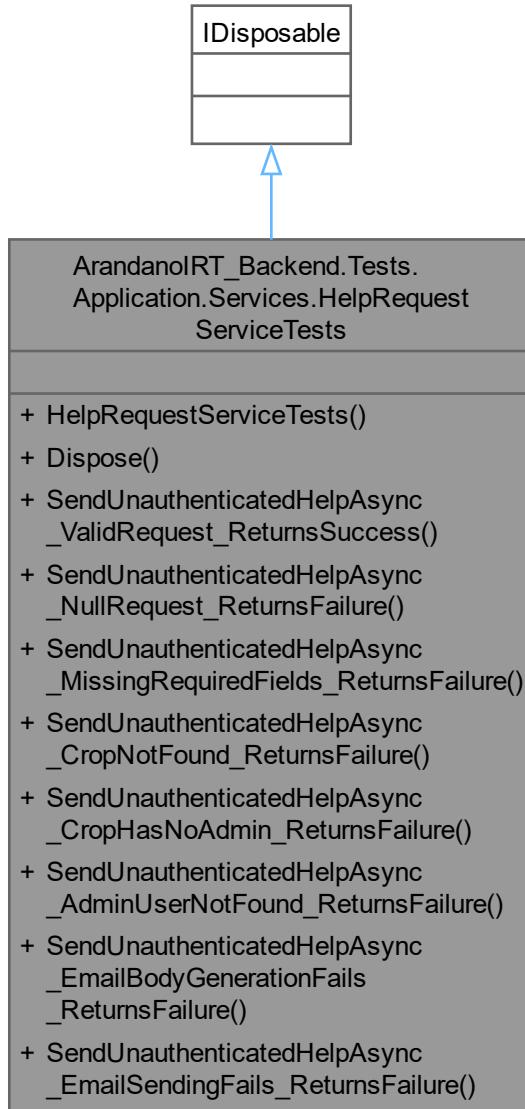
6.84 ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests Class Reference

Contains unit tests for the HelpRequestService class.

Inheritance diagram for ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests:



Public Member Functions

- [HelpRequestServiceTests \(\)](#)

Initializes a new instance of the HelpRequestServiceTests class, setting up all mock dependencies and the service under test.

- [void Dispose \(\)](#)

Disposes resources used by the test context, if any.

- async Task `SendUnauthenticatedHelpAsync_ValidRequest_ReturnsSuccess ()`
- async Task `SendUnauthenticatedHelpAsync_NullRequest_ReturnsFailure ()`
- async Task `SendUnauthenticatedHelpAsync_MissingRequiredFields_ReturnsFailure (string name, string email, string subject, string cropName, string message)`
- async Task `SendUnauthenticatedHelpAsync_CropNotFound_ReturnsFailure ()`
- async Task `SendUnauthenticatedHelpAsync_CropHasNoAdmin_ReturnsFailure ()`
- async Task `SendUnauthenticatedHelpAsync_AdminUserNotFound_ReturnsFailure ()`
- async Task `SendUnauthenticatedHelpAsync_EmailBodyGenerationFails_ReturnsFailure ()`
- async Task `SendUnauthenticatedHelpAsync_EmailSendingFails_ReturnsFailure ()`

6.84.1 Detailed Description

Contains unit tests for the HelpRequestService class.

6.84.2 Constructor & Destructor Documentation

6.84.2.1 HelpRequestServiceTests()

```
ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.HelpRequestServiceTests ( )  
[inline]
```

Initializes a new instance of the HelpRequestServiceTests class, setting up all mock dependencies and the service under test.

6.84.3 Member Function Documentation

6.84.3.1 Dispose()

```
void ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.Dispose ( ) [inline]
```

Disposes resources used by the test context, if any.

6.84.3.2 SendUnauthenticatedHelpAsync_AdminUserNotFound_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated←  
HelpAsync_AdminUserNotFound_ReturnsFailure ( ) [inline]
```

6.84.3.3 SendUnauthenticatedHelpAsync_CropHasNoAdmin_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated←  
HelpAsync_CropHasNoAdmin_ReturnsFailure ( ) [inline]
```

6.84.3.4 SendUnauthenticatedHelpAsync_CropNotFound_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_CropNotFound_ReturnsFailure ( ) [inline]
```

6.84.3.5 SendUnauthenticatedHelpAsync_EmailBodyGenerationFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_EmailBodyGenerationFails_ReturnsFailure ( ) [inline]
```

6.84.3.6 SendUnauthenticatedHelpAsync_EmailSendingFails_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_EmailSendingFails_ReturnsFailure ( ) [inline]
```

6.84.3.7 SendUnauthenticatedHelpAsync_MissingRequiredFields_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_MissingRequiredFields_ReturnsFailure (
    string name,
    string email,
    string subject,
    string cropName,
    string message ) [inline]
```

6.84.3.8 SendUnauthenticatedHelpAsync_NullRequest_ReturnsFailure()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_NullRequest_ReturnsFailure ( ) [inline]
```

6.84.3.9 SendUnauthenticatedHelpAsync_ValidRequest_ReturnsSuccess()

```
async Task ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests.SendUnauthenticated<→
HelpAsync_ValidRequest_ReturnsSuccess ( ) [inline]
```

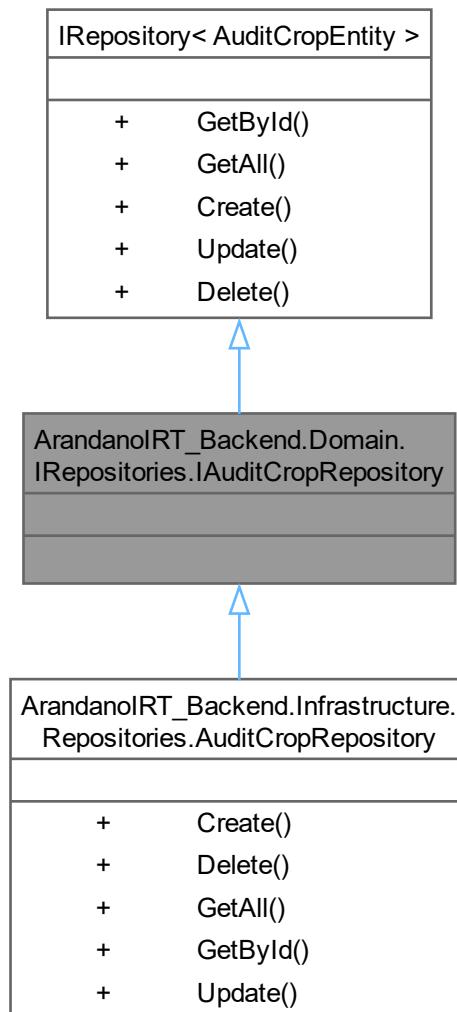
The documentation for this class was generated from the following file:

- Tests/Unit/Application/Services/[HelpRequestServiceTests.cs](#)

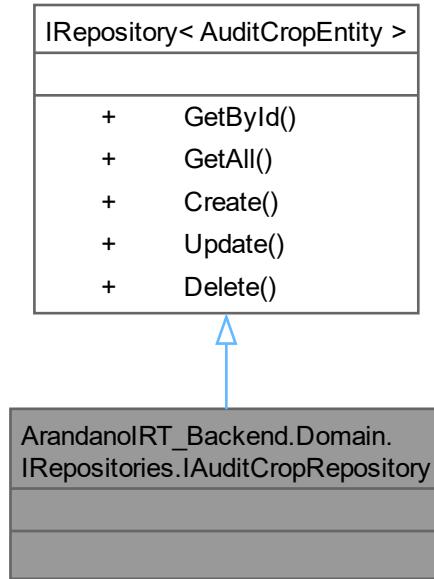
6.85 ArandanoIRT_Backend.Domain.IRepositories.IAuditCropRepository Interface Reference

Defines the repository contract specifically for managing AuditCropEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditCropRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditCropRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditCropEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.85.1 Detailed Description

Defines the repository contract specifically for managing AuditCropEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

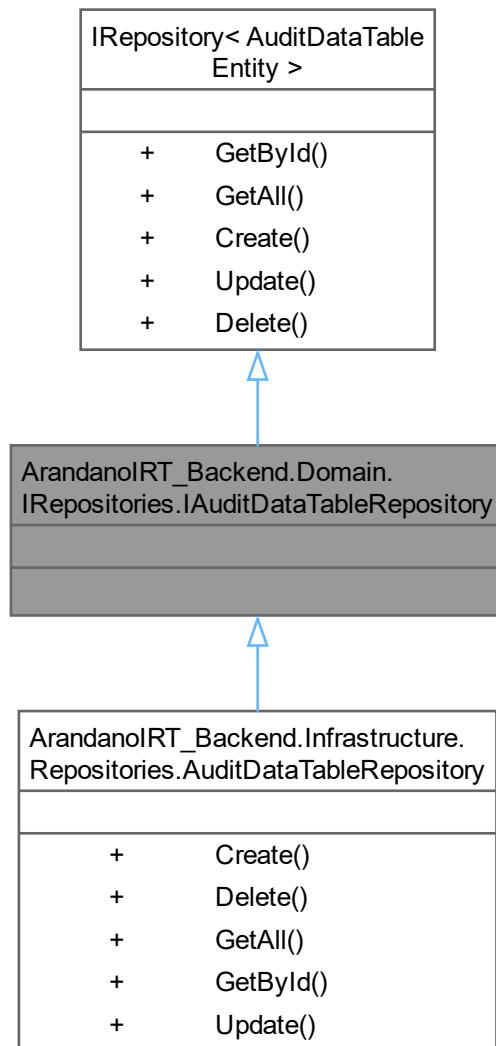
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IAuditCropRepository.cs](#)

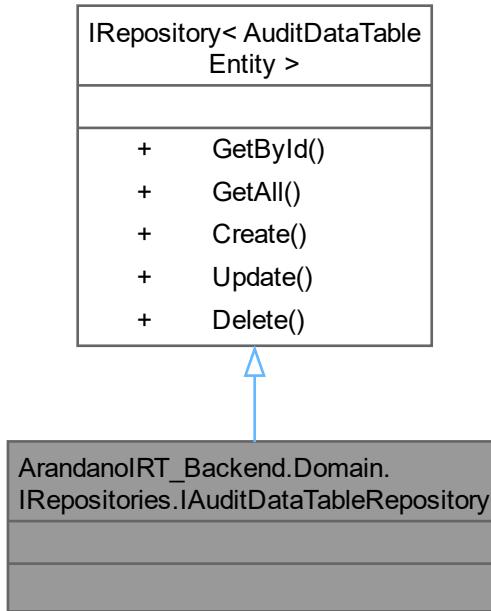
6.86 ArandanoIRT_Backend.Domain.IRepositories.IAuditDataTable ← Repository Interface Reference

Defines the repository contract specifically for managing AuditDataTableEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditDataTableRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditDataTableRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditDataTableEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.86.1 Detailed Description

Defines the repository contract specifically for managing AuditDataTableEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

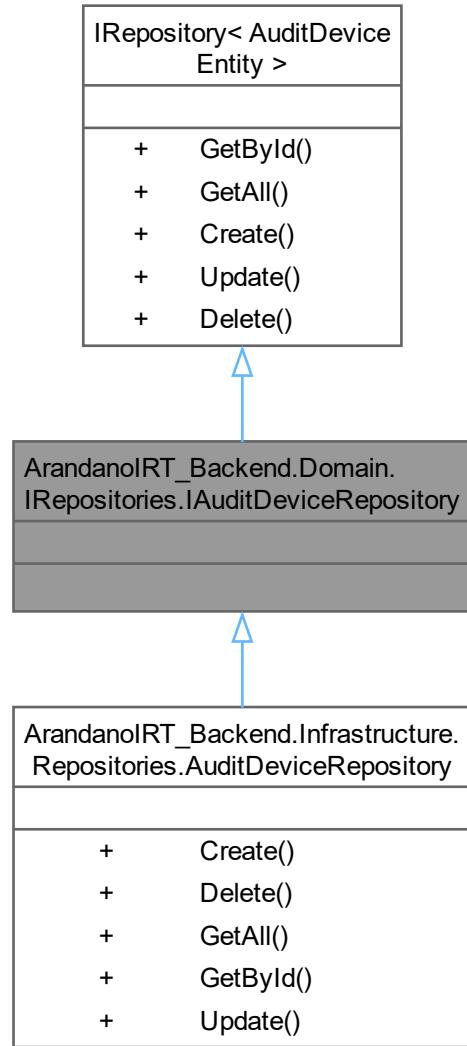
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IAuditDataTableRepository.cs](#)

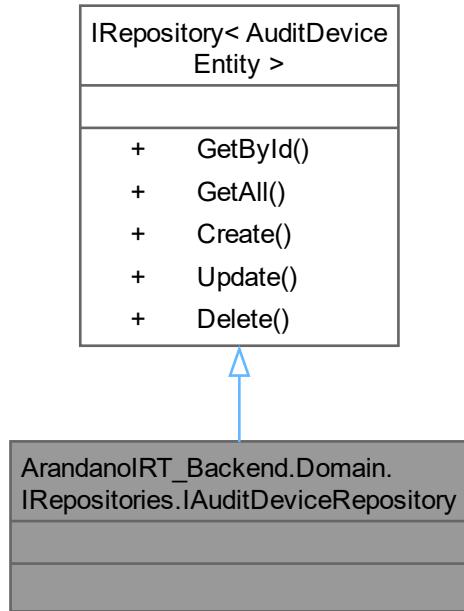
6.87 ArandanoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository Interface Reference

Defines the repository contract specifically for managing AuditDeviceEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditDeviceEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.87.1 Detailed Description

Defines the repository contract specifically for managing AuditDeviceEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

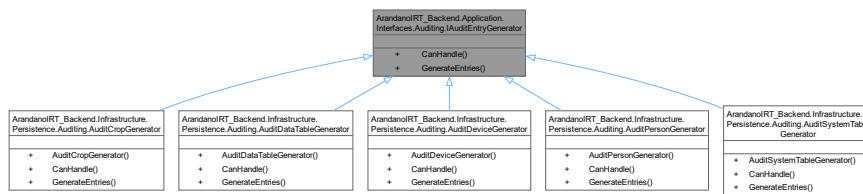
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IAuditDeviceRepository.cs

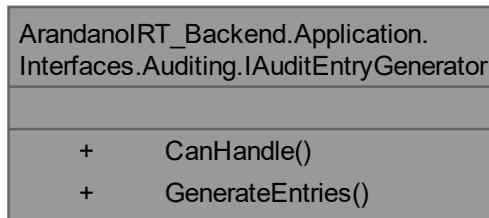
6.88 ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator ← Generator Interface Reference

Defines a contract for components responsible for generating audit log entries based on entity changes tracked by Entity Framework Core.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator:



Public Member Functions

- bool [CanHandle](#) (EntityEntry entry)
Determines whether this generator can process the specified entity entry for auditing.
- IEnumerable< object > [GenerateEntries](#) (EntityEntry entry, [AuditMetadata](#) metadata)
Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

6.88.1 Detailed Description

Defines a contract for components responsible for generating audit log entries based on entity changes tracked by Entity Framework Core.

6.88.2 Member Function Documentation

6.88.2.1 CanHandle()

```
bool ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator.CanHandle (
    EntityEntry entry )
```

Determines whether this generator can process the specified entity entry for auditing.

Parameters

<code>entry</code>	The Entity Framework Core EntityEntry representing the tracked change.
--------------------	--

Returns

`true` if this generator can handle the entry; otherwise, `false`.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator](#), [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator](#), [ArandanoIRT_Backend.Infrastructure.Persistence.AuditSystemTableGenerator](#) and [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator](#).

6.88.2.2 GenerateEntries()

```
IEnumerable< object > ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator.←
GenerateEntries (
    EntityEntry entry,
    AuditMetadata metadata )
```

Generates one or more audit log entries based on the changes in the specified entity entry and the provided metadata.

Parameters

<code>entry</code>	The EntityEntry containing the details of the entity change (e.g., original and current values).
<code>metadata</code>	The AuditMetadata containing context about the action being audited (user, time, etc.).

Returns

An `IEnumerable<T>` collection of objects, where each object represents a distinct audit log entry to be persisted. The specific type of these objects depends on the implementation.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator](#), [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator](#), [ArandanoIRT_Backend.Infrastructure.Persistence.AuditTableGenerator](#) and [ArandanoIRT_Backend.Infrastructure.Persistence.AuditSystemTableGenerator](#).

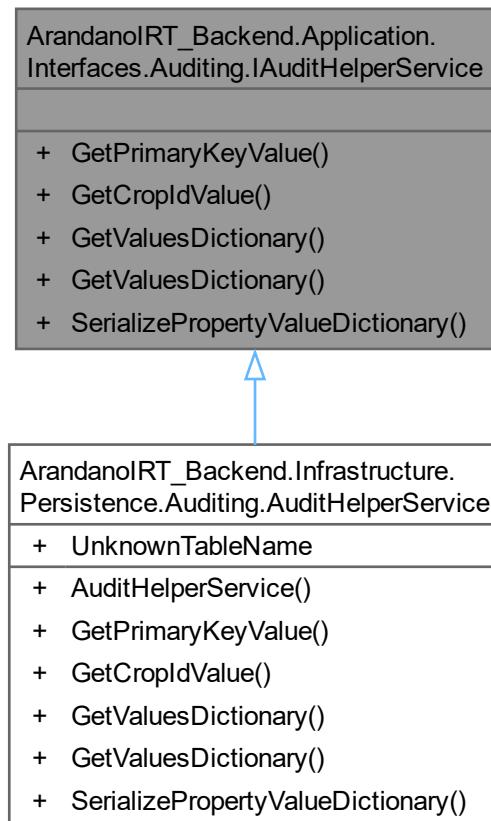
The documentation for this interface was generated from the following file:

- Application/Interfaces/Auditing/IAuditEntryGenerator.cs

6.89 ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelper ← Service Interface Reference

Defines a contract for a helper service providing common utility functions used during the audit logging process, particularly with Entity Framework Core entities.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService:

ArandanoIRT_Backend.Application. Interfaces.Auditing.IAuditHelperService
+ GetPrimaryKeyValue() + GetCropIdValue() + GetValuesDictionary() + GetValuesDictionary() + SerializePropertyValueDictionary()

Public Member Functions

- int? [GetPrimaryKeyValue](#) (EntityEntry entry)
Gets the primary key value of the entity represented by the specified entry, assuming the primary key is a single integer property.
- int? [GetCropIdValue](#) (EntityEntry entry)
Gets the 'CropId' value (or equivalent identifier related to a crop) from the entity represented by the specified entry, if available.
- IDictionary< string, object?>? [GetValuesDictionary](#) (PropertyValues? propertyValues)
Creates a dictionary mapping property names to their values from the provided PropertyValues object, excluding a predefined set of sensitive properties (like 'Password').
- IDictionary< string, object?>? [GetValuesDictionary](#) (PropertyValues? propertyValues, IEnumerable< string >? additionalExcludedProperties)
Creates a dictionary mapping property names to their values from the provided PropertyValues object, excluding both a predefined set of sensitive properties and any additionally specified properties.
- string? [SerializePropertyValueDictionary](#) (IDictionary< string, object?>? values)
Serializes the provided dictionary of property values into a string representation, typically JSON format.

6.89.1 Detailed Description

Defines a contract for a helper service providing common utility functions used during the audit logging process, particularly with Entity Framework Core entities.

6.89.2 Member Function Documentation

6.89.2.1 GetCropIdValue()

```
int? ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService.GetCropIdValue (
    EntityEntry entry )
```

Gets the 'CropId' value (or equivalent identifier related to a crop) from the entity represented by the specified entry, if available.

Parameters

<i>entry</i>	The EntityEntry for the tracked entity.
--------------	---

Returns

The integer 'CropId' value, or `null` if the property doesn't exist, is not an integer, or its value is null.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#).

6.89.2.2 GetPrimaryKeyValue()

```
int? ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService.GetPrimaryKeyValue (
    EntityEntry entry )
```

Gets the primary key value of the entity represented by the specified entry, assuming the primary key is a single integer property.

Parameters

<i>entry</i>	The EntityEntry for the tracked entity.
--------------	---

Returns

The integer primary key value of the entity if found and is a single integer key, otherwise `null`.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#).

6.89.2.3 GetValuesDictionary() [1/2]

```
IDictionary< string, object?>? ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService.GetValuesDictionary (
    PropertyValues? propertyValues )
```

Creates a dictionary mapping property names to their values from the provided `PropertyValues` object, excluding a predefined set of sensitive properties (like 'Password').

Parameters

<i>propertyValues</i>	The <code>PropertyValues</code> object (e.g., <code>OriginalValues</code> , <code>CurrentValues</code>) containing the entity's property values. Can be <code>null</code> .
-----------------------	--

Returns

A dictionary where keys are property names and values are the corresponding property values (as `object?`), excluding default sensitive properties. Returns `null` if `propertyValues` is `null` or the resulting dictionary is empty after exclusions.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#).

6.89.2.4 GetValuesDictionary() [2/2]

```
IDictionary< string, object?>? ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService.GetValuesDictionary (
    PropertyValues? propertyValues,
    IEnumerable< string >? additionalExcludedProperties )
```

Creates a dictionary mapping property names to their values from the provided `PropertyValues` object, excluding both a predefined set of sensitive properties and any additionally specified properties.

Parameters

<code>propertyValues</code>	The <code>PropertyValues</code> object (e.g., <code>OriginalValues</code> , <code>CurrentValues</code>) containing the entity's property values. Can be <code>null</code> .
<code>additionalExcludedProperties</code>	An optional collection of additional property names to exclude from the resulting dictionary, beyond the default exclusions. Can be <code>null</code> or empty.

//

Returns

A dictionary where keys are property names and values are the corresponding property values (as `object?`), excluding default and specified properties. Returns `null` if `propertyValues` is `null` or the resulting dictionary is empty after exclusions.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#).

6.89.2.5 SerializePropertyValueDictionary()

```
string? ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService.SerializePropertyValueDictionary (
    IDictionary< string, object?>? values )
```

Serializes the provided dictionary of property values into a string representation, typically JSON format.

Parameters

<code>values</code>	The dictionary containing property names and their values. Can be <code>null</code> .
---------------------	---

Returns

A string representation of the dictionary (e.g., JSON formatted), or `null` if the input dictionary is `null` or empty.
Returns an error placeholder string on serialization failure.

Implemented in [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#).

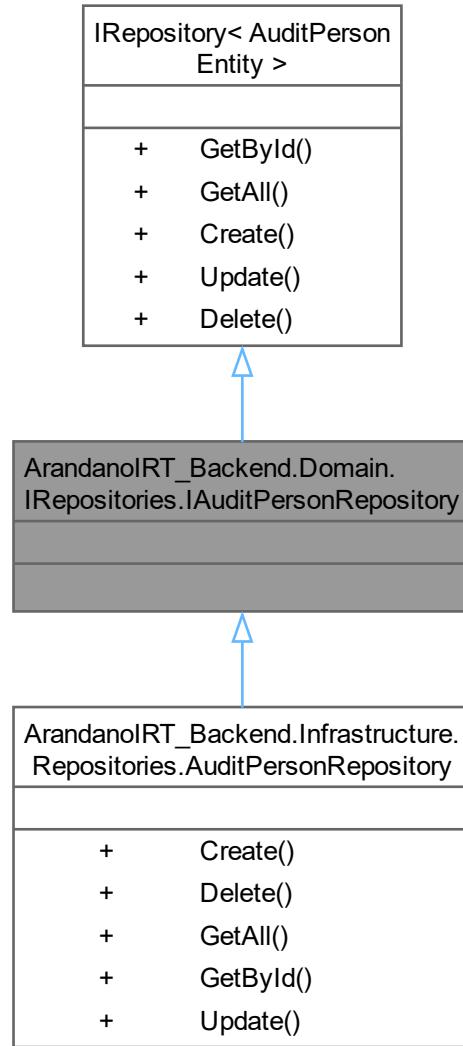
The documentation for this interface was generated from the following file:

- Application/Interfaces/Auditing/[IAuditHelperService.cs](#)

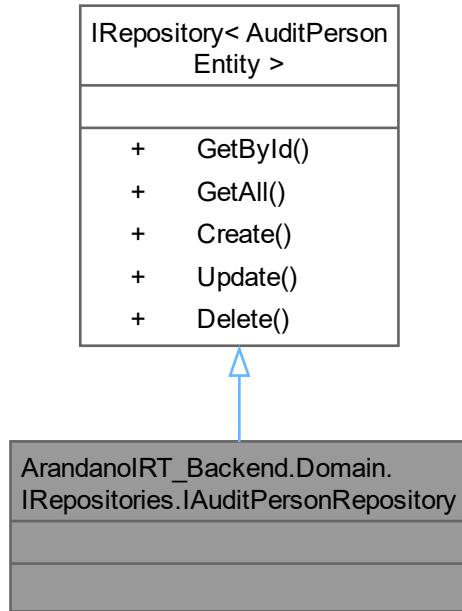
6.90 ArandanoIRT_Backend.Domain.IRepositories.IAuditPersonRepository Interface Reference

Defines the repository contract specifically for managing `AuditPersonEntity` instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditPersonRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditPersonRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditPersonEntity >](#)

- Task< Result< T > > [GetBy`Id`](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.90.1 Detailed Description

Defines the repository contract specifically for managing AuditPersonEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

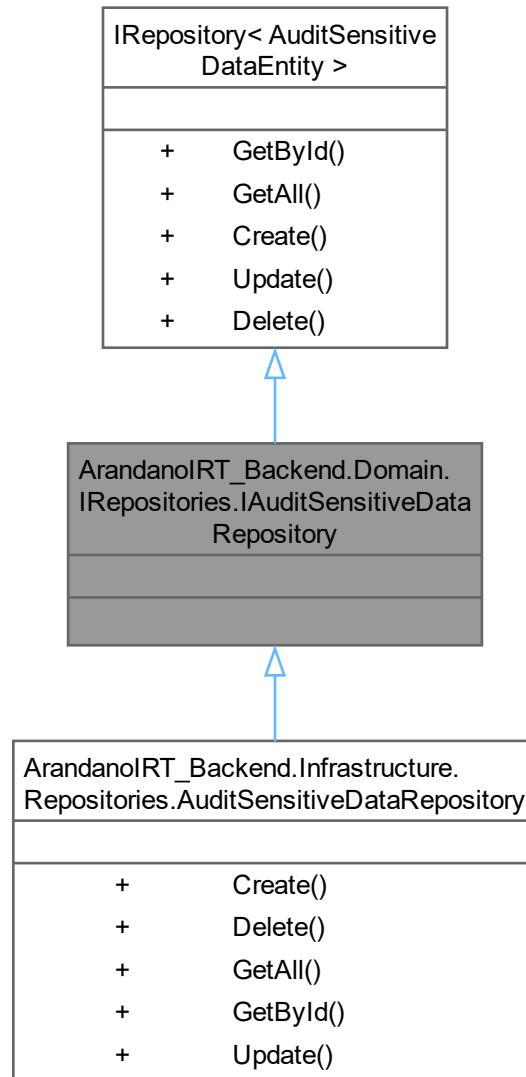
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IAuditPersonRepository.cs

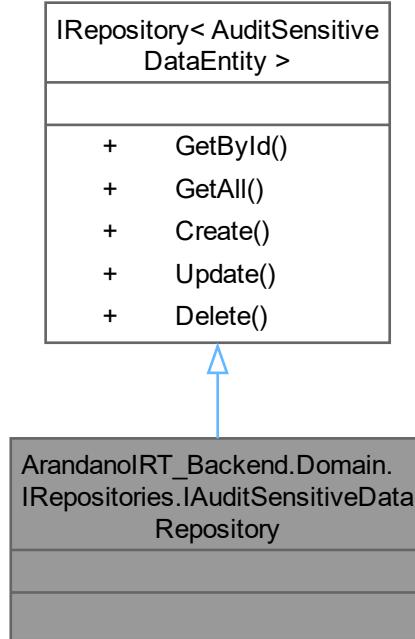
6.91 ArandanoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository Interface Reference

Defines the repository contract specifically for managing AuditSensitiveDataEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditSensitiveDataEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.91.1 Detailed Description

Defines the repository contract specifically for managing AuditSensitiveDataEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

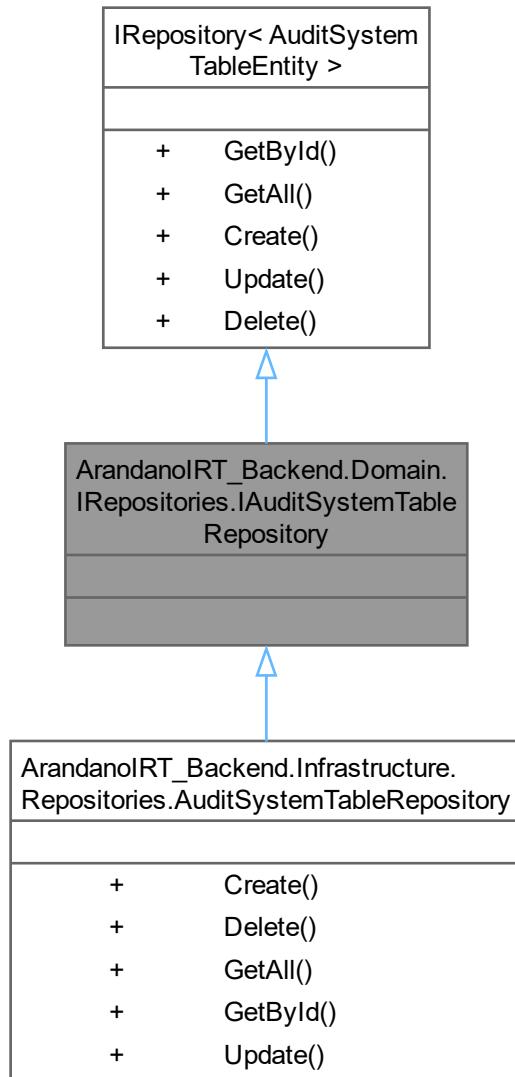
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IAuditSensitiveDataRepository.cs

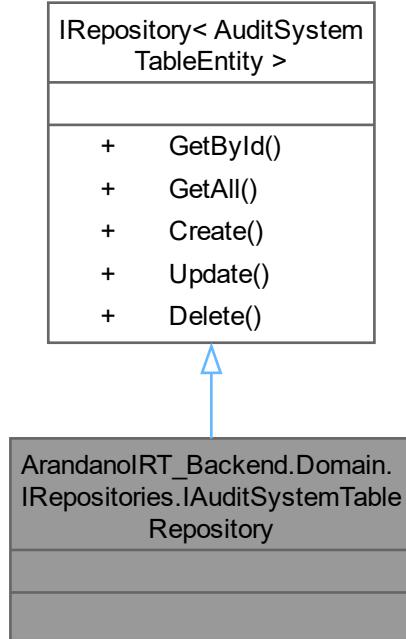
6.92 ArandanoIRT_Backend.Domain.IRepositories.IAuditSystemTable← Repository Interface Reference

Defines the repository contract specifically for managing AuditSystemTableEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditSystemTableRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IAuditSystemTableRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< AuditSystemTableEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.92.1 Detailed Description

Defines the repository contract specifically for managing AuditSystemTableEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

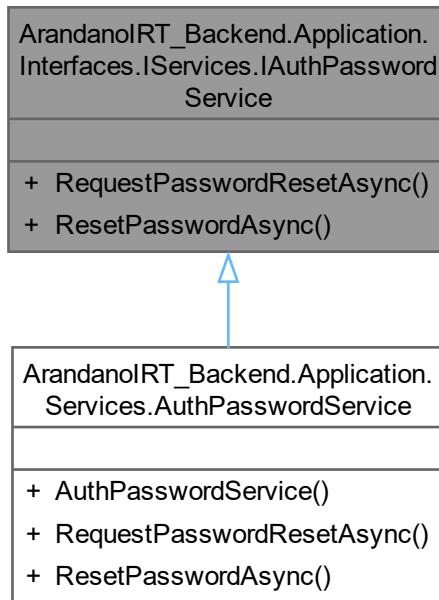
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IAuditSystemTableRepository.cs

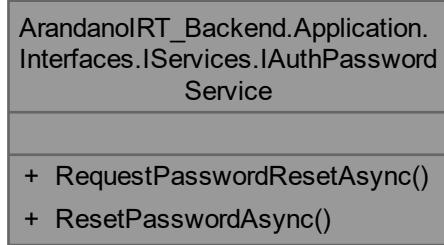
6.93 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService Interface Reference

Defines the contract for password management operations like reset requests and execution.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService:



Public Member Functions

- Task< [Result](#) > [RequestPasswordResetAsync](#) ([ForgotPasswordRequestDto](#) request)
Initiates the password reset process for a given email address.
- Task< [Result](#) > [ResetPasswordAsync](#) ([ChangePasswordRequestDto](#) request)
Resets the user's password using a valid reset token and a new password.

6.93.1 Detailed Description

Defines the contract for password management operations like reset requests and execution.

6.93.2 Member Function Documentation

6.93.2.1 RequestPasswordResetAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService.Request<-
PasswordResetAsync (
    ForgotPasswordRequestDto request )
```

Initiates the password reset process for a given email address.

Generates a reset token, stores it, and typically sends it via email. Protects against email enumeration attacks.

Parameters

<code>request</code>	DTO containing the user's email.
----------------------	----------------------------------

Returns

A Task representing the asynchronous operation. Always returns a success result to prevent email enumeration, even if the email doesn't exist or validation fails internally.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthPasswordService](#).

6.93.2.2 ResetPasswordAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService.Reset←  
PasswordAsync (   
    ChangePasswordRequestDto request )
```

Resets the user's password using a valid reset token and a new password.

Parameters

request	DTO containing the reset token, new password, and confirmation.
----------------	---

Returns

A Task representing the asynchronous operation, with a result indicating success or failure of the password reset.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthPasswordService](#).

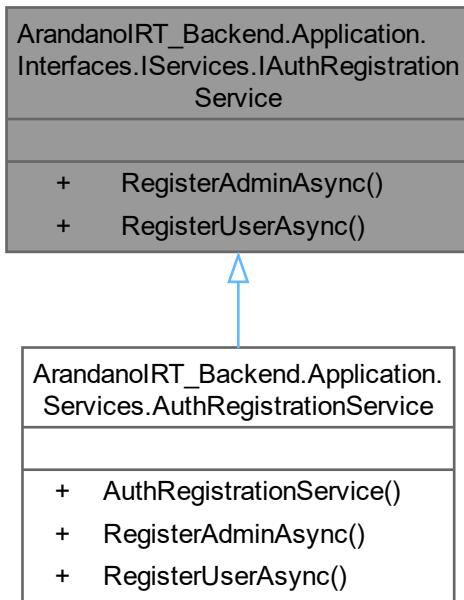
The documentation for this interface was generated from the following file:

- Application/Interfaces/IServices/IAuthPasswordService.cs

6.94 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuth← RegistrationService Interface Reference

Defines the contract for user registration operations.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService:



Public Member Functions

- Task< [Result](#) > [RegisterAdminAsync](#) ([RegisterAdminRequestDto](#) request)
Registers a new administrator user along with their initial crop.
- Task< [Result](#) > [RegisterUserAsync](#) ([RegisterUserRequestDto](#) request)
Registers a new regular user using an invitation code.

6.94.1 Detailed Description

Defines the contract for user registration operations.

6.94.2 Member Function Documentation

6.94.2.1 RegisterAdminAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService.←
RegisterAdminAsync (
    RegisterAdminRequestDto request )
```

Registers a new administrator user along with their initial crop.

Parameters

<code>request</code>	DTO containing admin and crop information.
----------------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthRegistrationService](#).

6.94.2.2 RegisterUserAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService.←
RegisterUserAsync (
    RegisterUserRequestDto request )
```

Registers a new regular user using an invitation code.

Parameters

<code>request</code>	DTO containing user information and the invitation code.
----------------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthRegistrationService](#).

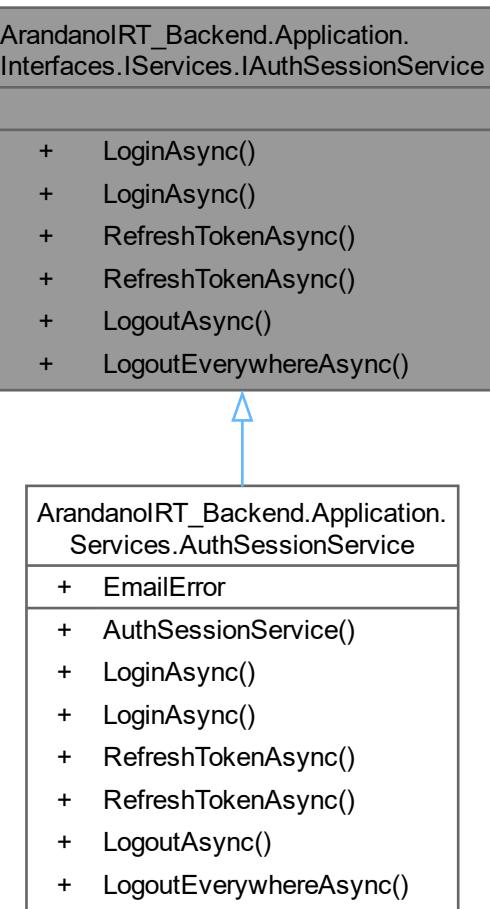
The documentation for this interface was generated from the following file:

- Application/Interfaces/IServices/IAuthRegistrationService.cs

6.95 ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService Interface Reference

Defines the contract for user session management operations, including login, token refresh, and logout.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService:

ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService
<ul style="list-style-type: none"> + LoginAsync() + LoginAsync() + RefreshTokenAsync() + RefreshTokenAsync() + LogoutAsync() + LogoutEverywhereAsync()

Public Member Functions

- Task< Result< LoginSuccessPayload > > [LoginAsync](#) (LoginRequestDto request, string ipAddress, string userAgent)

Authenticates a user based on email and password without device information.
- Task< Result< LoginSuccessPayload > > [LoginAsync](#) (LoginRequestDto request, string ipAddress, string userAgent, string? deviceInfo)

Authenticates a user based on email and password, including device information.
- Task< Result< TokenResponseDto > > [RefreshTokenAsync](#) (string refreshTokenValue, string ipAddress, string userAgent)

Refreshes the access token using a valid refresh token without device information.
- Task< Result< TokenResponseDto > > [RefreshTokenAsync](#) (string refreshTokenValue, string ipAddress, string userAgent, string? deviceInfo)

Refreshes the access token using a valid refresh token, including device information.
- Task< Result > [LogoutAsync](#) (string refreshTokenValue, string ipAddress)

Revokes the specific refresh token used for the current session (logout).
- Task< Result > [LogoutEverywhereAsync](#) (long sessionId, string ipAddress)

Revokes all active refresh tokens associated with a user's session ID (logout everywhere).

6.95.1 Detailed Description

Defines the contract for user session management operations, including login, token refresh, and logout.

6.95.2 Member Function Documentation

6.95.2.1 LoginAsync() [1/2]

```
Task< Result< LoginSuccessPayload > > ArandanoIRT_Backend.Application.Interfaces.IServices.←
IAuthSessionService.LoginAsync (
    LoginRequestDto request,
    string ipAddress,
    string userAgent )
```

Authenticates a user based on email and password without device information.

Parameters

<i>request</i>	DTO containing login credentials.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

6.95.2.2 LoginAsync() [2/2]

```
Task< Result< LoginSuccessPayload > > ArandanoIRT_Backend.Application.Interfaces.IServices.←
IAuthSessionService.LoginAsync (
    LoginRequestDto request,
    string ipAddress,
    string userAgent,
    string? deviceInfo )
```

Authenticates a user based on email and password, including device information.

Parameters

<i>request</i>	DTO containing login credentials.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.
<i>deviceInfo</i>	Specific device information (nullable).

Returns

A Task resulting in a Result containing the login payload (user details + tokens) on success, or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

6.95.2.3 LogoutAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService.Logout←
Async (
    string refreshTokenValue,
    string ipAddress )
```

Revokes the specific refresh token used for the current session (logout).

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string to revoke.
<i>ipAddress</i>	The IP address performing the logout.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

6.95.2.4 LogoutEverywhereAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService.Logout←
EverywhereAsync (
    long sessionId,
    string ipAddress )
```

Revokes all active refresh tokens associated with a user's session ID (logout everywhere).

Parameters

<i>sessionId</i>	The session ID (typically from the JWT 'sid' claim) to revoke tokens for.
<i>ipAddress</i>	The IP address performing the logout.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

6.95.2.5 RefreshTokenAsync() [1/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuth←
SessionService.RefreshTokenAsync (
    string refreshTokenValue,
```

```
        string ipAddress,
        string userAgent )
```

Refreshes the access token using a valid refresh token without device information.

Implements refresh token rotation via the underlying token service.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

6.95.2.6 RefreshTokenAsync() [2/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService.RefreshTokenAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent,
    string? deviceInfo )
```

Refreshes the access token using a valid refresh token, including device information.

Implements refresh token rotation via the underlying token service.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
<i>ipAddress</i>	The IP address of the requesting client.
<i>userAgent</i>	The user agent string of the requesting client.
<i>deviceInfo</i>	Specific device information (nullable).

Returns

A Task resulting in a Result containing the new access and refresh tokens on success, or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.AuthSessionService](#).

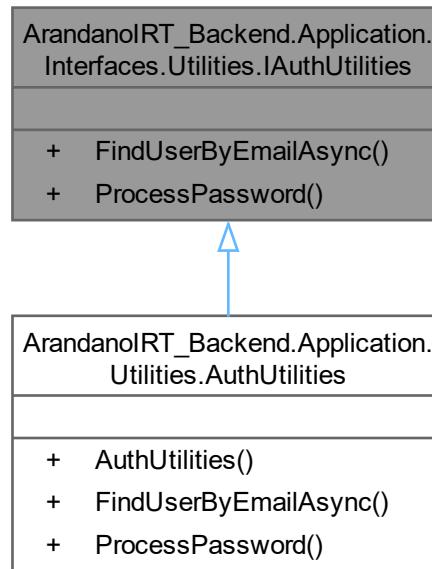
The documentation for this interface was generated from the following file:

- Application/Interfaces/IServices/IAuthSessionService.cs

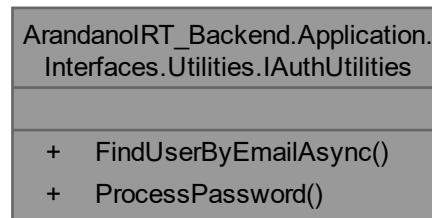
6.96 ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities Interface Reference

Defines the contract for shared utility functions used across authentication-related services.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities:



Public Member Functions

- Task< [Result< PersonEntity >](#) > [FindUserByEmailAsync](#) (string email)
Finds a user by email in the repository.
- [Result< string >](#) [ProcessPassword](#) (string encryptedPassword, string operationContextForLog)
Decrypts, validates policy, and hashes a password.

6.96.1 Detailed Description

Defines the contract for shared utility functions used across authentication-related services.

6.96.2 Member Function Documentation

6.96.2.1 FindUserByEmailAsync()

```
Task< Result< PersonEntity > > ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities.FindUserByEmailAsync (
    string email )
```

Finds a user by email in the repository.

Parameters

<i>email</i>	The email address to search for.
--------------	----------------------------------

Returns

A Task resulting in a Result containing the PersonEntity on success, or Failure if not found.

Implemented in [ArandanoIRT_Backend.Application.Interfaces.AuthUtilities](#).

6.96.2.2 ProcessPassword()

```
Result< string > ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities.ProcessPassword (
    string encryptedPassword,
    string operationContextForLog )
```

Decrypts, validates policy, and hashes a password.

Parameters

<i>encryptedPassword</i>	The encrypted password string.
<i>operationContextForLog</i>	A string indicating the context (e.g., "AdminRegistration", "PasswordReset") for logging purposes.

Returns

Result containing the hashed password on success, or Failure.

Implemented in [ArandanoIRT_Backend.Application.Utilities.AuthUtilities](#).

The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/[IAuthUtilities.cs](#)

6.97 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService Interface Reference

Defines a contract for a generic caching service.

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService:

ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService
+ Get< T >() + GetCollection< T >() + Set< T >() + SetCollection< T >() + Remove()

Public Member Functions

- T? [Get< T >](#) (string key)
Retrieves a cached item by its key.
- IEnumerable< T >? [GetCollection< T >](#) (string key)
Retrieves a cached collection of items by its key.
- void [Set< T >](#) (string key, T value, TimeSpan expirationTime)
Adds or updates an item in the cache with a specified key and relative expiration time.
- void [SetCollection< T >](#) (string key, IEnumerable< T > value, TimeSpan expirationTime)
Adds or updates a collection of items in the cache with a specified key and relative expiration time.
- void [Remove](#) (string key)
Removes a cached item or collection identified by the specified key from the cache.

6.97.1 Detailed Description

Defines a contract for a generic caching service.

Provides asynchronous operations to get, set (with expiration), and remove cached data items and collections.

6.97.2 Member Function Documentation

6.97.2.1 Get< T >()

```
T? ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService.Get< T > (
    string key )
```

Retrieves a cached item by its key.

Template Parameters

<i>T</i>	The expected type of the cached item.
----------	---------------------------------------

Parameters

<i>key</i>	The unique key identifying the cached item.
------------	---

Returns

The cached item of type *T* if found and not expired; otherwise, the default value for type *T* (typically `null` for reference types).

6.97.2.2 GetCollection< T >()

```
IEnumerable< T >? ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService.Get←
Collection< T > (
    string key )
```

Retrieves a cached collection of items by its key.

Template Parameters

<i>T</i>	The expected type of items in the cached collection.
----------	--

Parameters

<i>key</i>	The unique key identifying the cached collection.
------------	---

Returns

An `IEnumerable<T>` containing the cached collection if found and not expired; otherwise, `null` (or potentially an empty enumerable depending on implementation).

6.97.2.3 Remove()

```
void ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService.Remove (
    string key )
```

Removes a cached item or collection identified by the specified key from the cache.

If the key does not exist, the operation typically completes without error.

Parameters

<code>key</code>	The unique key of the item or collection to remove.
------------------	---

6.97.2.4 Set< T >()

```
void ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService.Set< T > (
    string key,
    T value,
    TimeSpan expirationTime )
```

Adds or updates an item in the cache with a specified key and relative expiration time.

If an item with the same key already exists, it is overwritten.

Template Parameters

<code>T</code>	The type of the item to cache.
----------------	--------------------------------

Parameters

<code>key</code>	The unique key to associate with the cached item.
<code>value</code>	The item to be stored in the cache.
<code>expirationTime</code>	The duration for which the item should remain valid in the cache, relative to the time it's set.

6.97.2.5 SetCollection< T >()

```
void ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService.SetCollection< T > (
    string key,
```

```
IEnumerable< T > value,
TimeSpan expirationTime )
```

Adds or updates a collection of items in the cache with a specified key and relative expiration time.

If a collection with the same key already exists, it is overwritten.

Template Parameters

<i>T</i>	The type of items in the collection to cache.
----------	---

Parameters

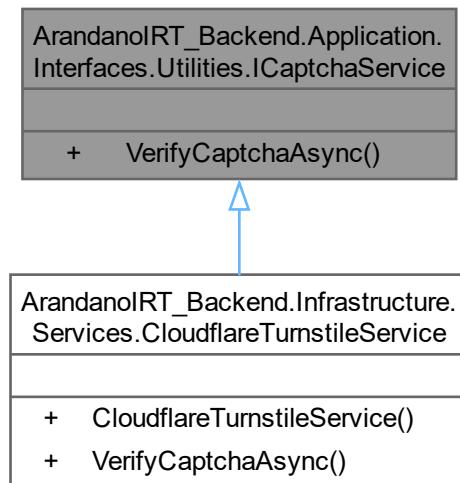
<i>key</i>	The unique key to associate with the cached collection.
<i>value</i>	The collection of items to be stored in the cache.
<i>expirationTime</i>	The duration for which the collection should remain valid in the cache, relative to the time it's set.

The documentation for this interface was generated from the following file:

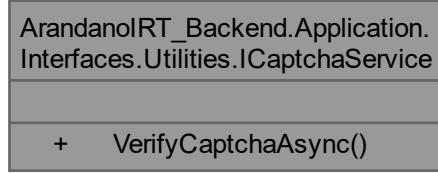
- Infrastructure/Interfaces/IServices/ICacheService.cs

6.98 ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService ↵ Service Interface Reference

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService:



Public Member Functions

- `Task< Result > VerifyCaptchaAsync (string token)`
Verifies the CAPTCHA token received from the client.

6.98.1 Member Function Documentation

6.98.1.1 VerifyCaptchaAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService.Verify<-->
CaptchaAsync (
    string token )
```

Verifies the CAPTCHA token received from the client.

Parameters

<code>token</code>	The CAPTCHA response token from the client.
--------------------	---

Returns

`Result` indicating success (verified) or failure.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService](#).

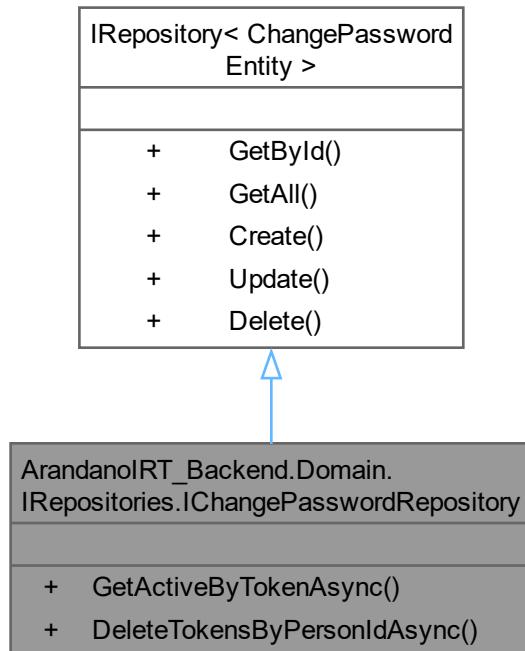
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/[ICaptchaService.cs](#)

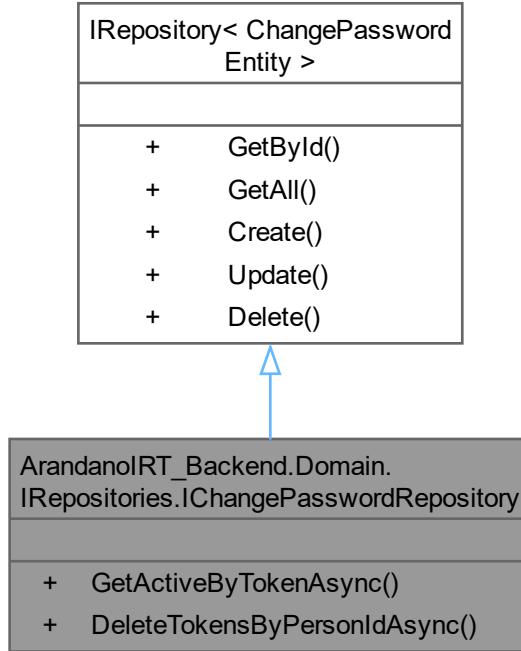
6.99 ArandanoIRT_Backend.Domain.IRepositories.IChangePassword Repository Interface Reference

Defines the repository contract specifically for managing ChangePasswordEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository:



Public Member Functions

- Task< Result< ChangePasswordEntity > > **GetActiveByTokenAsync** (string token)
Asynchronously retrieves an active (not expired and not used) change password entity by its reset token.
- Task< Result< bool > > **DeleteTokensByPersonIdAsync** (int personId)
Asynchronously deletes all change password tokens associated with a specific person.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< ChangePasswordEntity >](#)

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.99.1 Detailed Description

Defines the repository contract specifically for managing ChangePasswordEntity instances.

Includes operations for retrieving active tokens and cleaning up tokens by user, in addition to the standard CRUD-like operations inherited from IRepository<T>.

6.99.2 Member Function Documentation

6.99.2.1 DeleteTokensByPersonIdAsync()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository.<-->
DeleteTokensByPersonIdAsync (
    int personId )
```

Asynchronously deletes all change password tokens associated with a specific person.

Useful for cleanup or when a new token is requested, ensuring only one active token exists per user.

Parameters

<i>personId</i>	The ID of the person whose password reset tokens should be deleted.
-----------------	---

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the deletion was successful (`true`) or failed (`false`), along with any error information in case of failure.

6.99.2.2 GetActiveByTokenAsync()

```
Task< Result< ChangePasswordEntity > > ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository.<-->
GetActiveByTokenAsync (
    string token )
```

Asynchronously retrieves an active (not expired and not used) change password entity by its reset token.

Parameters

<i>token</i>	The password reset token string to search for.
--------------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the ChangePasswordEntity if found and active, or a failure result otherwise.

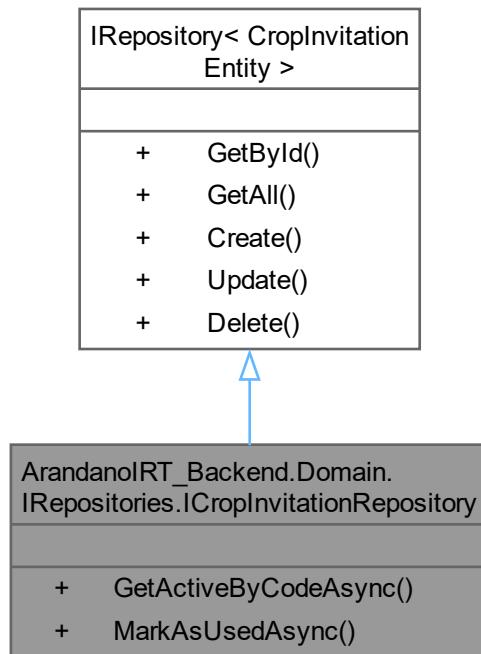
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IChangePasswordRepository.cs

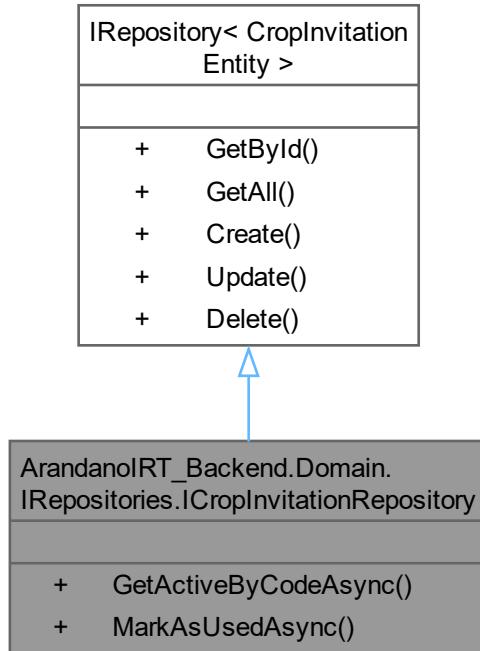
6.100 ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository Interface Reference

Defines the repository contract specifically for managing CropInvitationEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository:



Public Member Functions

- Task< Result< CropInvitationEntity > > **GetActiveByCodeAsync** (string accessCode)
Asynchronously retrieves an active (not used, not expired) crop invitation by its access code.
- Task< Result< bool > > **MarkAsUsedAsync** (int invitationId, int userId)
Asynchronously marks a specific crop invitation as used by associating a user ID and potentially updating its status.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories IRepository<CropInvitationEntity>](#)

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.100.1 Detailed Description

Defines the repository contract specifically for managing CropInvitationEntity instances.

Includes operations for retrieving active invitations by code and marking them as used, in addition to the standard CRUD-like operations inherited from IRepository<T>.

6.100.2 Member Function Documentation

6.100.2.1 GetActiveByCodeAsync()

```
Task< Result< CropInvitationEntity > > ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository.GetActiveByCodeAsync (
    string accessCode )
```

Asynchronously retrieves an active (not used, not expired) crop invitation by its access code.

Parameters

accessCode	The unique access code of the invitation.
------------	---

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the CropInvitationEntity if found and active, or a failure result otherwise.

6.100.2.2 MarkAsUsedAsync()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository.MarkAsUsedAsync (
    int invitationId,
    int userId )
```

Asynchronously marks a specific crop invitation as used by associating a user ID and potentially updating its status.

Parameters

invitationId	The ID of the invitation to mark as used.
userId	The ID of the person who used the invitation.

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the operation was successful (true) or failed (false), along with any error information in case of failure.

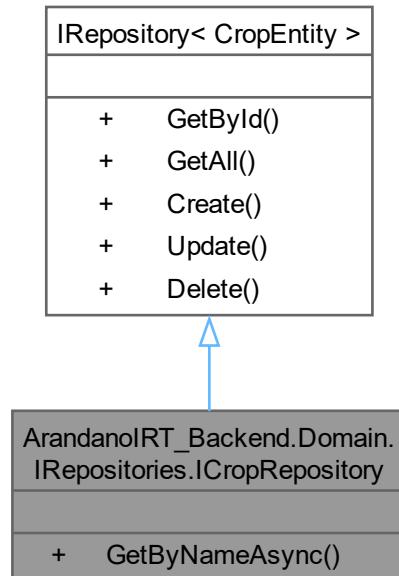
The documentation for this interface was generated from the following file:

- Domain/IRepositories/ICropInvitationRepository.cs

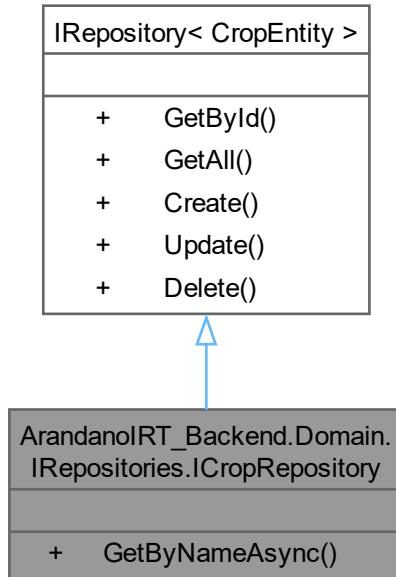
6.101 ArandanoIRT_Backend.Domain.IRepositories.ICropRepository Interface Reference

Defines the repository contract specifically for managing CropEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.ICropRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.ICropRepository:



Public Member Functions

- `Task< Result< CropEntity > > GetByNameAsync (string name)`

Retrieves a single crop entity by its unique name.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< CropEntity >](#)

- `Task< Result< T > > GetById (int id)`

Asynchronously retrieves an entity by its unique identifier.

- `Task< Result< IEnumerable< T > > > GetAll ()`

Asynchronously retrieves all entities of type T .

- `Task< Result< T > > Create (T entity)`

Asynchronously creates a new entity in the data store.

- `Task< Result< bool > > Update (T entity)`

Asynchronously updates an existing entity in the data store.

- `Task< Result< bool > > Delete (int id)`

Asynchronously deletes an entity from the data store by its unique identifier.

6.101.1 Detailed Description

Defines the repository contract specifically for managing CropEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

6.101.2 Member Function Documentation

6.101.2.1 GetByNameAsync()

```
Task< Result< CropEntity > > ArandanoIRT_Backend.Domain.IRepositories.ICropRepository.GetBy←
NameAsync ( ←
            string name )
```

Retrieves a single crop entity by its unique name.

Parameters

<i>name</i>	The name of the crop to retrieve.
-------------	-----------------------------------

Returns

A Task representing the asynchronous operation, containing a Result object. The Result is successful and contains the CropEntity if found; otherwise, the Result is a failure.

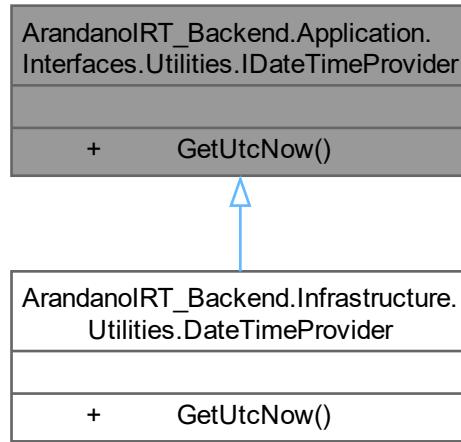
The documentation for this interface was generated from the following file:

- Domain/IRepositories/ICropRepository.cs

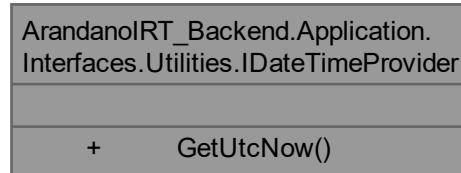
6.102 ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTime← Provider Interface Reference

Provides the current time, abstracting away direct DateTime usage for testability.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider:



Public Member Functions

- DateTime [GetUtcNow \(\)](#)
Gets the current UTC DateTime.

6.102.1 Detailed Description

Provides the current time, abstracting away direct DateTime usage for testability.

6.102.2 Member Function Documentation

6.102.2.1 GetUtcNow()

```
DateTime ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider.GetUtcNow ( )
```

Gets the current UTC DateTime.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider](#).

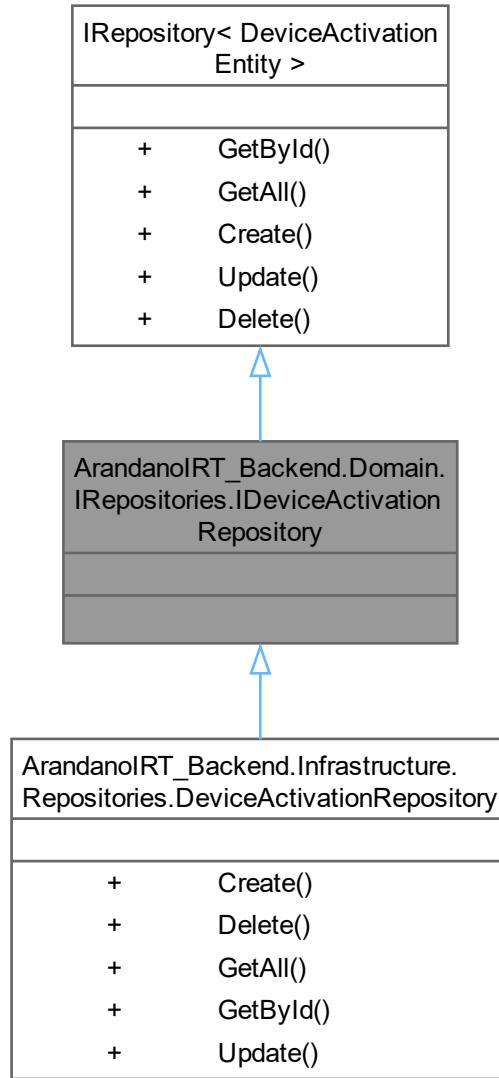
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/[IDateTimeProvider.cs](#)

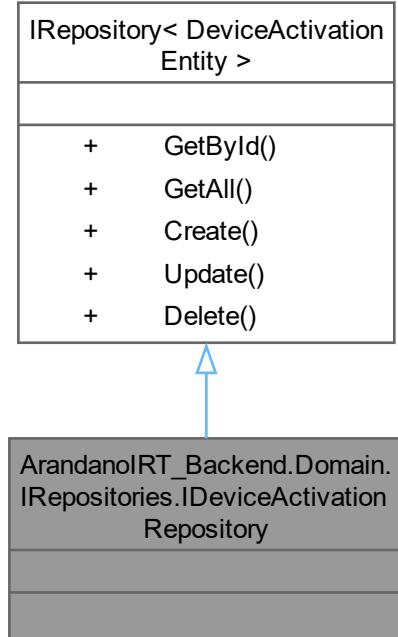
6.103 ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository Interface Reference

Defines the repository contract specifically for managing DeviceActivationEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceActivationEntity >](#)

- Task< [Result< T >](#) > [GetBy Id](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.103.1 Detailed Description

Defines the repository contract specifically for managing DeviceActivationEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

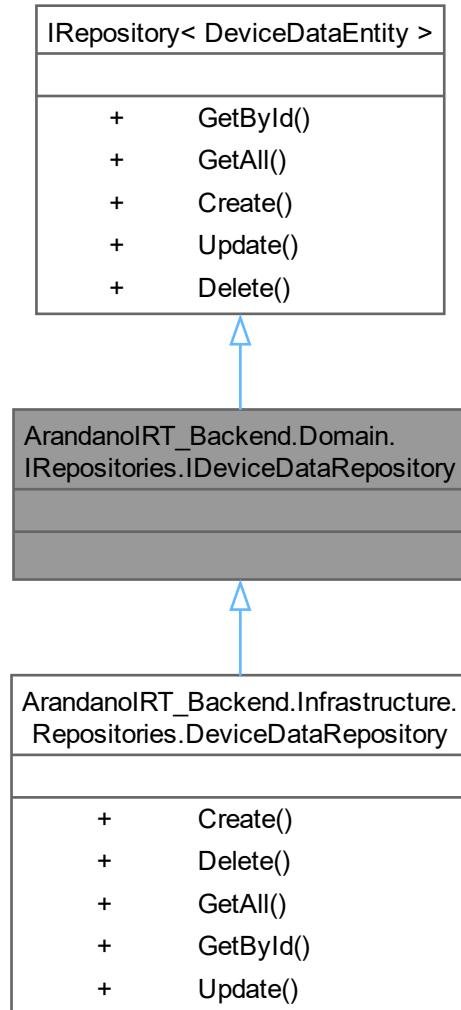
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IDeviceActivationRepository.cs](#)

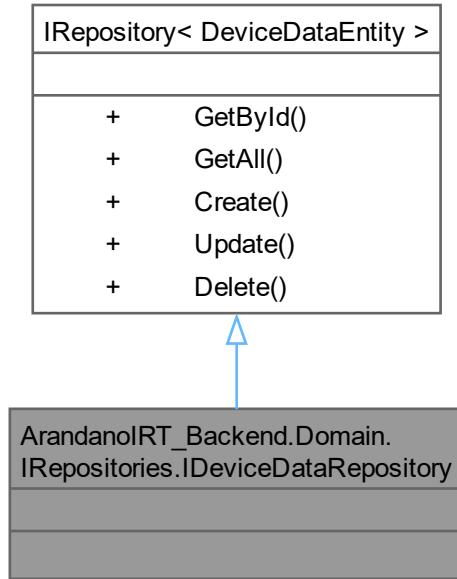
6.104 ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository Interface Reference

Defines the repository contract specifically for managing DeviceDataEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository<DeviceDataEntity>](#)

- `Task<Result<T>> GetById (int id)`
Asynchronously retrieves an entity by its unique identifier.
- `Task<Result<IEnumerable<T>>> GetAll ()`
Asynchronously retrieves all entities of type T.
- `Task<Result<T>> Create (T entity)`
Asynchronously creates a new entity in the data store.
- `Task<Result<bool>> Update (T entity)`
Asynchronously updates an existing entity in the data store.
- `Task<Result<bool>> Delete (int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.104.1 Detailed Description

Defines the repository contract specifically for managing `DeviceDataEntity` instances.

Inherits standard CRUD-like operations from `IRepository<T>`.

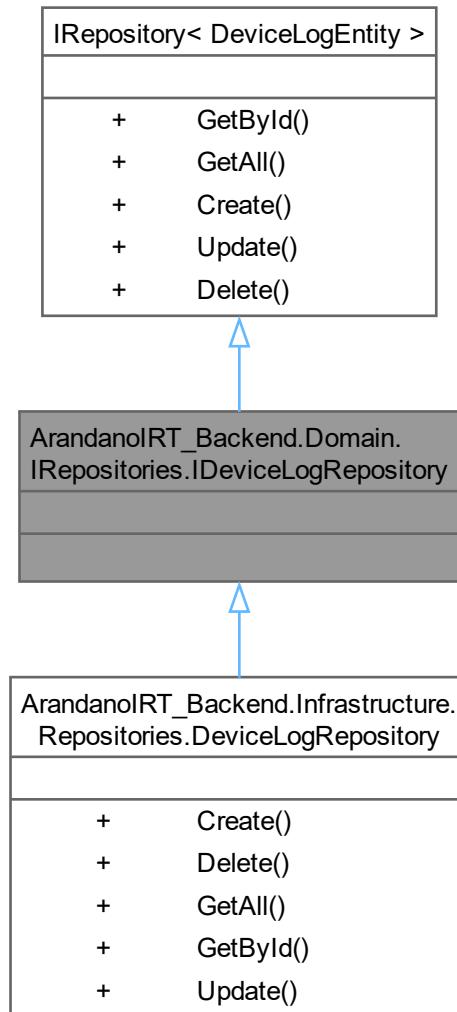
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IDeviceDataRepository.cs

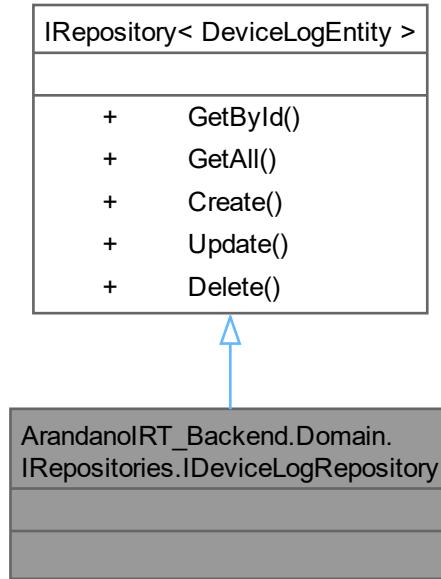
6.105 ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository Interface Reference

Defines the repository contract specifically for managing DeviceLogEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceLogEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.105.1 Detailed Description

Defines the repository contract specifically for managing DeviceLogEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

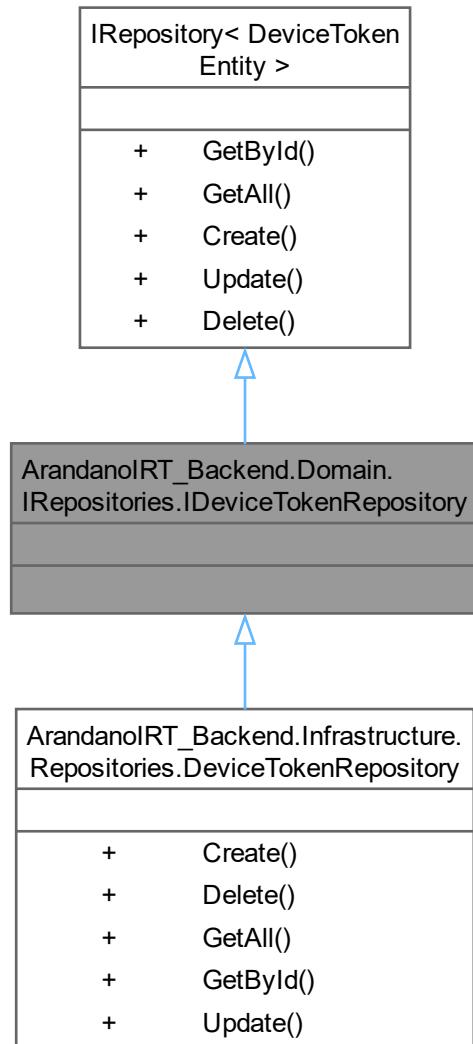
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IDeviceLogRepository.cs

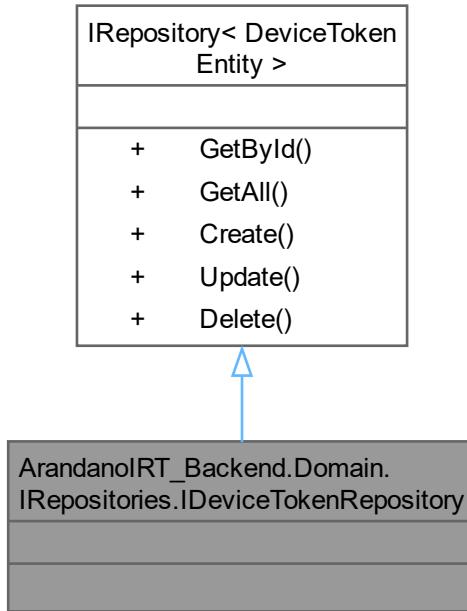
6.106 ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository Interface Reference

Defines the repository contract specifically for managing DeviceTokenEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< DeviceTokenEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.106.1 Detailed Description

Defines the repository contract specifically for managing DeviceTokenEntity instances.

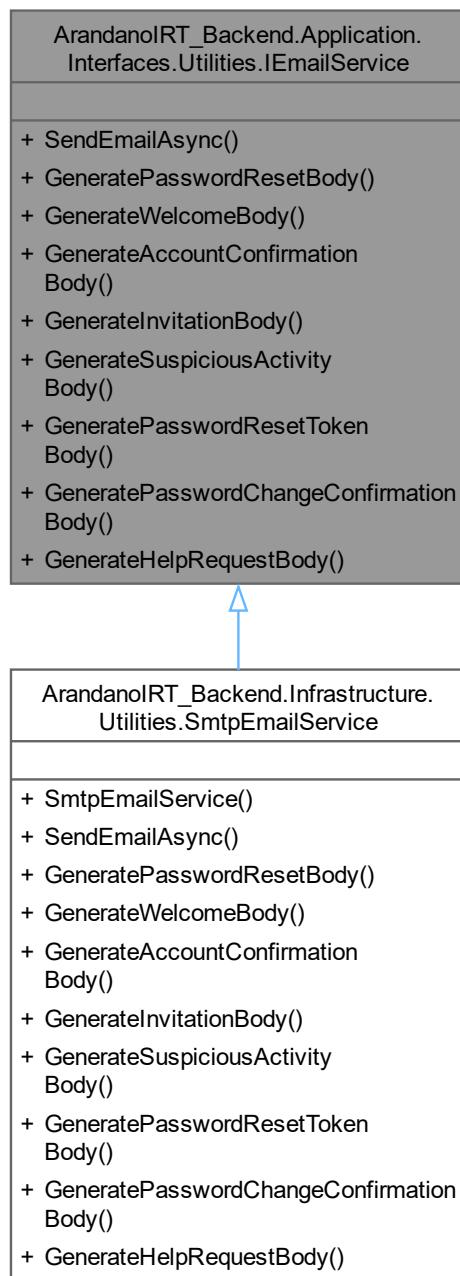
Inherits standard CRUD-like operations from IRepository<T>.

The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IDeviceTokenRepository.cs](#)

6.107 ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService Interface Reference

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService:

ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService
<ul style="list-style-type: none"> + SendEmailAsync() + GeneratePasswordResetBody() + GenerateWelcomeBody() + GenerateAccountConfirmationBody() + GenerateInvitationBody() + GenerateSuspiciousActivityBody() + GeneratePasswordResetTokenBody() + GeneratePasswordChangeConfirmationBody() + GenerateHelpRequestBody()

Public Member Functions

- Task< [Result](#) > [SendEmailAsync](#) (string to, string subject, string htmlBody)
Sends an email asynchronously to the specified recipient.
- string [GeneratePasswordResetBody](#) (string userName, string resetToken)
Generates the body for a password reset email, including the user's name and reset token.
- string [GenerateWelcomeBody](#) (string userName)
Generates the body for a welcome email, including the user's name.
- string [GenerateAccountConfirmationBody](#) (string userName, string confirmationLink)
Generates the body for an account confirmation email, including the user's name and confirmation link.
- string [GenerateInvitationBody](#) (string inviterName, string cropName, string accessCode, string expiryDate)
Generates the body for an invitation email, including the inviter's name, crop name, access code, and expiry date.
- string [GenerateSuspiciousActivityBody](#) (string userName, DateTime attemptTime, string ipAddress)
Generates the email body for a suspicious login activity warning.
- string [GeneratePasswordResetTokenBody](#) (string userName, string resetToken)
Generates the simple email body containing only the password reset token.
- string [GeneratePasswordChangeConfirmationBody](#) (string userName)
Generates the email body confirming a successful password change.
- string [GenerateHelpRequestBody](#) ([HelpRequestDto](#) request)
Generates the email body for a help request submitted by a user.

6.107.1 Member Function Documentation

6.107.1.1 GenerateAccountConfirmationBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GenerateAccountConfirmation←
Body (
    string userName,
    string confirmationLink )
```

Generates the body for an account confirmation email, including the user's name and confirmation link.

Parameters

<i>userName</i>	The name of the user confirming their account.
<i>confirmationLink</i>	The URL link that the user must follow to confirm their account.

Returns

HTML email body string for the account confirmation message.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.2 GenerateHelpRequestBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GenerateHelpRequestBody
(
    HelpRequestDto request )
```

Generates the email body for a help request submitted by a user.

Parameters

<i>request</i>	The help request data transfer object containing user details and message.
----------------	--

Returns

HTML email body string for the help request notification.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.3 GenerateInvitationBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GenerateInvitationBody (
    string inviterName,
```

```
    string cropName,  
    string accessCode,  
    string expiryDate )
```

Generates the body for an invitation email, including the inviter's name, crop name, access code, and expiry date.

Parameters

<i>inviterName</i>	The name of the person sending the invitation.
<i>cropName</i>	The name of the crop associated with the invitation.
<i>accessCode</i>	The access code required to join the invitation.
<i>expiryDate</i>	The date when the invitation expires.

Returns

HTML email body string for the invitation.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.4 GeneratePasswordChangeConfirmationBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GeneratePasswordChangeConfirmationBody (
    string userName )
```

Generates the email body confirming a successful password change.

Parameters

<i>userName</i>	The name of the user.
-----------------	-----------------------

Returns

HTML email body string.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.5 GeneratePasswordResetBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GeneratePasswordResetBody (
    string userName,
    string resetToken )
```

Generates the body for a password reset email, including the user's name and reset token.

Parameters

<i>userName</i>	The name of the user requesting the password reset.
<i>resetToken</i>	The token used to validate the password reset request.

Returns

HTML email body string for the password reset request.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.6 GeneratePasswordResetTokenBody()

```
string ArandanoIRT_Backend.Application.Interfaces.ILoggerService.GeneratePasswordResetTokenBody (
    string userName,
    string resetToken )
```

Generates the simple email body containing only the password reset token.

Parameters

<i>userName</i>	The name of the user.
<i>resetToken</i>	The password reset token string.

Returns

Plain text or simple HTML email body string.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.7 GenerateSuspiciousActivityBody()

```
string ArandanoIRT_Backend.Application.Interfaces.ILoggerService.GenerateSuspiciousActivityBody (
    string userName,
    DateTime attemptTime,
    string ipAddress )
```

Generates the email body for a suspicious login activity warning.

Parameters

<i>userName</i>	The name of the user.
<i>attemptTime</i>	The approximate time of the suspicious activity.
<i>ipAddress</i>	The IP address associated with the attempts.

Returns

HTML email body string for the suspicious activity alert.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.8 GenerateWelcomeBody()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.GenerateWelcomeBody (
    string userName )
```

Generates the body for a welcome email, including the user's name.

Parameters

<i>userName</i>	The name of the new user receiving the welcome message.
-----------------	---

Returns

HTML email body string for the welcome email.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

6.107.1.9 SendEmailAsync()

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService.SendEmailAsync (
    string to,
    string subject,
    string htmlBody )
```

Sends an email asynchronously to the specified recipient.

Parameters

<i>to</i>	The email address of the recipient.
<i>subject</i>	The subject line of the email.
<i>htmlBody</i>	The body of the email in HTML format.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService](#).

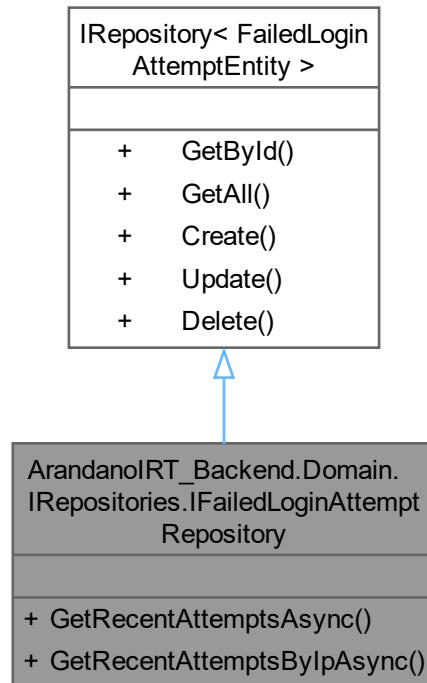
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/IEmailService.cs

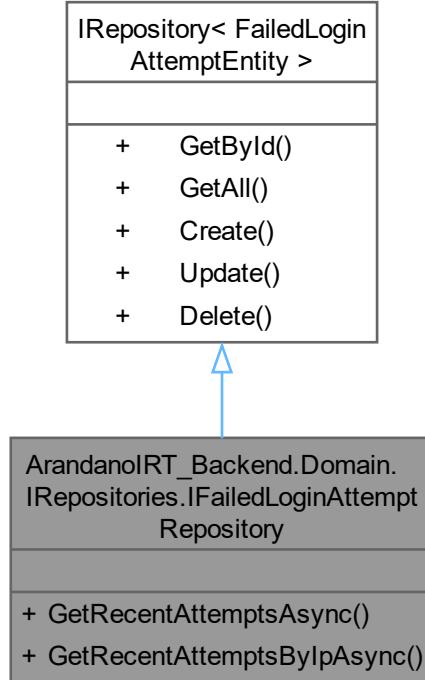
6.108 ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttempt Repository Interface Reference

Defines the repository contract specifically for managing FailedLoginAttemptEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository:



Public Member Functions

- Task< [Result< IEnumerable< FailedLoginAttemptEntity > >> > \[GetRecentAttemptsAsync\]\(#\) \(int personId, Date←Time since\)
Asynchronously retrieves recent failed login attempts for a specific person recorded since a given timestamp.](#)
- Task< [Result< IEnumerable< FailedLoginAttemptEntity > >> > \[GetRecentAttemptsByIpAsync\]\(#\) \(string ipAddress, DateTime since\)
Asynchronously retrieves recent failed login attempts originating from a specific IP address since a given timestamp.](#)

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< FailedLoginAttemptEntity >](#)

- Task< [Result< T > > \[GetById\]\(#\) \(int id\)
Asynchronously retrieves an entity by its unique identifier.](#)
- Task< [Result< IEnumerable< T > >> > \[GetAll\]\(#\) \(\)
Asynchronously retrieves all entities of type T .](#)
- Task< [Result< T > > \[Create\]\(#\) \(T entity\)
Asynchronously creates a new entity in the data store.](#)

- Task<`Result< bool >`>> `Update` (`T entity`)
Asynchronously updates an existing entity in the data store.
- Task<`Result< bool >`>> `Delete` (`int id`)
Asynchronously deletes an entity from the data store by its unique identifier.

6.108.1 Detailed Description

Defines the repository contract specifically for managing FailedLoginAttemptEntity instances.

Includes operations for retrieving recent attempts by user or IP, in addition to the standard CRUD-like operations inherited from `IRepository<T>`.

6.108.2 Member Function Documentation

6.108.2.1 GetRecentAttemptsAsync()

```
Task< Result< IEnumerable< FailedLoginAttemptEntity > > > ArandanoIRT_Backend.Domain.IRepositories.←
IFailedLoginAttemptRepository.GetRecentAttemptsAsync (
    int personId,
    DateTime since )
```

Asynchronously retrieves recent failed login attempts for a specific person recorded since a given timestamp.

Parameters

<code>personId</code>	The ID of the person whose failed attempts are to be retrieved.
<code>since</code>	The UTC timestamp indicating the start of the period to check (exclusive or inclusive depends on implementation).

Returns

A `Task<TResult>` representing the asynchronous operation, containing a `Result<T>` which holds an `IEnumerable<T>` of `FailedLoginAttemptEntity` if successful, or a failure result otherwise. The list may be empty if no recent attempts are found.

6.108.2.2 GetRecentAttemptsByIpAsync()

```
Task< Result< IEnumerable< FailedLoginAttemptEntity > > > ArandanoIRT_Backend.Domain.IRepositories.←
IFailedLoginAttemptRepository.GetRecentAttemptsByIpAsync (
    string ipAddress,
    DateTime since )
```

Asynchronously retrieves recent failed login attempts originating from a specific IP address since a given timestamp.

Parameters

<i>ipAddress</i>	The IP address from which the failed attempts originated.
<i>since</i>	The UTC timestamp indicating the start of the period to check (exclusive or inclusive depends on implementation).

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds an IEnumerable<T> of FailedLoginAttemptEntity if successful, or a failure result otherwise. The list may be empty if no recent attempts are found from this IP.

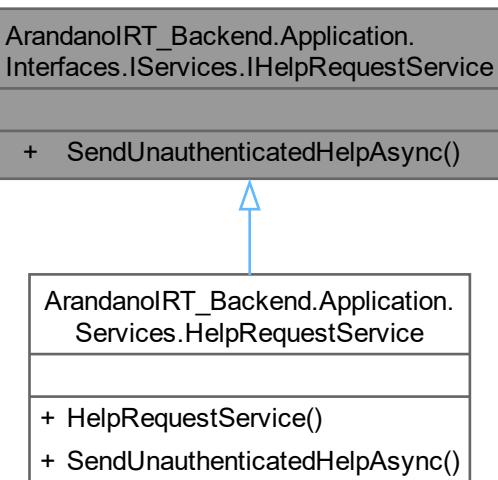
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IFailedLoginAttemptRepository.cs](#)

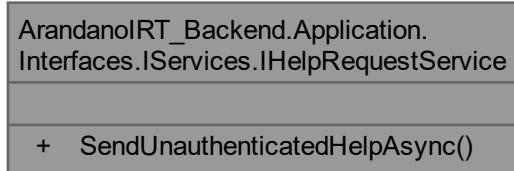
6.109 ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService Interface Reference

Defines the contract for handling help request operations.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService:



Public Member Functions

- Task< [Result](#) > [SendUnauthenticatedHelpAsync \(HelpRequestDto request\)](#)
Processes a help request submitted by an unauthenticated user.

6.109.1 Detailed Description

Defines the contract for handling help request operations.

6.109.2 Member Function Documentation

6.109.2.1 [SendUnauthenticatedHelpAsync\(\)](#)

```
Task< Result > ArandanoIRT_Backend.Application.Interfaces.IHelpRequestService.SendUnauthenticatedHelpAsync (
    HelpRequestDto request )
```

Processes a help request submitted by an unauthenticated user.

Validates the associated crop name and sends an email notification to the crop's administrator.

Parameters

<code>request</code>	The help request data transfer object containing user details, message, and crop name.
----------------------	--

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Implemented in [ArandanoIRT_Backend.Application.Services.HelpRequestService](#).

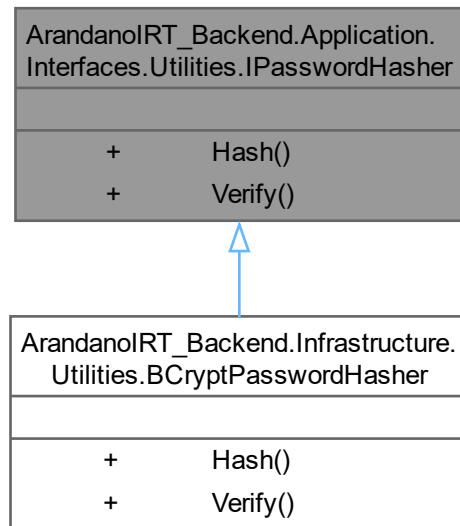
The documentation for this interface was generated from the following file:

- Application/Interfaces/IServices/IHelpRequestService.cs

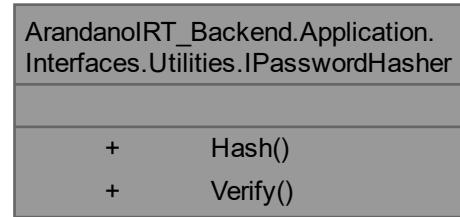
6.110 ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher Interface Reference

Service interface for hashing and verifying passwords.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher:



Public Member Functions

- string **Hash** (string password)
Creates a hash from a plain text password.
- bool **Verify** (string hash, string providedPassword)
Verifies a plain text password against a stored hash.

6.110.1 Detailed Description

Service interface for hashing and verifying passwords.

6.110.2 Member Function Documentation

6.110.2.1 Hash()

```
string ArandanoIRT_Backend.Application.Interfaces.IPasswordHasher.Hash (
    string password )
```

Creates a hash from a plain text password.

Parameters

<i>password</i>	The plain text password.
-----------------	--------------------------

Returns

The hashed password.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher](#).

6.110.2.2 Verify()

```
bool ArandanoIRT_Backend.Application.Interfaces.IPasswordHasher.Verify (
    string hash,
    string providedPassword )
```

Verifies a plain text password against a stored hash.

Parameters

<i>hash</i>	The stored password hash.
<i>providedPassword</i>	The plain text password provided by the user.

Returns

True if the password matches the hash, false otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher](#).

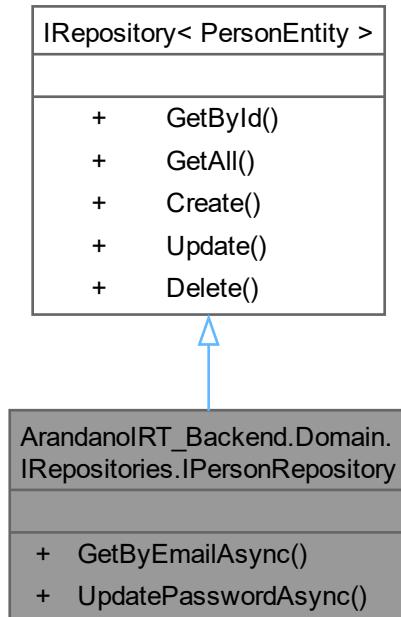
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/[IPasswordHasher.cs](#)

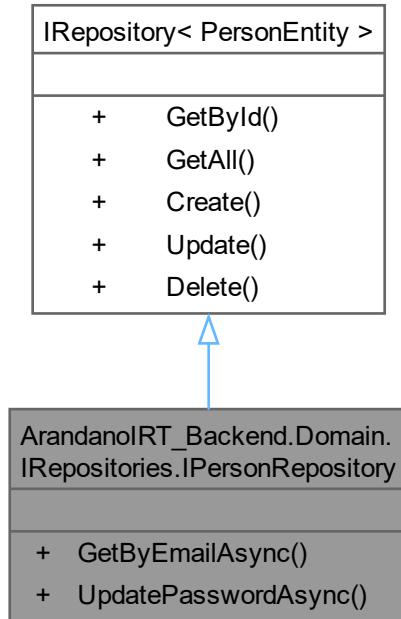
6.111 ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository Interface Reference

Defines the repository contract specifically for managing PersonEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository:



Public Member Functions

- Task< [Result< PersonEntity >](#) > > [GetByEmailAsync](#) (string email)
Asynchronously retrieves a person entity by their unique email address.
- Task< [Result< bool >](#) > > [UpdatePasswordAsync](#) (int personId, string newPasswordHash, DateTime changeTimestamp)
Asynchronously updates the password hash and the last password change timestamp for a specified person.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< PersonEntity >](#)

- Task< [Result< T >](#) > > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T >](#) > > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.111.1 Detailed Description

Defines the repository contract specifically for managing PersonEntity instances.

Includes operations for retrieving users by email and updating passwords, in addition to the standard CRUD-like operations inherited from IRepository<T>.

6.111.2 Member Function Documentation

6.111.2.1 GetByEmailAsync()

```
Task< Result< PersonEntity > > ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository.GetByEmailAsync (
    string email )
```

Asynchronously retrieves a person entity by their unique email address.

Parameters

<code>email</code>	The email address to search for.
--------------------	----------------------------------

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the Person Entity if found, or a failure result otherwise.

6.111.2.2 UpdatePasswordAsync()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository.UpdatePasswordAsync (
    int personId,
    string newPasswordHash,
    DateTime changeTimestamp )
```

Asynchronously updates the password hash and the last password change timestamp for a specified person.

Parameters

<code>personId</code>	The ID of the person entity whose password is to be updated.
<code>newPasswordHash</code>	The new, securely hashed password to store.
<code>changeTimestamp</code>	The UTC timestamp when the password change occurred.

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the update was successful (true) or failed (false), along with any error information in case of failure.

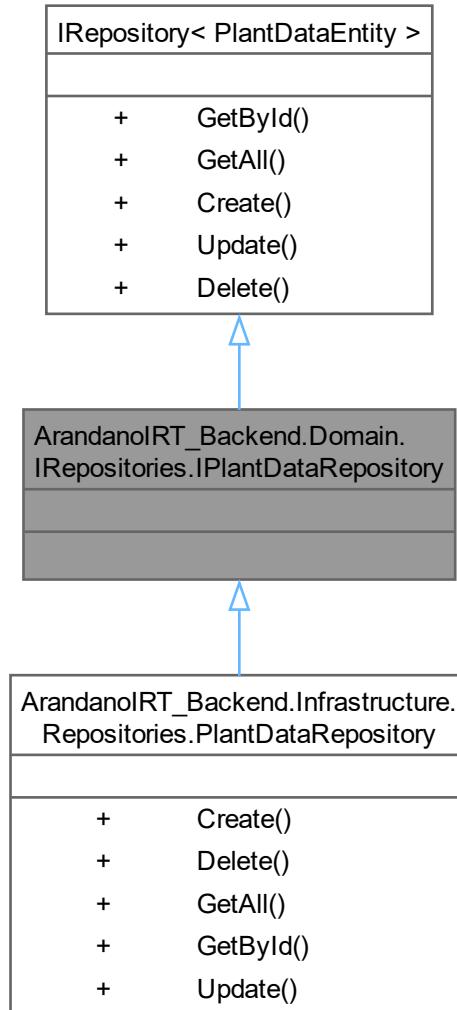
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IPersonRepository.cs

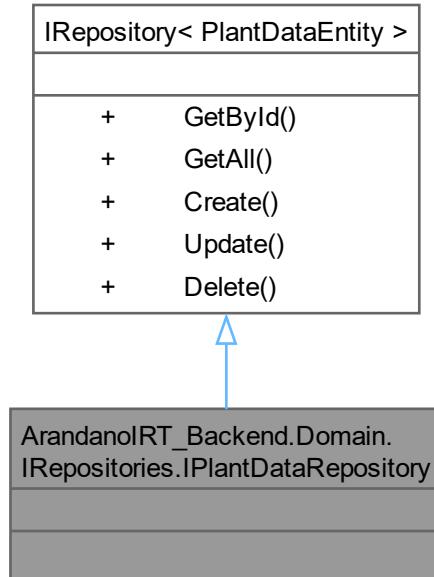
6.112 ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository Interface Reference

Defines the repository contract specifically for managing PlantDataEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantDataEntity >](#)

- `Task< Result< T > > GetById (int id)`
Asynchronously retrieves an entity by its unique identifier.
- `Task< Result< IEnumerable< T > > > GetAll ()`
Asynchronously retrieves all entities of type T .
- `Task< Result< T > > Create (T entity)`
Asynchronously creates a new entity in the data store.
- `Task< Result< bool > > Update (T entity)`
Asynchronously updates an existing entity in the data store.
- `Task< Result< bool > > Delete (int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.112.1 Detailed Description

Defines the repository contract specifically for managing `PlantDataEntity` instances.

Inherits standard CRUD-like operations from `IRepository<T>`.

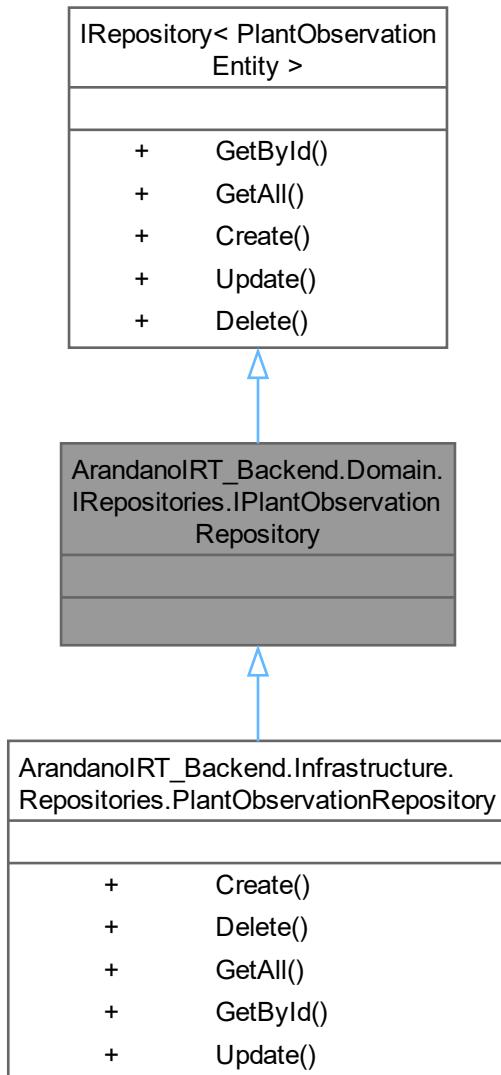
The documentation for this interface was generated from the following file:

- [Domain/IRepositories/IPlantDataRepository.cs](#)

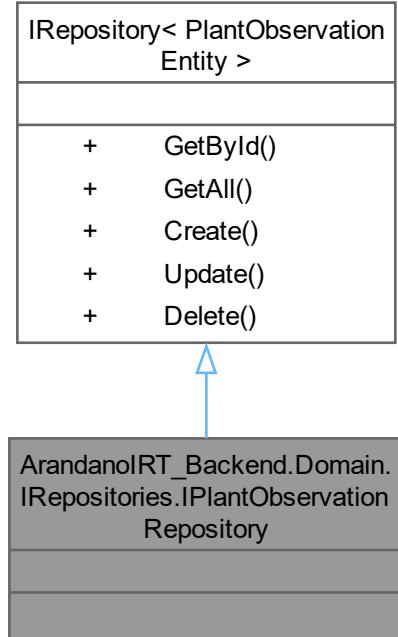
6.113 ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository Interface Reference

Defines the repository contract specifically for managing PlantObservationEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantObservationEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.113.1 Detailed Description

Defines the repository contract specifically for managing PlantObservationEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

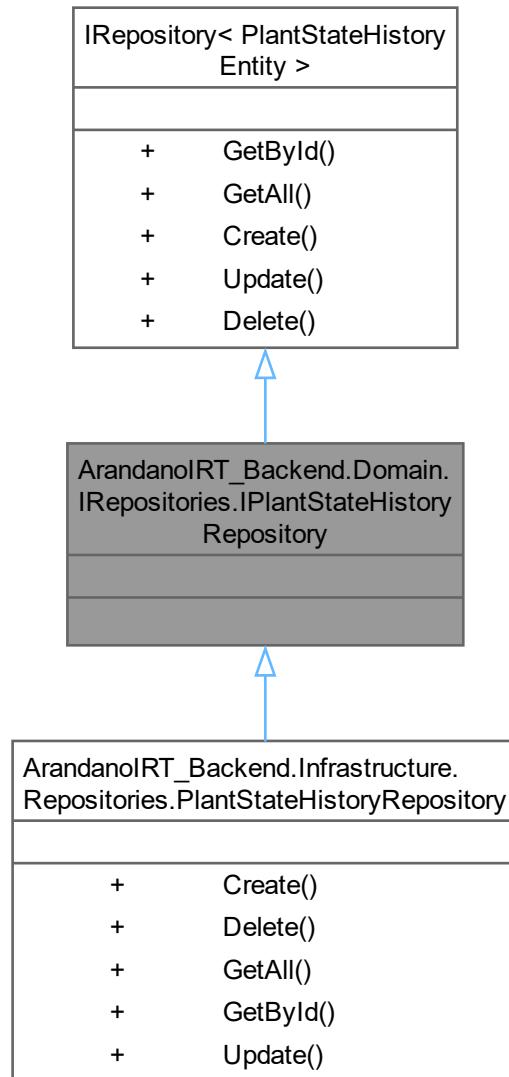
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IPlantObservationRepository.cs](#)

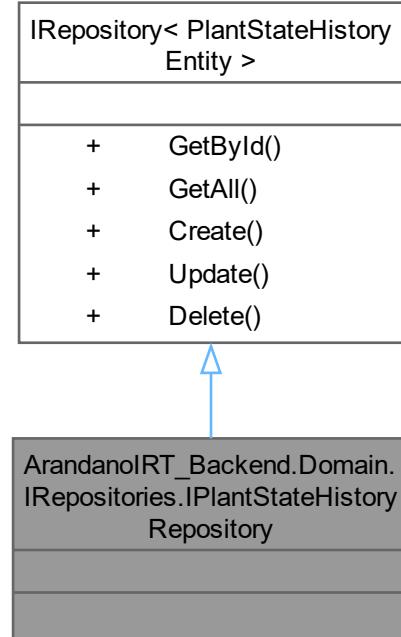
6.114 ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository Interface Reference

Defines the repository contract specifically for managing PlantStateHistoryEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< PlantStateHistoryEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.114.1 Detailed Description

Defines the repository contract specifically for managing PlantStateHistoryEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

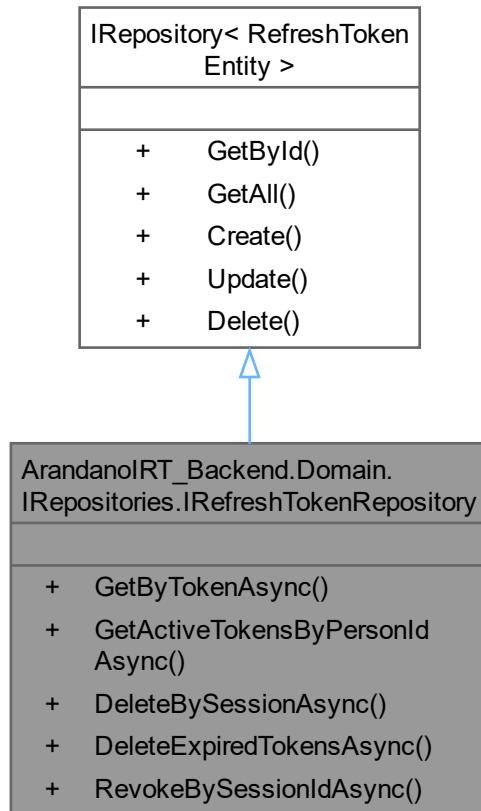
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IPlantStateHistoryRepository.cs

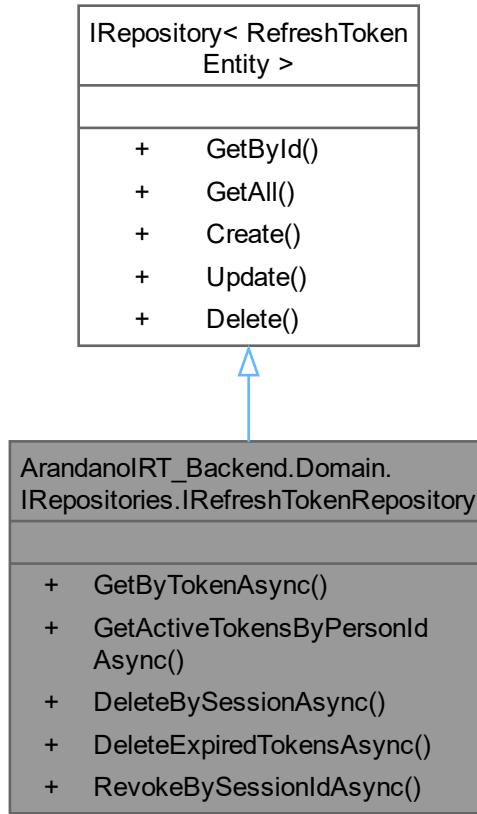
6.115 ArandanoIRT_Backend.Domain.IRepositories.IRefreshToken← Repository Interface Reference

Defines the repository contract specifically for managing RefreshTokenEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository:



Public Member Functions

- `Task< Result< RefreshTokenEntity > > GetByTokenAsync (string token)`
Asynchronously retrieves a refresh token entity by its unique token string.
- `Task< Result< IEnumerable< RefreshTokenEntity > > > GetActiveTokensByPersonIdAsync (int personId)`
Asynchronously retrieves all non-revoked refresh tokens associated with a specific user.
- `Task< Result< bool > > DeleteBySessionAsync (long sessionId)`
Asynchronously deletes all refresh tokens associated with a specific session identifier.
- `Task< Result< bool > > DeleteExpiredTokensAsync (int personId, DateTime olderThan)`
Asynchronously deletes all refresh tokens for a specific user that expired before a given date.
- `Task< Result< bool > > RevokeBySessionIdAsync (long sessionId, string revokedByIp)`
Asynchronously revokes (marks as invalid) all active refresh tokens associated with a specific session ID.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories IRepository< RefreshTokenEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T >](#) > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.115.1 Detailed Description

Defines the repository contract specifically for managing RefreshTokenEntity instances.

Includes specific operations for token retrieval, cleanup, and revocation, in addition to the standard CRUD-like operations inherited from IRepository<T>.

6.115.2 Member Function Documentation

6.115.2.1 DeleteBySessionAsync()

```
Task< Result< bool > > > ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository.DeleteBySessionAsync (
    long sessionId )
```

Asynchronously deletes all refresh tokens associated with a specific session identifier.

Parameters

<code>sessionId</code>	The session identifier whose tokens should be deleted.
------------------------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the deletion was successful (`true`) or failed (`false`), along with any error information in case of failure.

6.115.2.2 DeleteExpiredTokensAsync()

```
Task< Result< bool > > > ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository.DeleteExpiredTokensAsync (
```

```
    int personId,  
        DateTime olderThan )
```

Asynchronously deletes all refresh tokens for a specific user that expired before a given date.

Used for periodic cleanup.

Parameters

<i>personId</i>	The ID of the user whose expired tokens should be deleted.
<i>olderThan</i>	The cutoff timestamp. Tokens expiring before this timestamp will be deleted.

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the deletion was successful (true) or failed (false), along with any error information in case of failure.

6.115.2.3 GetActiveTokensByPersonIdAsync()

```
Task< Result< IEnumerable< RefreshTokenEntity > > > ArandanoIRT_Backend.Domain.IRepositories.←  
IRefreshTokenRepository.GetActiveTokensByPersonIdAsync (←  
    int personId )
```

Asynchronously retrieves all non-revoked refresh tokens associated with a specific user.

Parameters

<i>personId</i>	The ID of the user whose active tokens are to be retrieved.
-----------------	---

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds an IEnumerable<T> of active RefreshTokenEntity instances if successful, or a failure result otherwise. The list may be empty.

6.115.2.4 GetByTokenAsync()

```
Task< Result< RefreshTokenEntity > > ArandanoIRT_Backend.Domain.IRepositories.IRefreshToken←  
Repository.GetByTokenAsync (←  
    string token )
```

Asynchronously retrieves a refresh token entity by its unique token string.

Parameters

<i>token</i>	The refresh token string to search for.
--------------	---

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the RefreshTokenEntity if found, or a failure result otherwise.

6.115.2.5 RevokeBySessionIdAsync()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository.RevokeBySessionIdAsync (
    long sessionId,
    string revokedByIp )
```

Asynchronously revokes (marks as invalid) all active refresh tokens associated with a specific session ID.

Parameters

<i>sessionId</i>	The session ID for which all associated active tokens should be revoked.
<i>revokedByIp</i>	The IP address performing the revocation action.

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the revocation was successful (`true`) or failed (`false`), along with any error information in case of failure.

The documentation for this interface was generated from the following file:

- Domain/IRepositories/IRefreshTokenRepository.cs

6.116 ArandanoIRT_Backend.Domain.IRepositories.IRepository< T > Interface Template Reference

Defines a generic contract for repository operations for a specific entity type *T*.

Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IRepository< T >:

ArandanoIRT_Backend.Domain.IRepositories.IRepository< T >
+ GetBy <code>Id()</code> + GetAll() + Create() + Update() + Delete()

Public Member Functions

- Task< [Result< T >](#) > [GetBy`Id`](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T > >](#) > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.116.1 Detailed Description

Defines a generic contract for repository operations for a specific entity type *T*.

Provides standard asynchronous CRUD-like methods using the Result pattern.

Template Parameters

<i>T</i>	The type of the entity managed by the repository. Must be a class.
----------	--

Type Constraints

T : class

6.116.2 Member Function Documentation

6.116.2.1 Create()

```
Task< Result< T > > ArandanoIRT_Backend.Domain.IRepositories IRepository< T >.Create (   
    T entity )
```

Asynchronously creates a new entity in the data store.

Parameters

<i>entity</i>	The entity instance to create.
---------------	--------------------------------

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the created entity (potentially with updated properties like ID) if successful, or a failure result otherwise.

6.116.2.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories IRepository< T >.Delete (   
    int id )
```

Asynchronously deletes an entity from the data store by its unique identifier.

Parameters

<i>id</i>	The unique identifier of the entity to delete.
-----------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the deletion was successful (`true`) or failed (`false`), along with any error information in case of failure.

6.116.2.3 GetAll()

```
Task< Result< IEnumerable< T > > > ArandanoIRT_Backend.Domain.IRepositories IRepository< T >.GetAll ( )
```

Asynchronously retrieves all entities of type *T*.

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds an IEnumerable<T> of entities if successful, or a failure result otherwise.

6.116.2.4 GetById()

```
Task< Result< T > > ArandanoIRT_Backend.Domain.IRepositories IRepository< T >.GetById ( int id )
```

Asynchronously retrieves an entity by its unique identifier.

Parameters

<i>id</i>	The unique identifier of the entity to retrieve.
-----------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the entity if found, or a failure result otherwise.

6.116.2.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Domain.IRepositories IRepository< T >.Update ( T entity )
```

Asynchronously updates an existing entity in the data store.

Parameters

<i>entity</i>	The entity instance with updated values.
---------------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> of bool indicating whether the update was successful (`true`) or failed (`false`), along with any error information in case of failure.

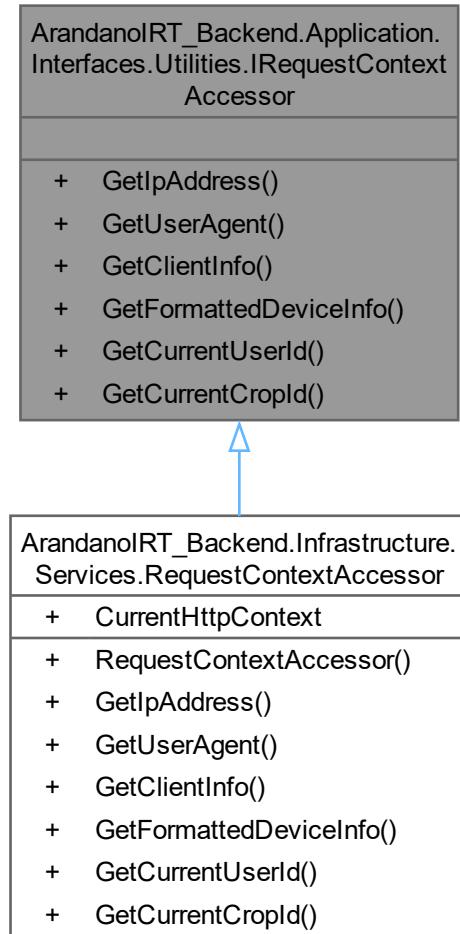
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[IRepository.cs](#)

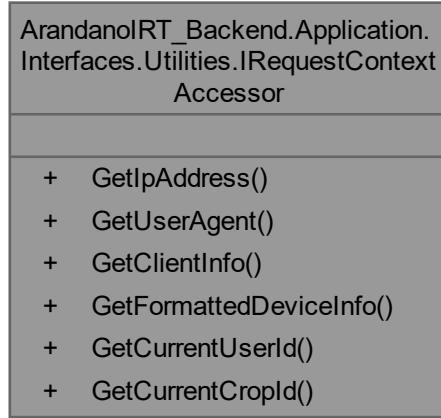
6.117 ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor Interface Reference

Defines a contract for accessing information related to the current HTTP request context, such as client details and authenticated user information.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.IRequestContextAccessor:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor:



Public Member Functions

- string [GetIpAddress \(\)](#)
Gets the IP address of the client making the current request.
- string [GetUserAgent \(\)](#)
Gets the User-Agent string sent by the client in the current request header.
- ClientInfo [GetClientInfo \(\)](#)
Gets structured information about the client (e.g., browser, OS, device) typically parsed from the User-Agent string.
- string [GetFormattedDeviceInfo \(\)](#)
Gets a formatted string summarizing the client's device and browser information.
- int? [GetCurrentUserId \(\)](#)
Gets the unique identifier of the currently authenticated user associated with the request.
- int? [GetCurrentCropId \(\)](#)
Gets the unique identifier of the crop associated with the current request context (e.g., based on the authenticated user or other request parameters).

6.117.1 Detailed Description

Defines a contract for accessing information related to the current HTTP request context, such as client details and authenticated user information.

6.117.2 Member Function Documentation

6.117.2.1 GetClientInfo()

```
ClientInfo ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.Get←
ClientInfo ( )
```

Gets structured information about the client (e.g., browser, OS, device) typically parsed from the User-Agent string.

Returns

A [ClientInfo](#) object containing parsed client details.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

6.117.2.2 GetCurrentCropId()

```
int? ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.GetCurrent←
CropId ( )
```

Gets the unique identifier of the crop associated with the current request context (e.g., based on the authenticated user or other request parameters).

Returns

The crop's ID as an integer, or `null` if no specific crop is associated with the context or it's unavailable.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

6.117.2.3 GetCurrentUserId()

```
int? ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.GetCurrent←
UserId ( )
```

Gets the unique identifier of the currently authenticated user associated with the request.

Returns

The user's ID as an integer, or `null` if the user is not authenticated or the ID cannot be determined.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

6.117.2.4 GetFormattedDeviceInfo()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.GetFormattedDeviceInfo ( )
```

Gets a formatted string summarizing the client's device and browser information.

Returns

A human-readable string describing the client's device and software.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

6.117.2.5 GetIpAddress()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.GetIpAddress ( )
```

Gets the IP address of the client making the current request.

Returns

The client's IP address as a string. May return an empty string or placeholder if unavailable.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

6.117.2.6 GetUserAgent()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor.GetUserAgent ( )
```

Gets the User-Agent string sent by the client in the current request header.

Returns

The client's User-Agent string. May return an empty string if unavailable.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#).

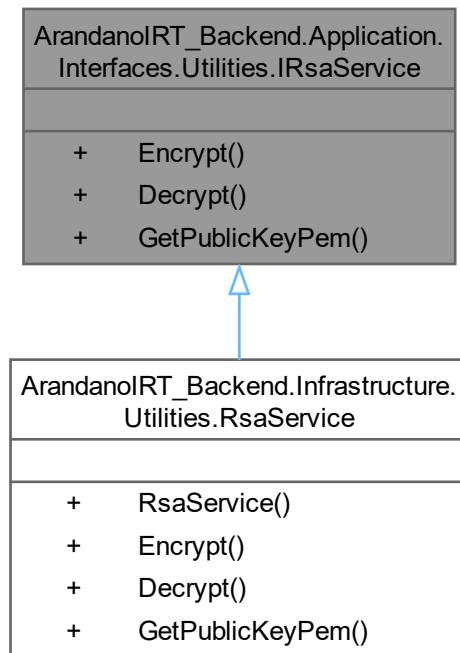
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/IRequestContextAccessor.cs

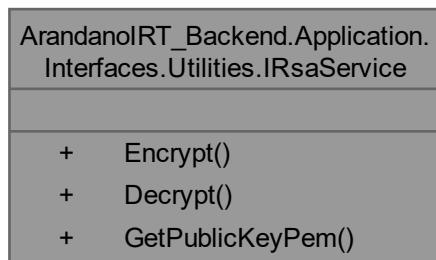
6.118 ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService Interface Reference

Service interface for RSA encryption and decryption operations.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService:



Public Member Functions

- string [Encrypt](#) (string plainText)
Encrypts data using the configured public key.
- string [Decrypt](#) (string encryptedText)
Decrypts data using the configured private key.
- string [GetPublicKeyPem](#) ()
Gets the public key in PEM format.

6.118.1 Detailed Description

Service interface for RSA encryption and decryption operations.

6.118.2 Member Function Documentation

6.118.2.1 Decrypt()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService.Decrypt (
    string encryptedText )
```

Decrypts data using the configured private key.

Parameters

<i>encryptedText</i>	The Base64 encoded encrypted string.
----------------------	--------------------------------------

Returns

The decrypted plain text.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.RsaService](#).

6.118.2.2 Encrypt()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService.Encrypt (
    string plainText )
```

Encrypts data using the configured public key.

Parameters

<i>plainText</i>	The data to encrypt.
------------------	----------------------

Returns

The Base64 encoded encrypted string.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.RsaService](#).

6.118.2.3 GetPublicKeyPem()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService.GetPublicKeyPem ( )
```

Gets the public key in PEM format.

Returns

The public key string.

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.RsaService](#).

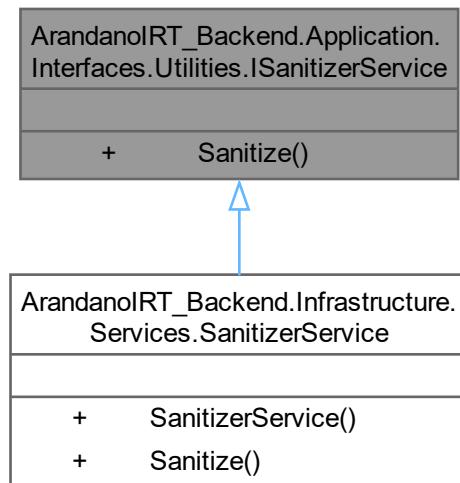
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/IRsaService.cs

6.119 ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizer ← Service Interface Reference

Defines the contract for a service that sanitizes input strings to remove potentially harmful content, such as HTML tags.

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService:



Public Member Functions

- string `Sanitize` (string? `input`)
Sanitizes the provided input string according to the implementation's policy.

6.119.1 Detailed Description

Defines the contract for a service that sanitizes input strings to remove potentially harmful content, such as HTML tags.

6.119.2 Member Function Documentation

6.119.2.1 Sanitize()

```
string ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService.Sanitize (
    string? input )
```

Sanitizes the provided input string according to the implementation's policy.

Parameters

<code>input</code>	The potentially unsafe input string.
--------------------	--------------------------------------

Returns

A sanitized string, safe for storage or specific uses.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.SanitizerService](#).

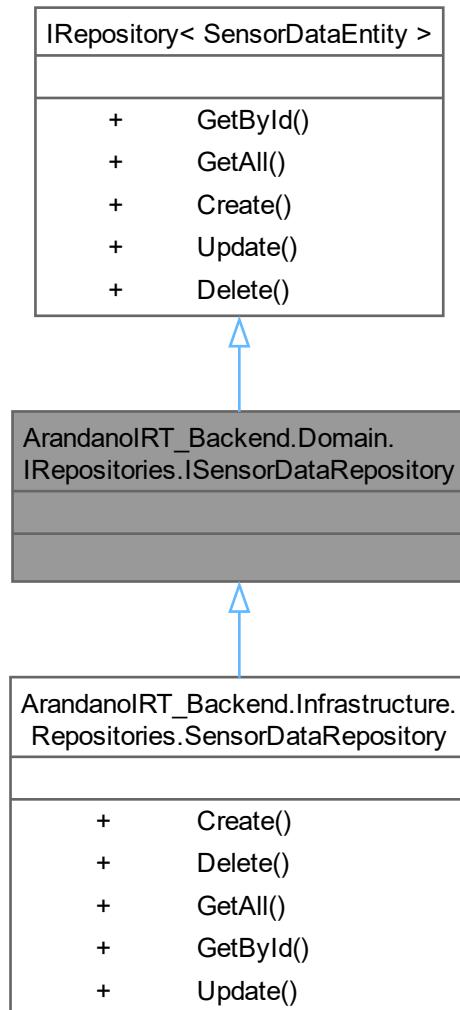
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/[ISanitizerService.cs](#)

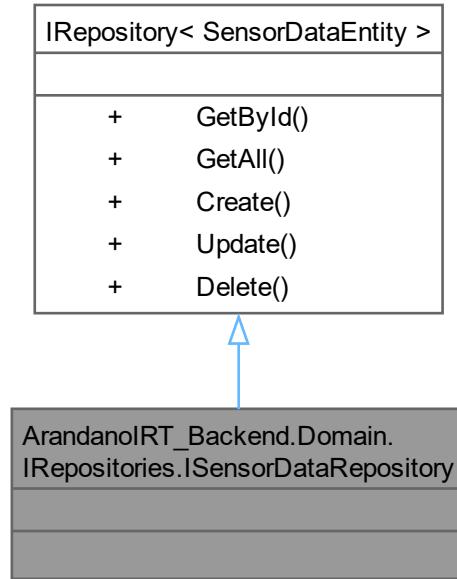
6.120 ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository Interface Reference

Defines the repository contract specifically for managing SensorDataEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< SensorDataEntity >](#)

- `Task< Result< T > > GetById (int id)`
Asynchronously retrieves an entity by its unique identifier.
- `Task< Result< IEnumerable< T > > > GetAll ()`
Asynchronously retrieves all entities of type T .
- `Task< Result< T > > Create (T entity)`
Asynchronously creates a new entity in the data store.
- `Task< Result< bool > > Update (T entity)`
Asynchronously updates an existing entity in the data store.
- `Task< Result< bool > > Delete (int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.120.1 Detailed Description

Defines the repository contract specifically for managing `SensorDataEntity` instances.

Inherits standard CRUD-like operations from `IRepository< T >`.

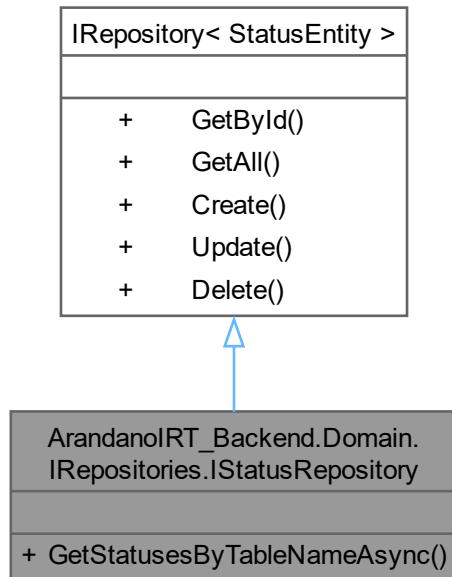
The documentation for this interface was generated from the following file:

- Domain/IRepositories/[ISensorDataRepository.cs](#)

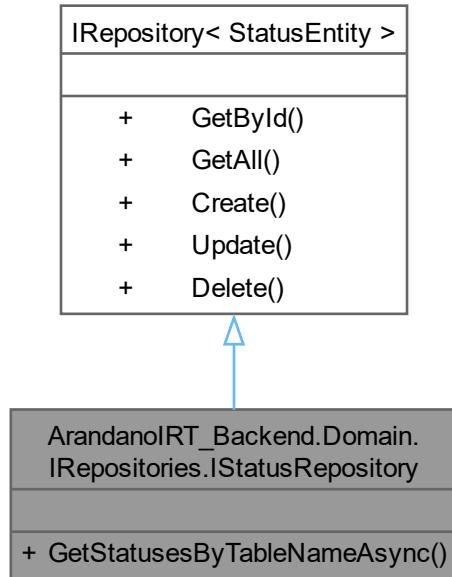
6.121 ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository Interface Reference

Defines the repository contract specifically for managing StatusEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository:



Public Member Functions

- Task< Result< IEnumerable< StatusEntity > > > > [GetStatusesByTableNameAsync](#) (string tableName)
Asynchronously retrieves all status entities associated with a specific table name, likely determined via the StatusEntity.TableRelationId.

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< StatusEntity >](#)

- Task< Result< T > > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > [GetAll](#) ()
Asynchronously retrieves all entities of type T.
- Task< Result< T > > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.121.1 Detailed Description

Defines the repository contract specifically for managing StatusEntity instances.

Includes an operation to retrieve statuses by associated table name, in addition to the standard CRUD-like operations inherited from IRepository<T>.

6.121.2 Member Function Documentation

6.121.2.1 GetStatusesByTableNameAsync()

```
Task< Result< IEnumerable< StatusEntity > > > ArandanoIRT_Backend.Domain.IRepositories.IStatus←
Repository.GetStatusesByTableNameAsync (
    string tableName )
```

Asynchronously retrieves all status entities associated with a specific table name, likely determined via the StatusEntity.TableRelationId.

Parameters

tableName	The name of the table for which to retrieve associated statuses.
-----------	--

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds an IEnumerable<T> of relevant StatusEntity instances if successful, or a failure result otherwise. The list may be empty.

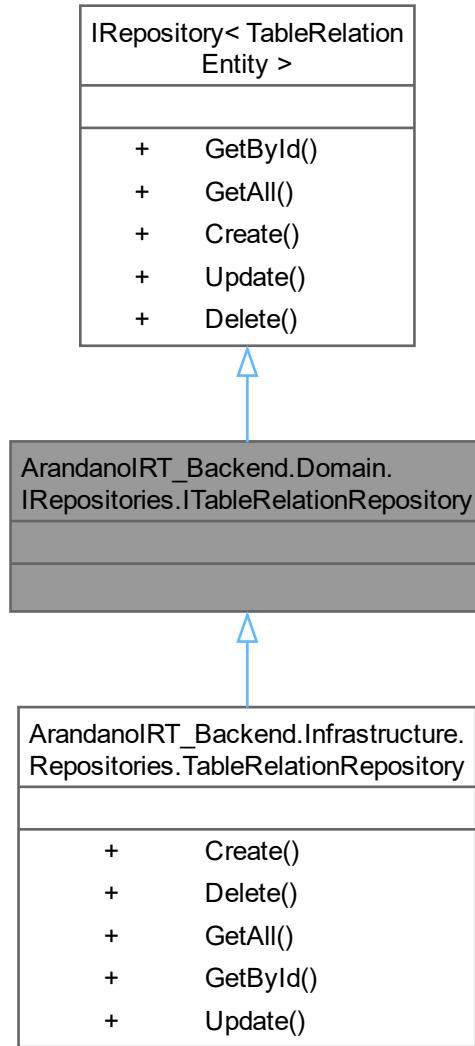
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IStatusRepository.cs

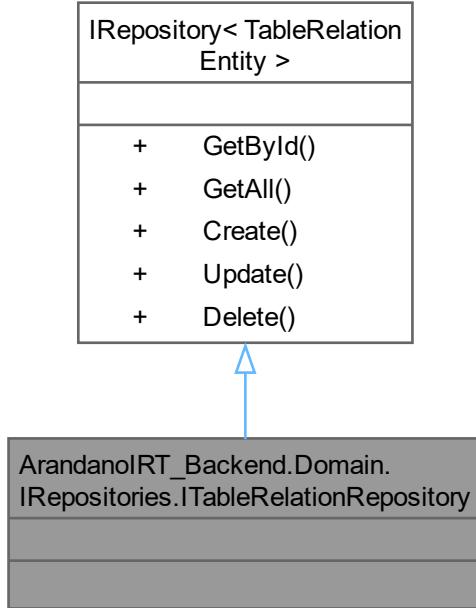
6.122 ArandanoIRT_Backend.Domain.IRepositories.ITableRelation← Repository Interface Reference

Defines the repository contract specifically for managing TableRelationEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.ITableRelationRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.ITableRelationRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< TableRelationEntity >](#)

- `Task< Result< T > > GetById (int id)`
Asynchronously retrieves an entity by its unique identifier.
- `Task< Result< IEnumerable< T > > > GetAll ()`
Asynchronously retrieves all entities of type T .
- `Task< Result< T > > Create (T entity)`
Asynchronously creates a new entity in the data store.
- `Task< Result< bool > > Update (T entity)`
Asynchronously updates an existing entity in the data store.
- `Task< Result< bool > > Delete (int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.122.1 Detailed Description

Defines the repository contract specifically for managing TableRelationEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

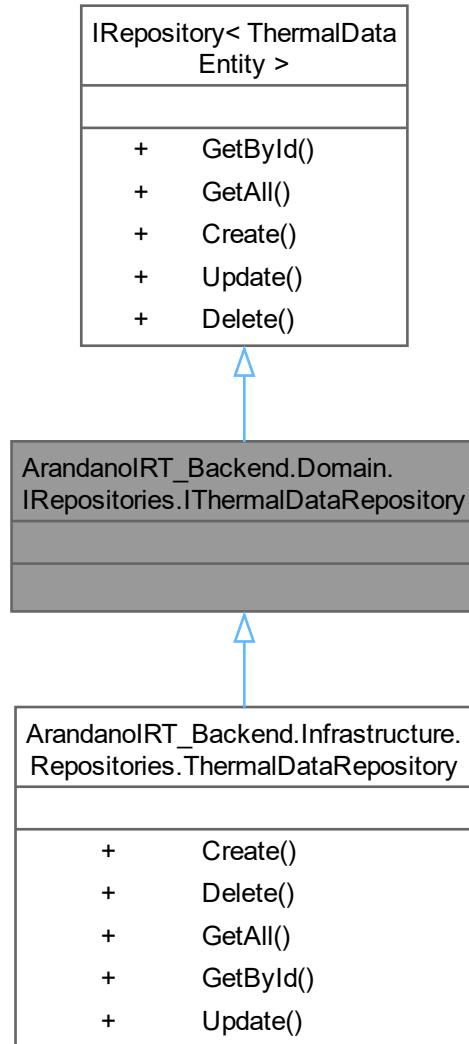
The documentation for this interface was generated from the following file:

- Domain/IRepositories/ITableRelationRepository.cs

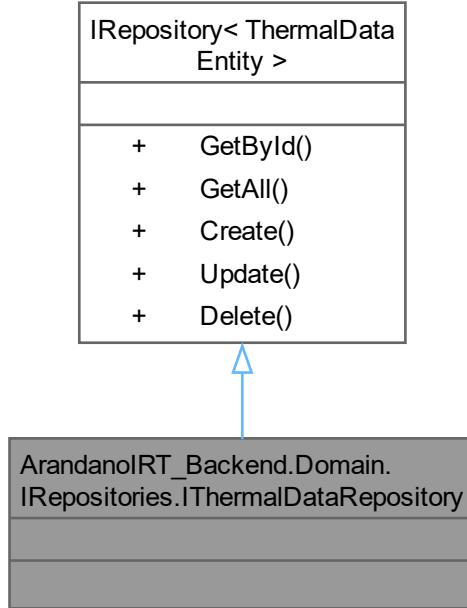
6.123 ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository Interface Reference

Defines the repository contract specifically for managing ThermalDataEntity instances.

Inheritance diagram for ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository:



Additional Inherited Members

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories.IRepository< ThermalDataEntity >](#)

- `Task< Result< T > > GetById (int id)`
Asynchronously retrieves an entity by its unique identifier.
- `Task< Result< IEnumerable< T > > > GetAll ()`
Asynchronously retrieves all entities of type T.
- `Task< Result< T > > Create (T entity)`
Asynchronously creates a new entity in the data store.
- `Task< Result< bool > > Update (T entity)`
Asynchronously updates an existing entity in the data store.
- `Task< Result< bool > > Delete (int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.123.1 Detailed Description

Defines the repository contract specifically for managing ThermalDataEntity instances.

Inherits standard CRUD-like operations from IRepository<T>.

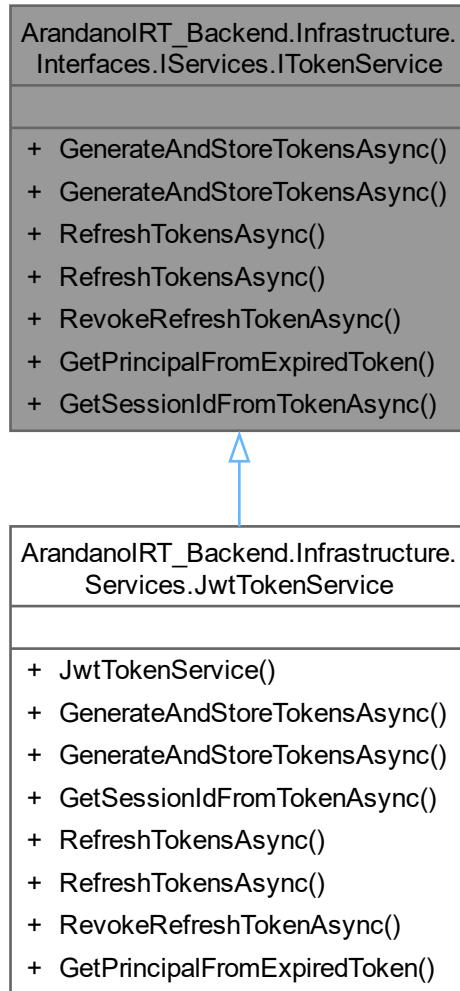
The documentation for this interface was generated from the following file:

- Domain/IRepositories/IThermalDataRepository.cs

6.124 ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.IToken ← Service Interface Reference

Defines a contract for services responsible for handling the lifecycle of authentication tokens, including the generation, storage, validation, refresh, and revocation of JWT access tokens and persistent refresh tokens.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService:

ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService
<ul style="list-style-type: none"> + GenerateAndStoreTokensAsync() + GenerateAndStoreTokensAsync() + RefreshTokensAsync() + RefreshTokensAsync() + RevokeRefreshTokenAsync() + GetPrincipalFromExpiredToken() + GetSessionIdFromTokenAsync()

Public Member Functions

- Task<[Result< TokenResponseDto >](#)> [GenerateAndStoreTokensAsync](#) ([PersonEntity](#) person, string ipAddress, string userAgent, string? deviceInfo)

Asynchronously generates a new set of access and refresh tokens for a specified user.
- Task<[Result< TokenResponseDto >](#)> [GenerateAndStoreTokensAsync](#) ([PersonEntity](#) person, string ipAddress, string userAgent)

Asynchronously generates a new set of access and refresh tokens for a specified user.
- Task<[Result< TokenResponseDto >](#)> [RefreshTokensAsync](#) (string refreshTokenValue, string ipAddress, string userAgent, string? deviceInfo)

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information including device info.
- Task<[Result< TokenResponseDto >](#)> [RefreshTokensAsync](#) (string refreshTokenValue, string ipAddress, string userAgent)

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information.
- Task<[Result >](#) [RevokeRefreshTokenAsync](#) (string refreshTokenValue, string ipAddress)

Asynchronously revokes a specific refresh token stored in the database, marking it as invalid.
- ClaimsPrincipal? [GetPrincipalFromExpiredToken](#) (string token)

Attempts to extract the claims principal from a JWT access token string, potentially ignoring lifetime validation (allowing expired tokens to be read) but still validating the signature.
- Task<[Result< long >](#)> [GetSessionIdFromTokenAsync](#) (string refreshTokenValue)

Asynchronously retrieves the session identifier associated with a given refresh token string from the database.

6.124.1 Detailed Description

Defines a contract for services responsible for handling the lifecycle of authentication tokens, including the generation, storage, validation, refresh, and revocation of JWT access tokens and persistent refresh tokens.

6.124.2 Member Function Documentation

6.124.2.1 GenerateAndStoreTokensAsync() [1/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.←
ITokenService.GenerateAndStoreTokensAsync (
    PersonEntity person,
    string ipAddress,
    string userAgent )
```

Asynchronously generates a new set of access and refresh tokens for a specified user.

Creates and stores the refresh token information (including IP, UserAgent) in the database. Device information will be stored as null or default.

Parameters

<i>person</i>	The PersonEntity for whom to generate tokens.
<i>ipAddress</i>	The IP address of the client requesting the tokens.
<i>userAgent</i>	The user agent string of the client requesting the tokens.

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.2 GenerateAndStoreTokensAsync() [2/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.←
ITokenService.GenerateAndStoreTokensAsync (
    PersonEntity person,
    string ipAddress,
    string userAgent,
    string? deviceInfo )
```

Asynchronously generates a new set of access and refresh tokens for a specified user.

Creates and stores the refresh token information (including IP, UserAgent, and device info) in the database.

Parameters

<i>person</i>	The PersonEntity for whom to generate tokens.
<i>ipAddress</i>	The IP address of the client requesting the tokens.
<i>userAgent</i>	The user agent string of the client requesting the tokens.
<i>deviceInfo</i>	Optional descriptive information about the client device.

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.3 GetPrincipalFromExpiredToken()

```
ClaimsPrincipal? ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService.Get←  
PrincipalFromExpiredToken (  
    string token )
```

Attempts to extract the claims principal from a JWT access token string, potentially ignoring lifetime validation (allowing expired tokens to be read) but still validating the signature.

Useful for retrieving user information from an expired token during the refresh process.

Parameters

<i>token</i>	The JWT access token string (can be expired but must be well-formed and have a valid signature).
--------------	--

Returns

The ClaimsPrincipal extracted from the token if parsing and signature validation succeed; otherwise, null.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.4 GetSessionIdFromTokenAsync()

```
Task< Result< long > > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService.←  
GetSessionIdFromTokenAsync (  
    string refreshTokenValue )
```

Asynchronously retrieves the session identifier associated with a given refresh token string from the database.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
--------------------------	----------------------------------

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the session ID (long) if the token is found and active, or a failure result otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.5 RefreshTokensAsync() [1/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.←
ITokenService.RefreshTokensAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent )
```

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information.

Device information will be stored as null or default.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string provided by the client.
<i>ipAddress</i>	The IP address of the client making the refresh request.
<i>userAgent</i>	The user agent string of the client making the refresh request.

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.6 RefreshTokensAsync() [2/2]

```
Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.←
ITokenService.RefreshTokensAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent,
    string? deviceInfo )
```

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information including device info.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string provided by the client.
<i>ipAddress</i>	The IP address of the client making the refresh request.
<i>userAgent</i>	The user agent string of the client making the refresh request.
<i>deviceInfo</i>	Optional descriptive information about the client device making the refresh request.

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

6.124.2.7 RevokeRefreshTokenAsync()

```
Task< Result > ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService.Revoke←  
RefreshTokenAsync ( string refreshTokenValue,  
                     string ipAddress )
```

Asynchronously revokes a specific refresh token stored in the database, marking it as invalid.

Parameters

refreshTokenValue	The opaque refresh token string to revoke.
ipAddress	The IP address initiating the revocation action (for auditing purposes).

Returns

A Task<TResult> representing the asynchronous operation, containing a Result indicating success or failure of the revocation operation.

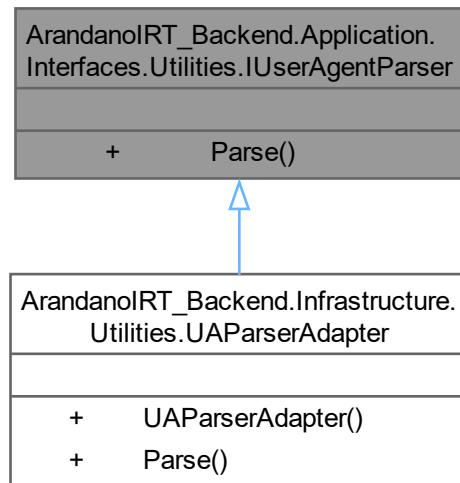
Implemented in [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#).

The documentation for this interface was generated from the following file:

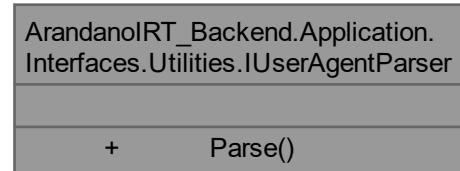
- Infrastructure/Interfaces/IServices/ITokenService.cs

6.125 ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser Interface Reference

Inheritance diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser:



Collaboration diagram for ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser:



Public Member Functions

- ClientInfo `Parse` (string? userAgentString)
Parses the user agent string and returns a ClientInfo object.

6.125.1 Member Function Documentation

6.125.1.1 Parse()

```
ClientInfo ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser.Parse (
    string? userAgentString )
```

Parses the user agent string and returns a ClientInfo object.

Parameters

<i>userAgentString</i>	
------------------------	--

Returns

Implemented in [ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter](#).

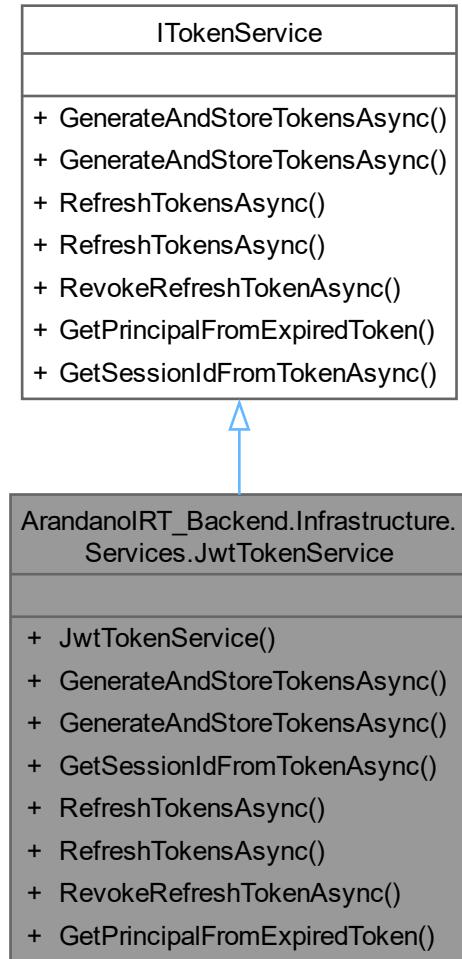
The documentation for this interface was generated from the following file:

- Application/Interfaces/Utilities/IUserAgentParser.cs

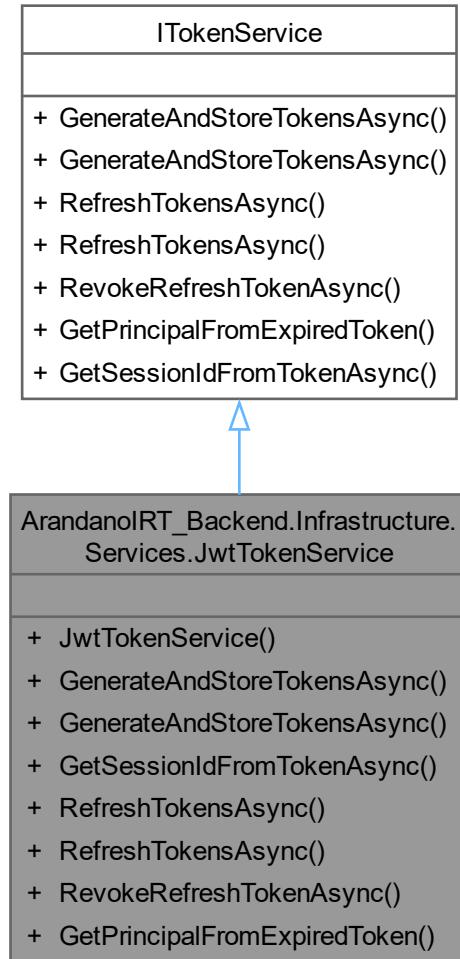
6.126 ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService Class Reference

Implements the ITokenService interface, providing services for generating, validating, refreshing, and revoking JWT access tokens and database-backed opaque refresh tokens.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService:



Public Member Functions

- `JwtTokenService (IConfiguration configuration, IRefreshTokenRepository refreshTokenRepository, IDateTimeProvider dateServiceProvider, IPersonRepository personRepository, ILogger<JwtTokenService> logger)`

Initializes a new instance of the JwtTokenService class.
- `async Task<Result<TokenResponseDto>> GenerateAndStoreTokensAsync (PersonEntity person, string ipAddress, string userAgent)`

*Asynchronously generates a new set of access and refresh tokens for a specified user.
Creates and stores the refresh token information (including IP, UserAgent) in the database. Device information will be stored as null or default.*

Parameters

person	<i>The PersonEntity for whom to generate tokens.</i>
ipAddress	<i>The IP address of the client requesting the tokens.</i>
userAgent	<i>The user agent string of the client requesting the tokens.</i>

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

- `async Task<Result<TokenResponseDto>> GenerateAndStoreTokensAsync (PersonEntity person, string ipAddress, string userAgent, string? deviceInfo)`

Asynchronously generates a new set of access and refresh tokens for a specified user.

Creates and stores the refresh token information (including IP, UserAgent, and device info) in the database.

Parameters

person	<i>The PersonEntity for whom to generate tokens.</i>
ipAddress	<i>The IP address of the client requesting the tokens.</i>
userAgent	<i>The user agent string of the client requesting the tokens.</i>
deviceInfo	<i>Optional descriptive information about the client device.</i>

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

- `async Task<Result<long>> GetSessionIdFromTokenAsync (string refreshTokenValue)`

Asynchronously retrieves the session identifier associated with a given refresh token string from the database.

Parameters

refreshTokenValue	<i>The opaque refresh token string.</i>
-------------------	---

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the session ID (long) if the token is found and active, or a failure result otherwise.

- `async Task<Result<TokenResponseDto>> RefreshTokensAsync (string refreshTokenValue, string ipAddress, string userAgent)`

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information.

Device information will be stored as null or default.

Parameters

refreshTokenValue	<i>The opaque refresh token string provided by the client.</i>
ipAddress	<i>The IP address of the client making the refresh request.</i>
userAgent	<i>The user agent string of the client making the refresh request.</i>

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

- `async Task<Result<TokenResponseDto>> RefreshTokensAsync (string refreshTokenValue, string ipAddress, string userAgent, string? deviceInfo)`

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information including device info.

Parameters

refreshTokenValue	<i>The opaque refresh token string provided by the client.</i>
ipAddress	<i>The IP address of the client making the refresh request.</i>
userAgent	<i>The user agent string of the client making the refresh request.</i>
deviceInfo	<i>Optional descriptive information about the client device making the refresh request.</i>

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

- async Task< Result > [RevokeRefreshTokenAsync](#) (string refreshTokenValue, string ipAddress)

Asynchronously revokes a specific refresh token stored in the database, marking it as invalid.

Parameters

refreshTokenValue	<i>The opaque refresh token string to revoke.</i>
ipAddress	<i>The IP address initiating the revocation action (for auditing purposes).</i>

Returns

A Task<TResult> representing the asynchronous operation, containing a Result indicating success or failure of the revocation operation.

- ClaimsPrincipal? [GetPrincipalFromExpiredToken](#) (string token)

Attempts to extract the claims principal from a JWT access token string, potentially ignoring lifetime validation (allowing expired tokens to be read) but still validating the signature.

Useful for retrieving user information from an expired token during the refresh process.

Parameters

token	<i>The JWT access token string (can be expired but must be well-formed and have a valid signature).</i>
-------	---

Returns

The ClaimsPrincipal extracted from the token if parsing and signature validation succeed; otherwise, null.

6.126.1 Detailed Description

Implements the ITokenService interface, providing services for generating, validating, refreshing, and revoking JWT access tokens and database-backed opaque refresh tokens.

6.126.2 Constructor & Destructor Documentation

6.126.2.1 JwtTokenService()

```
ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.JwtTokenService (
    IConfiguration configuration,
    IRefreshTokenRepository refreshTokenRepository,
    IDateTimeProvider dateTimeProvider,
    IPersonRepository personRepository,
    ILogger< JwtTokenService > logger ) [inline]
```

Initializes a new instance of the JwtTokenService class.

Exceptions

<i>ArgumentNullException</i>	Thrown if dependencies or required JWT config keys are null/missing.
<i>FormatException</i>	Thrown if JWT expiry/refresh config values cannot be parsed.

6.126.3 Member Function Documentation**6.126.3.1 GenerateAndStoreTokensAsync() [1/2]**

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.GenerateAndStoreTokensAsync (
    PersonEntity person,
    string ipAddress,
    string userAgent ) [inline]
```

Asynchronously generates a new set of access and refresh tokens for a specified user.

Creates and stores the refresh token information (including IP, UserAgent) in the database. Device information will be stored as null or default.

Parameters

<i>person</i>	The PersonEntity for whom to generate tokens.
<i>ipAddress</i>	The IP address of the client requesting the tokens.
<i>userAgent</i>	The user agent string of the client requesting the tokens.

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.2 GenerateAndStoreTokensAsync() [2/2]

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.GenerateAndStoreTokensAsync (
    PersonEntity person,
    string ipAddress,
    string userAgent,
    string? deviceInfo ) [inline]
```

Asynchronously generates a new set of access and refresh tokens for a specified user.

Creates and stores the refresh token information (including IP, UserAgent, and device info) in the database.

Parameters

<i>person</i>	The PersonEntity for whom to generate tokens.
<i>ipAddress</i>	The IP address of the client requesting the tokens.
<i>userAgent</i>	The user agent string of the client requesting the tokens.
<i>deviceInfo</i>	Optional descriptive information about the client device.

Returns

A Task<TResult> containing a Result<T> with a TokenResponseDto if successful, or failure otherwise.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.3 GetPrincipalFromExpiredToken()

```
ClaimsPrincipal? ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.GetPrincipalFromExpiredToken (
    string token ) [inline]
```

Attempts to extract the claims principal from a JWT access token string, potentially ignoring lifetime validation (allowing expired tokens to be read) but still validating the signature.

Useful for retrieving user information from an expired token during the refresh process.

Parameters

<i>token</i>	The JWT access token string (can be expired but must be well-formed and have a valid signature).
--------------	--

Returns

The ClaimsPrincipal extracted from the token if parsing and signature validation succeed; otherwise, null.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.4 GetSessionIdFromTokenAsync()

```
async Task< Result< long > > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.GetSessionIdFromTokenAsync (
    string refreshTokenValue ) [inline]
```

Asynchronously retrieves the session identifier associated with a given refresh token string from the database.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string.
--------------------------	----------------------------------

Returns

A Task<TResult> representing the asynchronous operation, containing a Result<T> which holds the session ID (long) if the token is found and active, or a failure result otherwise.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.5 RefreshTokensAsync() [1/2]

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.RefreshTokensAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent ) [inline]
```

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information.

Device information will be stored as null or default.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string provided by the client.
<i>ipAddress</i>	The IP address of the client making the refresh request.
<i>userAgent</i>	The user agent string of the client making the refresh request.

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.6 RefreshTokensAsync() [2/2]

```
async Task< Result< TokenResponseDto > > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.RefreshTokensAsync (
    string refreshTokenValue,
    string ipAddress,
    string userAgent,
    string? deviceInfo ) [inline]
```

Asynchronously validates a provided refresh token, generates new tokens, revokes the old one (Token Rotation), and stores the new refresh token information including device info.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string provided by the client.
<i>ipAddress</i>	The IP address of the client making the refresh request.
<i>userAgent</i>	The user agent string of the client making the refresh request.
<i>deviceInfo</i>	Optional descriptive information about the client device making the refresh request.

Returns

A Task<TResult> containing a Result<T> with a new TokenResponseDto if successful, or failure otherwise.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

6.126.3.7 RevokeRefreshTokenAsync()

```
async Task< Result > ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService.RevokeRefresh←
TokenAsync (
    string refreshTokenValue,
    string ipAddress ) [inline]
```

Asynchronously revokes a specific refresh token stored in the database, marking it as invalid.

Parameters

<i>refreshTokenValue</i>	The opaque refresh token string to revoke.
<i>ipAddress</i>	The IP address initiating the revocation action (for auditing purposes).

Returns

A Task<TResult> representing the asynchronous operation, containing a Result indicating success or failure of the revocation operation.

Implements [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#).

The documentation for this class was generated from the following file:

- Infrastructure/Services/[JwtTokenService.cs](#)

6.127 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtToken← ServiceTests Class Reference

Contains unit tests for the JwtTokenService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests:



Public Member Functions

- [JwtTokenServiceTests \(\)](#)
Initializes a new instance of the JwtTokenServiceTests class.
- [async Task GenerateAndStoreTokensAsync_WithValidUser_ShouldReturnSuccessResultWithTokens \(\)](#)

- `async Task GenerateAndStoreTokensAsync_WithNullPerson_ShouldReturnFailure ()`
Tests successful generation and storage of tokens for a valid user entity.
- `async Task GenerateAndStoreTokensAsync_WhenRepoCreateFails_ShouldReturnFailure ()`
Tests that token generation fails when the provided PersonEntity is null.
- `async Task RefreshTokensAsync_WithValidActiveToken_ShouldReturnNewTokensAndRevokeOld ()`
Tests that token generation fails if the RefreshTokenRepository's Create method fails.
- `async Task RefreshTokensAsync_WithNonExistentToken_ShouldReturnFailure ()`
Tests successful token refresh using a valid, active (not expired, not revoked) refresh token.
- `async Task RefreshTokensAsync_WithExpiredToken_ShouldReturnFailure ()`
Tests the refresh token process when the provided refresh token does not exist in the repository.
- `async Task RefreshTokensAsync_WithRevokedToken_ShouldReturnFailure ()`
Tests the refresh token process when the provided refresh token has expired.
- `async Task RefreshTokensAsync_WithValidTokenButUserNotFound_ShouldReturnFailureAndRevokeToken ()`
Tests the refresh token process when the token is valid but the associated user account cannot be found.
- `async Task RevokeRefreshTokenAsync_WithValidActiveToken_ShouldReturnSuccessAndUpdateToken ()`
Tests the successful revocation of a valid, active refresh token.
- `async Task RevokeRefreshTokenAsync_WithNonExistentToken_ShouldReturnSuccess ()`
Tests that attempting to revoke a non-existent token returns success gracefully.
- `async Task RevokeRefreshTokenAsync_WithAlreadyRevokedToken_ShouldReturnSuccess ()`
Tests that attempting to revoke an already revoked token returns success gracefully.
- `void GetPrincipalFromExpiredToken_WithValidToken_ShouldReturnPrincipalWithClaims ()`
Tests extracting claims principal from a JWT string that is structurally valid and signed correctly, even if the 'exp' claim indicates it's expired.
- `void GetPrincipalFromExpiredToken_WithInvalidSignature_ShouldReturnNull ()`
Tests that GetPrincipalFromExpiredToken returns null if the token signature is invalid.
- `void GetPrincipalFromExpiredToken_WithNullToken_ShouldReturnNull ()`
Tests that GetPrincipalFromExpiredToken returns null when the input token string is null.
- `void GetPrincipalFromExpiredToken_WithEmptyToken_ShouldReturnNull ()`
Tests that GetPrincipalFromExpiredToken returns null when the input token string is empty.
- `void GetPrincipalFromExpiredToken_WithMalformedToken_ShouldReturnNull ()`
Tests that GetPrincipalFromExpiredToken returns null when the input token string is malformed.
- `async Task GetSessionIdFromTokenAsync_WithValidActiveToken_ShouldReturnSessionId ()`
Tests retrieving the session ID from a valid, active (not expired, not revoked) refresh token.
- `async Task GetSessionIdFromTokenAsync_WithNonExistentToken_ShouldReturnFailure ()`
Tests retrieving session ID when the provided refresh token does not exist.
- `async Task GetSessionIdFromTokenAsync_WithRevokedToken_ShouldReturnFailure ()`
Tests retrieving session ID when the provided refresh token has been revoked.
- `async Task GetSessionIdFromTokenAsync_WithExpiredToken_ShouldReturnFailure ()`
Tests retrieving session ID when the provided refresh token has expired.

6.127.1 Detailed Description

Contains unit tests for the JwtTokenService class.

6.127.2 Constructor & Destructor Documentation

6.127.2.1 JwtTokenServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.JwtTokenServiceTests (
) [inline]
```

Initializes a new instance of the JwtTokenServiceTests class.

Sets up mock dependencies (Configuration, Repositories, DateTimeProvider) and instantiates the JwtTokenService for testing.

6.127.3 Member Function Documentation

6.127.3.1 GenerateAndStoreTokensAsync_WhenRepoCreateFails_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Generate←
AndStoreTokensAsync_WhenRepoCreateFails_ShouldReturnFailure () [inline]
```

Tests that token generation fails if the RefreshTokenRepository's Create method fails.

6.127.3.2 GenerateAndStoreTokensAsync_WithNullPerson_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Generate←
AndStoreTokensAsync_WithNullPerson_ShouldReturnFailure () [inline]
```

Tests that token generation fails when the provided PersonEntity is null.

6.127.3.3 GenerateAndStoreTokensAsync_WithValidUser_ShouldReturnSuccessResultWithTokens()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Generate←
AndStoreTokensAsync_WithValidUser_ShouldReturnSuccessResultWithTokens () [inline]
```

Tests successful generation and storage of tokens for a valid user entity.

Verifies the returned DTO and the interaction with the refresh token repository.

6.127.3.4 GetPrincipalFromExpiredToken_WithEmptyToken_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetPrincipal←
FromExpiredToken_WithEmptyToken_ShouldReturnNull () [inline]
```

Tests that GetPrincipalFromExpiredToken returns null when the input token string is empty.

6.127.3.5 GetPrincipalFromExpiredToken_WithInvalidSignature_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetPrincipal←
FromExpiredToken_WithInvalidSignature_ShouldReturnNull ( ) [inline]
```

Tests that GetPrincipalFromExpiredToken returns null if the token signature is invalid.

6.127.3.6 GetPrincipalFromExpiredToken_WithMalformedToken_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetPrincipal←
FromExpiredToken_WithMalformedToken_ShouldReturnNull ( ) [inline]
```

Tests that GetPrincipalFromExpiredToken returns null when the input token string is malformed.

6.127.3.7 GetPrincipalFromExpiredToken_WithNullToken_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetPrincipal←
FromExpiredToken_WithNullToken_ShouldReturnNull ( ) [inline]
```

Tests that GetPrincipalFromExpiredToken returns null when the input token string is null.

6.127.3.8 GetPrincipalFromExpiredToken_WithValidToken_ShouldReturnPrincipalWithClaims()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetPrincipal←
FromExpiredToken_WithValidToken_ShouldReturnPrincipalWithClaims ( ) [inline]
```

Tests extracting claims principal from a JWT string that is structurally valid and signed correctly, even if the 'exp' claim indicates it's expired.

6.127.3.9 GetSessionIdFromTokenAsync_WithExpiredToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetSession←
IdFromTokenAsync_WithExpiredToken_ShouldReturnFailure ( ) [inline]
```

Tests retrieving session ID when the provided refresh token has expired.

6.127.3.10 GetSessionIdFromTokenAsync_WithNonExistentToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetSession←
IdFromTokenAsync_WithNonExistentToken_ShouldReturnFailure ( ) [inline]
```

Tests retrieving session ID when the provided refresh token does not exist.

6.127.3.11 GetSessionIdFromTokenAsync_WithRevokedToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetSessionIdFromTokenAsync_WithRevokedToken_ShouldReturnFailure () [inline]
```

Tests retrieving session ID when the provided refresh token has been revoked.

6.127.3.12 GetSessionIdFromTokenAsync_WithValidActiveToken_ShouldReturnSessionId()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.GetSessionIdFromTokenAsync_WithValidActiveToken_ShouldReturnSessionId () [inline]
```

Tests retrieving the session ID from a valid, active (not expired, not revoked) refresh token.

6.127.3.13 RefreshTokensAsync_WithExpiredToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.RefreshTokensAsync_WithExpiredToken_ShouldReturnFailure () [inline]
```

Tests the refresh token process when the provided refresh token has expired.

6.127.3.14 RefreshTokensAsync_WithNonExistentToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.RefreshTokensAsync_WithNonExistentToken_ShouldReturnFailure () [inline]
```

Tests the refresh token process when the provided refresh token does not exist in the repository.

6.127.3.15 RefreshTokensAsync_WithRevokedToken_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.RefreshTokensAsync_WithRevokedToken_ShouldReturnFailure () [inline]
```

Tests the refresh token process when the provided refresh token has been revoked.

6.127.3.16 RefreshTokensAsync_WithValidActiveToken_ShouldReturnNewTokensAndRevokeOld()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.RefreshTokensAsync_WithValidActiveToken_ShouldReturnNewTokensAndRevokeOld () [inline]
```

Tests successful token refresh using a valid, active (not expired, not revoked) refresh token.

Verifies that new tokens are returned and the old refresh token is updated (revoked/replaced).

6.127.3.17 RefreshTokensAsync_WithValidTokenButUserNotFound_ShouldReturnFailureAndRevokeToken()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Refresh←  
TokensAsync_WithValidTokenButUserNotFound_ShouldReturnFailureAndRevokeToken ( ) [inline]
```

Tests the refresh token process when the token is valid but the associated user account cannot be found.

Expects the token to be revoked as a security measure.

6.127.3.18 RevokeRefreshTokenAsync_WithAlreadyRevokedToken_ShouldReturnSuccess()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Revoke←  
RefreshTokenAsync_WithAlreadyRevokedToken_ShouldReturnSuccess ( ) [inline]
```

Tests that attempting to revoke an already revoked token returns success gracefully.

6.127.3.19 RevokeRefreshTokenAsync_WithNonExistentToken_ShouldReturnSuccess()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Revoke←  
RefreshTokenAsync_WithNonExistentToken_ShouldReturnSuccess ( ) [inline]
```

Tests that attempting to revoke a non-existent token returns success gracefully.

6.127.3.20 RevokeRefreshTokenAsync_WithValidActiveToken_ShouldReturnSuccessAndUpdateToken()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests.Revoke←  
RefreshTokenAsync_WithValidActiveToken_ShouldReturnSuccessAndUpdateToken ( ) [inline]
```

Tests the successful revocation of a valid, active refresh token.

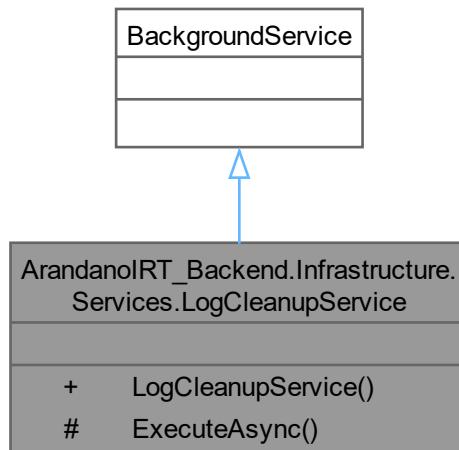
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[JwtTokenServiceTests.cs](#)

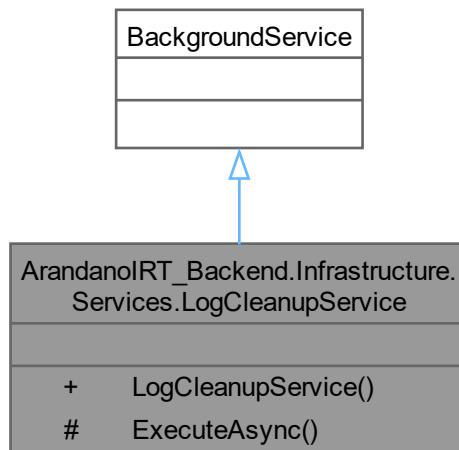
6.128 ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService Class Reference

A background service responsible for periodically deleting old log files from the configured log directory to manage disk space.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService:



Public Member Functions

- `LogCleanupService (ILogger< LogCleanupService > logger, IConfiguration configuration)`
Initializes a new instance of the LogCleanupService class.

Protected Member Functions

- override async Task `ExecuteAsync (CancellationToken stoppingToken)`
Executes the background log cleanup task.

6.128.1 Detailed Description

A background service responsible for periodically deleting old log files from the configured log directory to manage disk space.

Inherits from `BackgroundService` for long-running background task execution.

6.128.2 Constructor & Destructor Documentation

6.128.2.1 LogCleanupService()

```
ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService.LogCleanupService (
    ILogger< LogCleanupService > logger,
    IConfiguration configuration ) [inline]
```

Initializes a new instance of the `LogCleanupService` class.

Parameters

<code>logger</code>	The logger instance.
<code>configuration</code>	The application configuration (currently unused in this implementation but injected).

Exceptions

<code>ArgumentNullException</code>	Thrown if logger is null.
------------------------------------	---------------------------

6.128.3 Member Function Documentation

6.128.3.1 ExecuteAsync()

```
override async Task ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService.ExecuteAsync (
    CancellationToken stoppingToken ) [inline], [protected]
```

Executes the background log cleanup task.

Periodically calls the cleanup logic. This method is called when the IHostedService starts. The task returned should not complete until cancellation is requested.

Parameters

<i>stoppingToken</i>	A CancellationToken triggered when the application host requests shutdown.
----------------------	--

Returns

A Task representing the long-running background service execution.

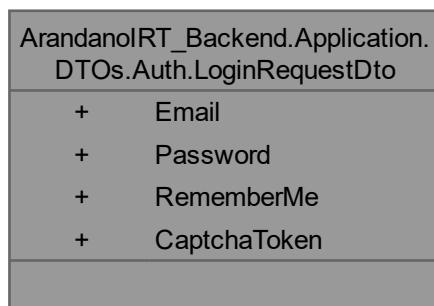
The documentation for this class was generated from the following file:

- Infrastructure/Services/[LogCleanupService.cs](#)

6.129 ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto Class Reference

Represents the request data for a user login attempt.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto:



Properties

- required string [Email](#) [get, set]
Gets or sets the user's email address.
- required string [Password](#) [get, set]
Gets or sets the user's password.
- bool [RememberMe](#) = false [get, set]
Gets or sets a value indicating whether the user wishes the login session to be persistent (e.g., "remember me").
- string? [CaptchaToken](#) [get, set]
Gets or sets the optional CAPTCHA verification token provided by the user.

6.129.1 Detailed Description

Represents the request data for a user login attempt.

6.129.2 Property Documentation

6.129.2.1 CaptchaToken

```
string? ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto.CaptchaToken [get], [set]
```

Gets or sets the optional CAPTCHA verification token provided by the user.

This may be required depending on security configurations or login attempt history.

6.129.2.2 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto.Email [get], [set]
```

Gets or sets the user's email address.

6.129.2.3 Password

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto.Password [get], [set]
```

Gets or sets the user's password.

string

6.129.2.4 RememberMe

```
bool ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto.RememberMe = false [get], [set]
```

Gets or sets a value indicating whether the user wishes the login session to be persistent (e.g., "remember me").

Defaults to `false`.

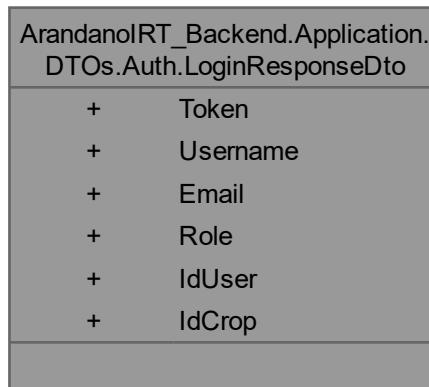
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[LoginRequestDto.cs](#)

6.130 ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto Class Reference

Represents the data returned after a successful user authentication.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto:



Properties

- required string `Token` [get, set]
Gets or sets the authentication token (e.g., JWT) issued for the session.
- required string `Username` [get, set]
Gets or sets the username of the authenticated user.
- required string `Email` [get, set]
Gets or sets the email address of the authenticated user.
- required string `Role` [get, set]
Gets or sets the role assigned to the authenticated user (e.g., "Admin", "User").
- required int `IdUser` [get, set]
Gets or sets the unique identifier of the authenticated user.
- required int `IdCrop` [get, set]
Gets or sets the unique identifier of the crop associated with the authenticated user.

6.130.1 Detailed Description

Represents the data returned after a successful user authentication.

6.130.2 Property Documentation

6.130.2.1 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.Email [get], [set]
```

Gets or sets the email address of the authenticated user.

6.130.2.2 IdCrop

```
required int ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.IdCrop [get], [set]
```

Gets or sets the unique identifier of the crop associated with the authenticated user.

This identifier likely links the user to a specific agricultural crop or related entity within the application domain.

6.130.2.3 IdUser

```
required int ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.IdUser [get], [set]
```

Gets or sets the unique identifier of the authenticated user.

6.130.2.4 Role

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.Role [get], [set]
```

Gets or sets the role assigned to the authenticated user (e.g., "Admin", "User").

6.130.2.5 Token

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.Token [get], [set]
```

Gets or sets the authentication token (e.g., JWT) issued for the session.

6.130.2.6 Username

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto.Username [get], [set]
```

Gets or sets the username of the authenticated user.

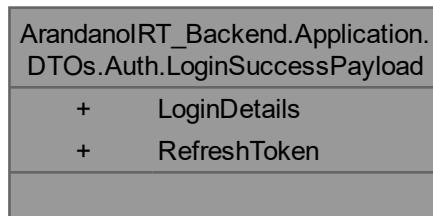
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[LoginResponseDto.cs](#)

6.131 ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload Class Reference

Represents the data returned after a successful login, including user details, access token, and the refresh token string.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload:



Properties

- required [LoginResponseDto](#) **LoginDetails** [get, set]
Standard login response details (user info, access token).
- required string **RefreshToken** [get, set]
The opaque refresh token string generated for this session.

6.131.1 Detailed Description

Represents the data returned after a successful login, including user details, access token, and the refresh token string.

6.131.2 Property Documentation

6.131.2.1 LoginDetails

```
required LoginResponseDto ArandanoIRT_Backend_ApplicationDTOsAuthLoginSuccessPayload.LoginDetails [get], [set]
```

Standard login response details (user info, access token).

6.131.2.2 RefreshToken

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload.RefreshToken [get],  
[set]
```

The opaque refresh token string generated for this session.

This should NOT typically be returned in the API response body, but is needed by the controller to set the HttpOnly cookie.

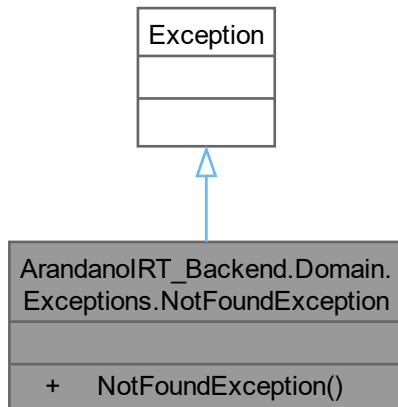
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[LoginSuccessPayload.cs](#)

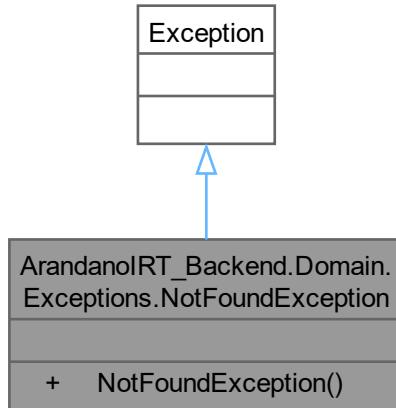
6.132 ArandanoIRT_Backend.Domain.Exceptions.NotFoundException Class Reference

Represents an error that occurs when a requested resource, entity, or record could not be found in the system or data store.

Inheritance diagram for ArandanoIRT_Backend.Domain.Exceptions.NotFoundException:



Collaboration diagram for ArandanoIRT_Backend.Domain.Exceptions.NotFoundException:



Public Member Functions

- `NotFoundException (string message)`

Initializes a new instance of the `NotFoundException` class with a specified error message that describes the resource that could not be found.

6.132.1 Detailed Description

Represents an error that occurs when a requested resource, entity, or record could not be found in the system or data store.

This exception inherits from the base `Exception` class.

6.132.2 Constructor & Destructor Documentation

6.132.2.1 NotFoundException()

```
ArandanoIRT_Backend.Domain.Exceptions.NotFoundException.NotFoundException (
    string message ) [inline]
```

Initializes a new instance of the `NotFoundException` class with a specified error message that describes the resource that could not be found.

Parameters

<i>message</i>	The message that describes the requested resource and indicates it was not found.
----------------	---

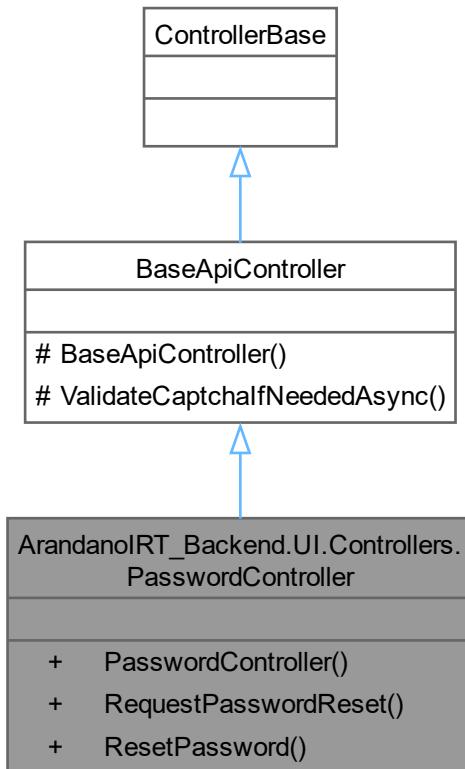
The documentation for this class was generated from the following file:

- Domain/Exceptions/[NotFoundException.cs](#)

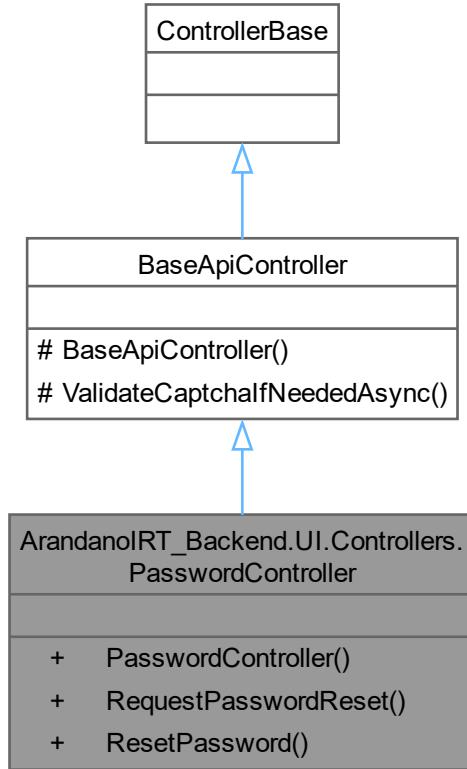
6.133 ArandanoIRT_Backend.UI.Controllers.PasswordController Class Reference

Handles endpoints related to user password management, like reset requests.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.PasswordController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.PasswordController:



Public Member Functions

- `>PasswordController (IAuthPasswordService passwordService, ICaptchaService captchaService, IWebHost< Environment environment, ILogger< PasswordController > logger)`
Initializes a new instance of the PasswordController class.
- `async Task< IActionResult > RequestPasswordReset ([FromBody] ForgotPasswordRequestDto request)`
Initiates the password reset process for the provided email address.
- `async Task< IActionResult > ResetPassword ([FromBody] ChangePasswordRequestDto request)`
Sets a new password using a valid reset code.

Additional Inherited Members

Protected Member Functions inherited from ArandanoIRT_Backend.UI.Controllers.BaseApiController

- `BaseApiController (ICaptchaService captchaService, IWebHostEnvironment environment)`

- Initializes a new instance of the BaseApiController class.*
- `async Task< IActionResult?> ValidateCaptchaIfNeededAsync (string? captchaToken, ILogger logger)`
Validates the CAPTCHA token if the application is not running in the Development environment.

6.133.1 Detailed Description

Handles endpoints related to user password management, like reset requests.

6.133.2 Constructor & Destructor Documentation

6.133.2.1 PasswordController()

```
ArandanoIRT_Backend.UI.Controllers.PasswordController.PasswordController (
    IAuthPasswordService passwordService,
    ICaptchaService captchaService,
    IWebHostEnvironment environment,
    ILogger< PasswordController > logger ) [inline]
```

Initializes a new instance of the PasswordController class.

Parameters

<code>passwordService</code>	The application service for password logic.
<code>captchaService</code>	The service for CAPTCHA validation (passed to base).
<code>environment</code>	The web hosting environment (passed to base).
<code>logger</code>	The logger for this controller.

6.133.3 Member Function Documentation

6.133.3.1 RequestPasswordReset()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.PasswordController.RequestPasswordReset (
    [FromBody] ForgotPasswordRequestDto request ) [inline]
```

Initiates the password reset process for the provided email address.

Sends a reset code to the email if it exists in the system. For security, always returns a successful response (200 OK) to prevent email enumeration.

Parameters

<code>request</code>	Contains the user's email and optional CAPTCHA token.
----------------------	---

Returns

A generic success message.

```
<response code="200">Request processed. An email will be sent if the address exists.</response>
<response code="400">Bad request (body is null, ideally should still return 200 for enumeration protection if
possible).</response> // Added 400 for null body case
```

6.133.3.2 ResetPassword()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.PasswordController.ResetPassword (
    [FromBody] ChangePasswordRequestDto request ) [inline]
```

Sets a new password using a valid reset code.

Expects the new password (`newPassword`, `confirmPassword`) to be RSA encrypted.

Parameters

<code>request</code>	Contains the reset code (<code>changeCode</code>) and the new encrypted password (and its confirmation).
----------------------	--

Returns

An `IActionResult` indicating success (204 No Content) or failure (400 Bad Request).

```
<response code="204">Password updated successfully.</response> <response code="400">Invalid request (e.g.,
invalid/expired code, passwords don't match, password policy violation).</response>
```

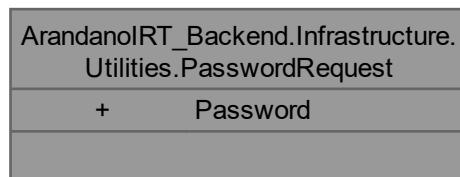
The documentation for this class was generated from the following file:

- UI/Controllers/[PasswordController.cs](#)

6.134 ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest Class Reference

Represents a simple request structure containing a password string.

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest:



Properties

- string **Password** = string.Empty [get, set]

Gets or sets the password string.

6.134.1 Detailed Description

Represents a simple request structure containing a password string.

This class appears somewhat out of place within RSA utilities but is included as provided. Its purpose might be related to receiving encrypted password data in API requests elsewhere.

6.134.2 Property Documentation

6.134.2.1 Password

```
string ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest.Password = string.Empty [get],  
[set]
```

Gets or sets the password string.

The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[RsaService.cs](#)

6.135 ArandanoIRT_Backend.Infrastructure.Data.Person Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Person:

ArandanoIRT_Backend.Infrastructure.Data.Person	
+	Idperson
+	Firstname
+	Lastname
+	Email
+	Password
+	Createdat
+	Updatedat
+	Lastloginat
+	Lastpasswordchangeat
+	Isadmin
and 13 more...	

Properties

- int `Idperson` [get, set]
- string `Firstname` = null! [get, set]
- string `Lastname` = null! [get, set]
- string `Email` = null! [get, set]
- string `Password` = null! [get, set]
- DateTime `Createdat` [get, set]
- DateTime? `Updatedat` [get, set]
- DateTime? `Lastloginat` [get, set]
- DateTime? `Lastpasswordchangeat` [get, set]
- bool `Isadmin` [get, set]
- bool `Allnotifications` [get, set]
- int? `Cropid` [get, set]
- virtual ICollection< `Changepassword` > `Changepassword` = new List< `Changepassword` >() [get, set]
- virtual ICollection< `Crop` > `Crop` = new List< `Crop` >() [get, set]
- virtual ? `Crop` `CropNavigation` [get, set]
- virtual ICollection< `Cropinvitation` > `CropinvitationCreatedbyNavigation` = new List< `Cropinvitation` >() [get, set]

- virtual ICollection< [Cropinvitation](#) > [CropinvitationUsedbyNavigation](#) = new List<[Cropinvitation](#)>() [get, set]
- virtual ICollection< [Devicedata](#) > [DevicedataRegisteredbyNavigation](#) = new List<[Devicedata](#)>() [get, set]
- virtual ICollection< [Devicedata](#) > [DevicedataUpdatedbyNavigation](#) = new List<[Devicedata](#)>() [get, set]
- virtual ICollection< [Failedloginattempt](#) > [Failedloginattempt](#) = new List<[Failedloginattempt](#)>() [get, set]
- virtual ICollection< [Plantobservation](#) > [Plantobservation](#) = new List<[Plantobservation](#)>() [get, set]
- virtual ICollection< [Plantstatehistory](#) > [Plantstatehistory](#) = new List<[Plantstatehistory](#)>() [get, set]
- virtual ICollection< [Refreshtoken](#) > [Refreshtoken](#) = new List<[Refreshtoken](#)>() [get, set]

6.135.1 Property Documentation

6.135.1.1 Allnotifications

```
bool ArandanoIRT_Backend.Infrastructure.Data.Person.Allnotifications [get], [set]
```

6.135.1.2 Changepassword

```
virtual ICollection<Changepassword> ArandanoIRT_Backend.Infrastructure.Data.Person.Changepassword
= new List<Changepassword>() [get], [set]
```

6.135.1.3 Createdat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Person.Createdat [get], [set]
```

6.135.1.4 Crop

```
virtual ICollection<Crop> ArandanoIRT_Backend.Infrastructure.Data.Person.Crop = new List<Crop>()
[get], [set]
```

6.135.1.5 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Person.Cropid [get], [set]
```

6.135.1.6 CropinvitationCreatedbyNavigation

```
virtual ICollection<Cropinvitation> ArandanoIRT_Backend.Infrastructure.Data.Person.Cropinvitation←
CreatedbyNavigation = new List<Cropinvitation>() [get], [set]
```

6.135.1.7 CropinvitationUsedbyNavigation

```
virtual ICollection<Cropinvitation> ArandanoIRT_Backend.Infrastructure.Data.Person.CropinvitationUsedbyNavigation = new List<Cropinvitation>() [get], [set]
```

6.135.1.8 CropNavigation

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Person.CropNavigation [get], [set]
```

6.135.1.9 DevicedataRegisteredbyNavigation

```
virtual ICollection<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.Person.DevicedataRegisteredbyNavigation = new List<Devicedata>() [get], [set]
```

6.135.1.10 DevicedataUpdatedbyNavigation

```
virtual ICollection<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.Person.DevicedataUpdatedbyNavigation = new List<Devicedata>() [get], [set]
```

6.135.1.11 Email

```
string ArandanoIRT_Backend.Infrastructure.Data.Person.Email = null! [get], [set]
```

6.135.1.12 Failedloginattempt

```
virtual ICollection<Failedloginattempt> ArandanoIRT_Backend.Infrastructure.Data.Person.Failedloginattempt = new List<Failedloginattempt>() [get], [set]
```

6.135.1.13 Firstname

```
string ArandanoIRT_Backend.Infrastructure.Data.Person.Firstname = null! [get], [set]
```

6.135.1.14 Idperson

```
int ArandanoIRT_Backend.Infrastructure.Data.Person.Idperson [get], [set]
```

6.135.1.15 Isadmin

```
bool ArandanoIRT_Backend.Infrastructure.Data.Person.Isadmin [get], [set]
```

6.135.1.16 Lastloginat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Person.Lastloginat [get], [set]
```

6.135.1.17 Lastname

```
string ArandanoIRT_Backend.Infrastructure.Data.Person.Lastname = null! [get], [set]
```

6.135.1.18 Lastpasswordchangeat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Person.Lastpasswordchangeat [get], [set]
```

6.135.1.19 Password

```
string ArandanoIRT_Backend.Infrastructure.Data.Person.Password = null! [get], [set]
```

6.135.1.20 Plantobservation

```
virtual ICollection<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.Person.Plantobservation  
= new List<Plantobservation>() [get], [set]
```

6.135.1.21 Plantstatehistory

```
virtual ICollection<Plantstatehistory> ArandanoIRT_Backend.Infrastructure.Data.Person.Plantstatehistory  
= new List<Plantstatehistory>() [get], [set]
```

6.135.1.22 RefreshToken

```
virtual ICollection<RefreshToken> ArandanoIRT_Backend.Infrastructure.Data.Person.RefreshToken  
= new List<RefreshToken>() [get], [set]
```

6.135.1.23 Updatedat

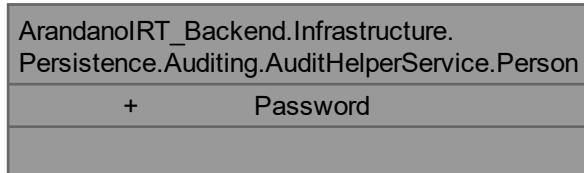
```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Person.Updatedat [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Person.cs](#)

6.136 ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person:



Properties

- string? [Password](#) [get, set]

6.136.1 Property Documentation

6.136.1.1 Password

```
string? ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person>Password [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Persistence/Auditing/[AuditHelperService.cs](#)

6.137 ArandanoIRT_Backend.Domain.Entities.PersonEntity Class Reference

Represents a user account in the system, containing personal details, credentials (typically hashed password), status flags, timestamps, and associations like the assigned crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.PersonEntity:

ArandanoIRT_Backend.Domain.Entities.PersonEntity
<ul style="list-style-type: none"> + IdPerson + FirstName + LastName + Email + Password + CreatedAt + UpdatedAt + LastLoginAt + LastPasswordChangeAt + IsAdmin + AllNotifications + Cropld + PersonEntity()

Public Member Functions

- [PersonEntity](#) (int idPerson, string firstName, string lastName, string email, string password, DateTime createdAt, bool isAdmin, bool allNotifications, int? cropld=null)

Initializes a new instance of the PersonEntity class, typically upon user creation.

Properties

- int [IdPerson](#) [get]
Gets the unique identifier for the person (user).
- string [FirstName](#) [get]
Gets the user's first name.
- string [LastName](#) [get]
Gets the user's last name.
- string [Email](#) [get]
Gets the user's email address, typically used for login.
- string [Password](#) [get]
Gets the user's password hash.
- DateTime [CreatedAt](#) [get]

- Gets the date and time (usually UTC) when the user account was created.
- DateTime? [UpdatedAt](#) [get, set]
 - Gets the date and time (usually UTC) when the user account details were last updated.
- DateTime? [LastLoginAt](#) [get, set]
 - Gets the date and time (usually UTC) of the user's last successful login.
- DateTime? [LastPasswordChangeAt](#) [get, set]
 - Gets the date and time (usually UTC) when the user's password was last changed.
- bool [IsAdmin](#) [get]
 - Gets a value indicating whether the user has administrative privileges.
- bool [AllNotifications](#) [get]
 - Gets a value indicating whether the user wishes to receive all types of notifications.
- int? [CropId](#) [get]
 - Gets the unique identifier of the crop this user is associated with.

6.137.1 Detailed Description

Represents a user account in the system, containing personal details, credentials (typically hashed password), status flags, timestamps, and associations like the assigned crop.

6.137.2 Constructor & Destructor Documentation

6.137.2.1 PersonEntity()

```
ArandanoIRT_Backend.Domain.Entities.PersonEntity.PersonEntity (
    int idPerson,
    string firstName,
    string lastName,
    string email,
    string password,
    DateTime createdAt,
    bool isAdmin,
    bool allNotifications,
    int? cropId = null ) [inline]
```

Initializes a new instance of the PersonEntity class, typically upon user creation.

Parameters

<i>idPerson</i>	The unique ID for the person (often generated by the database).
<i>firstName</i>	The user's first name.
<i>lastName</i>	The user's last name.
<i>email</i>	The user's email address.
<i>password</i>	The user's password (typically the hashed password).
<i>createdAt</i>	The creation timestamp.
<i>isAdmin</i>	Flag indicating if the user is an administrator.
<i>allNotifications</i>	Flag indicating notification preference.
<i>cropId</i>	The ID of the crop the user is associated with (optional, defaults to null).

Exceptions

<code>ArgumentException</code>	Thrown if <code>firstName</code> is null or whitespace (Spanish message: "El nombre es requerido."). Thrown if <code>lastName</code> is null or whitespace (Spanish message: "El apellido es requerido.").
--------------------------------	---

Note that `UpdatedAt`, `LastLoginAt`, and `LastPasswordChangeAt` properties are not set by this constructor and remain null initially.

6.137.3 Property Documentation

6.137.3.1 AllNotifications

```
bool ArandanoIRT_Backend.Domain.Entities.PersonEntity.AllNotifications [get]
```

Gets a value indicating whether the user wishes to receive all types of notifications.

6.137.3.2 CreatedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.PersonEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when the user account was created.

6.137.3.3 CropId

```
int? ArandanoIRT_Backend.Domain.Entities.PersonEntity.CropId [get]
```

Gets the unique identifier of the crop this user is associated with.

Nullable if not associated with a specific crop.

6.137.3.4 Email

```
string ArandanoIRT_Backend.Domain.Entities.PersonEntity.Email [get]
```

Gets the user's email address, typically used for login.

6.137.3.5 FirstName

```
string ArandanoIRT_Backend.Domain.Entities.PersonEntity.FirstName [get]
```

Gets the user's first name.

6.137.3.6 IdPerson

```
int ArandanoIRT_Backend.Domain.Entities.PersonEntity.IdPerson [get]
```

Gets the unique identifier for the person (user).

6.137.3.7 IsAdmin

```
bool ArandanoIRT_Backend.Domain.Entities.PersonEntity.IsAdmin [get]
```

Gets a value indicating whether the user has administrative privileges.

6.137.3.8 LastLoginAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.PersonEntity.LastLoginAt [get], [set]
```

Gets the date and time (usually UTC) of the user's last successful login.

Nullable if never logged in.

6.137.3.9 LastName

```
string ArandanoIRT_Backend.Domain.Entities.PersonEntity.LastName [get]
```

Gets the user's last name.

6.137.3.10 LastPasswordChangeAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.PersonEntity.LastPasswordChangeAt [get], [set]
```

Gets the date and time (usually UTC) when the user's password was last changed.

Nullable if never changed after creation.

6.137.3.11 Password

```
string ArandanoIRT_Backend.Domain.Entities.PersonEntity.Password [get]
```

Gets the user's password hash.

Direct access returns the stored hash, not the plain text password.

6.137.3.12 UpdatedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.PersonEntity.UpdatedAt [get], [set]
```

Gets the date and time (usually UTC) when the user account details were last updated.

Nullable if never updated.

The documentation for this class was generated from the following file:

- Domain/Entities/[PersonEntity.cs](#)

6.138 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests Class Reference

Contains unit tests for the PersonEntity class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests:

ArandanoIRT_Backend.Tests. Unit.Domain.Entities.PersonEntityTests
+ Constructor_ShouldThrowArgument Exception_WhenFirstNameIsNull() + Constructor_ShouldThrowArgument Exception_WhenFirstNameIsEmpty() + Constructor_ShouldThrowArgument Exception_WhenFirstNameIsWhitespace() + Constructor_ShouldThrowArgument Exception_WhenLastNameIsNull() + Constructor_ShouldThrowArgument Exception_WhenLastNameIsEmpty() + Constructor_ShouldThrowArgument Exception_WhenLastNameIsWhitespace() + Constructor_ShouldCreate Instance_WhenValidArgumentsProvided() + Constructor_ShouldCreate Instance_WhenOptionalCropIdIs Null()

Public Member Functions

- void [Constructor_ShouldThrowArgumentException_WhenFirstNameIsNull\(\)](#)
Verifies that the constructor throws ArgumentException when FirstName is null.
- void [Constructor_ShouldThrowArgumentException_WhenFirstNameIsEmpty\(\)](#)
Verifies that the constructor throws ArgumentException when FirstName is empty.
- void [Constructor_ShouldThrowArgumentException_WhenFirstNameIsWhitespace\(\)](#)
Verifies that the constructor throws ArgumentException when FirstName is whitespace.
- void [Constructor_ShouldThrowArgumentException_WhenLastNameIsNull\(\)](#)
Verifies that the constructor throws ArgumentException when LastName is null.
- void [Constructor_ShouldThrowArgumentException_WhenLastNameIsEmpty\(\)](#)
Verifies that the constructor throws ArgumentException when LastName is empty.
- void [Constructor_ShouldThrowArgumentException_WhenLastNameIsWhitespace\(\)](#)
Verifies that the constructor throws ArgumentException when LastName is whitespace.
- void [Constructor_ShouldCreateInstance_WhenValidArgumentsProvided\(\)](#)
Verifies that the constructor successfully creates an instance and assigns properties correctly when valid arguments are provided.
- void [Constructor_ShouldCreateInstance_WhenOptionalCropIdIsNull\(\)](#)
Verifies that the constructor works correctly when the optional CropId is null.

6.138.1 Detailed Description

Contains unit tests for the PersonEntity class.

Focuses on constructor validation logic.

6.138.2 Member Function Documentation

6.138.2.1 Constructor_ShouldCreateInstance_WhenOptionalCropIdIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldCreate←  
Instance_WhenOptionalCropIdIsNull ( ) [inline]
```

Verifies that the constructor works correctly when the optional CropId is null.

6.138.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldCreate←  
Instance_WhenValidArgumentsProvided ( ) [inline]
```

Verifies that the constructor successfully creates an instance and assigns properties correctly when valid arguments are provided.

6.138.2.3 Constructor_ShouldThrowArgumentException_WhenFirstNameIsEmpty()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenFirstNameIsEmpty ( ) [inline]
```

Verifies that the constructor throws ArgumentException when FirstName is empty.

6.138.2.4 Constructor_ShouldThrowArgumentException_WhenFirstNameIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenFirstNameIsNull ( ) [inline]
```

Verifies that the constructor throws ArgumentException when FirstName is null.

6.138.2.5 Constructor_ShouldThrowArgumentException_WhenFirstNameIsWhitespace()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenFirstNameIsWhitespace ( ) [inline]
```

Verifies that the constructor throws ArgumentException when FirstName is whitespace.

6.138.2.6 Constructor_ShouldThrowArgumentException_WhenLastNameIsEmpty()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenLastNameIsEmpty ( ) [inline]
```

Verifies that the constructor throws ArgumentException when LastName is empty.

6.138.2.7 Constructor_ShouldThrowArgumentException_WhenLastNameIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenLastNameIsNull ( ) [inline]
```

Verifies that the constructor throws ArgumentException when LastName is null.

6.138.2.8 Constructor_ShouldThrowArgumentException_WhenLastNameIsWhitespace()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenLastNameIsWhitespace ( ) [inline]
```

Verifies that the constructor throws ArgumentException when LastName is whitespace.

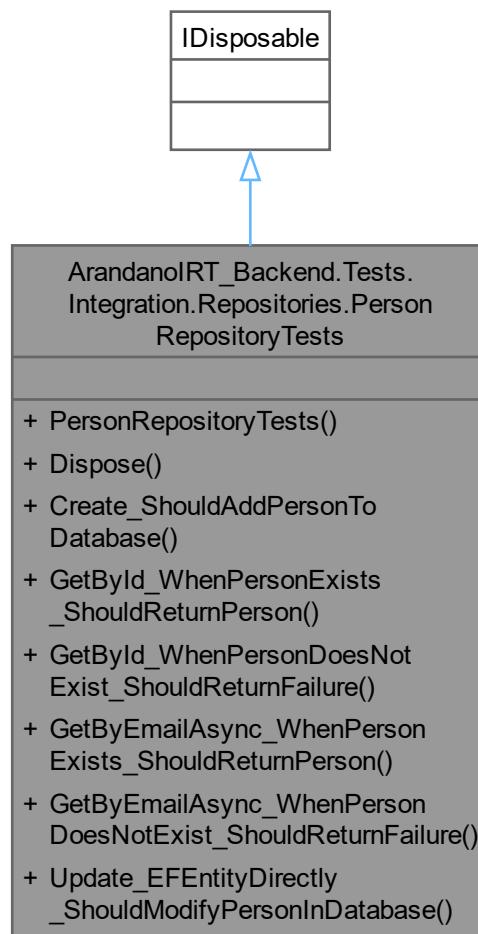
The documentation for this class was generated from the following file:

- Tests/Unit/Domain/Entities/[PersonEntityTests.cs](#)

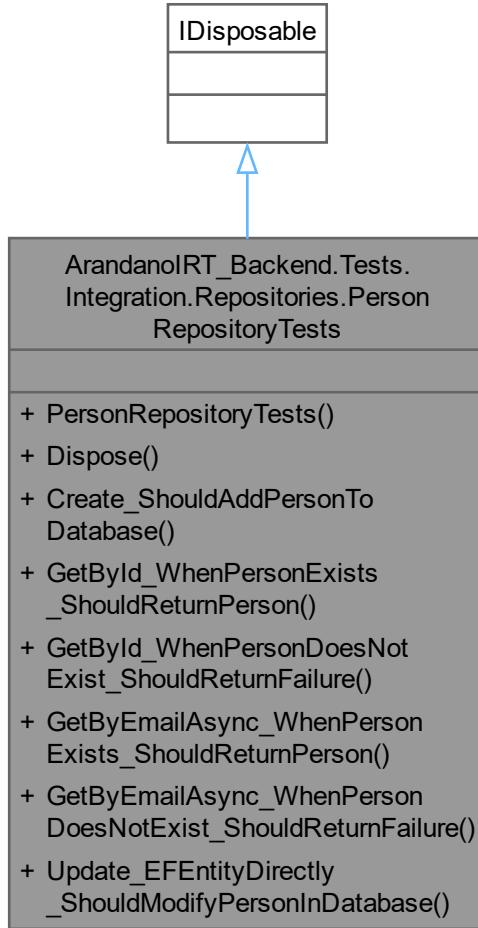
6.139 ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests Class Reference

Contains integration tests for the PersonRepository using an SQLite in-memory database.

Inheritance diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests:



Public Member Functions

- [PersonRepositoryTests \(\)](#)

Initializes a new instance of the PersonRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

- [void Dispose \(\)](#)

Disposes the database context and connection used by the test instance.

- [async Task Create_ShouldAddPersonToDatabase \(\)](#)
- [async Task GetById_WhenPersonExists_ShouldReturnPerson \(\)](#)
- [async Task GetById_WhenPersonDoesNotExist_ShouldReturnFailure \(\)](#)
- [async Task GetByEmailAsync_WhenPersonExists_ShouldReturnPerson \(\)](#)
- [async Task GetByEmailAsync_WhenPersonDoesNotExist_ShouldReturnFailure \(\)](#)
- [async Task Update_EFEntityDirectly_ShouldModifyPersonInDatabase \(\)](#)

6.139.1 Detailed Description

Contains integration tests for the PersonRepository using an SQLite in-memory database.

6.139.2 Constructor & Destructor Documentation

6.139.2.1 PersonRepositoryTests()

```
ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.PersonRepositoryTests ( )  
[inline]
```

Initializes a new instance of the PersonRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

6.139.3 Member Function Documentation

6.139.3.1 Create_ShouldAddPersonToDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.Create_ShouldAddPersonToDatabase ( ) [inline]
```

6.139.3.2 Dispose()

```
void ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.Dispose ( ) [inline]
```

Disposes the database context and connection used by the test instance.

6.139.3.3 GetByEmailAsync_WhenPersonDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.GetByEmailAsync_WhenPersonDoesNotExist_ShouldReturnFailure ( ) [inline]
```

6.139.3.4 GetByEmailAsync_WhenPersonExists_ShouldReturnPerson()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.GetByEmailAsync_WhenPersonExists_ShouldReturnPerson ( ) [inline]
```

6.139.3.5 GetById_WhenPersonDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.GetById_WhenPersonDoesNotExist_ShouldReturnFailure ( ) [inline]
```

6.139.3.6 GetById_WhenPersonExists_ShouldReturnPerson()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.GetById_When←  
PersonExists_ShouldReturnPerson () [inline]
```

6.139.3.7 Update_EFEntityDirectly_ShouldModifyPersonInDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests.Update_←  
EFEntityDirectly_ShouldModifyPersonInDatabase () [inline]
```

The documentation for this class was generated from the following file:

- Tests/Integration/Repositories/[PersonRepositoryTests.cs](#)

6.140 ArandanoIRT_Backend.Infrastructure.Data.Plantdata Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Plantdata:

ArandanoIRT_Backend.Infrastructure. Data.Plantdata	
+	Idplantdata
+	Nameplant
+	Registeredat
+	Updatedat
+	Statusid
+	Cropid
+	Crop
+	Devicedata
+	Plantobservation
+	Plantstatehistory
+	Sensordata
+	Status
+	Thermaldata

Properties

- int `Idplantdata` [get, set]
- string `Nameplant` = null! [get, set]
- DateTime `Registeredat` [get, set]
- DateTime? `Updatedat` [get, set]
- int? `Statusid` [get, set]
- int? `Cropid` [get, set]
- virtual ? `Crop Crop` [get, set]
- virtual ICollection<`Devicedata`> `Devicedata` = new List<`Devicedata`>() [get, set]
- virtual ICollection<`Plantobservation`> `Plantobservation` = new List<`Plantobservation`>() [get, set]
- virtual ICollection<`Plantstatehistory`> `Plantstatehistory` = new List<`Plantstatehistory`>() [get, set]
- virtual ICollection<`Sensordata`> `Sensordata` = new List<`Sensordata`>() [get, set]
- virtual ? `Status Status` [get, set]
- virtual ICollection<`Thermaldata`> `Thermaldata` = new List<`Thermaldata`>() [get, set]

6.140.1 Property Documentation

6.140.1.1 Crop

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Crop [get], [set]
```

6.140.1.2 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Cropid [get], [set]
```

6.140.1.3 Devicedata

```
virtual ICollection<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Devicedata =  
new List<Devicedata>() [get], [set]
```

6.140.1.4 Idplantdata

```
int ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Idplantdata [get], [set]
```

6.140.1.5 Nameplant

```
string ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Nameplant = null! [get], [set]
```

6.140.1.6 Plantobservation

```
virtual ICollection<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Plantobservation  
= new List<Plantobservation>() [get], [set]
```

6.140.1.7 Plantstatehistory

```
virtual ICollection<Plantstatehistory> ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Plantstatehistory = new List<Plantstatehistory>() [get], [set]
```

6.140.1.8 Registeredat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Registeredat [get], [set]
```

6.140.1.9 Sensordata

```
virtual ICollection<Sensordata> ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Sensordata = new List<Sensordata>() [get], [set]
```

6.140.1.10 Status

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Status [get], [set]
```

6.140.1.11 Statusid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Statusid [get], [set]
```

6.140.1.12 Thermaldata

```
virtual ICollection<Thermaldata> ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Thermaldata = new List<Thermaldata>() [get], [set]
```

6.140.1.13 Updatedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.Plantdata.Updatedat [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Plantdata.cs](#)

6.141 ArandanoIRT_Backend.Domain.Entities.PlantDataEntity Class Reference

Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.PlantDataEntity:

ArandanoIRT_Backend.Domain.Entities.PlantDataEntity
+ IdPlantState
+ NamePlant
+ RegisteredAt
+ UpdatedAt
+ StatusId
+ CropId
+ PlantDataEntity()

Public Member Functions

- [PlantDataEntity](#) (int idPlantState, string namePlant, DateTime registeredAt, int? statusId, int? cropId)
Initializes a new instance of the PlantDataEntity class, typically upon plant creation.

Properties

- int [IdPlantState](#) [get]
Gets the unique identifier for the plant record (property name is IdPlantState).
- string [NamePlant](#) [get]
Gets the name or identifier assigned to the plant.
- DateTime [RegisteredAt](#) [get]
Gets the date and time (usually UTC) when the plant record was created or registered.
- DateTime? [UpdatedAt](#) [get, set]
Gets the date and time (usually UTC) when the plant record was last updated.
- int? [StatusId](#) [get]
Gets the identifier representing the current status of the plant (e.g., Healthy, Diseased, Harvested).
- int? [CropId](#) [get]
Gets the unique identifier of the crop to which this plant belongs.

6.141.1 Detailed Description

Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop.

6.141.2 Constructor & Destructor Documentation

6.141.2.1 PlantDataEntity()

```
ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.PlantDataEntity (
    int idPlantState,
    string namePlant,
    DateTime registeredAt,
    int? statusId,
    int? cropId ) [inline]
```

Initializes a new instance of the PlantDataEntity class, typically upon plant creation.

Parameters

<i>idPlantState</i>	The unique ID for this plant record (often generated by the database).
<i>namePlant</i>	The name or identifier of the plant.
<i>registeredAt</i>	The registration timestamp.
<i>statusId</i>	The initial status ID (nullable).
<i>cropId</i>	The ID of the crop this plant belongs to (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>namePlant</i> is null or whitespace (Spanish message: "El nombre de la planta es requerido.").
--------------------------	---

Note that the UpdatedAt property is not set by this constructor and remains null initially.

6.141.3 Property Documentation

6.141.3.1 CropId

```
int? ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.CropId [get]
```

Gets the unique identifier of the crop to which this plant belongs.

Nullable.

6.141.3.2 IdPlantState

```
int ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.IdPlantState [get]
```

Gets the unique identifier for the plant record (property name is IdPlantState).

6.141.3.3 NamePlant

```
string ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.NamePlant [get]
```

Gets the name or identifier assigned to the plant.

6.141.3.4 RegisteredAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.RegisteredAt [get]
```

Gets the date and time (usually UTC) when the plant record was created or registered.

6.141.3.5 StatusId

```
int? ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.StatusId [get]
```

Gets the identifier representing the current status of the plant (e.g., Healthy, Diseased, Harvested).

Nullable.

6.141.3.6 UpdatedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.PlantDataEntity.UpdatedAt [get], [set]
```

Gets the date and time (usually UTC) when the plant record was last updated.

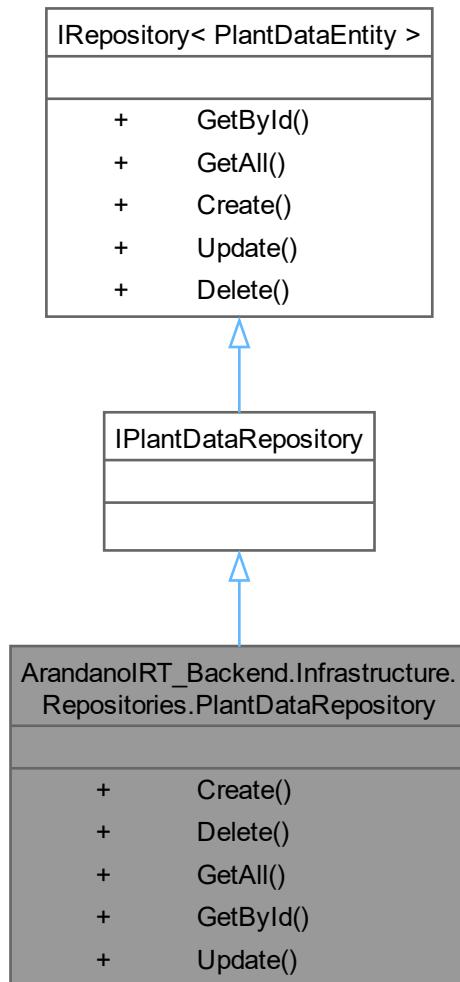
Nullable if never updated.

The documentation for this class was generated from the following file:

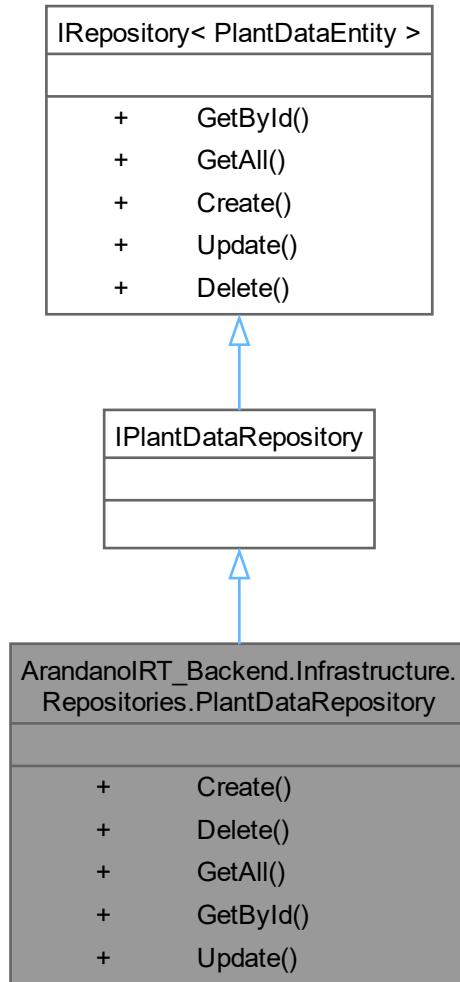
- Domain/Entities/[PlantDataEntity.cs](#)

6.142 ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository:



Public Member Functions

- `Task< Result< PlantDataEntity > > Create (PlantDataEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< PlantDataEntity > > > GetAll ()`
- `Task< Result< PlantDataEntity > > GetById (int id)`
- `Task< Result< bool > > Update (PlantDataEntity entity)`

Public Member Functions inherited from

`ArandanoIRT_Backend.Domain.IRepositories IRepository< PlantDataEntity >`

- `Task< Result< T > > GetById (int id)`

- Task< **Result**< I`Enumerable`< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< **Result**< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< **Result**< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< **Result**< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.142.1 Member Function Documentation

6.142.1.1 Create()

```
Task< Result< PlantDataEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantData←
Repository.Create (
    PlantDataEntity entity ) [inline]
```

6.142.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository.←
Delete (
    int id ) [inline]
```

6.142.1.3 GetAll()

```
Task< Result< IEnumerable< PlantDataEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.←
PlantDataRepository.GetAll ( ) [inline]
```

6.142.1.4 GetById()

```
Task< Result< PlantDataEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantData←
Repository.GetById (
    int id ) [inline]
```

6.142.1.5 Update()

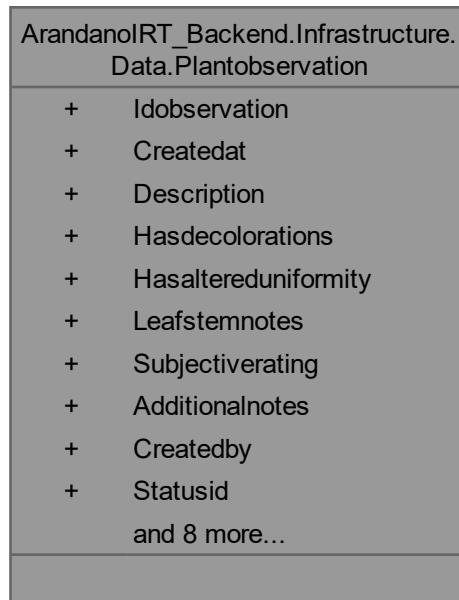
```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository.←
Update (
    PlantDataEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/[Repositories/PlantDataRepository.cs](#)

6.143 ArandanoIRT_Backend.Infrastructure.Data.Plantobservation Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Plantobservation:



Properties

- int **Idobservation** [get, set]
- DateTime **Createdat** [get, set]
- string **Description** = null! [get, set]
- bool **Hasdecolorations** [get, set]
- bool **Hasaltereduniformity** [get, set]
- string? **Leafstemnotes** [get, set]
- short **Subjectiverating** [get, set]
- string? **Additionalnotes** [get, set]
- int? **Createdby** [get, set]
- int? **Statusid** [get, set]
- int? **Plantid** [get, set]
- int? **Lastsensordataid** [get, set]
- int? **Lastthermaldataid** [get, set]
- virtual ? Person **CreatedbyNavigation** [get, set]
- virtual ? Sensordata **Lastsensordata** [get, set]
- virtual ? Thermaldata **Lastthermaldata** [get, set]
- virtual ? Plantdata **Plant** [get, set]
- virtual ? Status **Status** [get, set]

6.143.1 Property Documentation

6.143.1.1 Additionalnotes

```
string? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Additionalnotes [get], [set]
```

6.143.1.2 Createdat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Createdat [get], [set]
```

6.143.1.3 Createdby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Createdby [get], [set]
```

6.143.1.4 CreatedbyNavigation

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.CreatedbyNavigation  
[get], [set]
```

6.143.1.5 Description

```
string ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Description = null! [get], [set]
```

6.143.1.6 Hasaltereduniformity

```
bool ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Hasaltereduniformity [get], [set]
```

6.143.1.7 Hasdecolorations

```
bool ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Hasdecolorations [get], [set]
```

6.143.1.8 Idobservation

```
int ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Idobservation [get], [set]
```

6.143.1.9 Lastsensordata

```
virtual ? Sensordata ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Lastsensordata  
[get], [set]
```

6.143.1.10 Lastsensordataid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Lastsensordataid [get], [set]
```

6.143.1.11 Lastthermaldata

```
virtual ? Thermaldata ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Lastthermaldata [get], [set]
```

6.143.1.12 Lastthermaldataid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Lastthermaldataid [get], [set]
```

6.143.1.13 Leafstemnotes

```
string? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Leafstemnotes [get], [set]
```

6.143.1.14 Plant

```
virtual ? Plantdata ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Plant [get], [set]
```

6.143.1.15 Plantid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Plantid [get], [set]
```

6.143.1.16 Status

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Status [get], [set]
```

6.143.1.17 Statusid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Statusid [get], [set]
```

6.143.1.18 Subjectiverating

```
short ArandanoIRT_Backend.Infrastructure.Data.Plantobservation.Subjectiverating [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Plantobservation.cs](#)

6.144 ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity Class Reference

Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity:

ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity
<ul style="list-style-type: none"> + IdObservation + CreatedAt + Description + HasDecolorations + HasAlteredUniformity + LeafStemNotes + SubjectiveRating + AdditionalNotes + CreatedBy + StatusId + PlantId + LastSensorDataId + LastThermalDataId + PlantObservationEntity()

Public Member Functions

- [PlantObservationEntity](#) (int idObservation, DateTime createdAt, string description, bool hasDecolorations, bool hasAlteredUniformity, short subjectiveRating, int? createdBy=null, int? statusId=null, int? plantId=null, int? lastSensorDataId=null, int? lastThermalDataId=null, string? leafStemNotes=null, string? additionalNotes=null)
- Initializes a new instance of the PlantObservationEntity class.*

Properties

- int [IdObservation](#) [get, set]

Gets or sets the unique identifier for this plant observation record.
- DateTime [CreatedAt](#) [get, set]

- Gets or sets the date and time (usually UTC) when the observation was created or recorded.
- string **Description** = null! [get, set]
 - Gets or sets a general description of the observation.
- bool **HasDecolorations** [get, set]
 - Gets or sets a value indicating whether discolorations were observed on the plant.
- bool **HasAlteredUniformity** [get, set]
 - Gets or sets a value indicating whether altered uniformity (e.g., in size or shape) was observed.
- string? **LeafStemNotes** [get, set]
 - Gets or sets specific notes related to the plant's leaves or stems.
- short **SubjectiveRating** [get, set]
 - Gets or sets a subjective rating provided during the observation, expected to be between 1 and 3.
- string? **AdditionalNotes** [get, set]
 - Gets or sets any additional free-text notes related to the observation.
- int? **CreatedBy** [get, set]
 - Gets or sets the unique identifier of the user who created this observation record.
- int? **StatusId** [get, set]
 - Gets or sets the identifier representing the status assigned to this observation or the plant at the time of observation.
- int? **PlantId** [get, set]
 - Gets or sets the unique identifier of the plant to which this observation pertains.
- int? **LastSensorDataId** [get, set]
 - Gets or sets the identifier of the last relevant general sensor data record associated with this observation.
- int? **LastThermalDataId** [get, set]
 - Gets or sets the identifier of the last relevant thermal sensor data record associated with this observation.

6.144.1 Detailed Description

Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers.

Properties in this entity use public getters and setters, allowing modification after instantiation.

6.144.2 Constructor & Destructor Documentation

6.144.2.1 PlantObservationEntity()

```
ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.PlantObservationEntity (
    int idObservation,
    DateTime createdAt,
    string description,
    bool hasDecolorations,
    bool hasAlteredUniformity,
    short subjectiveRating,
    int? createdBy = null,
    int? statusId = null,
    int? plantId = null,
    int? lastSensorDataId = null,
    int? lastThermalDataId = null,
    string? leafStemNotes = null,
    string? additionalNotes = null ) [inline]
```

Initializes a new instance of the PlantObservationEntity class.

Parameters

<i>idObservation</i>	The unique ID for this observation record.
<i>createdAt</i>	The creation timestamp.
<i>description</i>	A general description of the observation.
<i>hasDecolorations</i>	Indicates if discolorations were observed.
<i>hasAlteredUniformity</i>	Indicates if altered uniformity was observed.
<i>subjectiveRating</i>	A subjective rating (must be between 1 and 3).
<i>createdBy</i>	The ID of the user creating the observation (optional).
<i>statusId</i>	The status ID associated with the observation (optional).
<i>plantId</i>	The ID of the observed plant (optional).
<i>lastSensorDataId</i>	ID of the last relevant sensor data record (optional).
<i>lastThermalDataId</i>	ID of the last relevant thermal data record (optional).
<i>leafStemNotes</i>	Specific notes about leaves/stems (optional).
<i>additionalNotes</i>	Additional free-text notes (optional).

Exceptions

<i>ArgumentException</i>	Thrown if <i>description</i> is null or empty (Spanish message: "La descripción no puede ser nula o vacía").
<i>ArgumentOutOfRangeException</i>	Thrown if <i>subjectiveRating</i> is not between 1 and 3 (Spanish message: "La calificación subjetiva debe estar entre 1 y 3").

6.144.3 Property Documentation

6.144.3.1 AdditionalNotes

string? ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.AdditionalNotes [get], [set]

Gets or sets any additional free-text notes related to the observation.

Nullable.

6.144.3.2 CreatedAt

DateTime ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.CreatedAt [get], [set]

Gets or sets the date and time (usually UTC) when the observation was created or recorded.

6.144.3.3 CreatedBy

int? ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.CreatedBy [get], [set]

Gets or sets the unique identifier of the user who created this observation record.

Nullable.

6.144.3.4 Description

```
string ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.Description = null! [get],  
[set]
```

Gets or sets a general description of the observation.

Initialized with null-forgiving operator; expected to be non-null after construction due to validation.

6.144.3.5 HasAlteredUniformity

```
bool ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.HasAlteredUniformity [get], [set]
```

Gets or sets a value indicating whether altered uniformity (e.g., in size or shape) was observed.

6.144.3.6 HasDecolorations

```
bool ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.HasDecolorations [get], [set]
```

Gets or sets a value indicating whether discolorations were observed on the plant.

6.144.3.7 IdObservation

```
int ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.IdObservation [get], [set]
```

Gets or sets the unique identifier for this plant observation record.

6.144.3.8 LastSensorDataId

```
int? ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.LastSensorDataId [get], [set]
```

Gets or sets the identifier of the last relevant general sensor data record associated with this observation.

Nullable.

6.144.3.9 LastThermalDataId

```
int? ArandanoIRT_Backend.Domain.Entities.PlanObservationEntity.LastThermalDataId [get], [set]
```

Gets or sets the identifier of the last relevant thermal sensor data record associated with this observation.

Nullable.

6.144.3.10 LeafStemNotes

```
string? ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.LeafStemNotes [get], [set]
```

Gets or sets specific notes related to the plant's leaves or stems.

Nullable.

6.144.3.11 PlantId

```
int? ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.PlantId [get], [set]
```

Gets or sets the unique identifier of the plant to which this observation pertains.

Nullable.

6.144.3.12 StatusId

```
int? ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.StatusId [get], [set]
```

Gets or sets the identifier representing the status assigned to this observation or the plant at the time of observation.

Nullable.

6.144.3.13 SubjectiveRating

```
short ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity.SubjectiveRating [get], [set]
```

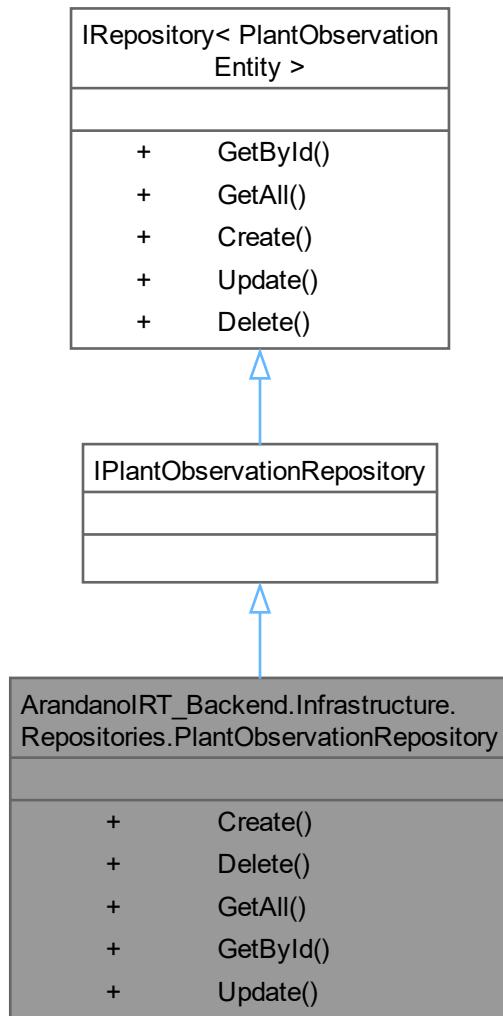
Gets or sets a subjective rating provided during the observation, expected to be between 1 and 3.

The documentation for this class was generated from the following file:

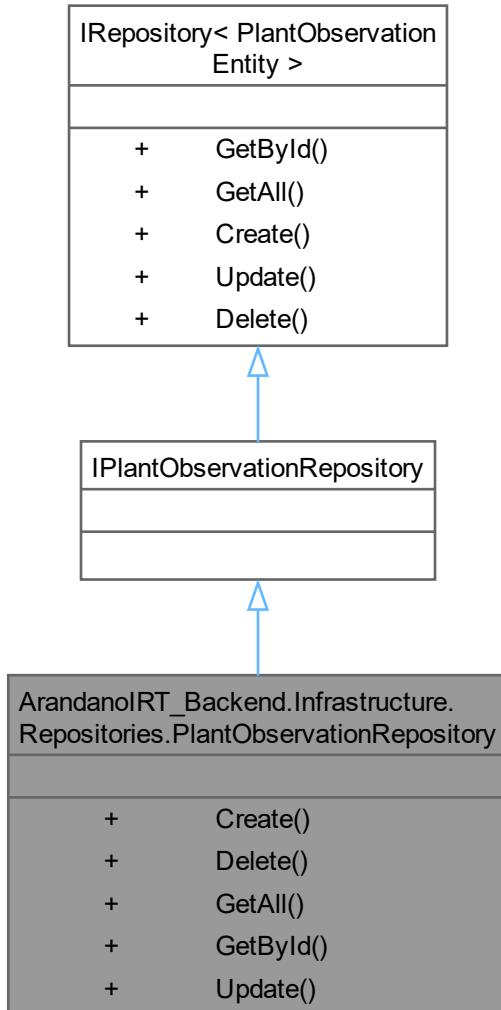
- Domain/Entities/[PlantObservationEntity.cs](#)

6.145 ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository:



Public Member Functions

- Task< [Result< PlantObservationEntity >](#) >> [Create](#) (`PlantObservationEntity` entity)
- Task< [Result< bool >](#) > [Delete](#) (int id)
- Task< [Result< IEnumerable< PlantObservationEntity >](#) >> [GetAll](#) ()
- Task< [Result< PlantObservationEntity >](#) >> [GetById](#) (int id)
- Task< [Result< bool >](#) > [Update](#) (`PlantObservationEntity` entity)

Public Member Functions inherited from
ArandanoIRT_Backend.Domain.IRepositories IRepository< PlantObservationEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.145.1 Member Function Documentation

6.145.1.1 Create()

```
Task< Result< PlantObservationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.Plant←
ObservationRepository.Create (
    PlantObservationEntity entity ) [inline]
```

6.145.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository.←
Delete (
    int id ) [inline]
```

6.145.1.3 GetAll()

```
Task< Result< IEnumerable< PlantObservationEntity > > > ArandanoIRT_Backend.Infrastructure.←
Repositories.PlantObservationRepository.GetAll ( ) [inline]
```

6.145.1.4 GetById()

```
Task< Result< PlantObservationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.Plant←
ObservationRepository.GetById (
    int id ) [inline]
```

6.145.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository.←
Update (
    PlantObservationEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/PlantObservationRepository.cs

6.146 ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory:

ArandanoIRT_Backend.Infrastructure. Data.Plantstatehistory	
+	Idplantdatahistory
+	Changedat
+	Changedby
+	Plantid
+	Statusid
+	ChangedbyNavigation
+	Plant
+	Status

Properties

- int **Idplantdatahistory** [get, set]
- DateTime **Changedat** [get, set]
- int? **Changedby** [get, set]
- int? **Plantid** [get, set]
- int? **Statusid** [get, set]
- virtual ? Person **ChangedbyNavigation** [get, set]
- virtual ? Plantdata **Plant** [get, set]
- virtual ? Status **Status** [get, set]

6.146.1 Property Documentation

6.146.1.1 Changedat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Changedat [get], [set]
```

6.146.1.2 Changedby

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Changedby [get], [set]
```

6.146.1.3 Changedbynavigation

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Changedbynavigation  
[get], [set]
```

6.146.1.4 Idplantdatahistory

```
int ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Idplantdatahistory [get], [set]
```

6.146.1.5 Plant

```
virtual ? Plantdata ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Plant [get], [set]
```

6.146.1.6 Plantid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Plantid [get], [set]
```

6.146.1.7 Status

```
virtual ? Status ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Status [get], [set]
```

6.146.1.8 Statusid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory.Statusid [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Plantstatehistory.cs](#)

6.147 ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity Class Reference

Represents a historical record documenting a change in a plant's status at a specific point in time, including who made the change.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity:

ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity
+ IdPlantStateHistory
+ ChangedAt
+ ChangedBy
+ PlantId
+ StatusId
+ PlantStateHistoryEntity()

Public Member Functions

- [PlantStateHistoryEntity](#) (int idPlantStateHistory, DateTime changedAt, int plantId, int? statusId, int? changedBy=null)

Initializes a new instance of the PlantStateHistoryEntity class.

Properties

- int [IdPlantStateHistory](#) [get]
Gets the unique identifier for this plant state history record.
- DateTime [ChangedAt](#) [get]
Gets the date and time (usually UTC) when the plant's status was changed.
- int? [ChangedBy](#) [get]
Gets the unique identifier of the user who made the status change.
- int [PlantId](#) [get]
Gets the unique identifier of the plant whose status was changed.
- int? [StatusId](#) [get]
Gets the identifier representing the status that was assigned to the plant at this point in time.

6.147.1 Detailed Description

Represents a historical record documenting a change in a plant's status at a specific point in time, including who made the change.

6.147.2 Constructor & Destructor Documentation

6.147.2.1 PlantStateHistoryEntity()

```
ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.PlantStateHistoryEntity (
    int idPlantStateHistory,
    DateTime changedAt,
    int plantId,
    int? statusId,
    int? changedBy = null ) [inline]
```

Initializes a new instance of the PlantStateHistoryEntity class.

Parameters

<i>idPlantStateHistory</i>	The unique ID for this history record (often generated by the database).
<i>changedAt</i>	The timestamp when the status change occurred.
<i>plantId</i>	The ID of the plant whose status changed.
<i>statusId</i>	The ID of the status assigned at this time (nullable).
<i>changedBy</i>	The ID of the user who made the change (optional, defaults to null).

6.147.3 Property Documentation

6.147.3.1 ChangedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.ChangedAt [get]
```

Gets the date and time (usually UTC) when the plant's status was changed.

6.147.3.2 ChangedBy

```
int? ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.ChangedBy [get]
```

Gets the unique identifier of the user who made the status change.

Nullable if changed by the system or unknown.

6.147.3.3 IdPlantStateHistory

```
int ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.IdPlantStateHistory [get]
```

Gets the unique identifier for this plant state history record.

6.147.3.4 PlantId

```
int ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.PlantId [get]
```

Gets the unique identifier of the plant whose status was changed.

6.147.3.5 StatusId

```
int? ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity.StatusId [get]
```

Gets the identifier representing the status that was assigned to the plant at this point in time.

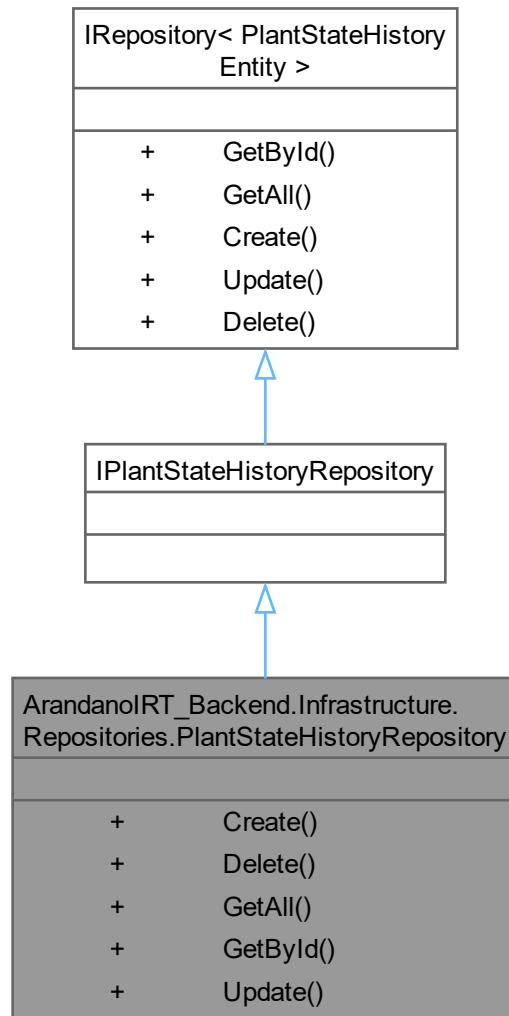
Nullable.

The documentation for this class was generated from the following file:

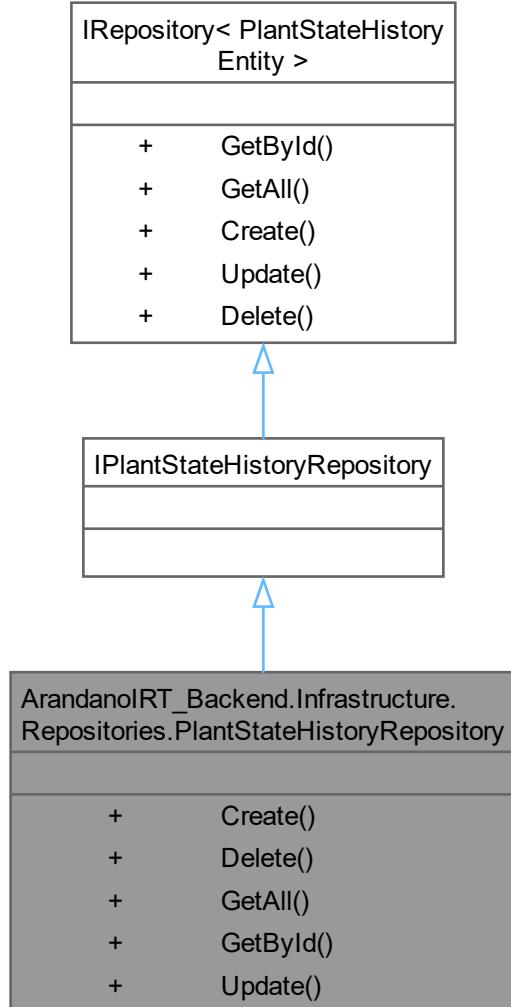
- Domain/Entities/[PlantStateHistoryEntity.cs](#)

6.148 ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository:



Public Member Functions

- Task< [Result< PlantStateHistoryEntity >](#) > > [Create](#) (`PlantStateHistoryEntity` entity)
- Task< [Result< bool >](#) > > [Delete](#) (int id)
- Task< [Result< IEnumerable< PlantStateHistoryEntity >](#) > > > [GetAll](#) ()
- Task< [Result< PlantStateHistoryEntity >](#) > > [GetById](#) (int id)
- Task< [Result< bool >](#) > > [Update](#) (`PlantStateHistoryEntity` entity)

Public Member Functions inherited from
ArandanoIRT_Backend.Domain.IRepositories IRepository< PlantStateHistoryEntity >

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.148.1 Member Function Documentation

6.148.1.1 Create()

```
Task< Result< PlantStateHistoryEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
PlantStateHistoryRepository.Create (
    PlantStateHistoryEntity entity ) [inline]
```

6.148.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository.←
Delete (
    int id ) [inline]
```

6.148.1.3 GetAll()

```
Task< Result< IEnumerable< PlantStateHistoryEntity > > > ArandanoIRT_Backend.Infrastructure.←
Repositories.PlantStateHistoryRepository.GetAll ( ) [inline]
```

6.148.1.4 GetById()

```
Task< Result< PlantStateHistoryEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.←
PlantStateHistoryRepository.GetById (
    int id ) [inline]
```

6.148.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository.←  
Update (   
    PlantStateHistoryEntity entity ) [inline]
```

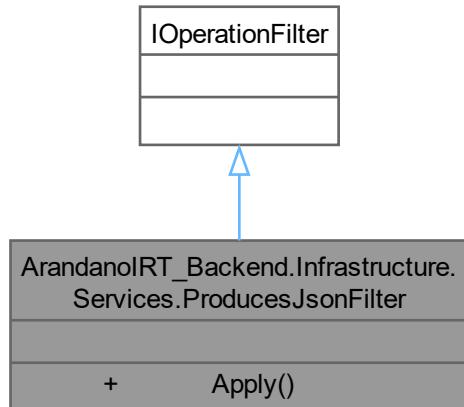
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/PlantStateHistoryRepository.cs

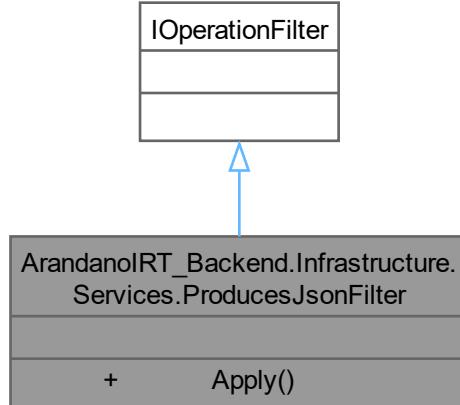
6.149 ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter Class Reference

An Swagger operation filter that attempts to determine the actual return type for 200 OK responses (handling Task and IActionResult with ProducesResponseType) and sets the corresponding JSON schema in the OpenAPI documentation.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter:



Public Member Functions

- void [Apply](#) (OpenApiOperation operation, OperationFilterContext context)
Applies the filter to the specified OpenAPI operation.

6.149.1 Detailed Description

An Swagger operation filter that attempts to determine the actual return type for 200 OK responses (handling Task and IActionResult with ProducesResponseType) and sets the corresponding JSON schema in the OpenAPI documentation.

6.149.2 Member Function Documentation

6.149.2.1 [Apply\(\)](#)

```
void ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter.Apply (
    OpenApiOperation operation,
    OperationFilterContext context ) [inline]
```

Applies the filter to the specified OpenAPI operation.

Parameters

<i>operation</i>	The OpenAPI operation to modify.
<i>context</i>	The context providing information about the operation, method, and schema generation.

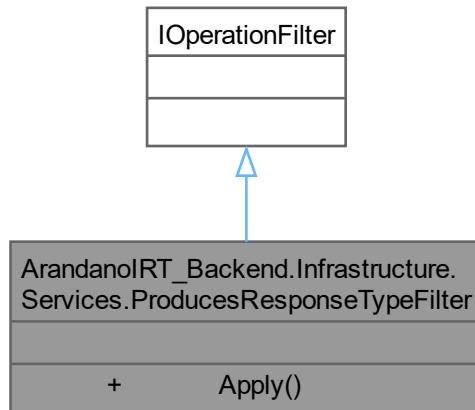
The documentation for this class was generated from the following file:

- Infrastructure/Services/[FiltersSwaggerService.cs](#)

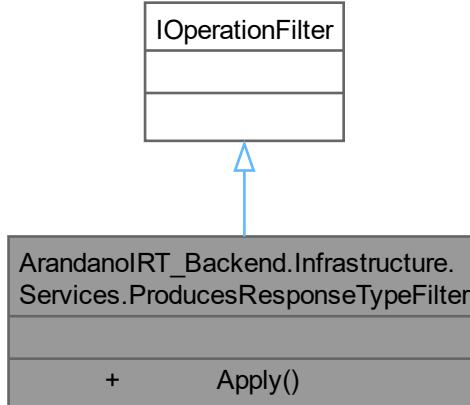
6.150 ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter Class Reference

An Swagger operation filter that adds standard HTTP error response types (400, 404, 429, 500) to every operation in the OpenAPI documentation.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter:



Public Member Functions

- void [Apply](#) (OpenApiOperation operation, OperationFilterContext context)
Applies the filter to the specified OpenAPI operation.

6.150.1 Detailed Description

An Swagger operation filter that adds standard HTTP error response types (400, 404, 429, 500) to every operation in the OpenAPI documentation.

6.150.2 Member Function Documentation

6.150.2.1 [Apply\(\)](#)

```
void ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter.Apply (
    OpenApiOperation operation,
    OperationFilterContext context ) [inline]
```

Applies the filter to the specified OpenAPI operation.

Parameters

<i>operation</i>	The OpenAPI operation to modify.
<i>context</i>	The context providing information about the operation.

The documentation for this class was generated from the following file:

- Infrastructure/Services/[FiltersSwaggerService.cs](#)

6.151 ArandanoIRT_Backend.Application.DTOs.Person.ProfileResponseDto Class Reference

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Person.ProfileResponseDto:

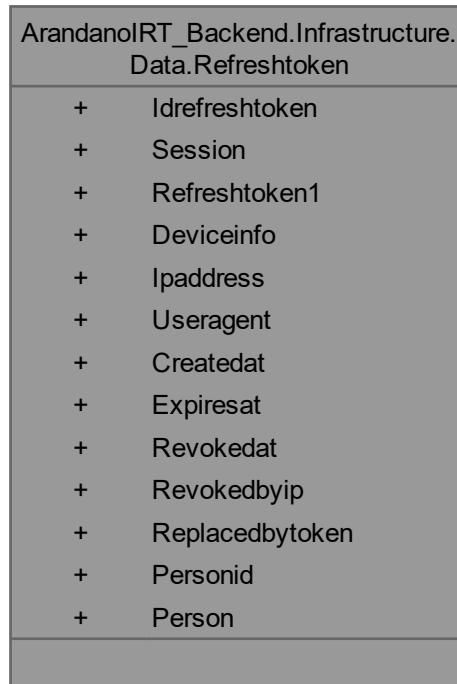


The documentation for this class was generated from the following file:

- Application/DTOs/Person/[ProfileResponseDto.cs](#)

6.152 ArandanoIRT_Backend.Infrastructure.Data.RefreshToken Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.RefreshToken:



Properties

- int **IdrefreshToken** [get, set]
- long **Session** [get, set]
- string **RefreshToken1** = null! [get, set]
- string **Deviceinfo** = null! [get, set]
- string **Ipadress** = null! [get, set]
- string **Useragent** = null! [get, set]
- DateTime **Createdat** [get, set]
- DateTime **Expiresat** [get, set]
- DateTime? **Revokedat** [get, set]
- string? **Revokedbyip** [get, set]
- string? **Replacedbytoken** [get, set]
- int? **Personid** [get, set]
- virtual ? Person **Person** [get, set]

6.152.1 Property Documentation

6.152.1.1 Createdat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Createdat [get], [set]
```

6.152.1.2 Deviceinfo

```
string ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Deviceinfo = null! [get], [set]
```

6.152.1.3 Expiresat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Expiresat [get], [set]
```

6.152.1.4 IdrefreshToken

```
int ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.IdrefreshToken [get], [set]
```

6.152.1.5 Ipaddress

```
string ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Ipaddress = null! [get], [set]
```

6.152.1.6 Person

```
virtual ? Person ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Person [get], [set]
```

6.152.1.7 Personid

```
int? ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Personid [get], [set]
```

6.152.1.8 RefreshToken1

```
string ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.RefreshToken1 = null! [get], [set]
```

6.152.1.9 Replacedbytoken

```
string? ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Replacedbytoken [get], [set]
```

6.152.1.10 Revokedat

```
DateTime? ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Revokedat [get], [set]
```

6.152.1.11 Revokedbyip

```
string? ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Revokedbyip [get], [set]
```

6.152.1.12 Session

```
long ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Session [get], [set]
```

6.152.1.13 Useragent

```
string ArandanoIRT_Backend.Infrastructure.Data.RefreshToken.Useragent = null! [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[RefreshToken.cs](#)

6.153 ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity Class Reference

Represents a refresh token used to obtain new access tokens, extending user sessions.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity:

ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity
<ul style="list-style-type: none"> + IdRefreshToken + Session + Token + DeviceInfo + IpAddress + UserAgent + CreatedAt + ExpiresAt + RevokedAt + RevokedByIp + ReplacedByToken + PersonId
<ul style="list-style-type: none"> + Revoke() + Replace() + RefreshTokenEntity()

Public Member Functions

- void **Revoke** (DateTime revokedAt, string? revokedByIp)

Marks the refresh token as revoked.
- void **Replace** (string replacedByToken)

Marks this refresh token as having been replaced by a new token, part of a rotation strategy.
- **RefreshTokenEntity** (int idRefreshToken, long session, string token, string deviceInfo, string ipAddress, string userAgent, DateTime createdAt, DateTime expiresAt, DateTime? revokedAt, string? revokedByIp, string? replacedByToken, int? personId)

Initializes a new instance of the RefreshTokenEntity class.

Properties

- int **IdRefreshToken** [get]

Gets the unique identifier for this refresh token record.
- long **Session** [get]

Gets an identifier grouping tokens belonging to the same logical session or user login instance.

- string **Token** [get]
Gets the unique, cryptographically secure refresh token string.
- string **DeviceInfo** [get]
Gets descriptive information about the device associated with this token at the time of issuance.
- string **IpAddress** [get]
Gets the IP address from which the token was originally requested.
- string **UserAgent** [get]
Gets the user agent string of the client from which the token was originally requested.
- DateTime **CreatedAt** [get]
Gets the date and time (usually UTC) when this refresh token was created.
- DateTime **ExpiresAt** [get]
Gets the date and time (usually UTC) when this refresh token expires and is no longer valid for obtaining new access tokens.
- DateTime? **RevokedAt** [get]
Gets the date and time (usually UTC) when this refresh token was revoked.
- string? **RevokedByIp** [get]
Gets the IP address that initiated the revocation action, if applicable.
- string? **ReplacedByToken** [get]
Gets the value (or reference) of the new refresh token that replaced this one, used in token rotation strategies.
- int? **PersonId** [get]
Gets the unique identifier of the person (user) to whom this refresh token belongs.

6.153.1 Detailed Description

Represents a refresh token used to obtain new access tokens, extending user sessions.

Contains information about the token itself, its lifecycle (creation, expiration, revocation), the associated user session, and the context (IP, device) in which it was issued.

6.153.2 Constructor & Destructor Documentation

6.153.2.1 RefreshTokenEntity()

```
ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.RefreshTokenEntity (
    int idRefreshToken,
    long session,
    string token,
    string deviceInfo,
    string ipAddress,
    string userAgent,
    DateTime createdAt,
    DateTime expiresAt,
    DateTime? revokedAt,
    string? revokedByIp,
    string? replacedByToken,
    int? personId ) [inline]
```

Initializes a new instance of the RefreshTokenEntity class.

Parameters

<i>idRefreshToken</i>	The unique ID for this token record (often generated by the database).
<i>session</i>	The session identifier associated with this token.
<i>token</i>	The refresh token string.
<i>deviceInfo</i>	Information about the device requesting the token.
<i>ipAddress</i>	The IP address of the client requesting the token.
<i>userAgent</i>	The user agent string of the client.
<i>createdAt</i>	The creation timestamp.
<i>expiresAt</i>	The expiration timestamp.
<i>revokedAt</i>	The revocation timestamp (nullable if active).
<i>revokedByIp</i>	The IP address causing revocation (nullable).
<i>replacedByToken</i>	The token that replaces this one (nullable).
<i>personId</i>	The ID of the user associated with this token (nullable).

Exceptions

<i>ArgumentException</i>	Thrown if <i>token</i> is null or whitespace (Spanish message: "El token es requerido."). Thrown if <i>deviceInfo</i> is null or whitespace (Spanish message: "La información del dispositivo es requerida."). Thrown if <i>ipAddress</i> is null or whitespace (Spanish message: "La dirección IP es requerida."). Thrown if <i>userAgent</i> is null or whitespace (Spanish message: "El user agent es requerido.").
--------------------------	--

6.153.3 Member Function Documentation**6.153.3.1 Replace()**

```
void ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.Replace (
    string replacedByToken) [inline]
```

Marks this refresh token as having been replaced by a new token, part of a rotation strategy.

Parameters

<i>replacedByToken</i>	The value (or reference) of the new token that replaces this one.
------------------------	---

6.153.3.2 Revoke()

```
void ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.Revoke (
    DateTime revokedAt,
    string? revokedByIp) [inline]
```

Marks the refresh token as revoked.

Parameters

<i>revokedAt</i>	The UTC timestamp when the token revocation occurred.
<i>revokedByIp</i>	The IP address associated with the revocation action (can be null).

6.153.4 Property Documentation**6.153.4.1 CreatedAt**

```
DateTime ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.CreatedAt [get]
```

Gets the date and time (usually UTC) when this refresh token was created.

6.153.4.2 DeviceInfo

```
string ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.DeviceInfo [get]
```

Gets descriptive information about the device associated with this token at the time of issuance.

6.153.4.3 ExpiresAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.ExpiresAt [get]
```

Gets the date and time (usually UTC) when this refresh token expires and is no longer valid for obtaining new access tokens.

6.153.4.4 IdRefreshToken

```
int ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.IdRefreshToken [get]
```

Gets the unique identifier for this refresh token record.

6.153.4.5 IpAddress

```
string ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.IpAddress [get]
```

Gets the IP address from which the token was originally requested.

6.153.4.6 PersonId

```
int? ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.PersonId [get]
```

Gets the unique identifier of the person (user) to whom this refresh token belongs.

Nullable if associated differently (e.g., device token).

6.153.4.7 ReplacedByToken

```
string? ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.ReplacedByToken [get]
```

Gets the value (or reference) of the new refresh token that replaced this one, used in token rotation strategies.

Nullable if not replaced.

6.153.4.8 RevokedAt

```
DateTime? ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.RevokedAt [get]
```

Gets the date and time (usually UTC) when this refresh token was revoked.

Nullable if the token is still active.

6.153.4.9 RevokedByIp

```
string? ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.RevokedByIp [get]
```

Gets the IP address that initiated the revocation action, if applicable.

Nullable if not revoked or IP not recorded.

6.153.4.10 Session

```
long ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.Session [get]
```

Gets an identifier grouping tokens belonging to the same logical session or user login instance.

6.153.4.11 Token

```
string ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.Token [get]
```

Gets the unique, cryptographically secure refresh token string.

6.153.4.12 UserAgent

```
string ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity.UserAgent [get]
```

Gets the user agent string of the client from which the token was originally requested.

The documentation for this class was generated from the following file:

- Domain/Entities/RefreshTokenEntity.cs

6.154 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests Class Reference

Contains unit tests for the RefreshTokenEntity class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests:

ArandanoIRT_Backend.Tests. Unit.Domain.Entities.RefreshToken EntityTests
+ Constructor_ShouldThrowArgument Exception_WhenTokenIsInvalid() + Constructor_ShouldThrowArgument Exception_WhenDeviceInfoIsInvalid() + Constructor_ShouldThrowArgument Exception_WhenIpAddressIsInvalid() + Constructor_ShouldThrowArgument Exception_WhenUserAgentIsInvalid() + Constructor_ShouldCreate Instance_WhenValidArgumentsProvided IncludingNulls() + Constructor_ShouldCreate Instance_WhenPersonIdIsNull() + Revoke_ShouldSetRevokedAt AndRevokedByIp() + Revoke_ShouldSetRevokedAt AndHandleNullRevokedByIp() + Replace_ShouldSetReplaced ByToken()

Public Member Functions

- void [Constructor_ShouldThrowArgumentException_WhenTokenIsInvalid](#) (string? invalidToken)
Verifies constructor throws ArgumentException if Token is null, empty, or whitespace.
- void [Constructor_ShouldThrowArgumentException_WhenDeviceInfoIsInvalid](#) (string? invalidDeviceInfo)
Verifies constructor throws ArgumentException if DeviceInfo is null, empty, or whitespace.
- void [Constructor_ShouldThrowArgumentException_WhenIpAddressIsInvalid](#) (string? invalidIpAddress)
Verifies constructor throws ArgumentException if IPAddress is null, empty, or whitespace.
- void [Constructor_ShouldThrowArgumentException_WhenUserAgentIsInvalid](#) (string? invalidUserAgent)
Verifies constructor throws ArgumentException if UserAgent is null, empty, or whitespace.
- void [Constructor_ShouldCreateInstance_WhenValidArgumentsProvidedIncludingNulls](#) ()
Verifies constructor successfully creates an instance and assigns properties correctly when all arguments are valid, including optional nulls for nullable properties.
- void [Constructor_ShouldCreateInstance_WhenPersonIdIsNull](#) ()
Verifies constructor successfully creates an instance when the optional PersonId is null.
- void [Revoke_ShouldSetRevokedAtAndRevokedByIp](#) ()
Verifies that the Revoke method correctly sets the RevokedAt and RevokedByIp properties.
- void [Revoke_ShouldSetRevokedAtAndHandleNullRevokedByIp](#) ()
Verifies that the Revoke method correctly sets RevokedAt and handles a null RevokedByIp argument.
- void [Replace_ShouldSetReplacedByToken](#) ()
Verifies that the Replace method correctly sets the ReplacedByToken property.

6.154.1 Detailed Description

Contains unit tests for the RefreshTokenEntity class.

Focuses on constructor validation and method logic (Revoke, Replace).

6.154.2 Member Function Documentation

6.154.2.1 Constructor_ShouldCreateInstance_WhenPersonIdIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
CreateInstance_WhenPersonIdIsNull ( ) [inline]
```

Verifies constructor successfully creates an instance when the optional PersonId is null.

6.154.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvidedIncludingNulls()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
CreateInstance_WhenValidArgumentsProvidedIncludingNulls ( ) [inline]
```

Verifies constructor successfully creates an instance and assigns properties correctly when all arguments are valid, including optional nulls for nullable properties.

6.154.2.3 Constructor_ShouldThrowArgumentException_WhenDeviceInfoIsInvalid()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
ThrowArgumentException_WhenDeviceInfoIsInvalid (
    string? invalidDeviceInfo ) [inline]
```

Verifies constructor throws ArgumentException if DeviceInfo is null, empty, or whitespace.

Parameters

<i>invalidDeviceInfo</i>	The invalid device info value.
--------------------------	--------------------------------

6.154.2.4 Constructor_ShouldThrowArgumentException_WhenIpAddressIsInvalid()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
ThrowArgumentException_WhenIpAddressIsInvalid (
    string? invalidIpAddress ) [inline]
```

Verifies constructor throws ArgumentException if ipAddress is null, empty, or whitespace.

Parameters

<i>invalidIpAddress</i>	The invalid IP address value.
-------------------------	-------------------------------

6.154.2.5 Constructor_ShouldThrowArgumentException_WhenTokenIsInvalid()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
ThrowArgumentException_WhenTokenIsInvalid (
    string? invalidToken ) [inline]
```

Verifies constructor throws ArgumentException if Token is null, empty, or whitespace.

Parameters

<i>invalidToken</i>	The invalid token value (null, empty, or whitespace).
---------------------	---

6.154.2.6 Constructor_ShouldThrowArgumentException_WhenUserAgentIsInvalid()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Constructor_Should←
ThrowArgumentException_WhenUserAgentIsInvalid (
    string? invalidUserAgent ) [inline]
```

Verifies constructor throws ArgumentException if UserAgent is null, empty, or whitespace.

Parameters

<i>invalidUserAgent</i>	The invalid user agent value.
-------------------------	-------------------------------

6.154.2.7 Replace_ShouldSetReplacedByToken()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Replace_ShouldSet←  
ReplacedByToken ( ) [inline]
```

Verifies that the Replace method correctly sets the ReplacedByToken property.

6.154.2.8 Revoke_ShouldSetRevokedAtAndHandleNullRevokedByIp()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Revoke_ShouldSet←  
RevokedAtAndHandleNullRevokedByIp ( ) [inline]
```

Verifies that the Revoke method correctly sets RevokedAt and handles a null RevokedByIp argument.

6.154.2.9 Revoke_ShouldSetRevokedAtAndRevokedByIp()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests.Revoke_ShouldSet←  
RevokedAtAndRevokedByIp ( ) [inline]
```

Verifies that the Revoke method correctly sets the RevokedAt and RevokedByIp properties.

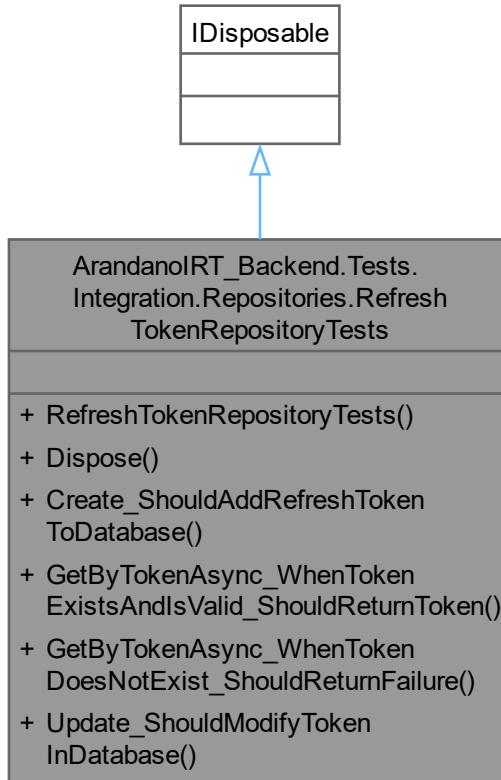
The documentation for this class was generated from the following file:

- Tests/Unit/Domain/Entities/RefreshTokenEntityTests.cs

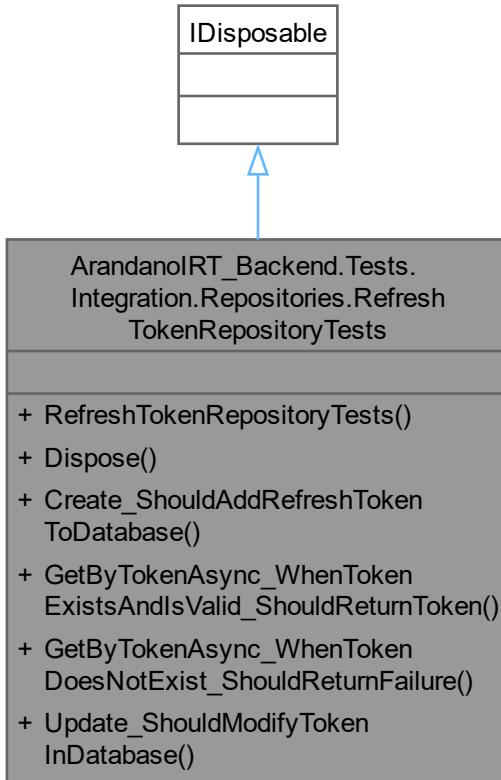
6.155 ArandanoIRT_Backend.Tests.Integration.Repositories.Refresh← TokenRepositoryTests Class Reference

Contains integration tests for the RefreshTokenRepository using an SQLite in-memory database.

Inheritance diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests:



Collaboration diagram for ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests:



Public Member Functions

- [RefreshTokenRepositoryTests \(\)](#)
Initializes a new instance of the RefreshTokenRepositoryTests class, setting up the in-memory database, mocks, and repository under test.
- [void Dispose \(\)](#)
Disposes the database context and connection used by the test instance.
- [async Task Create_ShouldAddRefreshTokenToDatabase \(\)](#)
- [async Task GetByTokenAsync_WhenTokenExistsAndIsValid_ShouldReturnToken \(\)](#)
- [async Task GetByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure \(\)](#)
- [async Task Update_ShouldModifyTokenInDatabase \(\)](#)

6.155.1 Detailed Description

Contains integration tests for the RefreshTokenRepository using an SQLite in-memory database.

6.155.2 Constructor & Destructor Documentation

6.155.2.1 RefreshTokenRepositoryTests()

```
ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.RefreshToken←  
RepositoryTests ( ) [inline]
```

Initializes a new instance of the RefreshTokenRepositoryTests class, setting up the in-memory database, mocks, and repository under test.

6.155.3 Member Function Documentation

6.155.3.1 Create_ShouldAddRefreshTokenToDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.Create←  
_ShouldAddRefreshTokenToDatabase ( ) [inline]
```

6.155.3.2 Dispose()

```
void ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.Dispose ( )  
[inline]
```

Disposes the database context and connection used by the test instance.

6.155.3.3 GetByTokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.GetBy←  
TokenAsync_WhenTokenDoesNotExist_ShouldReturnFailure ( ) [inline]
```

6.155.3.4 GetByTokenAsync_WhenTokenExistsAndIsValid_ShouldReturnToken()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.GetBy←  
TokenAsync_WhenTokenExistsAndIsValid_ShouldReturnToken ( ) [inline]
```

6.155.3.5 Update_ShouldModifyTokenInDatabase()

```
async Task ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests.Update←  
_ShouldModifyTokenInDatabase ( ) [inline]
```

The documentation for this class was generated from the following file:

- Tests/Integration/Repositories/[RefreshTokenRepositoryTests.cs](#)

6.156 ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto Class Reference

Represents the request data for registering a new administrator account along with associated crop information.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto:



Properties

- required [AdminInfoDto AdminInfo](#) [get, set]
Gets or sets the administrator's personal and credential information.
- required [CropInfoDto CropInfo](#) [get, set]
Gets or sets the information about the crop associated with this new administrator.
- string? [CaptchaToken](#) [get, set]
Gets or sets the optional CAPTCHA verification token provided during the registration process.

6.156.1 Detailed Description

Represents the request data for registering a new administrator account along with associated crop information.

6.156.2 Property Documentation

6.156.2.1 AdminInfo

required [AdminInfoDto](#) ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto.AdminInfo
[get], [set]

Gets or sets the administrator's personal and credential information.

See also

[AdminInfoDto](#)

6.156.2.2 CaptchaToken

```
string? ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto.CaptchaToken [get],  
[set]
```

Gets or sets the optional CAPTCHA verification token provided during the registration process.

6.156.2.3 CropInfo

```
required CropInfoDto ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto.CropInfo  
[get], [set]
```

Gets or sets the information about the crop associated with this new administrator.

See also

CropInfoDto

```
// Assuming CropInfoDto is defined in the referenced Objects namespace
```

The documentation for this class was generated from the following file:

- Application/DTOs/Auth/RegisterAdminRequestDto.cs

6.157 ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto Class Reference

Represents the request data for registering a new standard user account.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto:



Properties

- required [UserInfoDto](#) `UserInfo` [get, set]
Gets or sets the user's personal and credential information.
- string? `CaptchaToken` [get, set]
Gets or sets the optional CAPTCHA verification token provided during the registration process.

6.157.1 Detailed Description

Represents the request data for registering a new standard user account.

6.157.2 Property Documentation

6.157.2.1 `CaptchaToken`

```
string? ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto.CaptchaToken [get],  
[set]
```

Gets or sets the optional CAPTCHA verification token provided during the registration process.

6.157.2.2 `UserInfo`

```
required UserInfoDto ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto.UserInfo  
[get], [set]
```

Gets or sets the user's personal and credential information.

See also

[UserInfoDto](#)

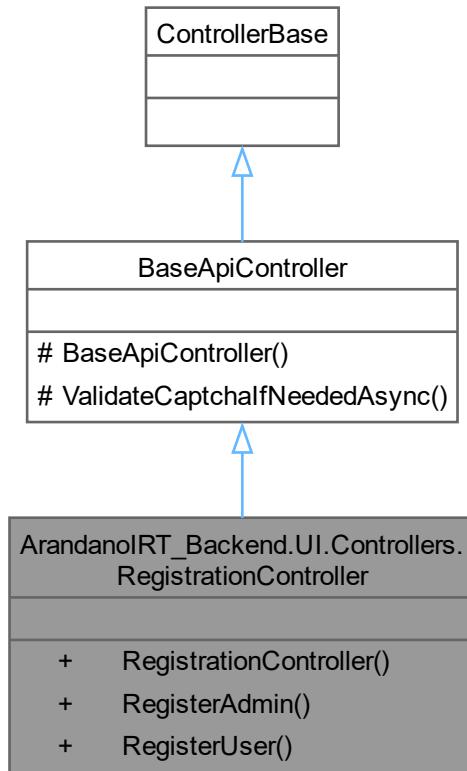
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[RegisterUserRequestDto.cs](#)

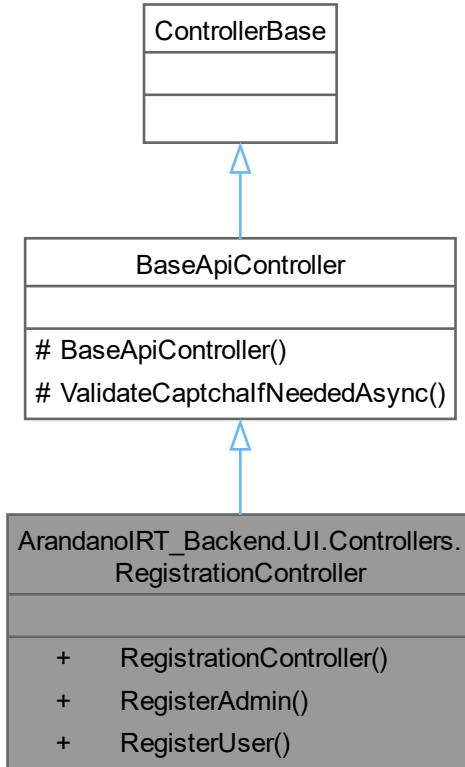
6.158 ArandanoIRT_Backend.UI.Controllers.RegistrationController Class Reference

Handles endpoints related to user and administrator registration.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.RegistrationController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.RegistrationController:



Public Member Functions

- `RegistrationController (IAuthRegistrationService registrationService, ICaptchaService captchaService, IWebHostEnvironment environment, ILogger<RegistrationController> logger)`
Initializes a new instance of the RegistrationController class.
- `async Task< IActionResult > RegisterAdmin ([FromBody] RegisterAdminRequestDto request)`
Registers a new administrator user along with their initial crop.
- `async Task< IActionResult > RegisterUser ([FromBody] RegisterUserRequestDto request)`
Registers a new regular user using an invitation code.

Additional Inherited Members

Protected Member Functions inherited from `ArandanoIRT_Backend.UI.Controllers.BaseApiController`

- `BaseApiController (ICaptchaService captchaService, IWebHostEnvironment environment)`

- Initializes a new instance of the BaseApiController class.*
- `async Task< IActionResult?> ValidateCaptchaIfNeededAsync (string? captchaToken, ILogger logger)`
Validates the CAPTCHA token if the application is not running in the Development environment.

6.158.1 Detailed Description

Handles endpoints related to user and administrator registration.

6.158.2 Constructor & Destructor Documentation

6.158.2.1 RegistrationController()

```
ArandanoIRT_Backend.UI.Controllers.RegistrationController.RegistrationController (
    IAuthRegistrationService registrationService,
    ICaptchaService captchaService,
    IWebHostEnvironment environment,
    ILogger< RegistrationController > logger ) [inline]
```

Initializes a new instance of the RegistrationController class.

Parameters

<code>registrationService</code>	The application service for registration logic.
<code>captchaService</code>	The service for CAPTCHA validation (passed to base).
<code>environment</code>	The web hosting environment (passed to base).
<code>logger</code>	The logger for this controller.

6.158.3 Member Function Documentation

6.158.3.1 RegisterAdmin()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.RegistrationController.RegisterAdmin (
    [FromBody] RegisterAdminRequestDto request ) [inline]
```

Registers a new administrator user along with their initial crop.

Parameters

<code>request</code>	Data for the administrator and crop to be registered.
----------------------	---

Returns

An IActionResult indicating success (204 No Content) or failure (400 Bad Request).

```
<response code="204">Admin and crop registration successful.</response> <response code="400">Invalid data  
(e.g., existing email, incomplete data, invalid CAPTCHA).</response>
```

6.158.3.2 RegisterUser()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.RegistrationController.Register←  
User ( [FromBody] RegisterUserRequestDto request ) [inline]
```

Registers a new regular user using an invitation code.

Parameters

<i>request</i>	Data for the user and the invitation code.
----------------	--

Returns

An IActionResult indicating success (204 No Content) or failure (400 Bad Request).

```
<response code="204">User registration successful.</response> <response code="400">Invalid data (e.g., existing  
email, invalid/expired code, invalid CAPTCHA).</response>
```

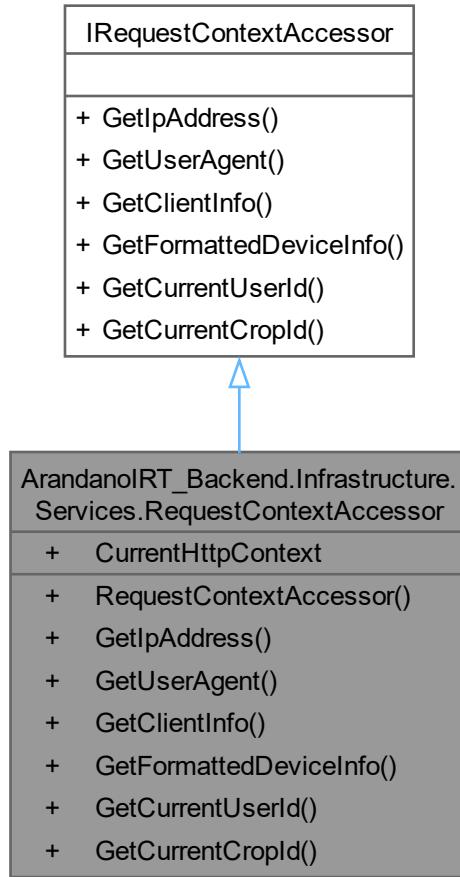
The documentation for this class was generated from the following file:

- UI/Controllers/[RegistrationController.cs](#)

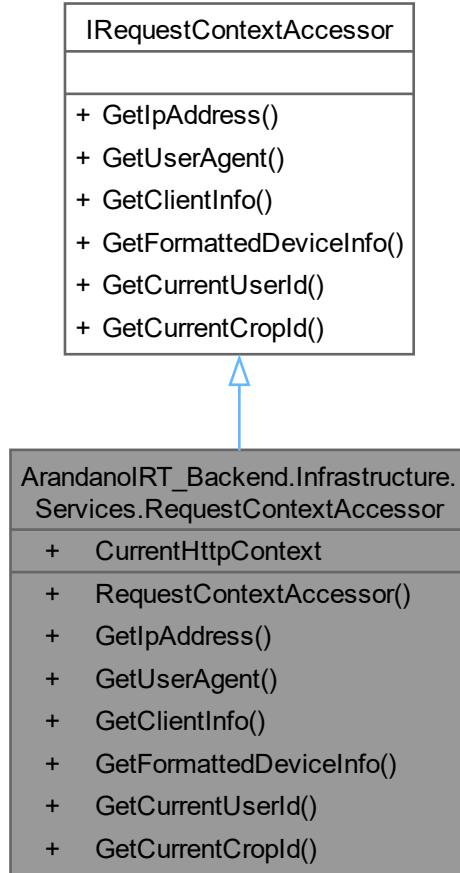
6.159 ArandanoIRT_Backend.Infrastructure.Services.RequestContext← Accessor Class Reference

Implements IRequestContextAccessor to provide access to common data from the current HTTP request context, such as IP address, User Agent details, and authenticated user information (ID, Crop ID).

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor:



Public Member Functions

- **RequestContextAccessor** (HttpContextAccessor httpContextAccessor, UserAgentParser userAgentParser)
Initializes a new instance of the RequestContextAccessor class.
- string **GetIpAddress** ()
Gets the IP address of the client making the current request.
Returns
The client's IP address as a string. May return an empty string or placeholder if unavailable.
- string **GetUserAgent** ()
Gets the User-Agent string sent by the client in the current request header.
Returns
The client's User-Agent string. May return an empty string if unavailable.
- ClientInfo **GetClientInfo** ()
Gets structured information about the client (e.g., browser, OS, device) typically parsed from the User-Agent string.

Returns

A [ClientInfo](#) object containing parsed client details.

- string [GetFormattedDeviceInfo \(\)](#)

Gets a formatted string summarizing the client's device and browser information.

Returns

A human-readable string describing the client's device and software.

- int? [GetCurrentUserId \(\)](#)

Gets the unique identifier of the currently authenticated user associated with the request.

Returns

The user's ID as an integer, or `null` if the user is not authenticated or the ID cannot be determined.

- int? [GetCurrentCropId \(\)](#)

Gets the unique identifier of the crop associated with the current request context (e.g., based on the authenticated user or other request parameters).

Returns

The crop's ID as an integer, or `null` if no specific crop is associated with the context or it's unavailable.

6.159.1 Detailed Description

Implements IRequestContextAccessor to provide access to common data from the current HTTP request context, such as IP address, User Agent details, and authenticated user information (ID, Crop ID).

Designed to be used within a request scope.

6.159.2 Constructor & Destructor Documentation

6.159.2.1 RequestContextAccessor()

```
ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.RequestContextAccessor (
    IHttpContextAccessor httpContextAccessor,
    IUserAgentParser userAgentParser ) [inline]
```

Initializes a new instance of the RequestContextAccessor class.

Parameters

<code>httpContextAccessor</code>	The accessor for the current HTTP context.
<code>userAgentParser</code>	The service for parsing User-Agent strings.

Exceptions

<code>ArgumentNullException</code>	Thrown if httpContextAccessor or userAgentParser is null.
------------------------------------	---

6.159.3 Member Function Documentation

6.159.3.1 GetClientInfo()

```
ClientInfo ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetClientInfo ( )  
[inline]
```

Gets structured information about the client (e.g., browser, OS, device) typically parsed from the User-Agent string.

Returns

A [ClientInfo](#) object containing parsed client details.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

6.159.3.2 GetCurrentCropId()

```
int? ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetCurrentCropId ( )  
[inline]
```

Gets the unique identifier of the crop associated with the current request context (e.g., based on the authenticated user or other request parameters).

Returns

The crop's ID as an integer, or `null` if no specific crop is associated with the context or it's unavailable.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

6.159.3.3 GetCurrentUserId()

```
int? ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetCurrentUserId ( )  
[inline]
```

Gets the unique identifier of the currently authenticated user associated with the request.

Returns

The user's ID as an integer, or `null` if the user is not authenticated or the ID cannot be determined.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

6.159.3.4 GetFormattedDeviceInfo()

```
string ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetFormattedDeviceInfo ( ) [inline]
```

Gets a formatted string summarizing the client's device and browser information.

Returns

A human-readable string describing the client's device and software.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

6.159.3.5 GetIpAddress()

```
string ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetIpAddress ( ) [inline]
```

Gets the IP address of the client making the current request.

Returns

The client's IP address as a string. May return an empty string or placeholder if unavailable.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

6.159.3.6 GetUserAgent()

```
string ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor.GetUserAgent ( ) [inline]
```

Gets the User-Agent string sent by the client in the current request header.

Returns

The client's User-Agent string. May return an empty string if unavailable.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#).

The documentation for this class was generated from the following file:

- Infrastructure/Services/[RequestContextAccessor.cs](#)

6.160 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests Class Reference

Contains unit tests for the RequestContextAccessor class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests:

ArandanoIRT_Backend.Tests. Unit.Infrastructure.Services.Request ContextAccessorTests
<ul style="list-style-type: none">+ RequestContextAccessorTests()+ GetIpAddress_WhenNoXForwardedFor_ShouldReturnRemotelpAddress()+ GetIpAddress_WhenXForwardedForExists_ShouldReturnFirstIpFromHeader()+ GetIpAddress_WhenXForwardedForHasMultipleIPs_ShouldReturnFirstIp()+ GetIpAddress_WhenHttpContextIsNull_ShouldReturnUnknown()+ GetUserAgent_WhenHeaderExists_ShouldReturnHeaderValue()+ GetUserAgent_WhenHeaderMissing_ShouldReturnUnknown()+ GetUserAgent_WhenHttpContextIsNull_ShouldReturnUnknown()+ GetClientInfo_ShouldCallUserAgentParserAndReturnResult()+ GetFormattedDeviceInfo_ShouldCallGetClientInfoAndFormat() and 11 more...

Public Member Functions

- [RequestContextAccessorTests \(\)](#)

Initializes a new instance of the RequestContextAccessorTests class.

- [void GetIpAddress_WhenNoXForwardedFor_ShouldReturnRemotelpAddress \(\)](#)
- [void GetIpAddress_WhenXForwardedForExists_ShouldReturnFirstIpFromHeader \(\)](#)

- void GetIpAddress_WhenXForwardedForHasMultipleIPs_ShouldReturnFirstIp ()
- void GetIpAddress_WhenHttpContextIsNull_ShouldReturnUnknown ()
- void GetUserAgent_WhenHeaderExists_ShouldReturnHeaderValue ()
- void GetUserAgent_WhenHeaderMissing_ShouldReturnUnknown ()
- void GetUserAgent_WhenHttpContextIsNull_ShouldReturnUnknown ()
- void GetClientInfo_ShouldCallUserAgentParserAndReturnResult ()
- void GetFormattedDeviceInfo_ShouldCallGetClientInfoAndFormat ()
- void GetClientInfo_WhenCalledMultipleTimes_ShouldReturnCachedResult ()
- void GetCurrentUserId_WhenUserIsAuthenticatedWithNameIdentifier_ShouldReturnUserId ()
- void GetCurrentUserId_WhenNameIdentifierClaimIsMissing_ShouldReturnNull ()
- void GetCurrentUserId_WhenNameIdentifierClaimIsNotInt_ShouldReturnNull ()
- void GetCurrentUserId_WhenUserIsNotAuthenticated_ShouldReturnNull ()
- void GetCurrentUserId_WhenHttpContextIsNull_ShouldReturnNull ()
- void GetCurrentCropId_WhenCropIdClaimExists_ShouldReturnCropId ()
- void GetCurrentCropId_WhenCropIdClaimIsMissing_ShouldReturnNull ()
- void GetCurrentCropId_WhenCropIdClaimIsNotInt_ShouldReturnNull ()
- void GetCurrentCropId_WhenUserIsNotAuthenticated_ShouldReturnNull ()
- void GetCurrentCropId_WhenHttpContextIsNull_ShouldReturnNull ()

6.160.1 Detailed Description

Contains unit tests for the RequestContextAccessor class.

6.160.2 Constructor & Destructor Documentation

6.160.2.1 RequestContextAccessorTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.RequestContextAccessorTests ( ) [inline]
```

Initializes a new instance of the RequestContextAccessorTests class.

Sets up mocks, creates a DefaultHttpContext, and instantiates the service under test.

6.160.3 Member Function Documentation

6.160.3.1 GetClientInfo_ShouldCallUserAgentParserAndReturnResult()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetClientInfo_ShouldCallUserAgentParserAndReturnResult ( ) [inline]
```

6.160.3.2 GetClientInfo_WhenCalledMultipleTimes_ShouldReturnCachedResult()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetClientInfo_WhenCalledMultipleTimes_ShouldReturnCachedResult ( ) [inline]
```

6.160.3.3 GetCurrentCropId_WhenCropIdClaimExists_ShouldReturnCropId()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentCropId_WhenCropIdClaimExists_ShouldReturnCropId () [inline]
```

6.160.3.4 GetCurrentCropId_WhenCropIdClaimIsMissing_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentCropId_WhenCropIdClaimIsMissing_ShouldReturnNull () [inline]
```

6.160.3.5 GetCurrentCropId_WhenCropIdClaimIsNotInt_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentCropId_WhenCropIdClaimIsNotInt_ShouldReturnNull () [inline]
```

6.160.3.6 GetCurrentCropId_WhenHttpContextIsNull_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentCropId_WhenHttpContextIsNull_ShouldReturnNull () [inline]
```

6.160.3.7 GetCurrentCropId_WhenUserIsNotAuthenticated_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentCropId_WhenUserIsNotAuthenticated_ShouldReturnNull () [inline]
```

6.160.3.8 GetCurrentUserId_WhenHttpContextIsNull_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentUser_Id_WhenHttpContextIsNull_ShouldReturnNull () [inline]
```

6.160.3.9 GetCurrentUserId_WhenNameIdentifierClaimIsMissing_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentUser_Id_WhenNameIdentifierClaimIsMissing_ShouldReturnNull () [inline]
```

6.160.3.10 GetCurrentUserId_WhenNameIdentifierClaimIsNotInt_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<-
CurrentUser_Id_WhenNameIdentifierClaimIsNotInt_ShouldReturnNull () [inline]
```

6.160.3.11 GetCurrentUserId_WhenUserIsAuthenticatedWithNamelIdentifier_ShouldReturnUserId()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<→
CurrentUserId_WhenUserIsAuthenticatedWithNameIdentifier_ShouldReturnUserId () [inline]
```

6.160.3.12 GetCurrentUserId_WhenUserIsNotAuthenticated_ShouldReturnNull()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<→
CurrentUserId_WhenUserIsNotAuthenticated_ShouldReturnNull () [inline]
```

6.160.3.13 GetFormattedDeviceInfo_ShouldCallGetClientInfoAndFormat()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.Get<→
FormattedDeviceInfo_ShouldCallGetClientInfoAndFormat () [inline]
```

6.160.3.14 GetIpAddress_WhenHttpContextIsNull_ShouldReturnUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetIp<→
Address_WhenHttpContextIsNull_ShouldReturnUnknown () [inline]
```

6.160.3.15 GetIpAddress_WhenNoXForwardedFor_ShouldReturnRemoteIpAddress()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetIp<→
Address_WhenNoXForwardedFor_ShouldReturnRemoteIpAddress () [inline]
```

6.160.3.16 GetIpAddress_WhenXForwardedForExists_ShouldReturnFirstIpFromHeader()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetIp<→
Address_WhenXForwardedForExists_ShouldReturnFirstIpFromHeader () [inline]
```

6.160.3.17 GetIpAddress_WhenXForwardedForHasMultipleIPs_ShouldReturnFirstIp()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetIp<→
Address_WhenXForwardedForHasMultipleIPs_ShouldReturnFirstIp () [inline]
```

6.160.3.18 GetUserAgent_WhenHeaderExists_ShouldReturnHeaderValue()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetUser<→
Agent_WhenHeaderExists_ShouldReturnHeaderValue () [inline]
```

6.160.3.19 GetUserAgent_WhenHeaderMissing_ShouldReturnUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetUser←  
Agent_WhenHeaderMissing_ShouldReturnUnknown ( ) [inline]
```

6.160.3.20 GetUserAgent_WhenHttpContextIsNull_ShouldReturnUnknown()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests.GetUser←  
Agent_WhenHttpContextIsNull_ShouldReturnUnknown ( ) [inline]
```

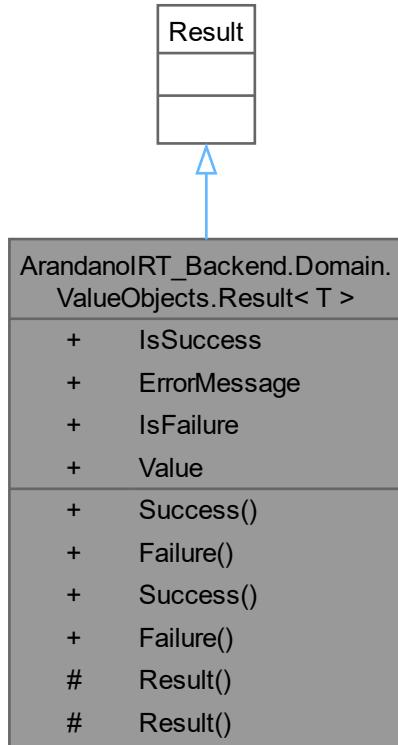
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[RequestContextAccessorTests.cs](#)

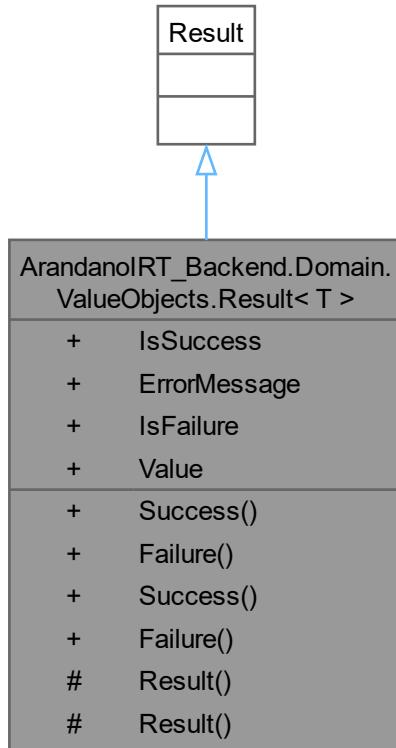
6.161 ArandanoIRT_Backend.Domain.ValueObjects.Result< T > Class Template Reference

Represents the outcome of an operation, indicating either success or failure.

Inheritance diagram for ArandanoIRT_Backend.Domain.ValueObjects.Result< T >:



Collaboration diagram for ArandanoIRT_Backend.Domain.ValueObjects.Result< T >:



Static Public Member Functions

- static [Result Success \(\)](#)
Creates a successful Result instance with no error message.
- static [Result Failure \(string errorMessage\)](#)
Creates a failed Result instance with the specified error message.
- static [Result< T > Success \(T value\)](#)
Creates a successful Result< T > instance with the specified value.
- static new [Result< T > Failure \(string errorMessage\)](#)
Creates a failed Result< T > instance with the specified error message.

Protected Member Functions

- [Result \(bool isSuccess, string errorMessage\)](#)
Initializes a new instance of the Result class.
- [Result \(T value, bool isSuccess, string errorMessage\)](#)
Initializes a new instance of the Result< T > class.

Properties

- bool `IsSuccess` [get]
Gets a value indicating whether the operation was successful.
- string `ErrorMessage` [get]
Gets the error message associated with a failed operation.
- bool `IsFailure` [get]
Gets a value indicating whether the operation failed.
- T `Value` [get]
Gets the value returned by the successful operation.

6.161.1 Detailed Description

Represents the outcome of an operation, indicating either success or failure.

Represents the outcome of an operation that returns a value upon success.

Contains an error message if the operation failed.

Inherits from Result and includes a Value property for the successful result.

Template Parameters

<code>T</code>	The type of the value returned upon success.
----------------	--

6.161.2 Constructor & Destructor Documentation

6.161.2.1 `Result()` [1/2]

```
ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Result (
    bool isSuccess,
    string errorMessage ) [inline], [protected]
```

Initializes a new instance of the Result class.

Protected constructor enforces consistency between success status and error message presence.

Parameters

<code>isSuccess</code>	Indicates if the operation was successful.
<code>errorMessage</code>	The error message if the operation failed; otherwise, string.Empty.

Exceptions

<i>InvalidOperationException</i>	Thrown if <i>isSuccess</i> is true but <i>errorMessage</i> is not null or empty, or if <i>isSuccess</i> is false but <i>errorMessage</i> is null or empty.
----------------------------------	--

6.161.2.2 Result() [2/2]

```
ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Result (
    T value,
    bool isSuccess,
    string errorMessage ) [inline], [protected]
```

Initializes a new instance of the *Result*<*T*> class.

Protected constructor ensures consistency.

Parameters

<i>value</i>	The value returned by the operation (meaningful only if <i>isSuccess</i> is true).
<i>isSuccess</i>	Indicates if the operation was successful.
<i>errorMessage</i>	The error message if the operation failed; otherwise, <code>string.Empty</code> .

6.161.3 Member Function Documentation**6.161.3.1 Failure() [1/2]**

```
static Result ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Failure (
    string errorMessage ) [static]
```

Creates a failed *Result* instance with the specified error message.

Parameters

<i>errorMessage</i>	The message describing the failure.
---------------------	-------------------------------------

Returns

A new failed *Result* with the given error message.

Exceptions

<i>ArgumentNullException</i>	Thrown if <i>errorMessage</i> is null (or becomes empty after check in constructor).
------------------------------	--

6.161.3.2 Failure() [2/2]

```
static new Result< T > ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Failure (
    string errorMessage ) [static]
```

Creates a failed Result<T> instance with the specified error message.

Hides the base Result.Failure method. The Value property will be initialized to the default value of type *T*.

Parameters

<i>errorMessage</i>	The message describing the failure.
---------------------	-------------------------------------

Returns

A new failed Result<T> with the given error message and default value.

Exceptions

<i>ArgumentNullException</i>	Thrown if <i>errorMessage</i> is null (or becomes empty after check in base constructor).
------------------------------	---

6.161.3.3 Success() [1/2]

```
static Result ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Success ( ) [static]
```

Creates a successful Result instance with no error message.

Returns

A new successful Result.

6.161.3.4 Success() [2/2]

```
static Result< T > ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Success (
    T value ) [static]
```

Creates a successful Result<T> instance with the specified value.

Parameters

<i>value</i>	The value returned by the successful operation.
--------------	---

Returns

A new successful Result<T> containing the value.

6.161.4 Property Documentation

6.161.4.1 ErrorMessage

```
string ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.ErrorMessage [get]
```

Gets the error message associated with a failed operation.

Is string.Empty if the operation was successful.

6.161.4.2 IsFailure

```
bool ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.IsFailure [get]
```

Gets a value indicating whether the operation failed.

Returns the opposite of IsSuccess.

6.161.4.3 IsSuccess

```
bool ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.IsSuccess [get]
```

Gets a value indicating whether the operation was successful.

6.161.4.4 Value

```
T ArandanoIRT_Backend.Domain.ValueObjects.Result< T >.Value [get]
```

Gets the value returned by the successful operation.

Accessing this property on a failed result may yield the default value for type *T*. Check Result.IsSuccess before accessing.

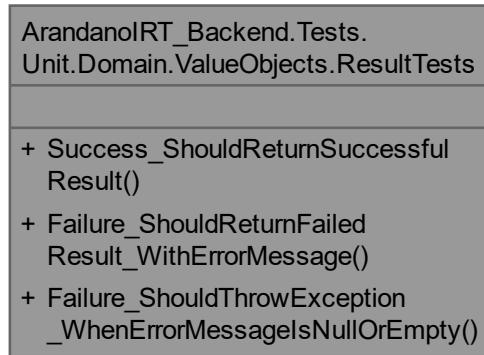
The documentation for this class was generated from the following file:

- Domain/ValueObjects/[ResultPattern.cs](#)

6.162 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests Class Reference

Contains unit tests for the non-generic Result class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests:



Public Member Functions

- void [Success_ShouldReturnSuccessfulResult \(\)](#)
Verifies that Result.Success() creates a successful Result instance.
- void [Failure_ShouldReturnFailedResult_WithErrorMessage \(\)](#)
Verifies that Result.Failure() creates a failed Result instance with the correct error message.
- void [Failure_ShouldThrowException_WhenErrorMessageIsNullOrEmpty \(string? invalidErrorMessage\)](#)
Verifies that Result.Failure() throws InvalidOperationException if the error message is null or empty.

6.162.1 Detailed Description

Contains unit tests for the non-generic Result class.

6.162.2 Member Function Documentation

6.162.2.1 Failure_ShouldReturnFailedResult_WithErrorMessage()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests.Failure_ShouldReturnFailedResult_WithErrorMessage ( ) [inline]
```

Verifies that Result.Failure() creates a failed Result instance with the correct error message.

6.162.2.2 Failure_ShouldThrowException_WhenErrorMessageIsNullOrEmpty()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests.Failure_ShouldThrowException←
_WhenErrorMessageIsNullOrEmpty (
    string? invalidErrorMessage ) [inline]
```

Verifies that Result.Failure() throws InvalidOperationException if the error message is null or empty.

The factory method prevents creating a Failure state without a meaningful error.

Parameters

<i>invalidErrorMessage</i>	The invalid error message (null or empty).
----------------------------	--

6.162.2.3 Success_ShouldReturnSuccessfulResult()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests.Success_ShouldReturnSuccessful←
Result ( ) [inline]
```

Verifies that Result.Success() creates a successful Result instance.

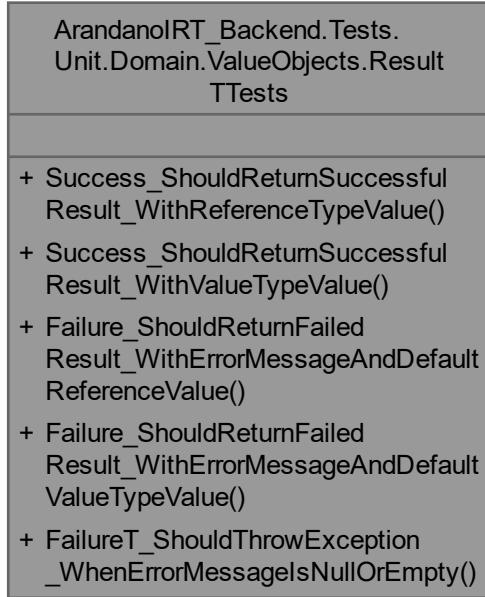
The documentation for this class was generated from the following file:

- Tests/Unit/Domain/ValueObjects/[ResultTests.cs](#)

6.163 ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.Result← TTests Class Reference

Contains unit tests for the generic Result<T> class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests:



Public Member Functions

- void [Success_ShouldReturnSuccessfulResult_WithReferenceTypeValue \(\)](#)
Verifies that Result{T}.Success(value) creates a successful Result instance with the correct value for a reference type.
- void [Success_ShouldReturnSuccessfulResult_WithValueTypeValue \(\)](#)
Verifies that Result{T}.Success(value) creates a successful Result instance with the correct value for a value type (int).
- void [Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultReferenceValue \(\)](#)
Verifies that Result{T}.Failure(errorMessage) creates a failed Result instance with the correct error message and the default value (null) for a reference type.
- void [Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultValueTypeValue \(\)](#)
Verifies that Result{T}.Failure(errorMessage) creates a failed Result instance with the correct error message and the default value (0) for a value type (int).
- void [FailureT_ShouldThrowException_WhenErrorMessageIsNullOrEmpty \(string? invalidErrorMessage\)](#)
Verifies that Result{T}.Failure() throws InvalidOperationException if the error message is null or empty.

6.163.1 Detailed Description

Contains unit tests for the generic Result<T> class.

6.163.2 Member Function Documentation

6.163.2.1 Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultReferenceValue()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests.Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultReferenceValue ( ) [inline]
```

Verifies that Result{T}.Failure(errorMessage) creates a failed Result instance with the correct error message and the default value (null) for a reference type.

6.163.2.2 Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultValueTypeValue()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests.Failure_ShouldReturnFailedResult_WithErrorMessageAndDefaultValueTypeValue ( ) [inline]
```

Verifies that Result{T}.Failure(errorMessage) creates a failed Result instance with the correct error message and the default value (0) for a value type (int).

6.163.2.3 FailureT_ShouldThrowException_WhenErrorMessageIsNullOrEmpty()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests.FailureT_ShouldThrowException_WhenErrorMessageIsNullOrEmpty (
    string? invalidErrorMessage ) [inline]
```

Verifies that Result{T}.Failure() throws InvalidOperationException if the error message is null or empty.

This check happens in the base class logic invoked by the generic factory method.

Parameters

<i>invalidErrorMessage</i>	The invalid error message (null or empty).
----------------------------	--

6.163.2.4 Success_ShouldReturnSuccessfulResult_WithReferenceTypeValue()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests.Success_ShouldReturnSuccessfulResult_WithReferenceTypeValue ( ) [inline]
```

Verifies that Result{T}.Success(value) creates a successful Result instance with the correct value for a reference type.

6.163.2.5 Success_ShouldReturnSuccessfulResult_WithValueTypeValue()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests.Success_ShouldReturnSuccessfulResult_WithValueTypeValue ( ) [inline]
```

Verifies that Result{T}.Success(value) creates a successful Result instance with the correct value for a value type (int).

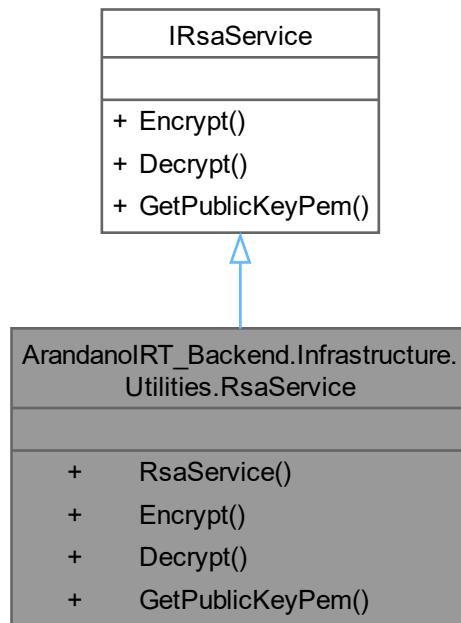
The documentation for this class was generated from the following file:

- Tests/Unit/Domain/ValueObjects/[ResultTests.cs](#)

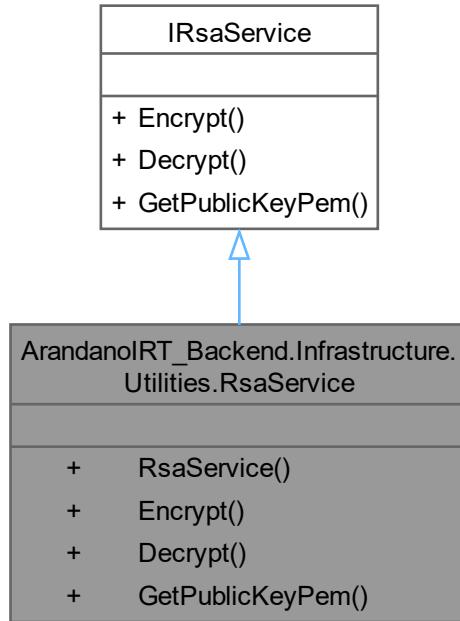
6.164 ArandanoIRT_Backend.Infrastructure.Utilities.RsaService Class Reference

Implements the IRsaService interface, providing functionalities for RSA encryption (using public key) and decryption (using private key).

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Utilities.RsaService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.RsaService:



Public Member Functions

- **RsaService** (ILogger< RsaService > logger)
Initializes a new instance of the RsaService class.
- string **Encrypt** (string plainText)
Encrypts data using the configured public key.

Parameters

plainText	<i>The data to encrypt.</i>
-----------	-----------------------------

Returns

The Base64 encoded encrypted string.

- string **Decrypt** (string encryptedText)
Decrypts data using the configured private key.

Parameters

encryptedText	<i>The Base64 encoded encrypted string.</i>
---------------	---

Returns

The decrypted plain text.

- string [GetPublicKeyPem \(\)](#)

Gets the public key in PEM format.

Returns

The public key string.

6.164.1 Detailed Description

Implements the `IRsaService` interface, providing functionalities for RSA encryption (using public key) and decryption (using private key).

Manages an internally generated RSA key pair.

6.164.2 Constructor & Destructor Documentation

6.164.2.1 `RsaService()`

```
ArandanoIRT_Backend.Infrastructure.Utilities.RsaService.RsaService (
    ILogger< RsaService > logger ) [inline]
```

Initializes a new instance of the `RsaService` class.

Creates an internal RSAUtility which generates a new RSA key pair upon instantiation.

Exceptions

<code>InvalidOperationException</code>	Thrown if RSA key pair generation fails during initialization.
--	--

6.164.3 Member Function Documentation

6.164.3.1 `Decrypt()`

```
string ArandanoIRT_Backend.Infrastructure.Utilities.RsaService.Decrypt (
    string encryptedText ) [inline]
```

Decrypts data using the configured private key.

Parameters

<code>encryptedText</code>	The Base64 encoded encrypted string.
----------------------------	--------------------------------------

Returns

The decrypted plain text.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService](#).

6.164.3.2 Encrypt()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.RsaService.Encrypt (
    string plainText ) [inline]
```

Encrypts data using the configured public key.

Parameters

<i>plainText</i>	The data to encrypt.
------------------	----------------------

Returns

The Base64 encoded encrypted string.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService](#).

6.164.3.3 GetPublicKeyPem()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.RsaService.GetPublicKeyPem ( ) [inline]
```

Gets the public key in PEM format.

Returns

The public key string.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService](#).

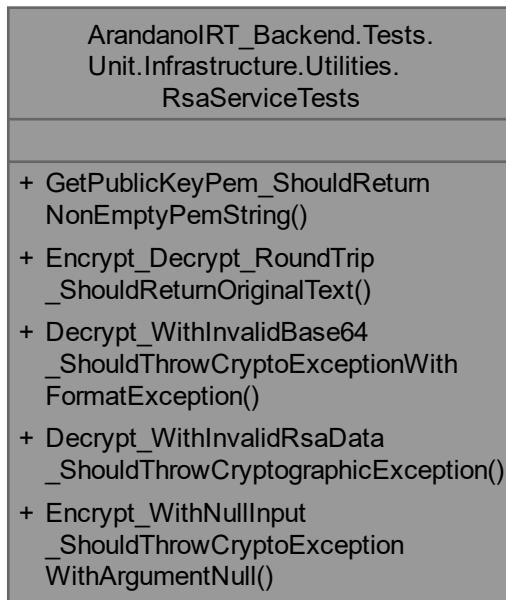
The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[RsaService.cs](#)

6.165 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests Class Reference

Contains unit tests for the RsaService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests:



Public Member Functions

- void [GetPublicKeyPem_ShouldReturnNonEmptyPemString \(\)](#)
Tests that GetPublicKeyPem returns a non-empty string formatted as a standard PEM public key.
- void [Encrypt_Decrypt_RoundTrip_ShouldReturnOriginalText \(string plainText\)](#)
Tests that text encrypted using the Encrypt method can be successfully decrypted back to the original text using the Decrypt method (round trip test).
- void [Decrypt_WithInvalidBase64_ShouldThrowCryptoExceptionWithFormatException \(\)](#)
Tests that Decrypt throws a CryptoException wrapping a FormatException when given input that is not valid Base64.
- void [Decrypt_WithInvalidRsaData_ShouldThrowCryptographicException \(\)](#)
Tests that Decrypt throws a CryptographicException when given data that is a valid Base64 string but does not represent valid RSA encrypted data (e.g., wrong format, padding issues, wrong key).
- void [Encrypt_WithNullInput_ShouldThrowCryptoExceptionWithArgumentNullException \(\)](#)
Tests that Encrypt throws a CryptographicException wrapping an ArgumentNullException when the input string is null.

6.165.1 Detailed Description

Contains unit tests for the RsaService class.

6.165.2 Member Function Documentation

6.165.2.1 Decrypt_WithInvalidBase64_ShouldThrowCryptoExceptionWithFormatException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests.Decrypt_WithInvalidBase64_ShouldThrowCryptoExceptionWithFormatException ( ) [inline]
```

Tests that Decrypt throws a CryptographicException wrapping a FormatException when given input that is not valid Base64.

6.165.2.2 Decrypt_WithInvalidRsaData_ShouldThrowCryptographicException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests.Decrypt_WithInvalidRsaData_ShouldThrowCryptographicException ( ) [inline]
```

Tests that Decrypt throws a CryptographicException when given data that is a valid Base64 string but does not represent valid RSA encrypted data (e.g., wrong format, padding issues, wrong key).

6.165.2.3 Encrypt_Decrypt_RoundTrip_ShouldReturnOriginalText()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests.Encrypt_Decrypt_RoundTrip_ShouldReturnOriginalText ( string plainText ) [inline]
```

Tests that text encrypted using the Encrypt method can be successfully decrypted back to the original text using the Decrypt method (round trip test).

Parameters

<code>plainText</code>	The plain text to encrypt and decrypt.
------------------------	--

6.165.2.4 Encrypt_WithNullInput_ShouldThrowCryptoExceptionWithArgumentNullException()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests.Encrypt_WithNullInput_ShouldThrowCryptoExceptionWithArgumentNullException ( ) [inline]
```

Tests that Encrypt throws a CryptographicException wrapping an ArgumentNullException when the input string is null.

6.165.2.5 GetPublicKeyPem_ShouldReturnNonEmptyPemString()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests.GetPublicKeyPem_←  
ShouldReturnNonEmptyPemString () [inline]
```

Tests that GetPublicKeyPem returns a non-empty string formatted as a standard PEM public key.

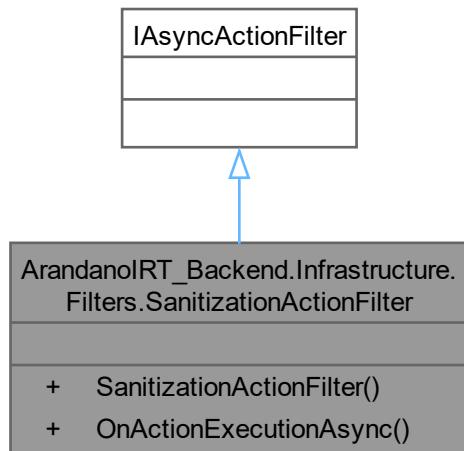
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Utilities/[RsaServiceTests.cs](#)

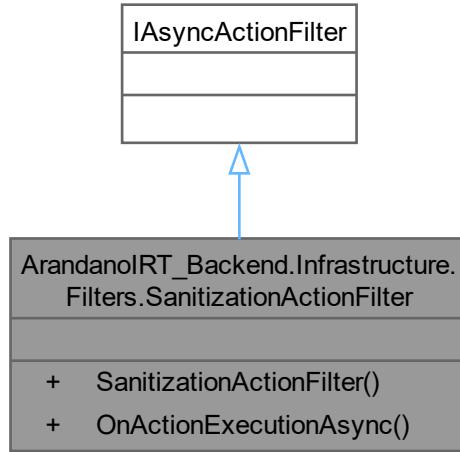
6.166 ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter Class Reference

An ASP.NET Core action filter that automatically sanitizes string properties marked with the SanitizeHtmlAttribute on action method arguments (typically DTOs).

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter:



Public Member Functions

- `SanitizationActionFilter (ISanitizerService sanitizer, ILogger< SanitizationActionFilter > logger)`
Initializes a new instance of the SanitizationActionFilter class.
- `async Task OnActionExecutionAsync (ActionExecutingContext context, ActionExecutionDelegate next)`
Called asynchronously before the action method executes.

6.166.1 Detailed Description

An ASP.NET Core action filter that automatically sanitizes string properties marked with the `SanitizeHtmlAttribute` on action method arguments (typically DTOs).

Uses the registered `ISanitizerService` to remove potentially harmful elements.

6.166.2 Constructor & Destructor Documentation

6.166.2.1 SanitizationActionFilter()

```
ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter.SanitizationActionFilter ( 
    ISanitizerService sanitizer,
    ILogger< SanitizationActionFilter > logger ) [inline]
```

Initializes a new instance of the `SanitizationActionFilter` class.

Parameters

<i>sanitizer</i>	The sanitizer service instance.
<i>logger</i>	The logger instance.

Exceptions

<i>ArgumentNullException</i>	Thrown if sanitizer or logger is null.
------------------------------	--

6.166.3 Member Function Documentation**6.166.3.1 OnActionExecutionAsync()**

```
async Task ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter.OnActionExecutionAsync (
    ActionExecutingContext context,
    ActionExecutionDelegate next ) [inline]
```

Called asynchronously before the action method executes.

Inspects action arguments and sanitizes marked string properties.

Parameters

<i>context</i>	The context for the action executing.
<i>next</i>	The delegate representing the remaining action filter pipeline and the action method itself.

Returns

A Task that on completion indicates the filter has executed.

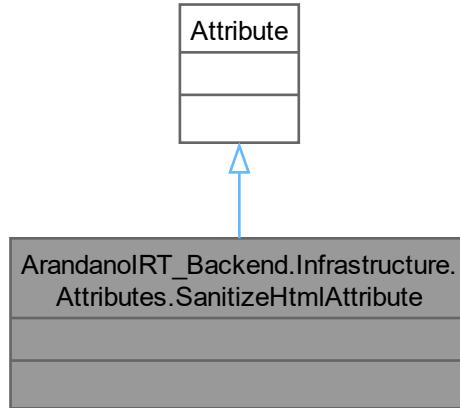
The documentation for this class was generated from the following file:

- Infrastructure/Filters/SanitizationActionFilter.cs

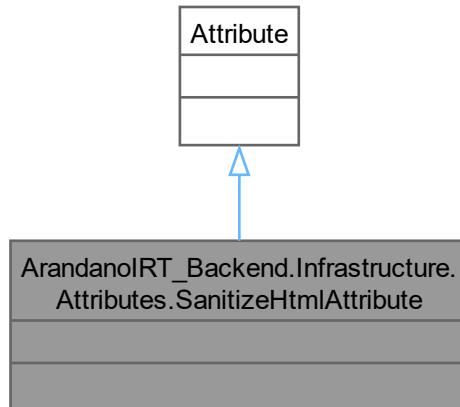
6.167 ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtml Attribute Class Reference

Attribute used to mark string properties on DTOs or models that should have HTML tags removed by the SanitizationActionFilter.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute:



6.167.1 Detailed Description

Attribute used to mark string properties on DTOs or models that should have HTML tags removed by the Sanitization ActionFilter.

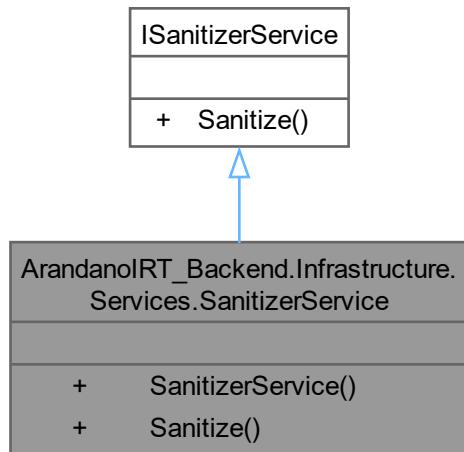
The documentation for this class was generated from the following file:

- Infrastructure/Attributes/[SanitizeHtmlAttribute.cs](#)

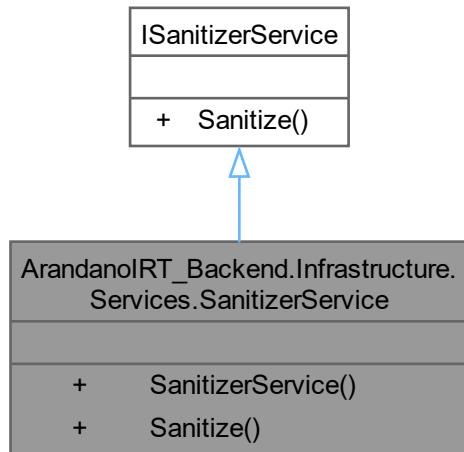
6.168 ArandanoIRT_Backend.Infrastructure.Services.SanitizerService Class Reference

Implements the ISanitizerService.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.SanitizerService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.SanitizerService:



Public Member Functions

- `SanitizerService (ILogger< SanitizerService > logger)`
Initializes a new instance of the SanitizerService class.
- `string Sanitize (string? input)`
Sanitizes the provided input string according to the implementation's policy.

Parameters

<code>input</code>	<i>The potentially unsafe input string.</i>
--------------------	---

Returns

A sanitized string, safe for storage or specific uses.

6.168.1 Detailed Description

Implements the ISanitizerService.

This implementation uses HtmlSanitizer for XSS protection (default config) and then decodes HTML entities to preserve original characters. Note: Default HtmlSanitizer configuration allows some safe HTML tags.

6.168.2 Constructor & Destructor Documentation

6.168.2.1 SanitizerService()

```
ArandanoIRT_Backend.Infrastructure.Services.SanitizerService.SanitizerService (
    ILogger< SanitizerService > logger ) [inline]
```

Initializes a new instance of the SanitizerService class.

Parameters

<code>logger</code>	<i>The logger instance.</i>
---------------------	-----------------------------

Exceptions

<code>ArgumentNullException</code>	<i>Thrown if logger is null.</i>
------------------------------------	----------------------------------

6.168.3 Member Function Documentation

6.168.3.1 Sanitize()

```
string ArandanoIRT_Backend.Infrastructure.Services.SanitizerService.Sanitize (
    string? input ) [inline]
```

Sanitizes the provided input string according to the implementation's policy.

Parameters

<i>input</i>	The potentially unsafe input string.
--------------	--------------------------------------

Returns

A sanitized string, safe for storage or specific uses.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService](#).

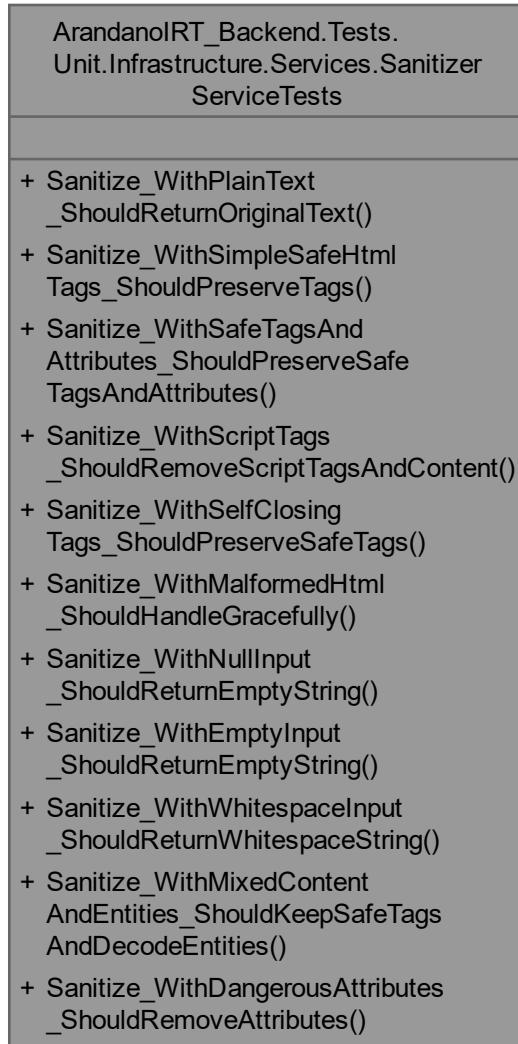
The documentation for this class was generated from the following file:

- Infrastructure/Services/[SanitizerService.cs](#)

6.169 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests Class Reference

Contains unit tests for the SanitizerService class (using default HtmlSanitizer + HtmlDecode).

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests:



Public Member Functions

- void `Sanitize_WithPlainText_ShouldReturnOriginalText` (string plainText)
Tests that plain text without any HTML tags remains unchanged after sanitization.
- void `Sanitize_WithSimpleSafeHtmlTags_ShouldPreserveTags` ()
Tests that simple, commonly safe HTML tags like 'p' and 'b' are preserved.
- void `Sanitize_WithSafeTagsAndAttributes_ShouldPreserveSafeTagsAndAttributes` ()
Tests that safe tags like 'a' with generally safe attributes like 'href' are preserved, while potentially unsafe attributes like 'class' might be removed by default rules.

- void [Sanitize_WithScriptTags_ShouldRemoveScriptTagsAndContent \(\)](#)
Tests that potentially dangerous 'script' tags and their entire content are removed.
- void [Sanitize_WithSelfClosingTags_ShouldPreserveSafeTags \(\)](#)
Tests that safe self-closing tags (like 'br', 'hr', 'img') and their safe attributes (like 'src') are preserved.
- void [Sanitize_WithMalformedHtml_ShouldHandleGracefully \(\)](#)
Tests that malformed HTML (e.g., unclosed tags) is handled gracefully according to the underlying sanitizer library's specific correction/removal logic.
- void [Sanitize_WithNullInput_ShouldReturnEmptyString \(\)](#)
Tests that null input results in an empty string.
- void [Sanitize_WithEmptyInput_ShouldReturnEmptyString \(\)](#)
Tests that empty string input results in an empty string output.
- void [Sanitize_WithWhitespaceInput_ShouldReturnWhitespaceString \(\)](#)
Tests that input consisting only of whitespace is returned unchanged.
- void [Sanitize_WithMixedContentAndEntities_ShouldKeepSafeTagsAndDecodeEntities \(\)](#)
Tests that safe HTML tags are preserved while HTML entities within the content are decoded back to their original characters.
- void [Sanitize_WithDangerousAttributes_ShouldRemoveAttributes \(\)](#)
Tests that dangerous attributes (like 'onclick') are removed from tags, while potentially safe attributes like 'style' might remain but could be normalized or restricted.

6.169.1 Detailed Description

Contains unit tests for the SanitizerService class (using default HtmlSanitizer + HtmlDecode).

Expects dangerous tags/attributes to be removed, safe tags/attributes to be preserved, and HTML entities to be decoded.

6.169.2 Member Function Documentation

6.169.2.1 Sanitize_WithDangerousAttributes_ShouldRemoveAttributes()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←
DangerousAttributes_ShouldRemoveAttributes ( ) [inline]
```

Tests that dangerous attributes (like 'onclick') are removed from tags, while potentially safe attributes like 'style' might remain but could be normalized or restricted.

6.169.2.2 Sanitize_WithEmptyInput_ShouldReturnEmptyString()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←
EmptyInput_ShouldReturnEmptyString ( ) [inline]
```

Tests that empty string input results in an empty string output.

6.169.2.3 Sanitize_WithMalformedHtml_ShouldHandleGracefully()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←  
MalformedHtml_ShouldHandleGracefully ( ) [inline]
```

Tests that malformed HTML (e.g., unclosed tags) is handled gracefully according to the underlying sanitizer library's specific correction/removal logic.

The exact output depends heavily on the library implementation.

6.169.2.4 Sanitize_WithMixedContentAndEntities_ShouldKeepSafeTagsAndDecodeEntities()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←  
MixedContentAndEntities_ShouldKeepSafeTagsAndDecodeEntities ( ) [inline]
```

Tests that safe HTML tags are preserved while HTML entities within the content are decoded back to their original characters.

6.169.2.5 Sanitize_WithNullInput_ShouldReturnEmptyString()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←  
NullInput_ShouldReturnEmptyString ( ) [inline]
```

Tests that null input results in an empty string.

6.169.2.6 Sanitize_WithPlainText_ShouldReturnOriginalText()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←  
PlainText_ShouldReturnOriginalText (   
    string plainText ) [inline]
```

Tests that plain text without any HTML tags remains unchanged after sanitization.

Parameters

<i>plainText</i>	The plain text input string.
------------------	------------------------------

6.169.2.7 Sanitize_WithSafeTagsAndAttributes_ShouldPreserveSafeTagsAndAttributes()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With←  
SafeTagsAndAttributes_ShouldPreserveSafeTagsAndAttributes ( ) [inline]
```

Tests that safe tags like 'a' with generally safe attributes like 'href' are preserved, while potentially unsafe attributes like 'class' might be removed by default rules.

Verifies that other potentially safe tags like 'span' are also preserved by default.

6.169.2.8 Sanitize_WithScriptTags_ShouldRemoveScriptTagsAndContent()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With<→
ScriptTags_ShouldRemoveScriptTagsAndContent ( ) [inline]
```

Tests that potentially dangerous 'script' tags and their entire content are removed.

6.169.2.9 Sanitize_WithSelfClosingTags_ShouldPreserveSafeTags()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With<→
SelfClosingTags_ShouldPreserveSafeTags ( ) [inline]
```

Tests that safe self-closing tags (like 'br', 'hr', 'img') and their safe attributes (like 'src') are preserved.

Note: Tag formatting might be normalized (e.g., becomes).

6.169.2.10 Sanitize_WithSimpleSafeHtmlTags_ShouldPreserveTags()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With<→
SimpleSafeHtmlTags_ShouldPreserveTags ( ) [inline]
```

Tests that simple, commonly safe HTML tags like 'p' and 'b' are preserved.

6.169.2.11 Sanitize_WithWhitespaceInput_ShouldReturnWhitespaceString()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests.Sanitize_With<→
WhitespaceInput_ShouldReturnWhitespaceString ( ) [inline]
```

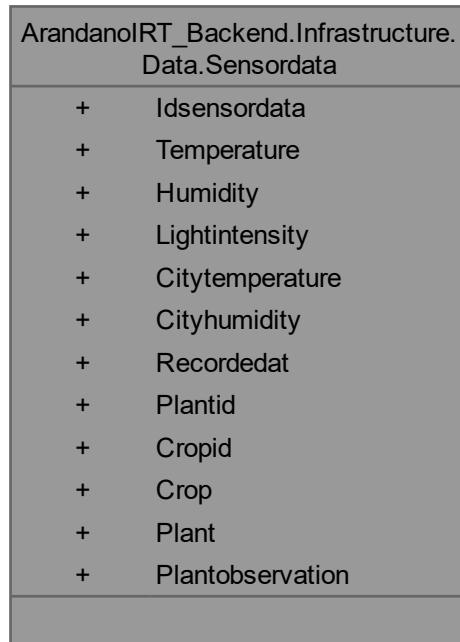
Tests that input consisting only of whitespace is returned unchanged.

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Services/[SanitizerServiceTests.cs](#)

6.170 ArandanoIRT_Backend.Infrastructure.Data.Sensordata Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Sensordata:



Properties

- int `Idsensordata` [get, set]
- float `Temperature` [get, set]
- float `Humidity` [get, set]
- float `Lightintensity` [get, set]
- float? `Citytemperature` [get, set]
- float? `Cityhumidity` [get, set]
- DateTime `Recordedat` [get, set]
- int? `Plantid` [get, set]
- int? `Cropid` [get, set]
- virtual ? `Crop Crop` [get, set]
- virtual ? `Plantdata Plant` [get, set]
- virtual ICollection< `Plantobservation` > `Plantobservation` = new List<`Plantobservation`>() [get, set]

6.170.1 Property Documentation

6.170.1.1 Cityhumidity

```
float? ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Cityhumidity [get], [set]
```

6.170.1.2 Citytemperature

```
float? ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Citytemperature [get], [set]
```

6.170.1.3 Crop

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Crop [get], [set]
```

6.170.1.4 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Cropid [get], [set]
```

6.170.1.5 Humidity

```
float ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Humidity [get], [set]
```

6.170.1.6 Idsensordata

```
int ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Idsensordata [get], [set]
```

6.170.1.7 Lightintensity

```
float ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Lightintensity [get], [set]
```

6.170.1.8 Plant

```
virtual ? Plantdata ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Plant [get], [set]
```

6.170.1.9 Plantid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Plantid [get], [set]
```

6.170.1.10 Plantobservation

```
virtual ICollection<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Plantobservation  
= new List<Plantobservation>() [get], [set]
```

6.170.1.11 Recordedat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Recordedat [get], [set]
```

6.170.1.12 Temperature

```
float ArandanoIRT_Backend.Infrastructure.Data.Sensordata.Temperature [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Sensordata.cs](#)

6.171 ArandanoIRT_Backend.Domain.Entities.SensorDataEntity Class Reference

Represents a single snapshot of environmental sensor readings (temperature, humidity, light intensity) recorded at a specific time.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.SensorDataEntity:

ArandanoIRT_Backend.Domain.Entities.SensorDataEntity
+ IdSensorData
+ Temperature
+ Humidity
+ LightIntensity
+ CityTemperature
+ CityHumidity
+ RecordedAt
+ PlantId
+ CropId
+ SensorDataEntity()

Public Member Functions

- `SensorDataEntity (int idSensorData, double temperature, double humidity, double lightIntensity, double? cityTemperature, double? cityHumidity, DateTime recordedAt, int? plantId, int? cropId)`
Initializes a new instance of the SensorDataEntity class.

Properties

- `int IdSensorData [get]`
Gets the unique identifier for this sensor data record.
- `double Temperature [get]`
Gets the temperature reading (units depend on sensor configuration, e.g., Celsius).
- `double Humidity [get]`
Gets the relative humidity reading (typically as a percentage, 0-100).
- `double LightIntensity [get]`
Gets the light intensity reading (units depend on sensor configuration, e.g., Lux).
- `double? CityTemperature [get]`
Gets the ambient temperature of the city or general area at the time of recording, if available.
- `double? CityHumidity [get]`
Gets the ambient relative humidity of the city or general area at the time of recording, if available.
- `DateTime RecordedAt [get]`
Gets the date and time (usually UTC) when these sensor readings were recorded.
- `int? PlantId [get]`
Gets the unique identifier of the plant associated with these sensor readings.
- `int? CropId [get]`
Gets the unique identifier of the crop associated with these sensor readings.

6.171.1 Detailed Description

Represents a single snapshot of environmental sensor readings (temperature, humidity, light intensity) recorded at a specific time.

May include external weather context and be associated with a plant or crop.

6.171.2 Constructor & Destructor Documentation

6.171.2.1 SensorDataEntity()

```
ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.SensorDataEntity (
    int idSensorData,
    double temperature,
    double humidity,
    double lightIntensity,
    double? cityTemperature,
    double? cityHumidity,
    DateTime recordedAt,
    int? plantId,
    int? cropId ) [inline]
```

Initializes a new instance of the SensorDataEntity class.

Parameters

<i>idSensorData</i>	The unique ID for this sensor data record (often generated by the database).
<i>temperature</i>	The temperature reading.
<i>humidity</i>	The humidity reading (expected 0-100).
<i>lightIntensity</i>	The light intensity reading (expected non-negative).
<i>cityTemperature</i>	The external city temperature (nullable).
<i>cityHumidity</i>	The external city humidity (nullable).
<i>recordedAt</i>	The timestamp when the readings were recorded.
<i>plantId</i>	The ID of the associated plant (nullable).
<i>cropId</i>	The ID of the associated crop (nullable).

Exceptions

<i>ArgumentOutOfRangeException</i>	Thrown if <i>humidity</i> is outside the range [0, 100] (Spanish message: "La humedad debe estar entre 0 y 100."). Thrown if <i>lightIntensity</i> is negative (Spanish message: "La intensidad de la luz no puede ser negativa.").
------------------------------------	---

6.171.3 Property Documentation

6.171.3.1 CityHumidity

```
double? ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.CityHumidity [get]
```

Gets the ambient relative humidity of the city or general area at the time of recording, if available.

Nullable.

6.171.3.2 CityTemperature

```
double? ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.CityTemperature [get]
```

Gets the ambient temperature of the city or general area at the time of recording, if available.

Nullable.

6.171.3.3 CropId

```
int? ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.CropId [get]
```

Gets the unique identifier of the crop associated with these sensor readings.

Nullable if not linked to a specific crop.

6.171.3.4 Humidity

```
double ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.Humidity [get]
```

Gets the relative humidity reading (typically as a percentage, 0-100).

6.171.3.5 IdSensorData

```
int ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.IdSensorData [get]
```

Gets the unique identifier for this sensor data record.

6.171.3.6 LightIntensity

```
double ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.LightIntensity [get]
```

Gets the light intensity reading (units depend on sensor configuration, e.g., Lux).

6.171.3.7 PlantId

```
int? ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.PlantId [get]
```

Gets the unique identifier of the plant associated with these sensor readings.

Nullable if not linked to a specific plant.

6.171.3.8 RecordedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.RecordedAt [get]
```

Gets the date and time (usually UTC) when these sensor readings were recorded.

6.171.3.9 Temperature

```
double ArandanoIRT_Backend.Domain.Entities.SensorDataEntity.Temperature [get]
```

Gets the temperature reading (units depend on sensor configuration, e.g., Celsius).

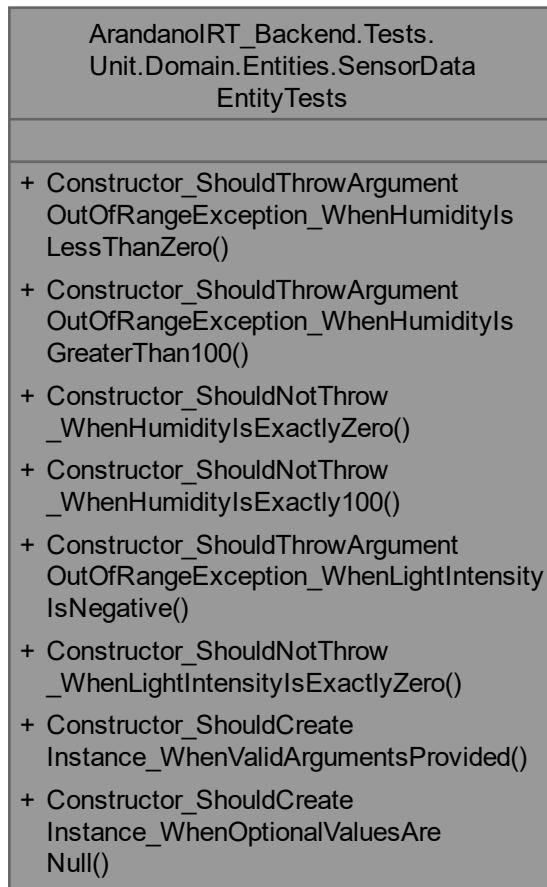
The documentation for this class was generated from the following file:

- Domain/Entities/SensorDataEntity.cs

6.172 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorData.EntityTests Class Reference

Contains unit tests for the SensorDataEntity class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests:



Public Member Functions

- void [Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsLessThanZero \(\)](#)
Verifies that the constructor throws ArgumentOutOfRangeException when humidity is less than 0.
- void [Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsGreaterThanOrEqualTo100 \(\)](#)
Verifies that the constructor throws ArgumentOutOfRangeException when humidity is greater than 100.
- void [Constructor_ShouldNotThrow_WhenHumidityIsExactlyZero \(\)](#)

- void [Constructor_ShouldNotThrow_WhenHumidityIsExactly100 \(\)](#)

Verifies that the constructor does NOT throw an exception when humidity is exactly 0 (lower boundary).
- void [Constructor_ShouldThrowArgumentOutOfRangeException_WhenLightIntensityIsNegative \(\)](#)

Verifies that the constructor does NOT throw an exception when light intensity is exactly 100 (upper boundary).
- void [Constructor_ShouldNotThrow_WhenLightIntensityIsExactlyZero \(\)](#)

Verifies that the constructor throws ArgumentOutOfRangeException when light intensity is negative.
- void [Constructor_ShouldCreateInstance_WhenValidArgumentsProvided \(\)](#)

Verifies that the constructor does NOT throw an exception when light intensity is exactly 0 (lower boundary).
- void [Constructor_ShouldCreateInstance_WhenOptionalValuesAreNull \(\)](#)

Verifies that the constructor successfully creates an instance and assigns properties correctly when all arguments are valid.
- void [Constructor_ShouldCreateInstance_WhenOptionalValuesAreNull \(\)](#)

Verifies that the constructor works correctly when optional arguments (CityTemperature, CityHumidity, PlantId, CropId) are null.

6.172.1 Detailed Description

Contains unit tests for the SensorDataEntity class.

Focuses on constructor validation logic, especially range checks.

6.172.2 Member Function Documentation

6.172.2.1 Constructor_ShouldCreateInstance_WhenOptionalValuesAreNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_Should←  
CreateInstance_WhenOptionalValuesAreNull ( ) [inline]
```

Verifies that the constructor works correctly when optional arguments (CityTemperature, CityHumidity, PlantId, CropId) are null.

6.172.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_Should←  
CreateInstance_WhenValidArgumentsProvided ( ) [inline]
```

Verifies that the constructor successfully creates an instance and assigns properties correctly when all arguments are valid.

6.172.2.3 Constructor_ShouldNotThrow_WhenHumidityIsExactly100()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_ShouldNot←  
Throw_WhenHumidityIsExactly100 ( ) [inline]
```

Verifies that the constructor does NOT throw an exception when humidity is exactly 100 (upper boundary).

6.172.2.4 Constructor_ShouldNotThrow_WhenHumidityIsExactlyZero()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_ShouldNot←  
Throw_WhenHumidityIsExactlyZero ( ) [inline]
```

Verifies that the constructor does NOT throw an exception when humidity is exactly 0 (lower boundary).

6.172.2.5 Constructor_ShouldNotThrow_WhenLightIntensityIsExactlyZero()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_ShouldNot←  
Throw_WhenLightIntensityIsExactlyZero ( ) [inline]
```

Verifies that the constructor does NOT throw an exception when light intensity is exactly 0 (lower boundary).

6.172.2.6 Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsGreaterThan100()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_Should←  
ThrowArgumentOutOfRangeException_WhenHumidityIsGreaterThan100 ( ) [inline]
```

Verifies that the constructor throws ArgumentOutOfRangeException when humidity is greater than 100.

6.172.2.7 Constructor_ShouldThrowArgumentOutOfRangeException_WhenHumidityIsLessThanZero()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_Should←  
ThrowArgumentOutOfRangeException_WhenHumidityIsLessThanZero ( ) [inline]
```

Verifies that the constructor throws ArgumentOutOfRangeException when humidity is less than 0.

6.172.2.8 Constructor_ShouldThrowArgumentOutOfRangeException_WhenLightIntensityIsNegative()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests.Constructor_Should←  
ThrowArgumentOutOfRangeException_WhenLightIntensityIsNegative ( ) [inline]
```

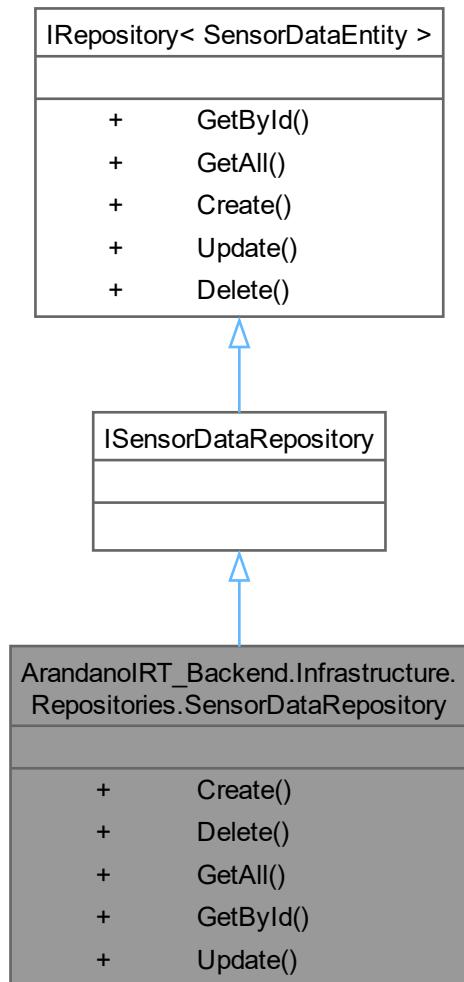
Verifies that the constructor throws ArgumentOutOfRangeException when light intensity is negative.

The documentation for this class was generated from the following file:

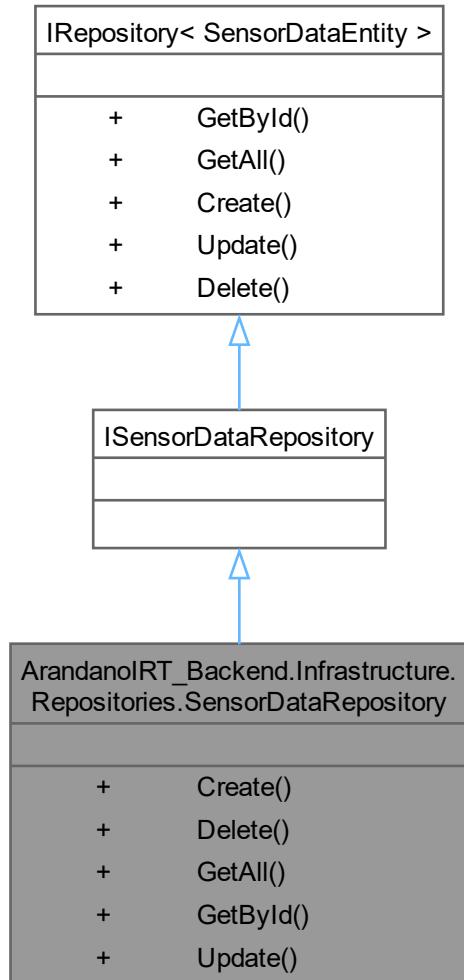
- Tests/Unit/Domain/Entities/[SensorDataEntityTests.cs](#)

6.173 ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository:



Public Member Functions

- `Task< Result< SensorDataEntity > > Create (SensorDataEntity entity)`
- `Task< Result< bool > > Delete (int id)`
- `Task< Result< IEnumerable< SensorDataEntity > > > GetAll ()`
- `Task< Result< SensorDataEntity > > GetById (int id)`
- `Task< Result< bool > > Update (SensorDataEntity entity)`

Public Member Functions inherited from

`ArandanoIRT_Backend.Domain.IRepositories.IRepository< SensorDataEntity >`

- `Task< Result< T > > GetById (int id)`

- Task<`Result< IEnumerable< T > >`> `GetAll()`
Asynchronously retrieves all entities of type T.
- Task<`Result< T >`> `Create(T entity)`
Asynchronously creates a new entity in the data store.
- Task<`Result< bool >`> `Update(T entity)`
Asynchronously updates an existing entity in the data store.
- Task<`Result< bool >`> `Delete(int id)`
Asynchronously deletes an entity from the data store by its unique identifier.

6.173.1 Member Function Documentation

6.173.1.1 Create()

```
Task< Result< SensorDataEntity >> ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository.Create (
    SensorDataEntity entity ) [inline]
```

6.173.1.2 Delete()

```
Task< Result< bool >> ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository.Delete (
    int id ) [inline]
```

6.173.1.3 GetAll()

```
Task< Result< IEnumerable< SensorDataEntity > >> ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository.GetAll () [inline]
```

6.173.1.4 GetById()

```
Task< Result< SensorDataEntity >> ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository.GetById (
    int id ) [inline]
```

6.173.1.5 Update()

```
Task< Result< bool >> ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository.Update (
    SensorDataEntity entity ) [inline]
```

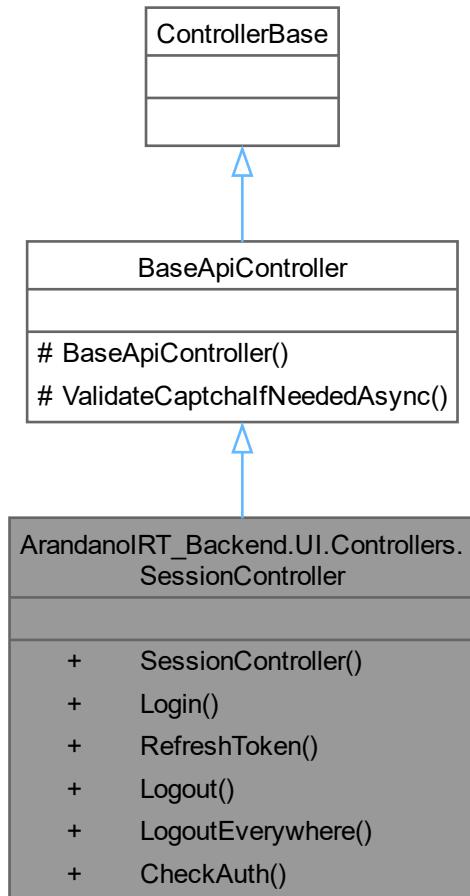
The documentation for this class was generated from the following file:

- Infrastructure/Repositories/[SensorDataRepository.cs](#)

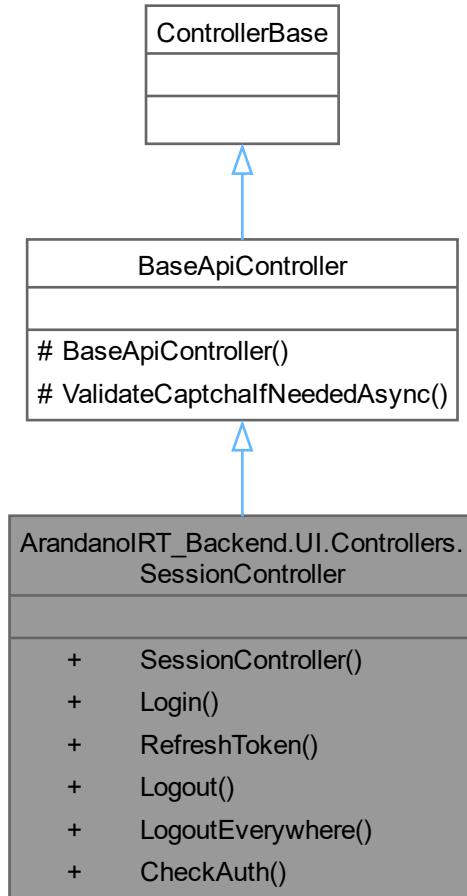
6.174 ArandanoIRT_Backend.UI.Controllers.SessionController Class Reference

Handles endpoints related to user session management, including login, token refresh, and logout operations.

Inheritance diagram for ArandanoIRT_Backend.UI.Controllers.SessionController:



Collaboration diagram for ArandanoIRT_Backend.UI.Controllers.SessionController:



Public Member Functions

- `SessionController (IAuthSessionService sessionService, CookiesService cookiesService, ITokenService tokenService, IRequestContextAccessor requestContextAccessor, ICaptchaService captchaService, IWebHostEnvironment environment, ILogger<SessionController> logger)`

Initializes a new instance of the SessionController class.
- `async Task< IActionResult > Login ([FromBody] LoginRequestDto request)`

Logs in an existing user using email and (encrypted) password.
- `async Task< IActionResult > RefreshToken ([FromHeader(Name="X-Refresh-Token")] string? xRefreshToken)`

Refreshes the JWT access token using a valid refresh token.
- `async Task< IActionResult > Logout ()`

Logs out the current user session.
- `async Task< IActionResult > LogoutEverywhere ()`

Logs out all active sessions for the current user.

- `IActionResult CheckAuth ()`

Example endpoint to check authentication status and retrieve claims.

Additional Inherited Members

Protected Member Functions inherited from `ArandanoIRT_Backend.UI.Controllers.BaseApiController`

- `BaseApiController (ICaptchaService captchaService, IWebHostEnvironment environment)`
Initializes a new instance of the BaseApiController class.
- `async Task< IActionResult?> ValidateCaptchaNeededAsync (string? captchaToken, ILogger logger)`
Validates the CAPTCHA token if the application is not running in the Development environment.

6.174.1 Detailed Description

Handles endpoints related to user session management, including login, token refresh, and logout operations.

6.174.2 Constructor & Destructor Documentation

6.174.2.1 SessionController()

```
ArandanoIRT_Backend.UI.Controllers.SessionController.SessionController (
    IAuthSessionService sessionService,
    CookiesService cookiesService,
    ITokenService tokenService,
    IRequestContextAccessor requestContextAccessor,
    ICaptchaService captchaService,
    IWebHostEnvironment environment,
    ILogger< SessionController > logger ) [inline]
```

Initializes a new instance of the SessionController class.

Parameters

<code>sessionService</code>	The application service for session logic.
<code>cookiesService</code>	Service for handling authentication cookies.
<code>tokenService</code>	Service for token operations (e.g., reading session ID).
<code>requestContextAccessor</code>	Accessor for request context information (IP, UserAgent).
<code>captchaService</code>	The service for CAPTCHA validation (passed to base).
<code>environment</code>	The web hosting environment (passed to base).
<code>logger</code>	The logger for this controller.

6.174.3 Member Function Documentation

6.174.3.1 CheckAuth()

```
IActionResult ArandanoIRT_Backend.UI.Controllers.SessionController.CheckAuth ( ) [inline]
```

Example endpoint to check authentication status and retrieve claims.

(For testing/debugging).

Returns

Authenticated user's claims information.

```
<response code="204">User is authenticated..</response> <response code="401">User is not authenticated.</response>
```

6.174.3.2 Login()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.SessionController.Login ( [FromBody] LoginRequestDto request ) [inline]
```

Logs in an existing user using email and (encrypted) password.

Expects password encrypted with the server's public RSA key. On success, sets HttpOnly cookies (jwt, refreshToken, stayLoggedIn) and returns basic user info + JWT.

Parameters

request	Login credentials (email, encrypted password, rememberMe, captcha).
---------	---

Returns

Basic user details and access token.

```
<response code="200">Login successful. Returns LoginResponseDto.</response> <response code="400">Invalid request (e.g., missing data, invalid CAPTCHA).</response> <response code="401">Unauthorized (invalid email or password).</response> <response code="500">Internal error (e.g., failed to set cookies).</response>
```

6.174.3.3 Logout()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.SessionController.Logout ( ) [inline]
```

Logs out the current user session.

Revokes the current refresh token on the server and removes auth cookies from the client. Requires the user to be authenticated.

Returns

An IActionResult indicating success (204 No Content).

```
<response code="204">Logout successful.</response> <response code="401">Unauthorized (user not authenticated).</response>
```

6.174.3.4 LogoutEverywhere()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.SessionController.LogoutEverywhere () [inline]
```

Logs out all active sessions for the current user.

Revokes all refresh tokens associated with the user's original session ID (obtained from the current token). Removes auth cookies from the current client. Requires authentication.

Returns

An IActionResult indicating success (204 No Content).

```
<response code="204">Logout everywhere successful.</response> <response code="401">Unauthorized (user not authenticated or token invalid).</response> <response code="400">Bad Request (e.g., unable to determine session ID from token).</response>
```

6.174.3.5 RefreshToken()

```
async Task< IActionResult > ArandanoIRT_Backend.UI.Controllers.SessionController.RefreshToken ([FromHeader(Name = "X-Refresh-Token")] string? xRefreshToken) [inline]
```

Refreshes the JWT access token using a valid refresh token.

Reads the refresh token from the 'refreshToken' HttpOnly cookie (preferred) or 'X-Refresh-Token' header. Implements token rotation (new access/refresh pair generated, old refresh token revoked). If the original token came from a cookie, cookies are updated. Otherwise, new tokens are returned in the body.

Parameters

<i>xRefreshToken</i>	(Optional) The refresh token passed via the 'X-Refresh-Token' header.
----------------------	---

Returns

New access and refresh tokens.

```
<response code="200">Tokens refreshed successfully. Returns TokenResponseDto.</response> <response code="400">Invalid request (e.g., refresh token missing).</response> <response code="401">Unauthorized (refresh token invalid, expired, or revoked).</response> <response code="500">Internal error (e.g., failed to renew cookies).</response>
```

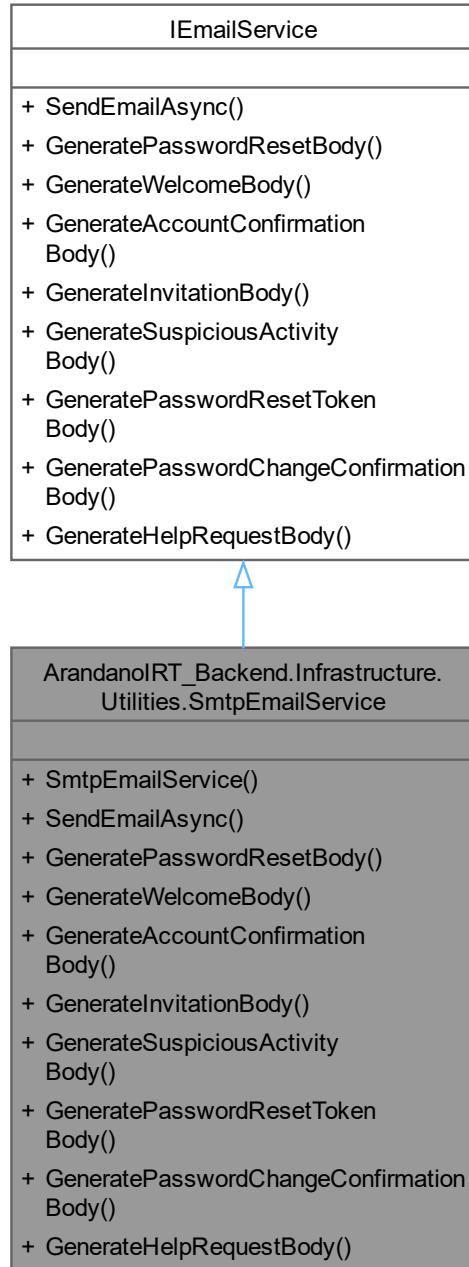
The documentation for this class was generated from the following file:

- UI/Controllers/[SessionController.cs](#)

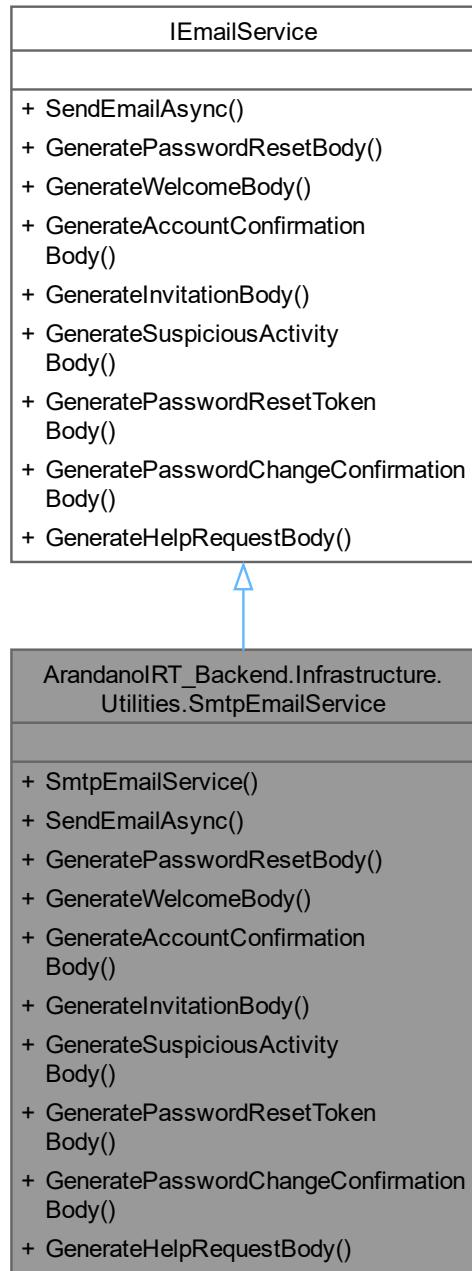
6.175 ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService Class Reference

Implements the `IEmailService` interface using an underlying `EmailSenderUtility` to send emails via SMTP.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService:



Public Member Functions

- `SmtpEmailService (IConfiguration configuration, ILogger< SmtpEmailService > logger)`

EmailSenderUtility instance for sending emails.

- `async Task< Result > SendEmailAsync (string to, string subject, string htmlBody)`

Sends an email asynchronously to the specified recipient.

Parameters

to	<i>The email address of the recipient.</i>
subject	<i>The subject line of the email.</i>
htmlBody	<i>The body of the email in HTML format.</i>

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

- `string GeneratePasswordResetBody (string userName, string resetToken)`

Generates the body for a password reset email, including the user's name and reset token.

Parameters

userName	<i>The name of the user requesting the password reset.</i>
resetToken	<i>The token used to validate the password reset request.</i>

Returns

HTML email body string for the password reset request.

- `string GenerateWelcomeBody (string userName)`

Generates the body for a welcome email, including the user's name.

Parameters

userName	<i>The name of the new user receiving the welcome message.</i>
----------	--

Returns

HTML email body string for the welcome email.

- `string GenerateAccountConfirmationBody (string userName, string confirmationLink)`

Generates the body for an account confirmation email, including the user's name and confirmation link.

Parameters

userName	<i>The name of the user confirming their account.</i>
confirmationLink	<i>The URL link that the user must follow to confirm their account.</i>

Returns

HTML email body string for the account confirmation message.

- `string GenerateInvitationBody (string inviterName, string cropName, string accessCode, string expiryDate)`

Generates the body for an invitation email, including the inviter's name, crop name, access code, and expiry date.

Parameters

inviterName	<i>The name of the person sending the invitation.</i>
cropName	<i>The name of the crop associated with the invitation.</i>
accessCode	<i>The access code required to join the invitation.</i>
expiryDate	<i>The date when the invitation expires.</i>

Returns

HTML email body string for the invitation.

- string [GenerateSuspiciousActivityBody](#) (string userName, DateTime attemptTime, string ipAddress)
Generates the email body for a suspicious login activity warning.

Parameters

userName	<i>The name of the user.</i>
attemptTime	<i>The approximate time of the suspicious activity.</i>
ipAddress	<i>The IP address associated with the attempts.</i>

Returns

HTML email body string for the suspicious activity alert.

- string [GeneratePasswordResetTokenBody](#) (string userName, string resetToken)
Generates the simple email body containing only the password reset token.

Parameters

userName	<i>The name of the user.</i>
resetToken	<i>The password reset token string.</i>

Returns

Plain text or simple HTML email body string.

- string [GeneratePasswordChangeConfirmationBody](#) (string userName)
Generates the email body confirming a successful password change.

Parameters

userName	<i>The name of the user.</i>
----------	------------------------------

Returns

HTML email body string.

- string [GenerateHelpRequestBody](#) ([HelpRequestDto](#) request)
Generates the email body for a help request submitted by a user.

Parameters

request	<i>The help request data transfer object containing user details and message.</i>
---------	---

Returns

HTML email body string for the help request notification.

6.175.1 Detailed Description

Implements the [IEmailService](#) interface using an underlying [EmailSenderUtility](#) to send emails via SMTP.

Also provides methods to generate standard email body content.

6.175.2 Constructor & Destructor Documentation

6.175.2.1 SmtpEmailService()

```
ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.SmtpEmailService (
    IConfiguration configuration,
    ILogger< SmtpEmailService > logger ) [inline]
```

EmailSenderUtility instance for sending emails.

6.175.3 Member Function Documentation

6.175.3.1 GenerateAccountConfirmationBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GenerateAccountConfirmationBody (
    string userName,
    string confirmationLink ) [inline]
```

Generates the body for an account confirmation email, including the user's name and confirmation link.

Parameters

<i>userName</i>	The name of the user confirming their account.
<i>confirmationLink</i>	The URL link that the user must follow to confirm their account.

Returns

HTML email body string for the account confirmation message.

Implements [ArandanoIRT_Backend.Application.Interfaces.IEmailService](#).

6.175.3.2 GenerateHelpRequestBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GenerateHelpRequestBody (
    HelpRequestDto request ) [inline]
```

Generates the email body for a help request submitted by a user.

Parameters

<i>request</i>	The help request data transfer object containing user details and message.
----------------	--

Returns

HTML email body string for the help request notification.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

6.175.3.3 GenerateInvitationBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GenerateInvitationBody (
    string inviterName,
    string cropName,
    string accessCode,
    string expiryDate ) [inline]
```

Generates the body for an invitation email, including the inviter's name, crop name, access code, and expiry date.

Parameters

<i>inviterName</i>	The name of the person sending the invitation.
<i>cropName</i>	The name of the crop associated with the invitation.
<i>accessCode</i>	The access code required to join the invitation.
<i>expiryDate</i>	The date when the invitation expires.

Returns

HTML email body string for the invitation.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

6.175.3.4 GeneratePasswordChangeConfirmationBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GeneratePasswordChangeConfirmationBody (
    string userName ) [inline]
```

Generates the email body confirming a successful password change.

Parameters

<i>userName</i>	The name of the user.
-----------------	-----------------------

Returns

HTML email body string.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

6.175.3.5 GeneratePasswordResetBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GeneratePasswordResetBody (
    string userName,
    string resetToken ) [inline]
```

Generates the body for a password reset email, including the user's name and reset token.

Parameters

<i>userName</i>	The name of the user requesting the password reset.
<i>resetToken</i>	The token used to validate the password reset request.

Returns

HTML email body string for the password reset request.

Implements [ArandanoIRT_Backend.Application.Interfaces.IEmailService](#).

6.175.3.6 GeneratePasswordResetTokenBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GeneratePasswordResetTokenBody (
    string userName,
    string resetToken ) [inline]
```

Generates the simple email body containing only the password reset token.

Parameters

<i>userName</i>	The name of the user.
<i>resetToken</i>	The password reset token string.

Returns

Plain text or simple HTML email body string.

Implements [ArandanoIRT_Backend.Application.Interfaces.IEmailService](#).

6.175.3.7 GenerateSuspiciousActivityBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GenerateSuspiciousActivityBody (
    string userName,
    DateTime attemptTime,
    string ipAddress ) [inline]
```

Generates the email body for a suspicious login activity warning.

Parameters

<i>userName</i>	The name of the user.
<i>attemptTime</i>	The approximate time of the suspicious activity.
<i>ipAddress</i>	The IP address associated with the attempts.

Returns

HTML email body string for the suspicious activity alert.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

6.175.3.8 GenerateWelcomeBody()

```
string ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.GenerateWelcomeBody (
    string userName) [inline]
```

Generates the body for a welcome email, including the user's name.

Parameters

<i>userName</i>	The name of the new user receiving the welcome message.
-----------------	---

Returns

HTML email body string for the welcome email.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

6.175.3.9 SendEmailAsync()

```
async Task< Result > ArandanoIRT_Backend.Infrastructure.Utilities.SmtpEmailService.SendEmailAsync
(
    string to,
    string subject,
    string htmlBody) [inline]
```

Sends an email asynchronously to the specified recipient.

Parameters

<i>to</i>	The email address of the recipient.
<i>subject</i>	The subject line of the email.
<i>htmlBody</i>	The body of the email in HTML format.

Returns

A Task representing the asynchronous operation, with a result indicating success or failure.

Performs email address validation before attempting to send. Wraps the potentially synchronous SendEmail method of EmailSenderUtility in Task.Run to conform to the async interface. Consider making EmailSenderUtility.SendEmail asynchronous using SmtpClient.SendMailAsync for better performance.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#).

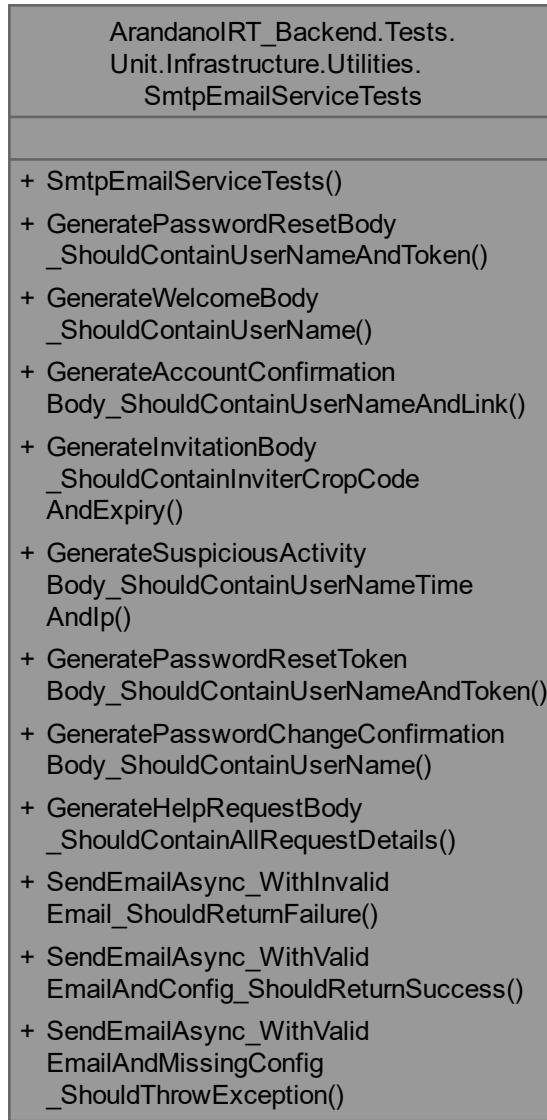
The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[SmtpEmailService.cs](#)

6.176 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests Class Reference

Contains unit tests for the SmtpEmailService class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests:



Public Member Functions

- [SmtpEmailServiceTests \(\)](#)
Initializes a new instance of the SmtpEmailServiceTests class.
- [void GeneratePasswordResetBody_ShouldContainUserNameAndToken \(\)](#)
- [void GenerateWelcomeBody_ShouldContainUserName \(\)](#)
- [void GenerateAccountConfirmationBody_ShouldContainUserNameAndLink \(\)](#)

- void [GenerateInvitationBody_ShouldContainInviterCropCodeAndExpiry\(\)](#)
- void [GenerateSuspiciousActivityBody_ShouldContainUserNameTimeAndIp\(\)](#)
- void [GeneratePasswordResetTokenBody_ShouldContainUserNameAndToken\(\)](#)
- void [GeneratePasswordChangeConfirmationBody_ShouldContainUserName\(\)](#)
- void [GenerateHelpRequestBody_ShouldContainAllRequestDetails\(\)](#)
- async Task [SendEmailAsync_WithInvalidEmail_ShouldReturnFailure\(\)](#)

Tests that SendEmailAsync returns a failure result when provided with an invalid recipient email address.
- async Task [SendEmailAsync_WithValidEmailAndConfig_ShouldReturnSuccess\(\)](#)

Tests that SendEmailAsync returns success for a valid email address when SMTP configuration is present.
- async Task [SendEmailAsync_WithValidEmailAndMissingConfig_ShouldThrowException\(\)](#)

Tests that SendEmailAsync (or its internal utility) throws an exception when required SMTP configuration is missing.

6.176.1 Detailed Description

Contains unit tests for the SmtpEmailService class.

6.176.2 Constructor & Destructor Documentation

6.176.2.1 SmtpEmailServiceTests()

```
ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.SmtpEmailService←
Tests ( ) [inline]
```

Initializes a new instance of the SmtpEmailServiceTests class.

Sets up mock configuration required by the SmtpEmailService.

6.176.3 Member Function Documentation

6.176.3.1 GenerateAccountConfirmationBody_ShouldContainUserNameAndLink()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
AccountConfirmationBody_ShouldContainUserNameAndLink ( ) [inline]
```

6.176.3.2 GenerateHelpRequestBody_ShouldContainAllRequestDetails()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.GenerateHelp←
RequestBody_ShouldContainAllRequestDetails ( ) [inline]
```

6.176.3.3 GenerateInvitationBody_ShouldContainInviterCropCodeAndExpiry()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
InvitationBody_ShouldContainInviterCropCodeAndExpiry ( ) [inline]
```

6.176.3.4 GeneratePasswordChangeConfirmationBody_ShouldContainUserName()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
PasswordChangeConfirmationBody_ShouldContainUserName ( ) [inline]
```

6.176.3.5 GeneratePasswordResetBody_ShouldContainUserNameAndToken()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
PasswordResetBody_ShouldContainUserNameAndToken ( ) [inline]
```

6.176.3.6 GeneratePasswordResetTokenBody_ShouldContainUserNameAndToken()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
PasswordResetTokenBody_ShouldContainUserNameAndToken ( ) [inline]
```

6.176.3.7 GenerateSuspiciousActivityBody_ShouldContainUserNameTimeAndIp()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
SuspiciousActivityBody_ShouldContainUserNameTimeAndIp ( ) [inline]
```

6.176.3.8 GenerateWelcomeBody_ShouldContainUserName()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Generate←
WelcomeBody_ShouldContainUserName ( ) [inline]
```

6.176.3.9 SendEmailAsync_WithInvalidEmail_ShouldReturnFailure()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Send←
EmailAsync_WithInvalidEmail_ShouldReturnFailure ( ) [inline]
```

Tests that SendEmailAsync returns a failure result when provided with an invalid recipient email address.

Relies on the validation performed by EmailValidatorUtility internally.

6.176.3.10 SendEmailAsync_WithValidEmailAndConfig_ShouldReturnSuccess()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Send←
EmailAsync_WithValidEmailAndConfig_ShouldReturnSuccess ( ) [inline]
```

Tests that SendEmailAsync returns success for a valid email address when SMTP configuration is present.

This test simulates the scenario where configuration is valid, and email sending *would* be attempted (though it doesn't actually send in the unit test environment).

6.176.3.11 SendEmailAsync_WithValidEmailAndMissingConfig_ShouldThrowException()

```
async Task ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests.Send←
EmailAsync_WithValidEmailAndMissingConfig_ShouldThrowException() [inline]
```

Tests that SendEmailAsync (or its internal utility) throws an exception when required SMTP configuration is missing.

This behavior depends on the implementation of the underlying email sending utility.

The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Utilities/[SmtpEmailServiceTests.cs](#)

6.177 ArandanoIRT_Backend.Infrastructure.Data.Status Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Status:

ArandanoIRT_Backend.Infrastructure.Data.Status	
+	Idstatus
+	Namestatus
+	Tablerelationid
+	Cropinvitation
+	Deviceactivation
+	Devicedata
+	Plantdata
+	Plantobservation
+	Plantstatehistory
+	Tablerelation

Properties

- int [Idstatus](#) [get, set]
- string [Namestatus](#) = null! [get, set]
- int? [Tablerelationid](#) [get, set]
- virtual ICollection<[Cropinvitation](#)> [Cropinvitation](#) = new List<[Cropinvitation](#)>() [get, set]
- virtual ICollection<[Deviceactivation](#)> [Deviceactivation](#) = new List<[Deviceactivation](#)>() [get, set]
- virtual ICollection<[Devicedata](#)> [Devicedata](#) = new List<[Devicedata](#)>() [get, set]
- virtual ICollection<[Plantdata](#)> [Plantdata](#) = new List<[Plantdata](#)>() [get, set]
- virtual ICollection<[Plantobservation](#)> [Plantobservation](#) = new List<[Plantobservation](#)>() [get, set]
- virtual ICollection<[Plantstatehistory](#)> [Plantstatehistory](#) = new List<[Plantstatehistory](#)>() [get, set]
- virtual ? [Tablerelation](#) [Tablerelation](#) [get, set]

6.177.1 Property Documentation

6.177.1.1 Cropinvitation

```
virtual ICollection<Cropinvitation> ArandanoIRT_Backend.Infrastructure.Data.Status.Cropinvitation  
= new List<Cropinvitation>() [get], [set]
```

6.177.1.2 Deviceactivation

```
virtual ICollection<Deviceactivation> ArandanoIRT_Backend.Infrastructure.Data.Status.Deviceactivation  
= new List<Deviceactivation>() [get], [set]
```

6.177.1.3 Devicedata

```
virtual ICollection<Devicedata> ArandanoIRT_Backend.Infrastructure.Data.Status.Devicedata = new  
List<Devicedata>() [get], [set]
```

6.177.1.4 Idstatus

```
int ArandanoIRT_Backend.Infrastructure.Data.Status.Idstatus [get], [set]
```

6.177.1.5 Namestatus

```
string ArandanoIRT_Backend.Infrastructure.Data.Status.Namestatus = null! [get], [set]
```

6.177.1.6 Plantdata

```
virtual ICollection<Plantdata> ArandanoIRT_Backend.Infrastructure.Data.Status.Plantdata = new  
List<Plantdata>() [get], [set]
```

6.177.1.7 Plantobservation

```
virtual ICollection<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.Status.Plantobservation  
= new List<Plantobservation>() [get], [set]
```

6.177.1.8 Plantstatehistory

```
virtual ICollection<Plantstatehistory> ArandanoIRT_Backend.Infrastructure.Data.Status.Plantstatehistory  
= new List<Plantstatehistory>() [get], [set]
```

6.177.1.9 Tablerelation

```
virtual ? Tablerelation ArandanoIRT_Backend.Infrastructure.Data.Status.Tablerelation [get],  
[set]
```

6.177.1.10 Tablerelationid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Status.Tablerelationid [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Status.cs](#)

6.178 ArandanoIRT_Backend.Domain.Entities.StatusEntity Class Reference

Represents a defined status value (e.g., "Active", "Pending", "Used") that can be applied to various entities within the application.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.StatusEntity:

ArandanoIRT_Backend.Domain.Entities.StatusEntity
+ IdStatus
+ NameStatus
+ TableRelationId
+ TableRelation
+ StatusEntity()

Public Member Functions

- [StatusEntity](#) (int idStatus, string nameStatus, int? tableRelationId=null)
Initializes a new instance of the StatusEntity class.

Properties

- int **IdStatus** [get]
Gets the unique identifier for the status.
- string **NameStatus** [get]
Gets the display name or code identifying the status (e.g., "Active", "Pending").
- int? **TableRelationId** [get, set]
Gets the optional foreign key linking this status to a specific table or entity type definition in the TableRelationEntity, indicating where this status is primarily used.
- **TableRelationEntity? TableRelation** [get]
Gets the optional navigation property to the related TableRelationEntity.

6.178.1 Detailed Description

Represents a defined status value (e.g., "Active", "Pending", "Used") that can be applied to various entities within the application.

It can optionally be linked to a specific table type via TableRelation.

6.178.2 Constructor & Destructor Documentation

6.178.2.1 StatusEntity()

```
ArandanoIRT_Backend.Domain.Entities.StatusEntity.StatusEntity (
    int idStatus,
    string nameStatus,
    int? tableRelationId = null ) [inline]
```

Initializes a new instance of the StatusEntity class.

Parameters

<i>idStatus</i>	The unique ID for the status.
<i>nameStatus</i>	The name or code identifying the status.
<i>tableRelationId</i>	The optional ID linking this status to a table/entity type (defaults to null).

Exceptions

<i>ArgumentException</i>	Thrown if <i>nameStatus</i> is null or whitespace (Spanish message: "El nombre del status es requerido.").
--------------------------	--

The TableRelation navigation property is not set by this constructor.

6.178.3 Property Documentation

6.178.3.1 IdStatus

```
int ArandanoIRT_Backend.Domain.Entities.StatusEntity.IdStatus [get]
```

Gets the unique identifier for the status.

6.178.3.2 NameStatus

```
string ArandanoIRT_Backend.Domain.Entities.StatusEntity.NameStatus [get]
```

Gets the display name or code identifying the status (e.g., "Active", "Pending").

6.178.3.3 TableRelation

```
TableRelationEntity? ArandanoIRT_Backend.Domain.Entities.StatusEntity.TableRelation [get]
```

Gets the optional navigation property to the related TableRelationEntity.

This property is typically populated by an ORM (like Entity Framework Core) and may be null otherwise.

6.178.3.4 TableRelationId

```
int? ArandanoIRT_Backend.Domain.Entities.StatusEntity.TableRelationId [get], [set]
```

Gets the optional foreign key linking this status to a specific table or entity type definition in the TableRelationEntity, indicating where this status is primarily used.

Nullable.

The documentation for this class was generated from the following file:

- Domain/Entities/[StatusEntity.cs](#)

6.179 ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests Class Reference

Contains unit tests for the StatusEntity class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests:

ArandanoIRT_Backend.Tests. Unit.Domain.Entities.StatusEntityTests
+ Constructor_ShouldThrowArgument Exception_WhenNameStatusIsNull() + Constructor_ShouldThrowArgument Exception_WhenNameStatusIsEmpty() + Constructor_ShouldThrowArgument Exception_WhenNameStatusIsWhitespace() + Constructor_ShouldCreate Instance_WhenValidArgumentsProvided() + Constructor_ShouldCreate Instance_WhenTableRelationId IsNull()

Public Member Functions

- void [Constructor_ShouldThrowArgumentException_WhenNameStatusIsNull \(\)](#)
Verifies that the constructor throws an ArgumentException when the status name provided is null.
- void [Constructor_ShouldThrowArgumentException_WhenNameStatusIsEmpty \(\)](#)
Verifies that the constructor throws an ArgumentException when the status name provided is empty.
- void [Constructor_ShouldThrowArgumentException_WhenNameStatusIsWhitespace \(\)](#)
Verifies that the constructor throws an ArgumentException when the status name provided consists only of whitespace.
- void [Constructor_ShouldCreateInstance_WhenValidArgumentsProvided \(\)](#)
Verifies that the constructor successfully creates an instance and assigns properties correctly when valid arguments are provided.
- void [Constructor_ShouldCreateInstance_WhenTableRelationIdIsNull \(\)](#)
Verifies that the constructor successfully creates an instance when the optional TableRelationId is null.

6.179.1 Detailed Description

Contains unit tests for the StatusEntity class.

Focuses on constructor validation logic.

6.179.2 Member Function Documentation

6.179.2.1 Constructor_ShouldCreateInstance_WhenTableRelationIdIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests.Constructor_ShouldCreate←  
Instance_WhenTableRelationIdIsNull ( ) [inline]
```

Verifies that the constructor successfully creates an instance when the optional TableRelationId is null.

6.179.2.2 Constructor_ShouldCreateInstance_WhenValidArgumentsProvided()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests.Constructor_ShouldCreate←  
Instance_WhenValidArgumentsProvided ( ) [inline]
```

Verifies that the constructor successfully creates an instance and assigns properties correctly when valid arguments are provided.

6.179.2.3 Constructor_ShouldThrowArgumentException_WhenNameStatusIsEmpty()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenNameStatusIsEmpty ( ) [inline]
```

Verifies that the constructor throws an ArgumentException when the status name provided is empty.

6.179.2.4 Constructor_ShouldThrowArgumentException_WhenNameStatusIsNull()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenNameStatusIsNull ( ) [inline]
```

Verifies that the constructor throws an ArgumentException when the status name provided is null.

6.179.2.5 Constructor_ShouldThrowArgumentException_WhenNameStatusIsWhitespace()

```
void ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests.Constructor_ShouldThrow←  
ArgumentException_WhenNameStatusIsWhitespace ( ) [inline]
```

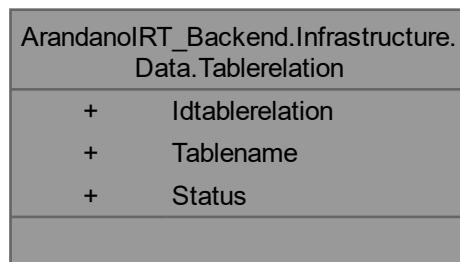
Verifies that the constructor throws an ArgumentException when the status name provided consists only of whitespace.

The documentation for this class was generated from the following file:

- Tests/Unit/Domain/Entities/[StatusEntityTests.cs](#)

6.180 ArandanoIRT_Backend.Infrastructure.Data.Tablerelation Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Tablerelation:



Properties

- int `Idtablerelation` [get, set]
- string `Tablename` = null! [get, set]
- virtual ICollection<`Status`> `Status` = new List<`Status`>() [get, set]

6.180.1 Property Documentation

6.180.1.1 `Idtablerelation`

```
int ArandanoIRT_Backend.Infrastructure.Data.Tablerelation.Idtablerelation [get], [set]
```

6.180.1.2 `Status`

```
virtual ICollection<Status> ArandanoIRT_Backend.Infrastructure.Data.Tablerelation.Status = new List<Status>() [get], [set]
```

6.180.1.3 `Tablename`

```
string ArandanoIRT_Backend.Infrastructure.Data.Tablerelation.Tablename = null! [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Tablerelation.cs](#)

6.181 ArandanoIRT_Backend.Domain.Entities.TableRelationEntity Class Reference

Represents a mapping between a unique identifier and a database table name.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.TableRelationEntity:

ArandanoIRT_Backend.Domain.Entities.TableRelationEntity
+ IdTableRelation
+ TableName
+ TableRelationEntity()

Public Member Functions

- [TableRelationEntity](#) (int idTableRelation, string tableName)

Initializes a new instance of the TableRelationEntity class.

Properties

- int [IdTableRelation](#) [get]
Gets the unique identifier for the table relation mapping.
- string [TableName](#) [get]
Gets the name of the database table being referenced.

6.181.1 Detailed Description

Represents a mapping between a unique identifier and a database table name.

This is likely used to associate other entities (like StatusEntity) with the specific type of table or entity they relate to.

6.181.2 Constructor & Destructor Documentation

6.181.2.1 TableRelationEntity()

```
ArandanoIRT_Backend.Domain.Entities.TableRelationEntity.TableRelationEntity (
    int idTableRelation,
    string tableName ) [inline]
```

Initializes a new instance of the TableRelationEntity class.

Parameters

<i>idTableRelation</i>	The unique ID for this table relation record.
<i>tableName</i>	The name of the database table.

Exceptions

<i>ArgumentException</i>	Thrown if <i>tableName</i> is null or whitespace (Spanish message: "El nombre de la tabla es requerido.").
--------------------------	--

6.181.3 Property Documentation

6.181.3.1 IdTableRelation

```
int ArandanoIRT_Backend.Domain.Entities.TableRelationEntity.IdTableRelation [get]
```

Gets the unique identifier for the table relation mapping.

6.181.3.2 TableName

```
string ArandanoIRT_Backend.Domain.Entities.TableRelationEntity.TableName [get]
```

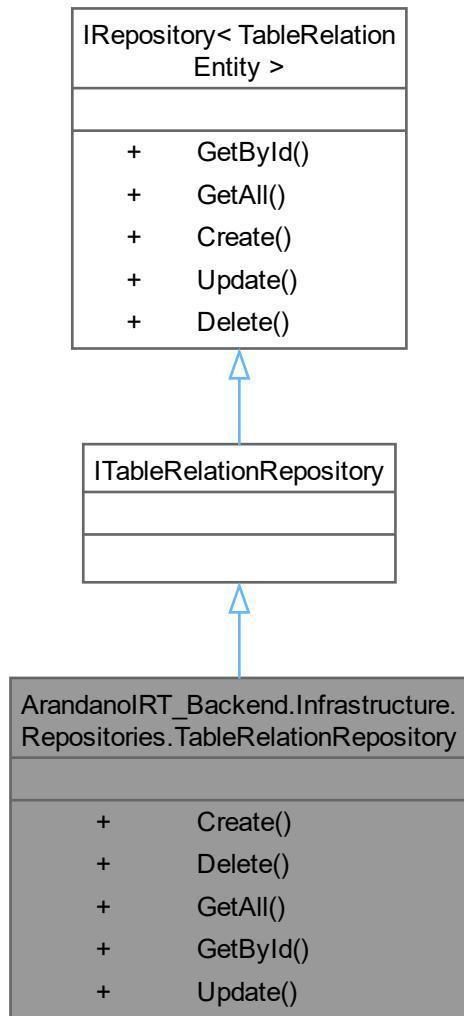
Gets the name of the database table being referenced.

The documentation for this class was generated from the following file:

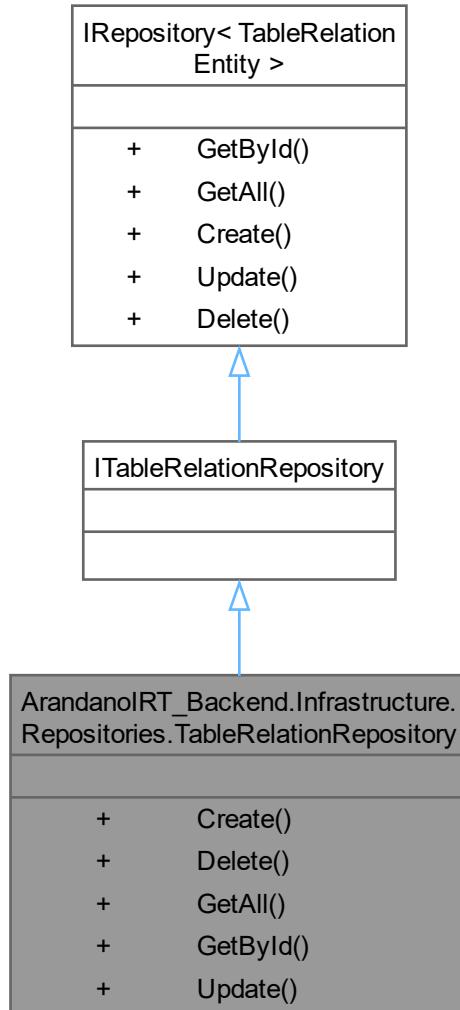
- Domain/Entities/[TableRelationEntity.cs](#)

6.182 ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository:



Public Member Functions

- Task<`Result<TableRelationEntity>`>> `Create (TableRelationEntity entity)`
- Task<`Result<bool>`>> `Delete (int id)`
- Task<`Result<IEnumerable<TableRelationEntity>`>>> `GetAll ()`
- Task<`Result<TableRelationEntity>`>> `GetById (int id)`
- Task<`Result<bool>`>> `Update (TableRelationEntity entity)`

Public Member Functions inherited from**ArandanoIRT_Backend.Domain.IRepositories IRepository< TableRelationEntity >**

- Task< Result< T > > **GetById** (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< Result< IEnumerable< T > > > **GetAll** ()
Asynchronously retrieves all entities of type T .
- Task< Result< T > > **Create** (T entity)
Asynchronously creates a new entity in the data store.
- Task< Result< bool > > **Update** (T entity)
Asynchronously updates an existing entity in the data store.
- Task< Result< bool > > **Delete** (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.182.1 Member Function Documentation**6.182.1.1 Create()**

```
Task< Result< TableRelationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository.Create (
    TableRelationEntity entity ) [inline]
```

6.182.1.2 Delete()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository.Delete (
    int id ) [inline]
```

6.182.1.3 GetAll()

```
Task< Result< IEnumerable< TableRelationEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository.GetAll ( ) [inline]
```

6.182.1.4 GetById()

```
Task< Result< TableRelationEntity > > ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository.GetById (
    int id ) [inline]
```

6.182.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository.←
Update (
    TableRelationEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/TableRelationRepository.cs

6.183 ArandanoIRT_Backend.Infrastructure.Data.Thermaldata Class Reference

Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Data.Thermaldata:

ArandanoIRT_Backend.Infrastructure. Data.Thermaldata	
+	Idthermaldata
+	Thermalimagedata
+	Rgbimagedata
+	Recordedat
+	Plantid
+	Cropid
+	Crop
+	Plant
+	Plantobservation

Properties

- int **Idthermaldata** [get, set]
- string **Thermalimagedata** = null! [get, set]
- byte?[] **Rgbimagedata** [get, set]
- DateTime **Recordedat** [get, set]
- int? **Plantid** [get, set]
- int? **Cropid** [get, set]
- virtual ? **Crop Crop** [get, set]
- virtual ? **Plantdata Plant** [get, set]
- virtual ICollection< **Plantobservation** > **Plantobservation** = new List<**Plantobservation**>() [get, set]

6.183.1 Property Documentation

6.183.1.1 Crop

```
virtual ? Crop ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Crop [get], [set]
```

6.183.1.2 Cropid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Cropid [get], [set]
```

6.183.1.3 Idthermaldata

```
int ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Idthermaldata [get], [set]
```

6.183.1.4 Plant

```
virtual ? Plantdata ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Plant [get], [set]
```

6.183.1.5 Plantid

```
int? ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Plantid [get], [set]
```

6.183.1.6 Plantobservation

```
virtual ICollection<Plantobservation> ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.←  
Plantobservation = new List<Plantobservation>() [get], [set]
```

6.183.1.7 Recordedat

```
DateTime ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Recordedat [get], [set]
```

6.183.1.8 Rgbimagedata

```
byte? [ ] ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Rgbimagedata [get], [set]
```

6.183.1.9 Thermalimagedata

```
string ArandanoIRT_Backend.Infrastructure.Data.Thermaldata.Thermalimagedata = null! [get], [set]
```

The documentation for this class was generated from the following file:

- Infrastructure/Data/[Thermaldata.cs](#)

6.184 ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity Class Reference

Represents a single record containing thermal image data and corresponding RGB image data, captured at a specific time and potentially associated with a plant or crop.

Collaboration diagram for ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity:

ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity
+ IdThermalData
+ ThermalImageData
+ RgbImageData
+ RecordedAt
+ PlantId
+ CropId
+ ThermalDataEntity()

Public Member Functions

- [ThermalDataEntity](#) (int idThermalData, string thermalImageData, byte[] rgbiImageData, DateTime recordedAt, int? plantId, int? cropId)

Initializes a new instance of the ThermalDataEntity class.

Properties

- int [IdThermalData](#) [get]
Gets the unique identifier for this thermal data record.
- string [ThermalImageData](#) [get]
Gets the thermal image data, likely stored as a Base64 encoded string, file path, or URL.
- IReadOnlyList< byte > [RgblImageData](#) [get]
Gets the corresponding RGB (standard color) image data as a read-only list of bytes.
- DateTime [RecordedAt](#) [get]
Gets the date and time (usually UTC) when the image data was recorded or captured.
- int? [PlantId](#) [get]
Gets the unique identifier of the plant associated with this image data.
- int? [CropId](#) [get]
Gets the unique identifier of the crop associated with this image data.

6.184.1 Detailed Description

Represents a single record containing thermal image data and corresponding RGB image data, captured at a specific time and potentially associated with a plant or crop.

6.184.2 Constructor & Destructor Documentation

6.184.2.1 ThermalDataEntity()

```
ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.ThermalDataEntity (
    int idThermalData,
    string thermalImageData,
    byte[ ] rgblImageData,
    DateTime recordedAt,
    int? plantId,
    int? cropId ) [inline]
```

Initializes a new instance of the ThermalDataEntity class.

Parameters

<i>idThermalData</i>	The unique ID for this thermal data record (often generated by the database).
<i>thermalImageData</i>	The thermal image data (e.g., Base64 string or path).
<i>rgblImageData</i>	The corresponding RGB image data as a byte array.
<i>recordedAt</i>	The timestamp when the data was recorded.
<i>plantId</i>	The ID of the associated plant (nullable).
<i>cropId</i>	The ID of the associated crop (nullable).

Exceptions

<code>ArgumentNullException</code>	Thrown if <code>rgblImageData</code> is null.
<code>ArgumentException</code>	Thrown if <code>thermalImageData</code> is null or whitespace.

6.184.3 Property Documentation**6.184.3.1 CropId**

```
int? ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.CropId [get]
```

Gets the unique identifier of the crop associated with this image data.

Nullable if not linked to a specific crop.

6.184.3.2 IdThermalData

```
int ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.IdThermalData [get]
```

Gets the unique identifier for this thermal data record.

6.184.3.3 PlantId

```
int? ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.PlantId [get]
```

Gets the unique identifier of the plant associated with this image data.

Nullable if not linked to a specific plant.

6.184.3.4 RecordedAt

```
DateTime ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.RecordedAt [get]
```

Gets the date and time (usually UTC) when the image data was recorded or captured.

6.184.3.5 RgbImageData

```
IReadOnlyList<byte> ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.RgbImageData [get]
```

Gets the corresponding RGB (standard color) image data as a read-only list of bytes.

Provides read-only access to prevent external modification of the internal array.

6.184.3.6 ThermalImageData

```
string ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity.ThermalImageData [get]
```

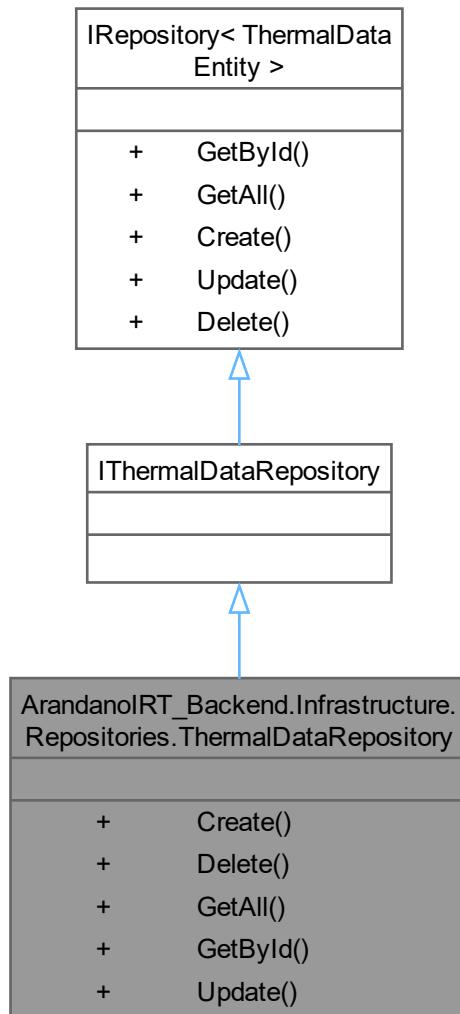
Gets the thermal image data, likely stored as a Base64 encoded string, file path, or URI.

The documentation for this class was generated from the following file:

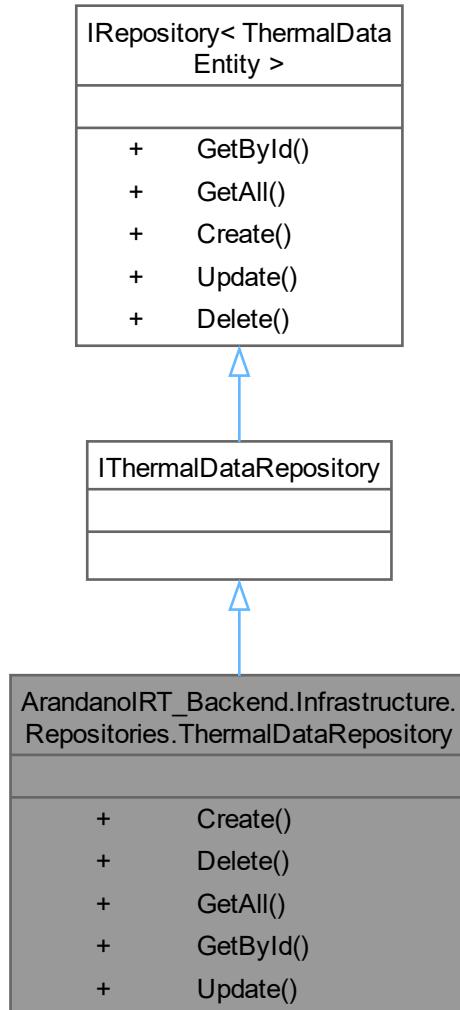
- Domain/Entities/[ThermalDataEntity.cs](#)

6.185 ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository Class Reference

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository:



Public Member Functions

- Task<`Result< ThermalDataEntity >`>> `Create (ThermalDataEntity entity)`
- Task<`Result< bool >`>> `Delete (int id)`
- Task<`Result< IEnumerable< ThermalDataEntity >`>>> `GetAll ()`
- Task<`Result< ThermalDataEntity >`>> `GetById (int id)`
- Task<`Result< bool >`>> `Update (ThermalDataEntity entity)`

Public Member Functions inherited from

[ArandanoIRT_Backend.Domain.IRepositories IRepository< ThermalDataEntity >](#)

- Task< [Result< T >](#) > [GetById](#) (int id)
Asynchronously retrieves an entity by its unique identifier.
- Task< [Result< IEnumerable< T >](#) > > [GetAll](#) ()
Asynchronously retrieves all entities of type T .
- Task< [Result< T >](#) > [Create](#) (T entity)
Asynchronously creates a new entity in the data store.
- Task< [Result< bool >](#) > [Update](#) (T entity)
Asynchronously updates an existing entity in the data store.
- Task< [Result< bool >](#) > [Delete](#) (int id)
Asynchronously deletes an entity from the data store by its unique identifier.

6.185.1 Member Function Documentation

6.185.1.1 Create()

```
Task< Result< ThermalDataEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository.Create (
    ThermalDataEntity entity ) [inline]
```

6.185.1.2 Delete()

```
Task< Result< bool > > > ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository.Delete (
    int id ) [inline]
```

6.185.1.3 GetAll()

```
Task< Result< IEnumerable< ThermalDataEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository.GetAll ( ) [inline]
```

6.185.1.4 GetById()

```
Task< Result< ThermalDataEntity > > > ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository.GetById (
    int id ) [inline]
```

6.185.1.5 Update()

```
Task< Result< bool > > ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository.←  
Update (   
    ThermalDataEntity entity ) [inline]
```

The documentation for this class was generated from the following file:

- Infrastructure/Repositories/[ThermalDataRepository.cs](#)

6.186 ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto Class Reference

Represents the response containing authentication tokens (access and refresh).

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto:

ArandanoIRT_Backend.Application. DTOs.Auth.TokenResponseDto
+ AccessToken + RefreshToken + AccessTokenExpiration

Properties

- required string **AccessToken** [get, set]
Gets or sets the access token used for authorizing API requests.
- required string **RefreshToken** [get, set]
Gets or sets the refresh token used to obtain a new access token without requiring the user to re-authenticate.
- DateTime **AccessTokenExpiration** [get, set]
Gets or sets the exact date and time (usually UTC) when the AccessToken expires.

6.186.1 Detailed Description

Represents the response containing authentication tokens (access and refresh).

6.186.2 Property Documentation

6.186.2.1 AccessToken

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto.AccessToken [get],  
[set]
```

Gets or sets the access token used for authorizing API requests.

This token typically has a short lifespan.

6.186.2.2 AccessTokenExpiration

```
DateTime ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto.AccessTokenExpiration [get],  
[set]
```

Gets or sets the exact date and time (usually UTC) when the AccessToken expires.

6.186.2.3 RefreshToken

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto.RefreshToken [get],  
[set]
```

Gets or sets the refresh token used to obtain a new access token without requiring the user to re-authenticate.

This token typically has a longer lifespan than the access token.

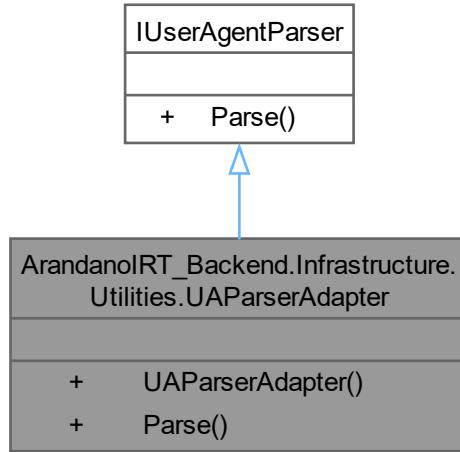
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/[TokenResponseDto.cs](#)

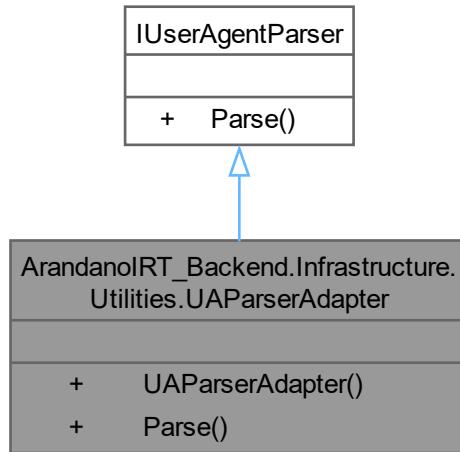
6.187 ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter Class Reference

Implements the IUserAgentParser interface by adapting the UAParser library (uap-csharp) to parse User-Agent strings into the application's Domain.ValueObjects.ClientInfo structure.

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter:



Public Member Functions

- `UAParserAdapter (ILogger< UAParserAdapter > logger)`

Initializes a new instance of the UAParserAdapter class.

- [Domain.ValueObjects.ClientInfo Parse \(string? userAgentString\)](#)

Parses the user agent string and returns a ClientInfo object.

Parameters

userAgentString	
-----------------	--

Returns

6.187.1 Detailed Description

Implements the IUserAgentParser interface by adapting the UAParser library (uap-csharp) to parse User-Agent strings into the application's Domain.ValueObjects.ClientInfo structure.

6.187.2 Constructor & Destructor Documentation

6.187.2.1 UAParserAdapter()

```
ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter.UAParserAdapter (
    ILogger< UAParserAdapter > logger ) [inline]
```

Initializes a new instance of the UAParserAdapter class.

Parameters

logger	The logger instance.
--------	----------------------

Exceptions

ArgumentNullException	Thrown if logger is null.
-----------------------	---------------------------

6.187.3 Member Function Documentation

6.187.3.1 Parse()

```
Domain.ValueObjects.ClientInfo ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter.Parse (
    string? userAgentString ) [inline]
```

Parses the user agent string and returns a ClientInfo object.

Parameters

<i>userAgentString</i>	<input type="button" value=""/>
------------------------	---------------------------------

Returns

Uses the UAParser library to parse the input string. Delegates mapping to a helper method. Returns default values if input is null/empty or parsing fails.

Implements [ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser](#).

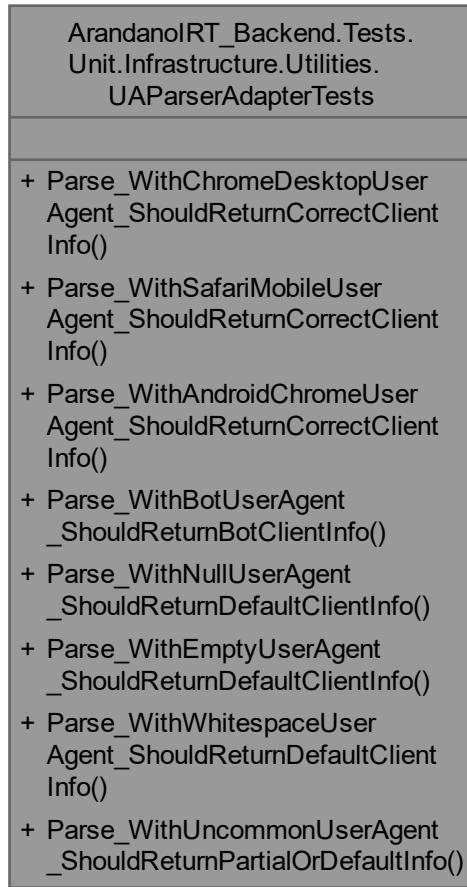
The documentation for this class was generated from the following file:

- Infrastructure/Utilities/[UAParserAdapter.cs](#)

6.188 ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests Class Reference

Contains unit tests for the UAParserAdapter class.

Collaboration diagram for ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests:



Public Member Functions

- void [Parse_WithChromeDesktopUserAgent_ShouldReturnCorrectClientInfo \(\)](#)
Tests parsing a typical User-Agent string from Chrome on Windows desktop.
- void [Parse_WithSafariMobileUserAgent_ShouldReturnCorrectClientInfo \(\)](#)
Tests parsing a typical User-Agent string from Safari on an iPhone.
- void [Parse_WithAndroidChromeUserAgent_ShouldReturnCorrectClientInfo \(\)](#)
Tests parsing a typical User-Agent string from Chrome on Android (Samsung device).
- void [Parse_WithBotUserAgent_ShouldReturnBotClientInfo \(\)](#)
Tests parsing a User-Agent string from a known bot (Googlebot).
- void [Parse_WithNullUserAgent_ShouldReturnDefaultClientInfo \(\)](#)
Tests parsing when the input User-Agent string is null.
- void [Parse_WithEmptyUserAgent_ShouldReturnDefaultClientInfo \(\)](#)

Tests parsing when the input User-Agent string is empty.

- void [Parse_WithWhitespaceUserAgent_ShouldReturnDefaultClientInfo \(\)](#)

Tests parsing when the input User-Agent string consists only of whitespace.

- void [Parse_WithUncommonUserAgent_ShouldReturnPartialOrDefaultInfo \(\)](#)

Tests parsing an uncommon or potentially unrecognized User-Agent string.

6.188.1 Detailed Description

Contains unit tests for the UAParserAdapter class.

6.188.2 Member Function Documentation

6.188.2.1 Parse_WithAndroidChromeUserAgent_ShouldReturnCorrectClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_With←
AndroidChromeUserAgent_ShouldReturnCorrectClientInfo ( ) [inline]
```

Tests parsing a typical User-Agent string from Chrome on Android (Samsung device).

Verifies the extracted browser, OS, and device information, and the formatted string output.

6.188.2.2 Parse_WithBotUserAgent_ShouldReturnBotClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_WithBot←
UserAgent_ShouldReturnBotClientInfo ( ) [inline]
```

Tests parsing a User-Agent string from a known bot (Googlebot).

Verifies the extracted bot name and device type ("Spider"), and the formatted string output.

6.188.2.3 Parse_WithChromeDesktopUserAgent_ShouldReturnCorrectClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_With←
ChromeDesktopUserAgent_ShouldReturnCorrectClientInfo ( ) [inline]
```

Tests parsing a typical User-Agent string from Chrome on Windows desktop.

Verifies the extracted browser, OS, and device information, and the formatted string output.

6.188.2.4 Parse_WithEmptyUserAgent_ShouldReturnDefaultClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_With←
EmptyUserAgent_ShouldReturnDefaultClientInfo ( ) [inline]
```

Tests parsing when the input User-Agent string is empty.

Expects a default ClientInfo object representing unknown information.

6.188.2.5 Parse_WithNullUserAgent_ShouldReturnDefaultClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_WithNullUserAgent_ShouldReturnDefaultClientInfo () [inline]
```

Tests parsing when the input User-Agent string is null.

Expects a default ClientInfo object representing unknown information.

6.188.2.6 Parse_WithSafariMobileUserAgent_ShouldReturnCorrectClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_WithSafariMobileUserAgent_ShouldReturnCorrectClientInfo () [inline]
```

Tests parsing a typical User-Agent string from Safari on an iPhone.

Verifies the extracted browser, OS, and device information, and the formatted string output.

6.188.2.7 Parse_WithUncommonUserAgent_ShouldReturnPartialOrDefaultInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_WithUncommonUserAgent_ShouldReturnPartialOrDefaultInfo () [inline]
```

Tests parsing an uncommon or potentially unrecognized User-Agent string.

Expects the parser to return default/fallback values ("Other") for unrecognized components.

6.188.2.8 Parse_WithWhitespaceUserAgent_ShouldReturnDefaultClientInfo()

```
void ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests.Parse_WithWhitespaceUserAgent_ShouldReturnDefaultClientInfo () [inline]
```

Tests parsing when the input User-Agent string consists only of whitespace.

Expects a default ClientInfo object representing unknown information.

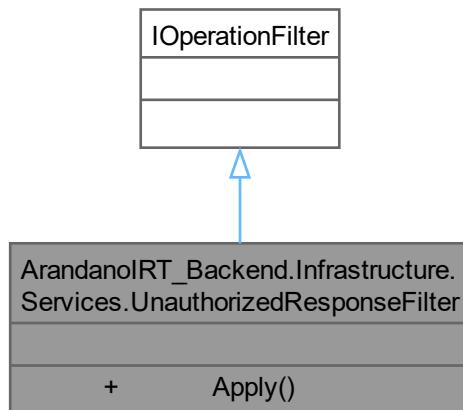
The documentation for this class was generated from the following file:

- Tests/Unit/Infrastructure/Utilities/[UAParserAdapterTests.cs](#)

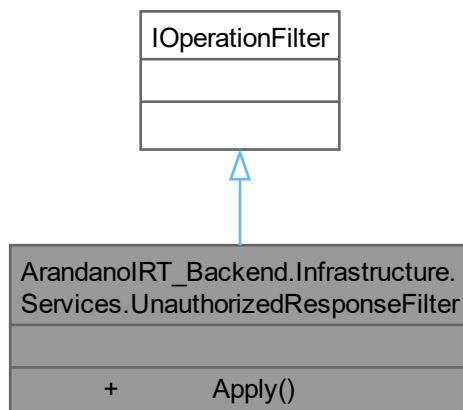
6.189 ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter Class Reference

An Swagger operation filter that adds the 401 Unauthorized response type to operations that are protected by an Authorize attribute (identified by name).

Inheritance diagram for ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter:



Collaboration diagram for ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter:



Public Member Functions

- void [Apply](#) (OpenApiOperation operation, OperationFilterContext context)
Applies the filter to the specified OpenAPI operation.

6.189.1 Detailed Description

An Swagger operation filter that adds the 401 Unauthorized response type to operations that are protected by an Authorize attribute (identified by name).

This filter identifies the Authorize attribute by comparing the attribute's type name as a string.

6.189.2 Member Function Documentation

6.189.2.1 Apply()

```
void ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter.Apply (
    OpenApiOperation operation,
    OperationFilterContext context ) [inline]
```

Applies the filter to the specified OpenAPI operation.

Parameters

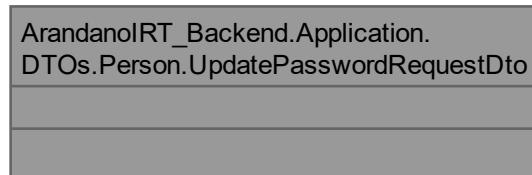
<i>operation</i>	The OpenAPI operation to modify.
<i>context</i>	The context providing information about the operation and method attributes.

The documentation for this class was generated from the following file:

- Infrastructure/Services/[FiltersSwaggerService.cs](#)

6.190 ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto Class Reference

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto:



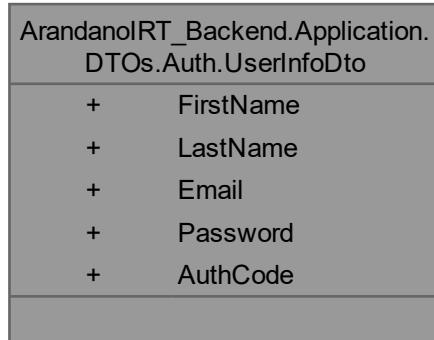
The documentation for this class was generated from the following file:

- Application/DTOs/Person/[UpdatePasswordRequestDto.cs](#)

6.191 ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto Class Reference

Represents the personal and credential information for a standard user during registration.

Collaboration diagram for ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto:



Properties

- required string `FirstName` [get, set]
Gets or sets the user's first name.
- required string `LastName` [get, set]
Gets or sets the user's last name.
- required string `Email` [get, set]
Gets or sets the user's email address, typically used for login.
- required string `Password` [get, set]
Gets or sets the user's initial password for the account.
- required string `AuthCode` [get, set]
Gets or sets the authorization code required for user registration.

6.191.1 Detailed Description

Represents the personal and credential information for a standard user during registration.

6.191.2 Property Documentation

6.191.2.1 AuthCode

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto.AuthCode [get], [set]
```

Gets or sets the authorization code required for user registration.

This code might be provided by an administrator or through an invitation process.

string

6.191.2.2 Email

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto.Email [get], [set]
```

Gets or sets the user's email address, typically used for login.

6.191.2.3 FirstName

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto.FirstName [get], [set]
```

Gets or sets the user's first name.

string

6.191.2.4 LastName

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto.LastName [get], [set]
```

Gets or sets the user's last name.

string

6.191.2.5 Password

```
required string ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto.Password [get], [set]
```

Gets or sets the user's initial password for the account.

///

string

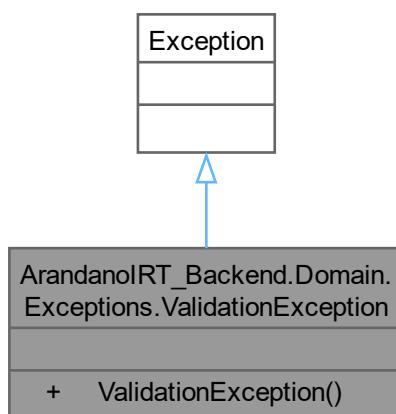
The documentation for this class was generated from the following file:

- Application/DTOs/Auth/RegisterUserRequestDto.cs

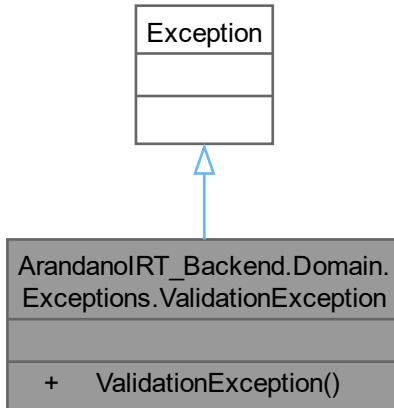
6.192 ArandanoIRT_Backend.Domain.Exceptions.ValidationException Class Reference

Represents a general error that occurs when input data or an object's state fails one or more validation checks.

Inheritance diagram for ArandanoIRT_Backend.Domain.Exceptions.ValidationException:



Collaboration diagram for ArandanoIRT_Backend.Domain.Exceptions.ValidationException:



Public Member Functions

- [ValidationException \(string message\)](#)

Initializes a new instance of the ValidationException class with a specified error message that describes the validation failure.

6.192.1 Detailed Description

Represents a general error that occurs when input data or an object's state fails one or more validation checks.

This exception inherits from the base `Exception` class. Consider using more specific exceptions (e.g., `ArgumentException`, `ArgumentNullException`, `ArgumentOutOfRangeException`) or custom exceptions inheriting from this one for more detailed error handling where appropriate.

6.192.2 Constructor & Destructor Documentation

6.192.2.1 ValidationException()

```
ArandanoIRT_Backend.Domain.Exceptions.ValidationException.ValidationException (
    string message ) [inline]
```

Initializes a new instance of the ValidationException class with a specified error message that describes the validation failure.

Parameters

<i>message</i>	The message that describes the validation error.
----------------	--

The documentation for this class was generated from the following file:

- Domain/Exceptions/[ValidationException.cs](#)

Chapter 7

File Documentation

7.1 Application/DTOs/Auth/ChangePasswordRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.ChangePasswordRequestDto](#)
Represents the request data for changing a user's password using a verification code.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.2 Application/DTOs/Auth/ForgotPasswordRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.ForgotPasswordRequestDto](#)
Represents the request data for initiating the password recovery process.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.3 Application/DTOs/Auth/HelpRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.HelpRequestDto](#)

Represents the data submitted through a help or support request form.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.4 Application/DTOs/Auth/LoginRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.LoginRequestDto](#)

Represents the request data for a user login attempt.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.5 Application/DTOs/Auth/LoginResponseDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.LoginResponseDto](#)

Represents the data returned after a successful user authentication.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.6 Application/DTOs/Auth/LoginSuccessPayload.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.LoginSuccessPayload](#)

Represents the data returned after a successful login, including user details, access token, and the refresh token string.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.7 Application/DTOs/Auth/RegisterAdminRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.AdminInfoDto](#)
Represents the personal and credential information for an administrator user.
- class [ArandanoIRT_Backend.Application.DTOs.Auth.RegisterAdminRequestDto](#)

Represents the request data for registering a new administrator account along with associated crop information.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.8 Application/DTOs/Auth/RegisterUserRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.UserInfoDto](#)
Represents the personal and credential information for a standard user during registration.
- class [ArandanoIRT_Backend.Application.DTOs.Auth.RegisterUserRequestDto](#)
Represents the request data for registering a new standard user account.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.9 Application/DTOs/Auth/TokenResponseDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Auth.TokenResponseDto](#)
Represents the response containing authentication tokens (access and refresh).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Auth](#)

7.10 Application/DTOs/Objects/CropInfoDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Objetcts.CropInfoDto](#)
Represents basic information about a specific crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Objetcts](#)

7.11 Application/DTOs/Person/ProfileResponseDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Person.ProfileResponseDto](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Person](#)

7.12 Application/DTOs/Person/UpdatePasswordRequestDto.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.DTOs.Person.UpdatePasswordRequestDto](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.DTOs](#)
- namespace [ArandanoIRT_Backend.Application.DTOs.Person](#)

7.13 Application/Interfaces/Auditing/AuditMetadata.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Auditing](#)

Functions

- record [ArandanoIRT_Backend.Application.Interfaces.Auditing.AuditMetadata](#) (int? UserId, string? IpAddress, string? UserAgent, DateTime PerformedAt)
Represents metadata associated with an audited event or action.

7.14 Application/Interfaces/Auditing/IAuditEntryGenerator.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditEntryGenerator](#)

Defines a contract for components responsible for generating audit log entries based on entity changes tracked by Entity Framework Core.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Auditing](#)

7.15 Application/Interfaces/Auditing/IAuditHelperService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Auditing.IAuditHelperService](#)

Defines a contract for a helper service providing common utility functions used during the audit logging process, particularly with Entity Framework Core entities.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Auditing](#)

7.16 Application/Interfaces/IServices/IAuthPasswordService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthPasswordService](#)

Defines the contract for password management operations like reset requests and execution.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.IServices](#)

7.17 Application/Interfaces/IServices/IAuthRegistrationService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthRegistrationService](#)

Defines the contract for user registration operations.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.IServices](#)

7.18 Application/Interfaces/IServices/IAuthSessionService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.IServices.IAuthSessionService](#)
Defines the contract for user session management operations, including login, token refresh, and logout.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.IServices](#)

7.19 Application/Interfaces/IServices/IHelpRequestService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.IServices.IHelpRequestService](#)
Defines the contract for handling help request operations.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.IServices](#)

7.20 Application/Interfaces/Utilities/IAuthUtilities.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IAuthUtilities](#)
Defines the contract for shared utility functions used across authentication-related services.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.21 Application/Interfaces/Utilities/ICaptchaService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.ICaptchaService](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.22 Application/Interfaces/Utilities/IDateTimeProvider.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IDateTimeProvider](#)
Provides the current time, abstracting away direct DateTime usage for testability.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.23 Application/Interfaces/Utilities/IEmailService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IEmailService](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.24 Application/Interfaces/Utilities/IPasswordHasher.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IPasswordHasher](#)
Service interface for hashing and verifying passwords.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.25 Application/Interfaces/Utilities/IRequestContextAccessor.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRequestContextAccessor](#)
Defines a contract for accessing information related to the current HTTP request context, such as client details and authenticated user information.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.26 Application/Interfaces/Utilities/IRsaService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IRsaService](#)
Service interface for RSA encryption and decryption operations.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.27 Application/Interfaces/Utilities/ISanitizerService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.ISanitizerService](#)

Defines the contract for a service that sanitizes input strings to remove potentially harmful content, such as HTML tags.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.28 Application/Interfaces/Utilities/IUserAgentParser.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Application.Interfaces.Utilities.IUserAgentParser](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces](#)
- namespace [ArandanoIRT_Backend.Application.Interfaces.Utilities](#)

7.29 Application/Mappings/AuthMapping.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Mappings.AuthMapping](#)

Provides static mapping methods for authentication related DTOs and Entities.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Mappings](#)

7.30 Application/Services/AuthPasswordService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Services.AuthPasswordService](#)
Implements services for handling password reset requests and password changes using tokens.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Services](#)

7.31 Application/Services/AuthRegistrationService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Services.AuthRegistrationService](#)
Implements services for registering new administrators and regular users.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Services](#)

7.32 Application/Services/AuthSessionService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Services.AuthSessionService](#)
Implements services for user session management including login, token refresh, and logout.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Services](#)

7.33 Application/Services/HelpRequestService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Services.HelpRequestService](#)
Implements services for handling user help requests.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Services](#)

7.34 Application/Utilities/AuthUtilities.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Utilities.AuthUtilities](#)
Implements shared utility functions for authentication services.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Utilities](#)

7.35 Application/Utilities/EmailSenderUtility.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Utilities.EmailSenderUtility](#)
Provides utility methods for sending emails using SMTP settings from configuration.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Utilities](#)

7.36 Application/Utilities/EmailValidatorUtility.cs File Reference

Classes

- class [ArandanoIRT_Backend.Application.Utilities.EmailValidatorUtility](#)

Provides static utility methods for validating email addresses, including syntax checks and DNS record verification (MX/A).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Application](#)
- namespace [ArandanoIRT_Backend.Application.Utilities](#)

7.37 Domain/Entities/AuditCropEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditCropEntity](#)

Represents a single audit trail record tracking changes specifically related to crop data.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.38 Domain/Entities/AuditDataTableEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditDataTableEntity](#)

Represents a generic audit trail record tracking changes made to specific columns within identified database tables, associated with a particular crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.39 Domain/Entities/AuditDeviceEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditDeviceEntity](#)

Represents a single audit trail record tracking changes specifically related to device entities or data, typically associated with a particular crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.40 Domain/Entities/AuditPersonEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditPersonEntity](#)

Represents a single audit trail record tracking changes specifically related to person (user) entities or data, associated with a particular crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.41 Domain/Entities/AuditSensitiveDataEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditSensitiveDataEntity](#)

Represents a single audit trail record specifically tracking actions involving access to, or operations on, sensitive data fields within specified database tables, associated with a crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.42 Domain/Entities/AuditSystemTableEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.AuditSystemTableEntity](#)

Represents a generic audit trail record tracking changes made to specific columns within identified system-level or configuration database tables, potentially associated with a crop context.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.43 Domain/Entities/ChangePasswordEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.ChangePasswordEntity](#)

Represents a record containing a temporary token generated to allow a user to securely reset their password.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.44 Domain/Entities/CropEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.CropEntity](#)

Represents an agricultural crop entity, containing its identification, name, location, and administrative details.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.45 Domain/Entities/CropInvitationEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.CropInvitationEntity](#)
Represents an invitation code record used to grant access or link a new user to a specific crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.46 Domain/Entities/DeviceActivationEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.DeviceActivationEntity](#)
Represents a record containing an activation code used to register, authorize, or link a physical device to the system.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.47 Domain/Entities/DeviceDataEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.DeviceDataEntity](#)
Represents a data-collecting device, containing its identification, description, configuration settings, registration details, and associations (crop, plant).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.48 Domain/Entities/DeviceLogEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.DeviceLogEntity](#)
Represents a single log entry generated by, or related to, a specific device.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.49 Domain/Entities/DeviceTokenEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.DeviceTokenEntity](#)
Represents a pair of authentication tokens (access and refresh) issued specifically for a device, along with metadata about the token's lifecycle and the device context.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.50 Domain/Entities/FailedLoginAttemptEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.FailedLoginAttemptEntity](#)
Represents a record logging a single failed login attempt against a specific user account.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.51 Domain/Entities/PersonEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.PersonEntity](#)

Represents a user account in the system, containing personal details, credentials (typically hashed password), status flags, timestamps, and associations like the assigned crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.52 Domain/Entities/PlantDataEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.PlantDataEntity](#)

Represents an individual plant within a crop, containing its identification, name, status, registration details, and association with a crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.53 Domain/Entities/PlantObservationEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.PlantObservationEntity](#)

Represents a single observation record made about a specific plant, detailing visual characteristics, subjective ratings, notes, and potentially linked sensor data identifiers.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.54 Domain/Entities/PlantStateHistoryEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.PlantStateHistoryEntity](#)

Represents a historical record documenting a change in a plant's status at a specific point in time, including who made the change.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.55 Domain/Entities/RefreshTokenEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.RefreshTokenEntity](#)

Represents a refresh token used to obtain new access tokens, extending user sessions.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.56 Domain/Entities/SensorDataEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.SensorDataEntity](#)

Represents a single snapshot of environmental sensor readings (temperature, humidity, light intensity) recorded at a specific time.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.57 Domain/Entities/StatusEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.StatusEntity](#)

Represents a defined status value (e.g., "Active", "Pending", "Used") that can be applied to various entities within the application.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.58 Domain/Entities/TableRelationEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.TableRelationEntity](#)

Represents a mapping between a unique identifier and a database table name.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.59 Domain/Entities/ThermalDataEntity.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Entities.ThermalDataEntity](#)

Represents a single record containing thermal image data and corresponding RGB image data, captured at a specific time and potentially associated with a plant or crop.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Entities](#)

7.60 Domain/Exceptions/BusinessRuleViolationException.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Exceptions.BusinessRuleViolationException](#)
Represents an error that occurs when an operation violates a specific business rule or constraint.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Exceptions](#)

7.61 Domain/Exceptions/DatabaseUpdateException.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Exceptions.DatabaseUpdateException](#)
Represents an error that occurs when a database update operation (e.g., insert, update, delete) fails unexpectedly or results in an inconsistent state.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Exceptions](#)

7.62 Domain/Exceptions/NotFoundException.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Exceptions.NotFoundException](#)
Represents an error that occurs when a requested resource, entity, or record could not be found in the system or data store.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Exceptions](#)

7.63 Domain/Exceptions/ValidationException.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.Exceptions.ValidationException](#)

Represents a general error that occurs when input data or an object's state fails one or more validation checks.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.Exceptions](#)

7.64 Domain/IRepositories/IAuditCropRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditCropRepository](#)

Defines the repository contract specifically for managing AuditCropEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.65 Domain/IRepositories/IAuditDataTableRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditDataTableRepository](#)

Defines the repository contract specifically for managing AuditDataTableEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.66 Domain/IRepositories/IAuditDeviceRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditDeviceRepository](#)

Defines the repository contract specifically for managing AuditDeviceEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.67 Domain/IRepositories/IAuditPersonRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditPersonRepository](#)

Defines the repository contract specifically for managing AuditPersonEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.68 Domain/IRepositories/IAuditSensitiveDataRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditSensitiveDataRepository](#)

Defines the repository contract specifically for managing AuditSensitiveDataEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.69 Domain/IRepositories/IAuditSystemTableRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IAuditSystemTableRepository](#)

Defines the repository contract specifically for managing AuditSystemTableEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.70 Domain/IRepositories/IChangePasswordRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IChangePasswordRepository](#)

Defines the repository contract specifically for managing ChangePasswordEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.71 Domain/IRepositories/ICropInvitationRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.ICropInvitationRepository](#)

Defines the repository contract specifically for managing CropInvitationEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.72 Domain/IRepositories/ICropRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.ICropRepository](#)

Defines the repository contract specifically for managing CropEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.73 Domain/IRepositories/IDeviceActivationRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IDeviceActivationRepository](#)

Defines the repository contract specifically for managing DeviceActivationEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.74 Domain/IRepositories/IDeviceDataRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IDeviceDataRepository](#)

Defines the repository contract specifically for managing DeviceDataEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.75 Domain/IRepositories/IDeviceLogRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IDeviceLogRepository](#)

Defines the repository contract specifically for managing DeviceLogEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.76 Domain/IRepositories/IDeviceTokenRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IDeviceTokenRepository](#)

Defines the repository contract specifically for managing DeviceTokenEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.77 Domain/IRepositories/IFailedLoginAttemptRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IFailedLoginAttemptRepository](#)

Defines the repository contract specifically for managing FailedLoginAttemptEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.78 Domain/IRepositories/IPersonRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IPersonRepository](#)

Defines the repository contract specifically for managing PersonEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.79 Domain/IRepositories/IPlantDataRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IPlantDataRepository](#)

Defines the repository contract specifically for managing PlantDataEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.80 Domain/IRepositories/IPlantObservationRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IPlantObservationRepository](#)

Defines the repository contract specifically for managing PlantObservationEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.81 Domain/IRepositories/IPlantStateHistoryRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IPlantStateHistoryRepository](#)

Defines the repository contract specifically for managing PlantStateHistoryEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.82 Domain/IRepositories/IRefreshTokenRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IRefreshTokenRepository](#)

Defines the repository contract specifically for managing RefreshTokenEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.83 Domain/IRepositories/IRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IRepository< T >](#)

Defines a generic contract for repository operations for a specific entity type T .

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.84 Domain/IRepositories/ISensorDataRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.ISensorDataRepository](#)

Defines the repository contract specifically for managing SensorDataEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.85 Domain/IRepositories/IStatusRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IStatusRepository](#)

Defines the repository contract specifically for managing StatusEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.86 Domain/IRepositories/ITableRelationRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.ITableRelationRepository](#)

Defines the repository contract specifically for managing TableRelationEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.87 Domain/IRepositories/IThermalDataRepository.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Domain.IRepositories.IThermalDataRepository](#)

Defines the repository contract specifically for managing ThermalDataEntity instances.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.IRepositories](#)

7.88 Domain/ValueObjects/ClientInfo.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.ValueObjects](#)

Functions

- record [ArandanoIRT_Backend.Domain.ValueObjects.ClientInfo](#) (string UserAgentFamily="Unknown", string UserAgentVersion="", string OSFamily="Unknown OS", string OSVersion="", string DeviceFamily="Unknown Device")

Holds parsed information about the client (e.g., browser, OS, device) making a request, typically derived from the User-Agent string.

7.89 Domain/ValueObjects/ResultPattern.cs File Reference

Classes

- class [ArandanoIRT_Backend.Domain.ValueObjects.Result< T >](#)

Represents the outcome of an operation, indicating either success or failure.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Domain](#)
- namespace [ArandanoIRT_Backend.Domain.ValueObjects](#)

7.90 Infrastructure/Attributes/SanitizeHtmlAttribute.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Attributes.SanitizeHtmlAttribute](#)

Attribute used to mark string properties on DTOs or models that should have HTML tags removed by the SanitizationActionFilter.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Attributes](#)

7.91 Infrastructure/Config/AuthConfig.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Config.AuthConfig](#)

Provides extension methods for configuring authentication services for the application.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.92 Infrastructure/Config/CORSConfig.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Config.CORSConfig](#)

Provides extension methods for configuring Cross-Origin Resource Sharing (CORS) policies for the application.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.93 Infrastructure/Config/DependencyInjection.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Config.DependencyInjection](#)

Provides extension methods for configuring dependency injection services for the Infrastructure and Application layers.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.94 Infrastructure/Config/JsonConfig.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Config.JsonConfig](#)

Provides extension methods for configuring JSON serialization options for the application, specifically for controllers using System.Text.Json.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.95 Infrastructure/Config/LoggingConfig.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Config.LoggingConfig](#)

Provides extension methods for configuring application logging using Serilog.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.96 Infrastructure/Config/RateLimitConfig.cs File Reference

Classes

- class **ArandanoIRT_Backend.Infrastructure.Config.RateLimitConfig**

Provides extension methods for configuring request rate limiting policies for the application using Microsoft.AspNetCore.← RateLimiting.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.97 Infrastructure/Config/ResponseCompressionConfig.cs File Reference

Classes

- class **ArandanoIRT_Backend.Infrastructure.Config.ResponseCompressionConfig**

Provides extension methods for configuring response compression services for the ASP.NET Core application pipeline.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.98 Infrastructure/Config/SwaggerConfig.cs File Reference

Classes

- class **ArandanoIRT_Backend.Infrastructure.Config.SwaggerConfig**

Provides extension methods for configuring Swashbuckle Swagger/OpenAPI generation services.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Config](#)

7.99 Infrastructure/Data/ApplicationDbContext.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.ApplicationDbContext](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.100 Infrastructure/Data/Auditcrop.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditcrop](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.101 Infrastructure/Data/Auditdatatable.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditdatatable](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.102 Infrastructure/Data/Auditdevice.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditdevice](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.103 Infrastructure/Data/Auditperson.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditperson](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.104 Infrastructure/Data/Auditsensitizeddata.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditsensitizeddata](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.105 Infrastructure/Data/Auditsystemtable.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Auditsystemtable](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.106 Infrastructure/Data/Changepassword.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Changepassword](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.107 Infrastructure/Data/Crop.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Crop](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.108 Infrastructure/Data/Cropinvitation.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Cropinvitation](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.109 Infrastructure/Data/Deviceactivation.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Deviceactivation](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.110 Infrastructure/Data/Devidedata.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Devidedata](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.111 Infrastructure/Data/DeviceLog.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.DeviceLog](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.112 Infrastructure/Data/DeviceToken.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Devicetoken](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.113 Infrastructure/Data/Failedloginattempt.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Failedloginattempt](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.114 Infrastructure/Data/Person.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Person](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.115 Infrastructure/Data/Plantdata.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Plantdata](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.116 Infrastructure/Data/Plantobservation.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Plantobservation](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.117 Infrastructure/Data/Plantstatehistory.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Plantstatehistory](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.118 Infrastructure/Data/RefreshToken.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.RefreshToken](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.119 Infrastructure/Data/Sensordata.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Sensordata](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.120 Infrastructure/Data/Status.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Status](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.121 Infrastructure/Data/Tablerelation.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Tablerelation](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.122 Infrastructure/Data/Thermaldata.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Data.Thermaldata](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Data](#)

7.123 Infrastructure/Filters/SanitizationActionFilter.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Filters.SanitizationActionFilter](#)

An ASP.NET Core action filter that automatically sanitizes string properties marked with the SanitizeHtmlAttribute on action method arguments (typically DTOs).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Filters](#)

7.124 Infrastructure/Interfaces/IServices/ICacheService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ICacheService](#)
Defines a contract for a generic caching service.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Interfaces](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices](#)

7.125 Infrastructure/Interfaces/IServices/ITokenService.cs File Reference

Classes

- interface [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices.ITokenService](#)
Defines a contract for services responsible for handling the lifecycle of authentication tokens, including the generation, storage, validation, refresh, and revocation of JWT access tokens and persistent refresh tokens.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Interfaces](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Interfaces.IServices](#)

7.126 Infrastructure/Persistence/Auditing/AuditCropGenerator.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditCropGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries specifically for changes made to Crop entities.

Namespaces

- namespace ArandanoIRT_Backend
- namespace ArandanoIRT_Backend.Infrastructure
- namespace ArandanoIRT_Backend.Infrastructure.Persistence
- namespace ArandanoIRT_Backend.Infrastructure.Persistence.Auditing

7.127 Infrastructure/Persistence/Auditing/AuditDataTableGenerator.cs File Reference

Classes

- class ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDataTableGenerator

Implements IAuditEntryGenerator to create audit trail entries in the Auditdatatable for changes made to various data entities like Sensordata, Thermaldata, and Plantdata.

Namespaces

- namespace ArandanoIRT_Backend
- namespace ArandanoIRT_Backend.Infrastructure
- namespace ArandanoIRT_Backend.Infrastructure.Persistence
- namespace ArandanoIRT_Backend.Infrastructure.Persistence.Auditing

7.128 Infrastructure/Persistence/Auditing/AuditDeviceGenerator.cs File Reference

Classes

- class ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditDeviceGenerator

Implements IAuditEntryGenerator to create audit trail entries in the Auditdevice table for changes made to Devicedata entities.

Namespaces

- namespace ArandanoIRT_Backend
- namespace ArandanoIRT_Backend.Infrastructure
- namespace ArandanoIRT_Backend.Infrastructure.Persistence
- namespace ArandanoIRT_Backend.Infrastructure.Persistence.Auditing

7.129 Infrastructure/Persistence/Auditing/AuditHelperService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService](#)
Implements IAuditHelperService providing utility methods commonly used during the generation of audit log entries, such as primary key retrieval, foreign key retrieval (CropId), and property value serialization.
- class [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditHelperService.Person](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing](#)

7.130 Infrastructure/Persistence/Auditing/AuditPersonGenerator.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditPersonGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditperson table for changes made to Person entities.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing](#)

7.131 Infrastructure/Persistence/Auditing/AuditSensitiveDataGenerator.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing](#)

7.132 Infrastructure/Persistence/Auditing/AuditSystemTableGenerator.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing.AuditSystemTableGenerator](#)
Implements IAuditEntryGenerator to create audit trail entries in the Auditsystemtable for changes made to specific "system" or related data entities like Plantstatehistory and Plantobservation.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence.Auditing](#)

7.133 Infrastructure/Persistence/Interceptors/AuditSaveChangesInterceptor.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors.AuditSaveChangesInterceptor](#)
An SaveChangesInterceptor that automatically generates audit log entries for entity changes detected by the DbContext's ChangeTracker before saving.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Persistence.Interceptors](#)

7.134 Infrastructure/Repositories/AuditCropRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditCropRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.135 Infrastructure/Repositories/AuditDataTableRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditDataTableRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.136 Infrastructure/Repositories/AuditDeviceRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditDeviceRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.137 Infrastructure/Repositories/AuditPersonRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditPersonRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.138 Infrastructure/Repositories/AuditSensitiveDataRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditSensitiveDataRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.139 Infrastructure/Repositories/AuditSystemTableRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.AuditSystemTableRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.140 Infrastructure/Repositories/ChangePasswordRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.141 Infrastructure/Repositories/CropInvitationRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.142 Infrastructure/Repositories/CropRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.143 Infrastructure/Repositories/DeviceActivationRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.DeviceActivationRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.144 Infrastructure/Repositories/DeviceDataRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.DeviceDataRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.145 Infrastructure/Repositories/DeviceLogRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.DeviceLogRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.146 Infrastructure/Repositories/DeviceTokenRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.DeviceTokenRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.147 Infrastructure/Repositories/FailedLoginAttemptRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.148 Infrastructure/Repositories/PersonRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.149 Infrastructure/Repositories/PlantDataRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.PlantDataRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.150 Infrastructure/Repositories/PlantObservationRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.PlantObservationRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.151 Infrastructure/Repositories/PlantStateHistoryRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.PlantStateHistoryRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.152 Infrastructure/Repositories/RefreshTokenRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.153 Infrastructure/Repositories/SensorDataRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.SensorDataRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.154 Infrastructure/Repositories/StatusRepository.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

Functions

- class [ArandanoIRT_Backend.Infrastructure.Repositories.StatusRepository](#) ([ApplicationDbContext](#) context, [ICacheService](#) cacheService, [IDateTimeProvider](#) dateServiceProvider, [ILogger< StatusRepository >](#) logger)
Implements the [IStatusRepository](#) interface, providing data access logic for status entities ([StatusEntity](#)) using Entity Framework Core.

7.155 Infrastructure/Repositories/TableRelationRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.TableRelationRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.156 Infrastructure/Repositories/ThermalDataRepository.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Repositories.ThermalDataRepository](#)

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Repositories](#)

7.157 Infrastructure/Services/CacheService.cs File Reference

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.158 Infrastructure/Services/CaptchaService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.CloudflareTurnstileService](#)

Implements the ICaptchaService interface using the Cloudflare Turnstile API for verifying CAPTCHA challenges.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.159 Infrastructure/Services/CookiesService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.CookiesService](#)

Provides methods for managing authentication-related HTTP cookies, such as setting, renewing, retrieving, and removing JWT, Refresh Token, and StayLoggedIn cookies.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.160 Infrastructure/Services/FiltersSwaggerService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.ProducesResponseTypeFilter](#)
An Swagger operation filter that adds standard HTTP error response types (400, 404, 429, 500) to every operation in the OpenAPI documentation.
- class [ArandanoIRT_Backend.Infrastructure.Services.UnauthorizedResponseFilter](#)
An Swagger operation filter that adds the 401 Unauthorized response type to operations that are protected by an Authorize attribute (identified by name).
- class [ArandanoIRT_Backend.Infrastructure.Services.ProducesJsonFilter](#)
An Swagger operation filter that attempts to determine the actual return type for 200 OK responses (handling Task and IActionResult with ProducesResponseType) and sets the corresponding JSON schema in the OpenAPI documentation.
- class [ArandanoIRT_Backend.Infrastructure.Services.AuthOperationFilter](#)
An Swagger operation filter that adds a security requirement definition (referencing "cookieAuth") to operations protected by the AuthorizeAttribute.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.161 Infrastructure/Services/JwtTokenService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.JwtTokenService](#)
Implements the ITokenService interface, providing services for generating, validating, refreshing, and revoking JWT access tokens and database-backed opaque refresh tokens.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.162 Infrastructure/Services/LogCleanupService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.LogCleanupService](#)
A background service responsible for periodically deleting old log files from the configured log directory to manage disk space.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.163 Infrastructure/Services/RequestContextAccessor.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.RequestContextAccessor](#)

Implements IRequestContextAccessor to provide access to common data from the current HTTP request context, such as IP address, User Agent details, and authenticated user information (ID, Crop ID).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.164 Infrastructure/Services/SanitizerService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Services.SanitizerService](#)

Implements the ISanitizerService.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Services](#)

7.165 Infrastructure/Utilities/BCryptPasswordHasher.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Utilities.BCryptPasswordHasher](#)

Implements the IPasswordHasher interface using the BCrypt algorithm provided by the BCrypt.Net library.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Utilities](#)

7.166 Infrastructure/Utilities/DateTimeProvider.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Utilities.DateTimeProvider](#)
Implements the `IDateTimeProvider` interface.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Utilities](#)

7.167 Infrastructure/Utilities/RsaService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Utilities.RsaService](#)
Implements the `IRsaService` interface, providing functionalities for RSA encryption (using public key) and decryption (using private key).
- class [ArandanoIRT_Backend.Infrastructure.Utilities.RSAUtility](#)
Internal helper class responsible for generating an RSA key pair and performing core encryption and decryption operations using that key pair.
- class [ArandanoIRT_Backend.Infrastructure.Utilities.RsaKeys](#)
Internal container class to hold the RSA private and public keys as strings.
- class [ArandanoIRT_Backend.Infrastructure.Utilities.PasswordRequest](#)
Represents a simple request structure containing a password string.
- class [ArandanoIRT_Backend.Infrastructure.Utilities.RSAExtensions](#)
Provides internal static extension methods related to RSA key formatting, primarily for PEM encoding.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Utilities](#)

7.168 Infrastructure/Utilities/SsmtpEmailService.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Utilities.SsmtpEmailService](#)

Implements the `IEmailService` interface using an underlying `EmailSenderUtility` to send emails via SMTP.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Utilities](#)

7.169 Infrastructure/Utilities/UAParserAdapter.cs File Reference

Classes

- class [ArandanoIRT_Backend.Infrastructure.Utilities.UAParserAdapter](#)

Implements the `IUserAgentParser` interface by adapting the `UAParser` library (`uap-csharp`) to parse User-Agent strings into the application's `Domain.ValueObjects.ClientInfo` structure.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Infrastructure.Utilities](#)

7.170 Tests/Integration/Common/DbContextTestHelper.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Common.DbContextTestHelper](#)

Provides helper methods for creating `DbContext` instances for integration tests using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Common](#)

7.171 Tests/Integration/Repositories/ChangePasswordRepositoryTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Repositories.ChangePasswordRepositoryTests](#)
Contains integration tests for the ChangePasswordRepository using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Repositories](#)

7.172 Tests/Integration/Repositories/CropRepositoryTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Repositories.CropRepositoryTests](#)
Contains integration tests for the CropRepository using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Repositories](#)

7.173 Tests/Integration/Repositories/FailedLoginAttemptRepositoryTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Repositories.FailedLoginAttemptRepositoryTests](#)
Contains integration tests for the FailedLoginAttemptRepository using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Repositories](#)

7.174 Tests/Integration/Repositories/PersonRepositoryTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Repositories.PersonRepositoryTests](#)
Contains integration tests for the PersonRepository using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Repositories](#)

7.175 Tests/Integration/Repositories/RefreshTokenRepositoryTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Integration.Repositories.RefreshTokenRepositoryTests](#)
Contains integration tests for the RefreshTokenRepository using an SQLite in-memory database.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Integration](#)
- namespace [ArandanoIRT_Backend.Tests.Integration.Repositories](#)

7.176 Tests/Unit/Application/Services/AuthPasswordServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Services.AuthPasswordServiceTests](#)
Contains unit tests for the AuthPasswordService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Services](#)

7.177 Tests/Application/Services/AuthRegistrationServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Services.AuthRegistrationServiceTests](#)
Contains unit tests for the AuthRegistrationService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Services](#)

7.178 Tests/Application/Services/AuthSessionServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Services.AuthSessionServiceTests](#)
Contains unit tests for the AuthSessionService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Services](#)

7.179 Tests/Application/Services/HelpRequestServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Services.HelpRequestServiceTests](#)
Contains unit tests for the HelpRequestService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Services](#)

7.180 Tests/Unit/Application/Utilities/AuthUtilitiesTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Utilities.AuthUtilitiesTests](#)

Contains unit tests for the AuthUtilities class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Utilities](#)

7.181 Tests/Unit/Application/Utilities/EmailValidatorUtilityTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Application.Utilities.EmailValidatorUtilityTests](#)

Contains unit tests for the static EmailValidatorUtility class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Application](#)
- namespace [ArandanoIRT_Backend.Tests.Application.Utilities](#)

7.182 Tests/Unit/Domain/Entities/PersonEntityTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.Entities.PersonEntityTests](#)

Contains unit tests for the PersonEntity class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.Entities](#)

7.183 Tests/Unit/Domain/Entities/RefreshTokenEntityTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.Entities.RefreshTokenEntityTests](#)
Contains unit tests for the RefreshTokenEntity class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.Entities](#)

7.184 Tests/Unit/Domain/Entities/SensorDataEntityTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.Entities.SensorDataEntityTests](#)
Contains unit tests for the SensorDataEntity class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.Entities](#)

7.185 Tests/Unit/Domain/Entities/StatusEntityTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.Entities.StatusEntityTests](#)
Contains unit tests for the StatusEntity class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.Entities](#)

7.186 Tests/Unit/Domain/ValueObjects/ClientInfoTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ClientInfoTests](#)

Contains unit tests for the [ClientInfo](#) value object.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects](#)

7.187 Tests/Unit/Domain/ValueObjects/ResultTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTests](#)
Contains unit tests for the non-generic [Result](#) class.
- class [ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects.ResultTTests](#)
Contains unit tests for the generic [Result<T>](#) class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Domain.ValueObjects](#)

7.188 Tests/Unit/Infrastructure/Persistence/AuditHelperServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence.AuditHelperServiceTests](#)
Contains unit tests for the [AuditHelperService](#) class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Persistence](#)

7.189 Tests/Unit/Infrastructure/Services/CacheServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CacheServiceTests](#)
Contains unit tests for the CacheService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.190 Tests/Unit/Infrastructure/Services/CloudflareTurnstileServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CloudflareTurnstileServiceTests](#)
Contains unit tests for the CloudflareTurnstileService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.191 Tests/Unit/Infrastructure/Services/CookiesServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.CookiesServiceTests](#)
Contains unit tests for the CookiesService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.192 Tests/Unit/Infrastructure/Services/JwtTokenServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.JwtTokenServiceTests](#)
Contains unit tests for the JwtTokenService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.193 Tests/Unit/Infrastructure/Services/RequestContextAccessorTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.RequestContextAccessorTests](#)
Contains unit tests for the RequestContextAccessor class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.194 Tests/Unit/Infrastructure/Services/SanitizerServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services.SanitizerServiceTests](#)
Contains unit tests for the SanitizerService class (using default HtmlSanitizer + HtmlDecode).

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Services](#)

7.195 Tests/Unit/Infrastructure/Utilities/BCryptPasswordHasherTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.BCryptPasswordHasherTests](#)
Contains unit tests for the BCryptPasswordHasher class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities](#)

7.196 Tests/Unit/Infrastructure/Utilities/DateTimeProviderTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.DateTimeProviderTests](#)
Contains unit tests for the DateTimeProvider class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities](#)

7.197 Tests/Unit/Infrastructure/Utilities/RsaServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.RsaServiceTests](#)
Contains unit tests for the RsaService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities](#)

7.198 Tests/Unit/Infrastructure/Utilities/SmtpEmailServiceTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.SmtpEmailServiceTests](#)
Contains unit tests for the SmtpEmailService class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities](#)

7.199 Tests/Unit/Infrastructure/Utilities/UAParserAdapterTests.cs File Reference

Classes

- class [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities.UAParserAdapterTests](#)
Contains unit tests for the UAParserAdapter class.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.Tests](#)
- namespace [ArandanoIRT_Backend.Tests.Unit](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure](#)
- namespace [ArandanoIRT_Backend.Tests.Unit.Infrastructure.Utilities](#)

7.200 UI/Controllers/BaseApiController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.BaseApiController](#)
Base controller for API endpoints providing shared functionality.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.201 UI/Controllers/CryptoController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.CryptoController](#)
Handles cryptographic utility endpoints.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.202 UI/Controllers/HelpRequestsController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.HelpRequestsController](#)
Handles endpoints related to user help requests.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.203 UI/Controllers/PasswordController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.PasswordController](#)
Handles endpoints related to user password management, like reset requests.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.204 UI/Controllers/RegistrationController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.RegistrationController](#)
Handles endpoints related to user and administrator registration.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.205 UI/Controllers/SessionController.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Controllers.SessionController](#)
Handles endpoints related to user session management, including login, token refresh, and logout operations.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Controllers](#)

7.206 UI/Middleware/ExceptionHandlingMiddleware.cs File Reference

Classes

- class [ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddleware](#)
ASP.NET Core middleware for centralized exception handling.
- class [ArandanoIRT_Backend.UI.Middleware.ExceptionHandlingMiddlewareExtensions](#)
Provides an extension method for easily registering the ExceptionHandlingMiddleware in the ASP.NET Core application pipeline.

Namespaces

- namespace [ArandanoIRT_Backend](#)
- namespace [ArandanoIRT_Backend.UI](#)
- namespace [ArandanoIRT_Backend.UI.Middleware](#)