

Grip

1.0

Generated by Doxygen 1.9.1

1 Namespace Index	1
1.1 Packages	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	9
4.1 File List	9
5 Namespace Documentation	11
5.1 Grip Namespace Reference	11
5.2 Grip.Api Namespace Reference	11
5.3 Grip.Api.Controllers Namespace Reference	11
5.4 Grip.Api.Hubs Namespace Reference	12
5.5 Grip.Bll Namespace Reference	12
5.6 Grip.Bll.DTO Namespace Reference	12
5.6.1 Function Documentation	12
5.6.1.1 ActiveAttendanceDTO()	12
5.6.1.2 PassiveAttendanceDTO()	13
5.6.1.3 StudentDetailDTO()	13
5.7 Grip.Bll.Exceptions Namespace Reference	13
5.8 Grip.Bll.Providers Namespace Reference	14
5.9 Grip.Bll.Services Namespace Reference	14
5.10 Grip.Bll.Services.Interfaces Namespace Reference	14
5.11 Grip.DAL Namespace Reference	15
5.12 Grip.DAL.Migrations Namespace Reference	15
5.13 Grip.DAL.Model Namespace Reference	15
5.14 Grip.Utills Namespace Reference	15
6 Class Documentation	17
6.1 Grip.DAL.Migrations.AddClassesAndAttendance Class Reference	17
6.1.1 Detailed Description	18
6.1.2 Member Function Documentation	18
6.1.2.1 BuildTargetModel()	18
6.1.2.2 Down()	18
6.1.2.3 Up()	18
6.2 Grip.DAL.Migrations.AddExempts Class Reference	19
6.2.1 Detailed Description	19
6.2.2 Member Function Documentation	19
6.2.2.1 BuildTargetModel()	20
6.2.2.2 Down()	20

6.2.2.3 Up()	20
6.3 ApiKeyValidationMiddleware Class Reference	20
6.3.1 Detailed Description	20
6.3.2 Constructor & Destructor Documentation	21
6.3.2.1 ApiKeyValidationMiddleware()	21
6.3.3 Member Function Documentation	21
6.3.3.1 InvokeAsync()	21
6.4 ApplicationDbContext Class Reference	22
6.4.1 Detailed Description	23
6.4.2 Constructor & Destructor Documentation	23
6.4.2.1 ApplicationDbContext()	23
6.4.3 Member Function Documentation	23
6.4.3.1 OnConfiguring()	24
6.4.3.2 OnModelCreating()	24
6.4.4 Property Documentation	24
6.4.4.1 Attendances	24
6.4.4.2 Classes	24
6.4.4.3 Exempts	25
6.4.4.4 Groups	25
6.4.4.5 PassiveTags	25
6.4.4.6 Stations	25
6.5 Grip.DAL.Migrations.ApplicationDbContextModelSnapshot Class Reference	26
6.5.1 Detailed Description	26
6.5.2 Member Function Documentation	26
6.5.2.1 BuildModel()	27
6.6 Grip.DAL.Model.Attendance Class Reference	27
6.6.1 Detailed Description	27
6.6.2 Property Documentation	27
6.6.2.1 Id	27
6.6.2.2 Station	28
6.6.2.3 Time	28
6.6.2.4 User	28
6.7 AttendanceController Class Reference	28
6.7.1 Detailed Description	29
6.7.2 Constructor & Destructor Documentation	29
6.7.2.1 AttendanceController()	29
6.7.3 Member Function Documentation	30
6.7.3.1 ActiveAuthentication()	30
6.7.3.2 GetAttendanceForDay()	30
6.7.3.3 PassiveAuthentication()	30
6.8 AttendanceService Class Reference	31
6.8.1 Detailed Description	32

6.8.2 Constructor & Destructor Documentation	32
6.8.2.1 AttendanceService()	32
6.8.3 Member Function Documentation	32
6.8.3.1 GetAttendanceForDay()	32
6.8.3.2 VerifyPassiveScan()	33
6.8.3.3 VerifyPhoneScan()	34
6.9 Grip.Bll.Exceptions.BadRequestException Class Reference	34
6.9.1 Detailed Description	35
6.9.2 Constructor & Destructor Documentation	35
6.9.2.1 BadRequestException()	35
6.10 Grip.Api.Controllers.ClassController Class Reference	36
6.10.1 Detailed Description	37
6.10.2 Constructor & Destructor Documentation	37
6.10.2.1 ClassController()	37
6.10.3 Member Function Documentation	37
6.10.3.1 DeleteClass()	37
6.10.3.2 GetClass() [1/2]	38
6.10.3.3 GetClass() [2/2]	38
6.10.3.4 GetClassOnDay()	38
6.10.3.5 PostClass()	39
6.10.3.6 PutClass()	39
6.11 ClassService Class Reference	40
6.11.1 Detailed Description	41
6.11.2 Constructor & Destructor Documentation	41
6.11.2.1 ClassService()	41
6.11.3 Member Function Documentation	41
6.11.3.1 Create()	42
6.11.3.2 Delete()	42
6.11.3.3 Get()	42
6.11.3.4 GetAll()	44
6.11.3.5 GetClassesForUserOnDay()	45
6.11.3.6 Update()	45
6.12 Grip.DAL.Migrations.ClassStationRelationAdded Class Reference	46
6.12.1 Detailed Description	46
6.12.2 Member Function Documentation	46
6.12.2.1 BuildTargetModel()	47
6.12.2.2 Down()	47
6.12.2.3 Up()	47
6.13 Grip.Bll.Providers.CurrentTimeProvider Class Reference	47
6.13.1 Detailed Description	48
6.13.2 Property Documentation	48
6.13.2.1 Now	48

6.14 Grip.Bll.Exceptions.DbConcurrencyException Class Reference	49
6.14.1 Detailed Description	49
6.14.2 Constructor & Destructor Documentation	50
6.14.2.1 DbConcurrencyException()	50
6.15 DeChunkingMiddleware Class Reference	50
6.15.1 Detailed Description	50
6.15.2 Constructor & Destructor Documentation	50
6.15.2.1 DeChunkingMiddleware()	50
6.15.3 Member Function Documentation	51
6.15.3.1 InvokeAsync()	51
6.16 EmailService Class Reference	51
6.16.1 Detailed Description	52
6.16.2 Constructor & Destructor Documentation	52
6.16.2.1 EmailService()	52
6.16.3 Member Function Documentation	53
6.16.3.1 SendEmailAsync() [1/2]	53
6.16.3.2 SendEmailAsync() [2/2]	53
6.17 Grip.DAL.Model.Exempt Class Reference	54
6.17.1 Detailed Description	54
6.17.2 Property Documentation	54
6.17.2.1 Id	54
6.17.2.2 IssuedBy	54
6.17.2.3 IssuedById	55
6.17.2.4 IssuedTo	55
6.17.2.5 IssuedTold	55
6.17.2.6 ValidFrom	55
6.17.2.7 ValidTo	55
6.18 Grip.Api.Controllers.ExemptController Class Reference	56
6.18.1 Detailed Description	57
6.18.2 Constructor & Destructor Documentation	57
6.18.2.1 ExemptController()	57
6.18.3 Member Function Documentation	57
6.18.3.1 DeleteExempt()	57
6.18.3.2 GetExempt() [1/2]	58
6.18.3.3 GetExempt() [2/2]	58
6.18.3.4 PostExempt()	58
6.19 Grip.Bll.Services.ExemptService Class Reference	59
6.19.1 Detailed Description	60
6.19.2 Constructor & Destructor Documentation	61
6.19.2.1 ExemptService()	61
6.19.3 Member Function Documentation	61
6.19.3.1 Create() [1/2]	61

6.19.3.2 Create() [2/2]	61
6.19.3.3 Delete()	62
6.19.3.4 Get()	62
6.19.3.5 GetAll()	63
6.19.3.6 GetAllForUser()	63
6.19.3.7 GetByUser()	63
6.19.3.8 Update()	64
6.20 Grip.DAL.Model.Group Class Reference	64
6.20.1 Detailed Description	65
6.20.2 Property Documentation	65
6.20.2.1 Classes	65
6.20.2.2 Id	65
6.20.2.3 Name	65
6.20.2.4 Users	65
6.21 GroupController Class Reference	66
6.21.1 Detailed Description	67
6.21.2 Constructor & Destructor Documentation	67
6.21.2.1 GroupController()	67
6.21.3 Member Function Documentation	67
6.21.3.1 AddUserToGroup()	67
6.21.3.2 DeleteGroup()	68
6.21.3.3 GetGroup() [1/2]	68
6.21.3.4 GetGroup() [2/2]	68
6.21.3.5 PostGroup()	69
6.21.3.6 PutGroup()	69
6.21.3.7 RemoveUserFromGroup()	70
6.22 Grip.Bll.Services.GroupService Class Reference	71
6.22.1 Detailed Description	72
6.22.2 Constructor & Destructor Documentation	72
6.22.2.1 GroupService()	72
6.22.3 Member Function Documentation	72
6.22.3.1 AddUserToGroup()	73
6.22.3.2 Create()	73
6.22.3.3 Delete()	74
6.22.3.4 Get()	75
6.22.3.5 GetAll()	75
6.22.3.6 GetUsersInGroup()	76
6.22.3.7 RemoveUserFromGroup()	77
6.22.3.8 Update()	77
6.23 HMACTokenProvider Class Reference	78
6.23.1 Detailed Description	78
6.23.2 Member Function Documentation	79

6.23.2.1 GenerateToken()	79
6.23.2.2 ValidateToken()	79
6.24 IAttendanceService Interface Reference	80
6.24.1 Detailed Description	80
6.24.2 Member Function Documentation	80
6.24.2.1 GetAttendanceForDay()	80
6.24.2.2 VerifyPassiveScan()	81
6.24.2.3 VerifyPhoneScan()	81
6.25 IClassService Interface Reference	81
6.25.1 Detailed Description	82
6.25.2 Member Function Documentation	83
6.25.2.1 GetClassesForUserOnDay()	83
6.26 Grip.Bll.Providers.ICurrentTimeProvider Interface Reference	83
6.26.1 Detailed Description	84
6.26.2 Property Documentation	84
6.26.2.1 Now	84
6.27 Grip.DAL.Migrations.Identity Class Reference	84
6.27.1 Detailed Description	85
6.27.2 Member Function Documentation	85
6.27.2.1 BuildTargetModel()	85
6.27.2.2 Down()	85
6.27.2.3 Up()	86
6.28 Grip.Bll.Services.Interfaces.IEmailService Interface Reference	86
6.28.1 Detailed Description	86
6.28.2 Member Function Documentation	86
6.28.2.1 SendEmailAsync() [1/2]	87
6.28.2.2 SendEmailAsync() [2/2]	87
6.29 IExemptService Interface Reference	88
6.29.1 Detailed Description	88
6.29.2 Member Function Documentation	89
6.29.2.1 Create()	89
6.30 IGroupService Interface Reference	89
6.30.1 Detailed Description	90
6.30.2 Member Function Documentation	90
6.30.2.1 AddUserToGroup()	90
6.30.2.2 GetUsersInGroup()	91
6.30.2.3 RemoveUserFromGroup()	91
6.31 IPassiveTagService Interface Reference	92
6.31.1 Detailed Description	92
6.32 Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdateDTO, GetDTO > Interface Template Reference	93
6.32.1 Detailed Description	93

6.32.2 Member Function Documentation	93
6.32.2.1 Create()	93
6.32.2.2 Delete()	94
6.32.2.3 Get()	94
6.32.2.4 GetAll()	94
6.32.2.5 Update()	95
6.33 Grip.Api.Hubs.IStationClient Interface Reference	95
6.33.1 Detailed Description	95
6.33.2 Member Function Documentation	95
6.33.2.1 ReceiveScan()	95
6.34 Grip.Bll.Services.Interfaces.IStationService Interface Reference	96
6.34.1 Detailed Description	96
6.34.2 Member Function Documentation	96
6.34.2.1 GetSecretKey()	96
6.35 IStationTokenProvider Interface Reference	97
6.35.1 Detailed Description	97
6.35.2 Member Function Documentation	97
6.35.2.1 GenerateToken()	97
6.35.2.2 ValidateToken()	98
6.36 Grip.Bll.Services.Interfaces.IStudentService Interface Reference	98
6.36.1 Detailed Description	99
6.36.2 Member Function Documentation	99
6.36.2.1 GetStudentDetailsAsync()	99
6.36.2.2 SearchStudentsAsync()	99
6.37 Grip.Bll.Services.Interfaces.IUserService Interface Reference	100
6.37.1 Detailed Description	101
6.37.2 Member Function Documentation	101
6.37.2.1 AddRole()	101
6.37.2.2 ConfirmEmail()	102
6.37.2.3 ForgotPassword()	102
6.37.2.4 Login()	102
6.37.2.5 RemoveRole()	103
6.37.2.6 ResetPassword()	103
6.38 Grip.DAL.MappingProfile Class Reference	104
6.38.1 Detailed Description	104
6.38.2 Constructor & Destructor Documentation	104
6.38.2.1 MappingProfile()	105
6.39 NotChunked Class Reference	105
6.39.1 Detailed Description	106
6.40 Grip.Bll.Exceptions.NotFoundException Class Reference	106
6.40.1 Detailed Description	107
6.40.2 Constructor & Destructor Documentation	107

6.40.2.1 NotFoundException()	107
6.41 Grip.DAL.Model.PassiveTag Class Reference	107
6.41.1 Detailed Description	107
6.41.2 Property Documentation	108
6.41.2.1 Id	108
6.41.2.2 SerialNumber	108
6.41.2.3 User	108
6.42 Grip.Api.Controllers.PassiveTagController Class Reference	109
6.42.1 Detailed Description	110
6.42.2 Constructor & Destructor Documentation	110
6.42.2.1 PassiveTagController()	110
6.42.3 Member Function Documentation	110
6.42.3.1 DeletePassiveTag()	110
6.42.3.2 GetPassiveTag()	111
6.42.3.3 GetPassiveTags()	111
6.42.3.4 PostPassiveTag()	111
6.42.3.5 PutPassiveTag()	112
6.43 Grip.Bll.Services.PassiveTagService Class Reference	112
6.43.1 Detailed Description	114
6.43.2 Constructor & Destructor Documentation	114
6.43.2.1 PassiveTagService()	114
6.43.3 Member Function Documentation	114
6.43.3.1 Create()	114
6.43.3.2 Delete()	115
6.43.3.3 Get()	115
6.43.3.4 GetAll()	116
6.43.3.5 Update()	116
6.44 Role Class Reference	117
6.44.1 Detailed Description	118
6.44.2 Member Data Documentation	118
6.44.2.1 Admin	118
6.44.2.2 Doorman	118
6.44.2.3 Student	118
6.44.2.4 Teacher	119
6.45 Grip.DAL.Model.Station Class Reference	119
6.45.1 Detailed Description	119
6.45.2 Property Documentation	119
6.45.2.1 Attendances	119
6.45.2.2 Classes	120
6.45.2.3 Id	120
6.45.2.4 Name	120
6.45.2.5 SecretKey	120

6.45.2.6 StationNumber	120
6.46 StationController Class Reference	121
6.46.1 Detailed Description	121
6.46.2 Constructor & Destructor Documentation	122
6.46.2.1 StationController()	122
6.46.3 Member Function Documentation	122
6.46.3.1 GetKey()	122
6.47 Grip.Api.Hubs.StationHub Class Reference	123
6.47.1 Detailed Description	123
6.47.2 Member Function Documentation	123
6.47.2.1 SelectStation()	124
6.48 Grip.Bll.Services.StationService Class Reference	124
6.48.1 Detailed Description	125
6.48.2 Constructor & Destructor Documentation	125
6.48.2.1 StationService()	125
6.48.3 Member Function Documentation	125
6.48.3.1 GetSecretKey()	125
6.49 Grip.Api.Controllers.StudentController Class Reference	126
6.49.1 Detailed Description	127
6.49.2 Constructor & Destructor Documentation	127
6.49.2.1 StudentController()	127
6.49.3 Member Function Documentation	128
6.49.3.1 GetStudentDetails()	128
6.49.3.2 GetStudents()	128
6.50 Grip.Bll.Services.StudentService Class Reference	129
6.50.1 Detailed Description	129
6.50.2 Constructor & Destructor Documentation	130
6.50.2.1 StudentService()	130
6.50.3 Member Function Documentation	130
6.50.3.1 GetStudentDetailsAsync()	130
6.50.3.2 SearchStudentsAsync()	131
6.51 Grip.Bll.Exceptions.UnauthorizedException Class Reference	132
6.51.1 Detailed Description	132
6.51.2 Constructor & Destructor Documentation	133
6.51.2.1 UnauthorizedException()	133
6.52 Grip.DAL.Model.User Class Reference	133
6.52.1 Detailed Description	134
6.52.2 Property Documentation	134
6.52.2.1 Attendances	134
6.52.2.2 Exemptions	135
6.52.2.3 Groups	135
6.52.2.4 IssuedExemptions	135

6.52.2.5 PassiveTags	135
6.53 UserController Class Reference	136
6.53.1 Detailed Description	137
6.53.2 Constructor & Destructor Documentation	137
6.53.2.1 UserController()	137
6.53.3 Member Function Documentation	138
6.53.3.1 AddRole()	138
6.53.3.2 ConfirmEmail()	138
6.53.3.3 Delete()	139
6.53.3.4 ForgotPassword()	140
6.53.3.5 Get() [1/2]	140
6.53.3.6 Get() [2/2]	141
6.53.3.7 Login()	141
6.53.3.8 Register()	141
6.53.3.9 RemoveRole()	142
6.53.3.10 ResetPassword()	142
6.53.3.11 Update()	143
6.54 Grip.Bll.Services.UserService Class Reference	143
6.54.1 Detailed Description	145
6.54.2 Constructor & Destructor Documentation	145
6.54.2.1 UserService()	145
6.54.3 Member Function Documentation	145
6.54.3.1 AddRole()	145
6.54.3.2 ConfirmEmail()	146
6.54.3.3 Create()	146
6.54.3.4 Delete()	147
6.54.3.5 ForgotPassword()	147
6.54.3.6 Get()	148
6.54.3.7 GetAll()	148
6.54.3.8 Login()	149
6.54.3.9 RemoveRole()	150
6.54.3.10 ResetPassword()	150
6.54.3.11 Update()	150
6.55 ValidateApiKey Class Reference	151
6.55.1 Detailed Description	152
7 File Documentation	153
7.1 Server/Api/Controllers/AttendanceController.cs File Reference	153
7.2 Server/Api/Controllers/ClassController.cs File Reference	153
7.3 Server/Api/Controllers/ExemptController.cs File Reference	153
7.4 Server/Api/Controllers/GroupController.cs File Reference	154
7.5 Server/Api/Controllers/PassiveTagController.cs File Reference	154

7.6 Server/Api/Controllers/StationController.cs File Reference	154
7.7 Server/Api/Controllers/StudentController.cs File Reference	154
7.8 Server/Api/Controllers/UserController.cs File Reference	155
7.9 Server/Api/Hubs/IStationClient.cs File Reference	155
7.10 Server/Api/Hubs/StationHub.cs File Reference	155
7.11 Server/Api/Middleware/ApiKeyValidationMiddleware.cs File Reference	155
7.12 Server/Api/Middleware/DeChunkingMiddleware.cs File Reference	156
7.13 Server/Api/Middleware/NotChunked.cs File Reference	156
7.14 Server/Api/Middleware/ValidateApiKey.cs File Reference	156
7.15 Server/Bll/DTO/Attendance/ActiveAttendanceDTO.cs File Reference	156
7.16 Server/Bll/DTO/Attendance/AttendanceDTO.cs File Reference	156
7.16.1 Function Documentation	157
7.16.1.1 AttendanceDTO()	157
7.17 Server/Bll/DTO/Attendance/PassiveAttendanceDTO.cs File Reference	157
7.18 Server/Bll/DTO/Class/ClassDTO.cs File Reference	157
7.18.1 Function Documentation	157
7.18.1.1 ClassDTO()	158
7.19 Server/Bll/DTO/Class/CreateClassDTO.cs File Reference	158
7.20 Server/Bll/DTO/Exempt/CreateExemptDTO.cs File Reference	158
7.21 Server/Bll/DTO/Exempt/ExemptDTO.cs File Reference	159
7.22 Server/Bll/DTO/Group/GroupDTO.cs File Reference	159
7.22.1 Function Documentation	159
7.22.1.1 GroupDTO()	159
7.23 Server/Bll/DTO/PassiveTag/CreatePassiveTagDTO.cs File Reference	159
7.23.1 Function Documentation	160
7.23.1.1 CreatePassiveTagDTO()	160
7.24 Server/Bll/DTO/PassiveTag/PassiveTagDTO.cs File Reference	160
7.24.1 Function Documentation	160
7.24.1.1 PassiveTagDTO()	160
7.25 Server/Bll/DTO/PassiveTag/UpdatePassiveTagDTO.cs File Reference	161
7.25.1 Function Documentation	161
7.25.1.1 UpdatePassiveTagDTO()	161
7.26 Server/Bll/DTO/Station/StationScanDTO.cs File Reference	161
7.27 Server/Bll/DTO/Station/StationSecretKeyDTO.cs File Reference	162
7.27.1 Function Documentation	162
7.27.1.1 StationSecretKeyDTO()	162
7.28 Server/Bll/DTO/Student/AbsenceDTO.cs File Reference	162
7.28.1 Function Documentation	162
7.28.1.1 AbsenceDTO()	162
7.29 Server/Bll/DTO/Student/StudentAttendanceDTO.cs File Reference	163
7.29.1 Function Documentation	163
7.29.1.1 StudentAttendanceDTO()	163

7.30 Server/BI/DTO/Student/StudentDetailDTO.cs File Reference	163
7.31 Server/BI/DTO/User/ConfirmEmailDTO.cs File Reference	163
7.31.1 Function Documentation	164
7.31.1.1 ConfirmEmailDTO()	164
7.32 Server/BI/DTO/User/ForgotPasswordDTO.cs File Reference	164
7.32.1 Function Documentation	164
7.32.1.1 ForgotPasswordDTO()	164
7.33 Server/BI/DTO/User/LoginResultDTO.cs File Reference	165
7.33.1 Function Documentation	165
7.33.1.1 LoginResultDTO()	165
7.34 Server/BI/DTO/User/LoginUserDTO.cs File Reference	165
7.34.1 Function Documentation	165
7.34.1.1 LoginUserDTO()	165
7.35 Server/BI/DTO/User/RegisterUserDTO.cs File Reference	166
7.35.1 Function Documentation	166
7.35.1.1 RegisterUserDTO()	166
7.36 Server/BI/DTO/User/ResetPasswordDTO.cs File Reference	166
7.36.1 Function Documentation	167
7.36.1.1 ResetPasswordDTO()	167
7.37 Server/BI/DTO/User/UserDTO.cs File Reference	167
7.37.1 Function Documentation	167
7.37.1.1 UserDTO()	167
7.38 Server/BI/DTO/User/UserInfoDTO.cs File Reference	168
7.38.1 Function Documentation	168
7.38.1.1 UserInfoDTO()	168
7.39 Server/BI/Exceptions/BadRequestException.cs File Reference	169
7.40 Server/BI/Exceptions/DbConcurrencyException.cs File Reference	169
7.41 Server/BI/Exceptions/NotFoundException.cs File Reference	169
7.42 Server/BI/Exceptions/UnauthorizedException.cs File Reference	170
7.43 Server/BI/Providers/CurrentTimeProvider.cs File Reference	170
7.44 Server/BI/Providers/HMACTokenProvider.cs File Reference	170
7.45 Server/BI/Providers/ICurrentTimeProvider.cs File Reference	170
7.46 Server/BI/Providers/IStationTokenProvider.cs File Reference	171
7.47 Server/BI/Services/AttendanceService.cs File Reference	171
7.48 Server/BI/Services/ClassService.cs File Reference	171
7.49 Server/BI/Services/EmailService.cs File Reference	171
7.50 Server/BI/Services/ExemptService.cs File Reference	171
7.51 Server/BI/Services/GroupService.cs File Reference	172
7.52 Server/BI/Services/Interfaces/IAttendanceService.cs File Reference	172
7.53 Server/BI/Services/Interfaces/IClassService.cs File Reference	172
7.54 Server/BI/Services/Interfaces/IEmailService.cs File Reference	172
7.55 Server/BI/Services/Interfaces/IExemptService.cs File Reference	173

7.56 Server/BI/Services/Interfaces/IGroupService.cs File Reference	173
7.57 Server/BI/Services/Interfaces/IPassiveTagService.cs File Reference	173
7.58 Server/BI/Services/Interfaces/IRestInterface.cs File Reference	173
7.59 Server/BI/Services/Interfaces/IStationService.cs File Reference	173
7.60 Server/BI/Services/Interfaces/IStudentService.cs File Reference	174
7.61 Server/BI/Services/Interfaces/IUserService.cs File Reference	174
7.62 Server/BI/Services/PassiveTagService.cs File Reference	174
7.63 Server/BI/Services/StationService.cs File Reference	175
7.64 Server/BI/Services/StudentService.cs File Reference	175
7.65 Server/BI/Services/UserService.cs File Reference	175
7.66 Server/BI/Utils/Consts.cs File Reference	176
7.67 Server/DAL/ApplicationDbContext.cs File Reference	176
7.68 Server/DAL/MappingProfile.cs File Reference	176
7.69 Server/DAL/Migrations/20230317172813_Identity.cs File Reference	176
7.70 Server/DAL/Migrations/20230317172813_Identity.Designer.cs File Reference	177
7.71 Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.cs File Reference	177
7.72 Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.Designer.cs File Reference	177
7.73 Server/DAL/Migrations/20230501161544_AddExempts.cs File Reference	177
7.74 Server/DAL/Migrations/20230501161544_AddExempts.Designer.cs File Reference	178
7.75 Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.cs File Reference	178
7.76 Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.Designer.cs File Reference	178
7.77 Server/DAL/Migrations/ApplicationDbContextModelSnapshot.cs File Reference	179
7.78 Server/DAL/Model/Attendance.cs File Reference	179
7.79 Server/DAL/Model/Class.cs File Reference	179
7.80 Server/DAL/Model/Exempt.cs File Reference	179
7.81 Server/DAL/Model/Group.cs File Reference	180
7.82 Server/DAL/Model/PassiveTag.cs File Reference	180
7.83 Server/DAL/Model/Role.cs File Reference	180
7.84 Server/DAL/Model/Station.cs File Reference	180
7.85 Server/DAL/Model/User.cs File Reference	181
Index	183

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Grip	11
Grip.Api	11
Grip.Api.Controllers	11
Grip.Api.Hubs	12
Grip.Bll	12
Grip.Bll.DTO	12
Grip.Bll.Exceptions	13
Grip.Bll.Providers	14
Grip.Bll.Services	14
Grip.Bll.Services.Interfaces	14
Grip.DAL	15
Grip.DAL.Migrations	15
Grip.DAL.Model	15
Grip.Utills	15

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

ApiKeyValidationMiddleware	20
Grip.DAL.Model.Attendance	27
Attribute	
NotChunked	105
ValidateApiKey	151
ControllerBase	
AttendanceController	28
Grip.Api.Controllers.ClassController	36
Grip.Api.Controllers.ExemptController	56
Grip.Api.Controllers.PassiveTagController	109
Grip.Api.Controllers.StudentController	126
GroupController	66
StationController	121
UserController	136
DeChunkingMiddleware	50
Exception	
Grip.Bll.Exceptions.BadRequestException	34
Grip.Bll.Exceptions.DbConcurrencyException	49
Grip.Bll.Exceptions.NotFoundException	106
Grip.Bll.Exceptions.UnauthorizedException	132
Grip.DAL.Model.Exempt	54
Grip.DAL.Model.Group	64
Hub	
Grip.Api.Hubs.StationHub	123
IAttendanceService	80
AttendanceService	31
Grip.Bll.Providers.ICurrentTimeProvider	83
Grip.Bll.Providers.CurrentTimeProvider	47
IdentityDbContext	
ApplicationDbContext	22
IdentityRole	
Role	117
IdentityUser	
Grip.DAL.Model.User	133
Grip.Bll.Services.Interfaces.IEmailService	86

EmailService	51
Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >	93
IRestInterface	
IClassService	81
ClassService	40
IExemptService	88
Grip.Bll.Services.ExemptService	59
IGroupService	89
Grip.Bll.Services.GroupService	71
IPassiveTagService	92
Grip.Bll.Services.PassiveTagService	112
Grip.Bll.Services.Interfaces.IRestInterface< RegisterUserDTO, UserDTO, UserDTO >	93
Grip.Bll.Services.Interfaces.IUserService	100
Grip.Bll.Services.UserService	143
Grip.Api.Hubs.IStationClient	95
Grip.Bll.Services.Interfaces.IStationService	96
Grip.Bll.Services.StationService	124
IStationTokenProvider	97
HMACTokenProvider	78
Grip.Bll.Services.Interfaces.IStudentService	98
Grip.Bll.Services.StudentService	129
Migration	
Grip.DAL.Migrations.AddClassesAndAttendance	17
Grip.DAL.Migrations.AddExempts	19
Grip.DAL.Migrations.ClassStationRelationAdded	46
Grip.DAL.Migrations.Identity	84
ModelSnapshot	
Grip.DAL.Migrations.ApplicationDbContextModelSnapshot	26
Grip.DAL.Model.PassiveTag	107
Profile	
Grip.DAL.MappingProfile	104
Grip.DAL.Model.Station	119

Chapter 3

Class Index

3.1 Class List

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

Grip.DAL.Migrations.AddClassesAndAttendance	17
Grip.DAL.Migrations.AddExempts	19
ApiKeyValidationMiddleware	
Validates the api key in the header of the request	20
ApplicationDbContext	
Represents the application's database context	22
Grip.DAL.Migrations.ApplicationDbContextModelSnapshot	26
Grip.DAL.Model.Attendance	
Represents the attendance information of a user at a station	27
AttendanceController	
Controller for handling attendance	28
AttendanceService	
Provides functionality related to attendance management	31
Grip.Bll.Exceptions.BadRequestException	
Exception thrown when a request is invalid: HTTP 400	34
Grip.Api.Controllers.ClassController	
Endpoint for managing classes	36
ClassService	
Service class for managing classes	40
Grip.DAL.Migrations.ClassStationRelationAdded	46
Grip.Bll.Providers.CurrentTimeProvider	
Class for abstracting the current time for testing purposes	47
Grip.Bll.Exceptions.DbConcurrencyException	
Exception thrown when a db update failed due to concurrency: HTTP 409	49
DeChunkingMiddleware	
Middleware to de-chunk HTTP requests, because ucontroller doesn't support chunked responses	50
EmailService	
Service class for sending emails	51
Grip.DAL.Model.Exempt	
Represents an exemption in the database	54
Grip.Api.Controllers.ExemptController	
Endpoint for managing exempt items	56

Grip.Bll.Services.ExemptService	
Service class for managing exempts	59
Grip.DAL.Model.Group	
Represents a group in the database	64
GroupController	
Represents a controller for managing groups	66
Grip.Bll.Services.GroupService	
Service class for managing groups	71
HMACTokenProvider	
Generates and validates HMAC tokens for stations	78
IAttendanceService	
Interface for the attendance service	80
IClassService	
Interface for the class service	81
Grip.Bll.Providers.ICurrentTimeProvider	
Interface for abstracting the current time for testing purposes	83
Grip.DAL.Migrations.Identity	
84	
Grip.Bll.Services.Interfaces.IEmailService	
Represents a service for sending emails	86
IExemptService	
Represents an interface for the exempt service	88
IGroupService	
Interface representing the Group Service, which provides CRUD operations for groups as well as specific group operations	89
IPassiveTagService	
Represents an interface for the passive tag service	92
Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >	
Represents a generic REST interface for CRUD operations	93
Grip.Api.Hubs.IStationClient	
This interface is used to define the methods that can be called from the station hub on the client	95
Grip.Bll.Services.Interfaces.IStationService	
Represents an interface for the station service	96
IStationTokenProvider	
Interface for generating and validating tokens for stations	97
Grip.Bll.Services.Interfaces.IStudentService	
Interface for the student service	98
Grip.Bll.Services.Interfaces.IUserService	
Represents an interface for the user service	100
Grip.DAL.MappingProfile	
Mapping profile for AutoMapper	104
NotChunked	
Attribute for marking a controller method write response in a not chunked manner	105
Grip.Bll.Exceptions.NotFoundException	
Exception thrown when a resource is not found: HTTP 404	106
Grip.DAL.Model.PassiveTag	
Represents a passive tag in the database	107
Grip.Api.Controllers.PassiveTagController	
API controller for managing passive tags	109
Grip.Bll.Services.PassiveTagService	
Service class for managing passive tags	112
Role	
Represents a role in the database	117
Grip.DAL.Model.Station	
Represents a station in the database	119
StationController	
Represents a controller for managing stations	121

Grip.Api.Hubs.StationHub	
SignalR Hub for station related events	123
Grip.Bll.Services.StationService	
Provides methods for managing stations	124
Grip.Api.Controllers.StudentController	
Controller for managing students	126
Grip.Bll.Services.StudentService	
Service for managing students	129
Grip.Bll.Exceptions.UnauthorizedException	
Exception thrown when a request needs authorization: HTTP 401	132
Grip.DAL.Model.User	
Represents a user in the database	133
UserController	
API endpoints for managing user accounts	136
Grip.Bll.Services.UserService	
Represents a service for managing users	143
ValidateApiKey	
Attribute for marking a controller method as requiring a valid API key	151

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Server/Api/Controllers/AttendanceController.cs	153
Server/Api/Controllers/ClassController.cs	153
Server/Api/Controllers/ExemptController.cs	153
Server/Api/Controllers/GroupController.cs	154
Server/Api/Controllers/PassiveTagController.cs	154
Server/Api/Controllers/StationController.cs	154
Server/Api/Controllers/StudentController.cs	154
Server/Api/Controllers/UserController.cs	155
Server/Api/Hubs/IStationClient.cs	155
Server/Api/Hubs/StationHub.cs	155
Server/Api/Middleware/ApiKeyValidationMiddleware.cs	155
Server/Api/Middleware/DeChunkingMiddleware.cs	156
Server/Api/Middleware/NotChunked.cs	156
Server/Api/Middleware/ValidateApiKey.cs	156
Server/Bll/DTO/Attendance/ActiveAttendanceDTO.cs	156
Server/Bll/DTO/Attendance/AttendanceDTO.cs	156
Server/Bll/DTO/Attendance/PassiveAttendanceDTO.cs	157
Server/Bll/DTO/Class/ClassDTO.cs	157
Server/Bll/DTO/Class/CreateClassDTO.cs	158
Server/Bll/DTO/Exempt/CreateExemptDTO.cs	158
Server/Bll/DTO/Exempt/ExemptDTO.cs	159
Server/Bll/DTO/Group/GroupDTO.cs	159
Server/Bll/DTO/PassiveTag/CreatePassiveTagDTO.cs	159
Server/Bll/DTO/PassiveTag/PassiveTagDTO.cs	160
Server/Bll/DTO/PassiveTag/UpdatePassiveTagDTO.cs	161
Server/Bll/DTO/Station/StationScanDTO.cs	161
Server/Bll/DTO/Station/StationSecretKeyDTO.cs	162
Server/Bll/DTO/Student/AbsenceDTO.cs	162
Server/Bll/DTO/Student/StudentAttendanceDTO.cs	163
Server/Bll/DTO/Student/StudentDetailDTO.cs	163
Server/Bll/DTO/User/ConfirmEmailDTO.cs	163
Server/Bll/DTO/User/ForgotPasswordDTO.cs	164
Server/Bll/DTO/User/LoginResultDTO.cs	165
Server/Bll/DTO/User/LoginUserDTO.cs	165
Server/Bll/DTO/User/RegisterUserDTO.cs	166

Server/Bll/DTO/User/ResetPasswordDTO.cs	166
Server/Bll/DTO/User/UserDTO.cs	167
Server/Bll/DTO/User/UserInfoDTO.cs	168
Server/Bll/Exceptions/BadRequestException.cs	169
Server/Bll/Exceptions/DbConcurrencyException.cs	169
Server/Bll/Exceptions/NotFoundException.cs	169
Server/Bll/Exceptions/UnauthorizedException.cs	170
Server/Bll/Providers/CurrentTimeProvider.cs	170
Server/Bll/Providers/HMACTokenProvider.cs	170
Server/Bll/Providers/ICurrentTimeProvider.cs	170
Server/Bll/Providers/IStationTokenProvider.cs	171
Server/Bll/Services/AttendanceService.cs	171
Server/Bll/Services/ClassService.cs	171
Server/Bll/Services/EmailService.cs	171
Server/Bll/Services/ExemptService.cs	171
Server/Bll/Services/GroupService.cs	172
Server/Bll/Services/PassiveTagService.cs	174
Server/Bll/Services/StationService.cs	175
Server/Bll/Services/StudentService.cs	175
Server/Bll/Services/UserService.cs	175
Server/Bll/Services/Interfaces/IAttendanceService.cs	172
Server/Bll/Services/Interfaces/IClassService.cs	172
Server/Bll/Services/Interfaces/IEmailService.cs	172
Server/Bll/Services/Interfaces/IExemptService.cs	173
Server/Bll/Services/Interfaces/IGroupService.cs	173
Server/Bll/Services/Interfaces/IPassiveTagService.cs	173
Server/Bll/Services/Interfaces/IRestInterface.cs	173
Server/Bll/Services/Interfaces/IStationService.cs	173
Server/Bll/Services/Interfaces/IStudentService.cs	174
Server/Bll/Services/Interfaces/IUserService.cs	174
Server/Bll/Utils/Consts.cs	176
Server/DAL/ApplicationDbContext.cs	176
Server/DAL/MappingProfile.cs	176
Server/DAL/Migrations/20230317172813_Identity.cs	176
Server/DAL/Migrations/20230317172813_Identity.Designer.cs	177
Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.cs	177
Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.Designer.cs	177
Server/DAL/Migrations/20230501161544_AddExempts.cs	177
Server/DAL/Migrations/20230501161544_AddExempts.Designer.cs	178
Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.cs	178
Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.Designer.cs	178
Server/DAL/Migrations/ApplicationDbContextModelSnapshot.cs	179
Server/DAL/Model/Attendance.cs	179
Server/DAL/Model/Class.cs	179
Server/DAL/Model/Exempt.cs	179
Server/DAL/Model/Group.cs	180
Server/DAL/Model/PassiveTag.cs	180
Server/DAL/Model/Role.cs	180
Server/DAL/Model/Station.cs	180
Server/DAL/Model/User.cs	181

Chapter 5

Namespace Documentation

5.1 Grip Namespace Reference

Namespaces

- namespace [Api](#)
- namespace [Bll](#)
- namespace [DAL](#)
- namespace [Utils](#)

5.2 Grip.Api Namespace Reference

Namespaces

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

5.3 Grip.Api.Controllers Namespace Reference

Classes

- class [ClassController](#)
Endpoint for managing classes
- class [ExemptController](#)
Endpoint for managing exempt items
- class [PassiveTagController](#)
API controller for managing passive tags.
- class [StudentController](#)
Controller for managing students.

5.4 Grip.Api.Hubs Namespace Reference

Classes

- interface [IStationClient](#)
This interface is used to define the methods that can be called from the station hub on the client
- class [StationHub](#)
SignalR Hub for station related events

5.5 Grip.Bll Namespace Reference

Namespaces

- namespace [DTO](#)
- namespace [Exceptions](#)
- namespace [Providers](#)
- namespace [Services](#)

5.6 Grip.Bll.DTO Namespace Reference

Functions

- record [ActiveAttendanceDTO](#) ([Required][RegularExpression(@"^\d*_\d*_\d*\$")] string Message, [Required] string Token)
DTO for the active attendance registration
- record [PassiveAttendanceDTO](#) ([Required] int StationId, [Required] Int64 SerialNumber)
DTO for the passive attendance registration
- record [StudentDetailDTO](#) ()
Represents the data transfer object for a student detail.

5.6.1 Function Documentation

5.6.1.1 ActiveAttendanceDTO()

```
record Grip.Bll.DTO.ActiveAttendanceDTO (
    [RegularExpression(@"^\d*_\d*_\d*$")] string Message,
    [Required] string Token )
```

[DTO](#) for the active attendance registration

Parameters

<i>Message</i>	Message provided by the station
<i>Token</i>	Message signed by the station

5.6.1.2 PassiveAttendanceDTO()

```
record Grip.Bll.DTO.PassiveAttendanceDTO (
    [Required] int StationId,
    [Required] Int64 SerialNumber )
```

DTO for the passive attendance registration

Parameters

<i>StationId</i>	Id of the station of the scan
<i>SerialNumber</i>	Serial number of the passive NFC tag

5.6.1.3 StudentDetailDTO()

```
record Grip.Bll.DTO.StudentDetailDTO ( )
```

Represents the data transfer object for a student detail.

Id of the user

Email address of the user

Username of the user

Absences of the user

Definition at line 12 of file StudentDetailDTO.cs.

5.7 Grip.Bll.Exceptions Namespace Reference

Classes

- class [BadRequestException](#)
Exception thrown when a request is invalid: HTTP 400.
- class [DbConcurrencyException](#)
Exception thrown when a db update failed due to concurrency: HTTP 409.
- class [NotFoundException](#)
Exception thrown when a resource is not found: HTTP 404.
- class [UnauthorizedException](#)
Exception thrown when a request needs authorization: HTTP 401.

5.8 Grip.Bll.Providers Namespace Reference

Classes

- class [CurrentTimeProvider](#)
Class for abstracting the current time for testing purposes.
- interface [ICurrentTimeProvider](#)
Interface for abstracting the current time for testing purposes.

5.9 Grip.Bll.Services Namespace Reference

Namespaces

- namespace [Interfaces](#)

Classes

- class [ExemptService](#)
Service class for managing exempts.
- class [GroupService](#)
Service class for managing groups.
- class [PassiveTagService](#)
Service class for managing passive tags.
- class [StationService](#)
Provides methods for managing stations.
- class [StudentService](#)
Service for managing students
- class [UserService](#)
Represents a service for managing users.

5.10 Grip.Bll.Services.Interfaces Namespace Reference

Classes

- interface [IEmailService](#)
Represents a service for sending emails.
- interface [IRestInterface](#)
Represents a generic REST interface for CRUD operations.
- interface [IStationService](#)
Represents an interface for the station service.
- interface [IStudentService](#)
Interface for the student service
- interface [IUserService](#)
Represents an interface for the user service.

5.11 Grip.DAL Namespace Reference

Namespaces

- namespace [Migrations](#)
- namespace [Model](#)

Classes

- class [MappingProfile](#)
Mapping profile for AutoMapper.

5.12 Grip.DAL.Migrations Namespace Reference

Classes

- class [Identity](#)
- class [AddClassesAndAttendance](#)
- class [AddExempts](#)
- class [ClassStationRelationAdded](#)
- class [ApplicationDbContextModelSnapshot](#)

5.13 Grip.DAL.Model Namespace Reference

Classes

- class [Attendance](#)
Represents the attendance information of a user at a station.
- class [Exempt](#)
Represents an exemption in the database.
- class [Group](#)
Represents a group in the database.
- class [PassiveTag](#)
Represents a passive tag in the database.
- class [Station](#)
Represents a station in the database.
- class [User](#)
Represents a user in the database.

5.14 Grip.Utils Namespace Reference

Classes

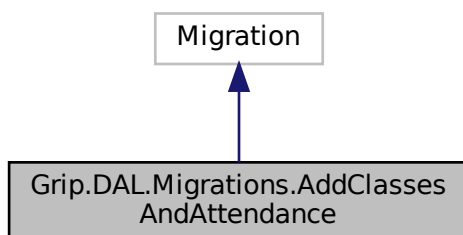
- class **Consts**
Contains constants

Chapter 6

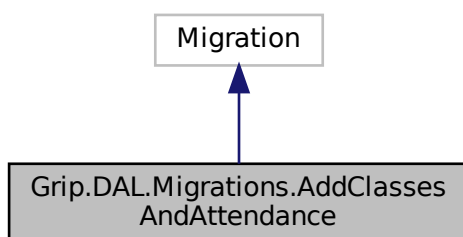
Class Documentation

6.1 Grip.DAL.Migrations.AddClassesAndAttendance Class Reference

Inheritance diagram for Grip.DAL.Migrations.AddClassesAndAttendance:



Collaboration diagram for Grip.DAL.Migrations.AddClassesAndAttendance:



Protected Member Functions

- override void [Up](#) (MigrationBuilder migrationBuilder)
- override void [Down](#) (MigrationBuilder migrationBuilder)
- override void [BuildTargetModel](#) (ModelBuilder modelBuilder)

6.1.1 Detailed Description

Definition at line 10 of file 20230411164235_AddClassesAndAttendance.cs.

6.1.2 Member Function Documentation

6.1.2.1 BuildTargetModel()

```
override void Grip.DAL.Migrations.AddClassesAndAttendance.BuildTargetModel (  
    modelBuilder modelBuilder ) [protected]
```

Definition at line 19 of file 20230411164235_AddClassesAndAttendance.Designer.cs.

6.1.2.2 Down()

```
override void Grip.DAL.Migrations.AddClassesAndAttendance.Down (  
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 174 of file 20230411164235_AddClassesAndAttendance.cs.

6.1.2.3 Up()

```
override void Grip.DAL.Migrations.AddClassesAndAttendance.Up (  
    MigrationBuilder migrationBuilder ) [protected]
```

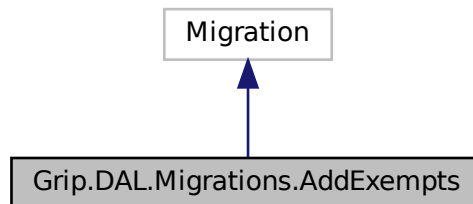
Definition at line 13 of file 20230411164235_AddClassesAndAttendance.cs.

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

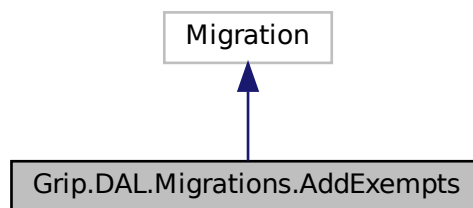
- Server/DAL/Migrations/[20230411164235_AddClassesAndAttendance.cs](#)
- Server/DAL/Migrations/[20230411164235_AddClassesAndAttendance.Designer.cs](#)

6.2 Grip.DAL.Migrations.AddExempts Class Reference

Inheritance diagram for Grip.DAL.Migrations.AddExempts:



Collaboration diagram for Grip.DAL.Migrations.AddExempts:



Protected Member Functions

- override void [Up](#) (MigrationBuilder migrationBuilder)
- override void [Down](#) (MigrationBuilder migrationBuilder)
- override void [BuildTargetModel](#) (ModelBuilder modelBuilder)

6.2.1 Detailed Description

Definition at line 10 of file 20230501161544_AddExempts.cs.

6.2.2 Member Function Documentation

6.2.2.1 BuildTargetModel()

```
override void Grip.DAL.Migrations.AddExempts.BuildTargetModel (
    modelBuilder modelBuilder ) [protected]
```

Definition at line 19 of file 20230501161544_AddExempts.Designer.cs.

6.2.2.2 Down()

```
override void Grip.DAL.Migrations.AddExempts.Down (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 83 of file 20230501161544_AddExempts.cs.

6.2.2.3 Up()

```
override void Grip.DAL.Migrations.AddExempts.Up (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 13 of file 20230501161544_AddExempts.cs.

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

- [Server/DAL/Migrations/20230501161544_AddExempts.cs](#)
- [Server/DAL/Migrations/20230501161544_AddExempts.Designer.cs](#)

6.3 ApiKeyValidationMiddleware Class Reference

Validates the api key in the header of the request.

Public Member Functions

- [ApiKeyValidationMiddleware](#) (RequestDelegate next, ILogger< [ApiKeyValidationMiddleware](#) > logger, IConfiguration configuration)
Initializes a new instance of the [ApiKeyValidationMiddleware](#) class.
- async Task [InvokeAsync](#) (HttpContext context)
Invokes the API key validation middleware.

6.3.1 Detailed Description

Validates the api key in the header of the request.

Definition at line 13 of file ApiKeyValidationMiddleware.cs.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 ApiKeyValidationMiddleware()

```
ApiKeyValidationMiddleware.ApiKeyValidationMiddleware (
    RequestDelegate next,
    ILogger< ApiKeyValidationMiddleware > logger,
    IConfiguration configuration )
```

Initializes a new instance of the [ApiKeyValidationMiddleware](#) class.

Parameters

<i>next</i>	The next middleware delegate.
<i>logger</i>	The logger.
<i>configuration</i>	The configuration.

Definition at line 25 of file ApiKeyValidationMiddleware.cs.

6.3.3 Member Function Documentation

6.3.3.1 InvokeAsync()

```
async Task ApiKeyValidationMiddleware.InvokeAsync (
    HttpContext context )
```

Invokes the API key validation middleware.

Parameters

<i>context</i>	The HTTP context.
----------------	-------------------

Returns

A task representing the asynchronous middleware invocation.

Definition at line 37 of file ApiKeyValidationMiddleware.cs.

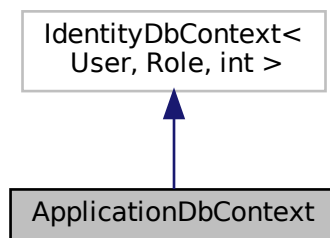
The documentation for this class was generated from the following file:

- Server/Api/Middleware/[ApiKeyValidationMiddleware.cs](#)

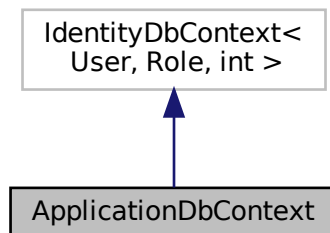
6.4 ApplicationDbContext Class Reference

Represents the application's database context.

Inheritance diagram for ApplicationDbContext:



Collaboration diagram for ApplicationDbContext:



Public Member Functions

- [ApplicationDbContext](#) (DbContextOptions< [ApplicationDbContext](#) > options, IConfiguration configuration)
Initializes a new instance of the [ApplicationDbContext](#) class.

Protected Member Functions

- override void [OnConfiguring](#) (DbContextOptionsBuilder optionsBuilder)
Configures the DbContext options before they are used to connect to the database.
- override void [OnModelCreating](#) (ModelBuilder builder)
Configures the model and relationships between entities.

Properties

- `DbSet< PassiveTag > PassiveTags = null!` [get, set]
Gets or sets the DbSet for the [PassiveTag](#) entity.
- `DbSet< Attendance > Attendances = null!` [get, set]
Gets or sets the DbSet for the [Attendance](#) entity.
- `DbSet< Group > Groups = null!` [get, set]
Gets or sets the DbSet for the [Group](#) entity.
- `DbSet< Class > Classes = null!` [get, set]
Gets or sets the DbSet for the [Class](#) entity.
- `DbSet< Station > Stations = null!` [get, set]
Gets or sets the DbSet for the [Station](#) entity.
- `DbSet< Exempt > Exempts = null!` [get, set]
Gets or sets the DbSet for the [Exempt](#) entity.

6.4.1 Detailed Description

Represents the application's database context.

Definition at line 11 of file `ApplicationDbContext.cs`.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 ApplicationDbContext()

```
ApplicationDbContext.ApplicationDbContext (
    DbContextOptions< ApplicationDbContext > options,
    IConfiguration configuration )
```

Initializes a new instance of the [ApplicationDbContext](#) class.

Parameters

<i>options</i>	The DbContext options.
<i>configuration</i>	The application configuration.

Definition at line 50 of file `ApplicationDbContext.cs`.

6.4.3 Member Function Documentation

6.4.3.1 OnConfiguring()

```
override void ApplicationDbContext.OnConfiguring (
    DbContextOptionsBuilder optionsBuilder ) [protected]
```

Configures the DbContext options before they are used to connect to the database.

Parameters

<i>optionsBuilder</i>	The options builder.
-----------------------	----------------------

Definition at line 60 of file ApplicationDbContext.cs.

6.4.3.2 OnModelCreating()

```
override void ApplicationDbContext.OnModelCreating (
    modelBuilder builder ) [protected]
```

Configures the model and relationships between entities.

Parameters

<i>builder</i>	The model builder.
----------------	--------------------

Definition at line 72 of file ApplicationDbContext.cs.

6.4.4 Property Documentation

6.4.4.1 Attendances

```
DbSet<Attendance> ApplicationDbContext.Attendances = null! [get], [set]
```

Gets or sets the DbSet for the Attendance entity.

Definition at line 21 of file ApplicationDbContext.cs.

6.4.4.2 Classes

```
DbSet<Class> ApplicationDbContext.Classes = null! [get], [set]
```

Gets or sets the DbSet for the Class entity.

Definition at line 31 of file ApplicationDbContext.cs.

6.4.4.3 Exempts

```
DbSet<Exempt> ApplicationDbContext.Exempts = null! [get], [set]
```

Gets or sets the DbSet for the Exempt entity.

Definition at line 41 of file ApplicationDbContext.cs.

6.4.4.4 Groups

```
DbSet<Group> ApplicationDbContext.Groups = null! [get], [set]
```

Gets or sets the DbSet for the Group entity.

Definition at line 26 of file ApplicationDbContext.cs.

6.4.4.5 PassiveTags

```
DbSet<PassiveTag> ApplicationDbContext.PassiveTags = null! [get], [set]
```

Gets or sets the DbSet for the PassiveTag entity.

Definition at line 16 of file ApplicationDbContext.cs.

6.4.4.6 Stations

```
DbSet<Station> ApplicationDbContext.Stations = null! [get], [set]
```

Gets or sets the DbSet for the Station entity.

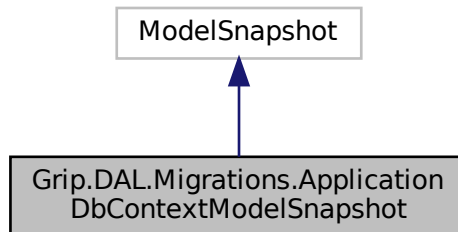
Definition at line 36 of file ApplicationDbContext.cs.

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

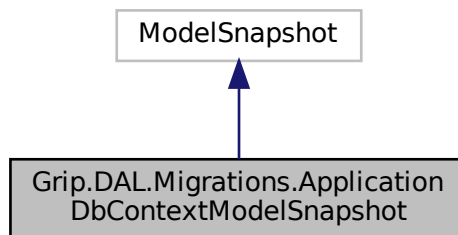
- [Server/DAL/ApplicationDbContext.cs](#)

6.5 Grip.DAL.Migrations.ApplicationDbContextModelSnapshot Class Reference

Inheritance diagram for Grip.DAL.Migrations.ApplicationDbContextModelSnapshot:



Collaboration diagram for Grip.DAL.Migrations.ApplicationDbContextModelSnapshot:



Protected Member Functions

- override void [BuildModel](#) (ModelBuilder modelBuilder)

6.5.1 Detailed Description

Definition at line 14 of file `ApplicationDbContextModelSnapshot.cs`.

6.5.2 Member Function Documentation

6.5.2.1 BuildModel()

```
override void Grip.DAL.Migrations.ApplicationDbContextModelSnapshot.BuildModel (
    modelBuilder modelBuilder ) [protected]
```

Definition at line 16 of file ApplicationDbContextModelSnapshot.cs.

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

- Server/DAL/Migrations/[ApplicationDbContextModelSnapshot.cs](#)

6.6 Grip.DAL.Model.Attendance Class Reference

Represents the attendance information of a user at a station.

Properties

- int [Id](#) [get, set]
Gets or sets the ID of the attendance record.
- [User User](#) = null! [get, set]
Gets or sets the user associated with the attendance.
- [Station Station](#) = null! [get, set]
Gets or sets the station associated with the attendance.
- DateTime [Time](#) [get, set]
Gets or sets the timestamp of the attendance.

6.6.1 Detailed Description

Represents the attendance information of a user at a station.

Definition at line 8 of file Attendance.cs.

6.6.2 Property Documentation

6.6.2.1 Id

```
int Grip.DAL.Model.Attendance.Id [get], [set]
```

Gets or sets the ID of the attendance record.

Definition at line 13 of file Attendance.cs.

6.6.2.2 Station

```
Station Grip.DAL.Model.Attendance.Station = null! [get], [set]
```

Gets or sets the station associated with the attendance.

Definition at line 23 of file Attendance.cs.

6.6.2.3 Time

```
DateTime Grip.DAL.Model.Attendance.Time [get], [set]
```

Gets or sets the timestamp of the attendance.

Definition at line 28 of file Attendance.cs.

6.6.2.4 User

```
User Grip.DAL.Model.Attendance.User = null! [get], [set]
```

Gets or sets the user associated with the attendance.

Definition at line 18 of file Attendance.cs.

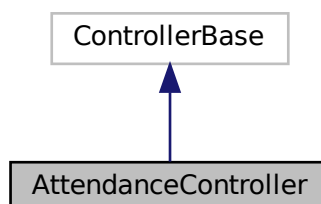
The documentation for this class was generated from the following file:

- Server/DAL/Model/[Attendance.cs](#)

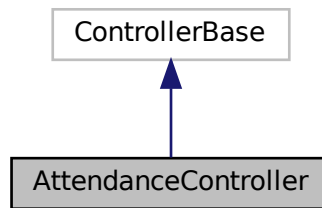
6.7 AttendanceController Class Reference

Controller for handling attendance

Inheritance diagram for AttendanceController:



Collaboration diagram for AttendanceController:



Public Member Functions

- [AttendanceController](#) (ILogger< [AttendanceController](#) > logger, UserManager< DAL.Model.User > userManager, IConfiguration configuration, [ApplicationDbContext](#) context, [IAttendanceService](#) attendanceService)
Constructor for [AttendanceController](#)
- async Task< IActionResult > [ActiveAuthentication](#) ([FromBody] ActiveAttendanceDTO request)
This endpoint is used to authenticate a user when they are physically present at a station with their phone as identification method
- async Task< IActionResult > [PassiveAuthentication](#) ([FromBody] PassiveAttendanceDTO request, [FromHeader] string apiKey)
This endpoint is used to authenticate a user when they are physically present at a station with their passive id tag as identification method
- async Task< IActionResult > [GetAttendanceForDay](#) (DateOnly date)
This endpoint is used to get the logged in users classes for given day and whether they are present or not

6.7.1 Detailed Description

Controller for handling attendance

Definition at line 19 of file AttendanceController.cs.

6.7.2 Constructor & Destructor Documentation

6.7.2.1 AttendanceController()

```

AttendanceController.AttendanceController (
    ILogger< AttendanceController > logger,
    UserManager< DAL.Model.User > userManager,
    IConfiguration configuration,
    ApplicationDbContext context,
    IAttendanceService attendanceService )
  
```

Constructor for [AttendanceController](#)

Parameters

<i>logger</i>	Logger for logging
<i>userManager</i>	User manager
<i>configuration</i>	Configuration
<i>context</i>	Database context
<i>attendanceService</i>	Attendance service

Definition at line 35 of file AttendanceController.cs.

6.7.3 Member Function Documentation

6.7.3.1 ActiveAuthentication()

```
async Task<IActionResult> AttendanceController.ActiveAuthentication (
    [FromBody] ActiveAttendanceDTO request )
```

This endpoint is used to authenticate a user when they are physically present at a station with their phone as identification method

Definition at line 52 of file AttendanceController.cs.

6.7.3.2 GetAttendanceForDay()

```
async Task<IActionResult> AttendanceController.GetAttendanceForDay (
    DateOnly date )
```

This endpoint is used to get the logged in users classes for given day and wetaher they are present or not

Definition at line 92 of file AttendanceController.cs.

6.7.3.3 PassiveAuthentication()

```
async Task<IActionResult> AttendanceController.PassiveAuthentication (
    [FromBody] PassiveAttendanceDTO request,
    [FromHeader] string apiKey )
```

This endpoint is used to authenticate a user when they are physically present at a station with their passive id tag as identification method

Definition at line 75 of file AttendanceController.cs.

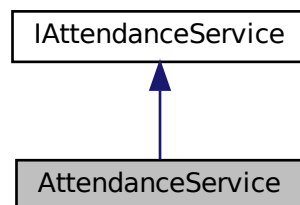
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[AttendanceController.cs](#)

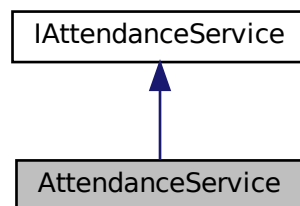
6.8 AttendanceService Class Reference

Provides functionality related to attendance management.

Inheritance diagram for AttendanceService:



Collaboration diagram for AttendanceService:



Public Member Functions

- `AttendanceService` (`ILogger< AttendanceService > logger`, `ApplicationDbContext context`, `IStationTokenProvider stationTokenProvider`, `IMapper mapper`, `IHubContext< StationHub, IStationClient > signalrHub`, `ICurrentTimeProvider currentTimeProvider`)
Initializes a new instance of the `AttendanceService` class.
- `async Task VerifyPhoneScan` (`ActiveAttendanceDTO request`, `DAL.Model.User user`)
Verifies phone scan attendance for a user.
- `async Task VerifyPassiveScan` (`PassiveAttendanceDTO request`)
Verifies passive scan attendance for a user.
- `async Task< IEnumerable< AttendanceDTO > > GetAttendanceForDay` (`User user`, `DateOnly date`)
Retrieves attendance information for a user on a specific day.

6.8.1 Detailed Description

Provides functionality related to attendance management.

Definition at line 17 of file AttendanceService.cs.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 AttendanceService()

```
AttendanceService.AttendanceService (
    ILogger< AttendanceService > logger,
    ApplicationDbContext context,
    IStationTokenProvider stationTokenProvider,
    IMapper mapper,
    IHubContext< StationHub, IStationClient > signalrHub,
    ICurrentTimeProvider currentTimeProvider )
```

Initializes a new instance of the [AttendanceService](#) class.

Parameters

<i>logger</i>	The logger instance.
<i>context</i>	The application database context.
<i>stationTokenProvider</i>	The station token provider.
<i>mapper</i>	The mapper instance.
<i>signalrHub</i>	The SignalR hub context.
<i>currentTimeProvider</i>	The current time provider.

Definition at line 35 of file AttendanceService.cs.

6.8.3 Member Function Documentation

6.8.3.1 GetAttendanceForDay()

```
async Task<IEnumerable<AttendanceDTO> > AttendanceService.GetAttendanceForDay (
    User user,
    DateOnly date )
```

Retrieves attendance information for a user on a specific day.

Parameters

<i>user</i>	The user to retrieve attendance information for.
<i>date</i>	The date to retrieve attendance information for.

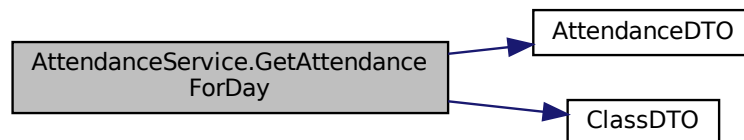
Returns

A collection of attendance information for the user on the specified day.

Implements [IAttendanceService](#).

Definition at line 131 of file AttendanceService.cs.

Here is the call graph for this function:

**6.8.3.2 VerifyPassiveScan()**

```
async Task AttendanceService.VerifyPassiveScan (
    PassiveAttendanceDTO request )
```

Verifies passive scan attendance for a user.

Parameters

<i>request</i>	The passive attendance request data.
----------------	--------------------------------------

Implements [IAttendanceService](#).

Definition at line 97 of file AttendanceService.cs.

Here is the call graph for this function:



6.8.3.3 VerifyPhoneScan()

```
async Task AttendanceService.VerifyPhoneScan (
    ActiveAttendanceDTO request,
    DAL.Model.User user )
```

Verifies phone scan attendance for a user.

Parameters

<i>request</i>	The active attendance request data.
<i>user</i>	The user associated with the attendance request.

Definition at line 50 of file AttendanceService.cs.

Here is the call graph for this function:



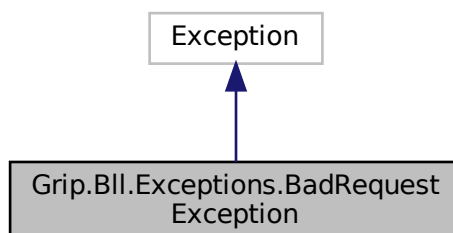
The documentation for this class was generated from the following file:

- Server/Bll/Services/[AttendanceService.cs](#)

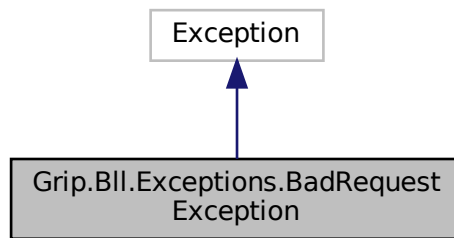
6.9 Grip.Bll.Exceptions.BadRequestException Class Reference

Exception thrown when a request is invalid: HTTP 400.

Inheritance diagram for `Grip.Bll.Exceptions.BadRequestException`:



Collaboration diagram for Grip.Bll.Exceptions.BadRequestException:



Public Member Functions

- [BadRequestException](#) (string? message=null)
Creates an instance of the exception.

6.9.1 Detailed Description

Exception thrown when a request is invalid: HTTP 400.

Definition at line 11 of file `BadRequestException.cs`.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 BadRequestException()

```
Grip.Bll.Exceptions.BadRequestException.BadRequestException (
    string? message = null )
```

Creates an instance of the exception.

Parameters

<i>message</i>	Description of the cause of the exception
----------------	---

Definition at line 17 of file `BadRequestException.cs`.

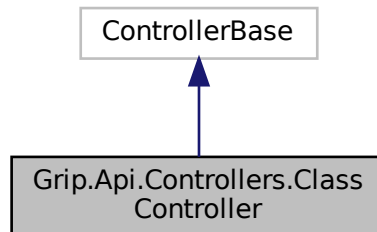
The documentation for this class was generated from the following file:

- `Server/Bll/Exceptions/BadRequestException.cs`

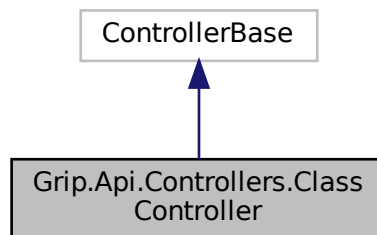
6.10 Grip.Api.Controllers.ClassController Class Reference

Endpoint for managing classes

Inheritance diagram for Grip.Api.Controllers.ClassController:



Collaboration diagram for Grip.Api.Controllers.ClassController:



Public Member Functions

- [ClassController](#) ([ApplicationDbContext](#) context, UserManager< [User](#) > userManager, IMapper mapper, [IClassService](#) classService)
Constructor for the class controller
- async Task< ActionResult< IEnumerable< [ClassDTO](#) > > > [GetClass](#) ()
Get all classes
- async Task< ActionResult< [ClassDTO](#) > > [GetClass](#) (int id)
Get a specific class
- async Task< IActionResult > [PutClass](#) (int id, [FromBody] [ClassDTO](#) @class)
Update a class
- async Task< ActionResult< [ClassDTO](#) > > [PostClass](#) ([FromBody] CreateClassDTO @class)
Creates a new class
- async Task< IActionResult > [DeleteClass](#) (int id)
Deletes a class
- async Task< ActionResult< IEnumerable< [ClassDTO](#) > > > [GetClassOnDay](#) (DateOnly date)
Gets all classes for the logged in user on the given day

6.10.1 Detailed Description

Endpoint for managing classes

Definition at line 23 of file ClassController.cs.

6.10.2 Constructor & Destructor Documentation

6.10.2.1 ClassController()

```
Grip.Api.Controllers.ClassController.ClassController (
    ApplicationDbContext context,
    UserManager< User > userManager,
    IMapper mapper,
    IClassService classService )
```

Constructor for the class controller

Parameters

<i>context</i>	Database context
<i>userManager</i>	User manager
<i>mapper</i>	Auto mappre
<i>classService</i>	Class service

Definition at line 37 of file ClassController.cs.

6.10.3 Member Function Documentation

6.10.3.1 DeleteClass()

```
async Task<IActionResult> Grip.Api.Controllers.ClassController.DeleteClass (
    int id )
```

Deletes a class

Parameters

<i>id</i>	Identifyer of the class
-----------	-------------------------

Definition at line 123 of file ClassController.cs.

6.10.3.2 GetClass() [1/2]

```
async Task<ActionResult<IEnumerable<ClassDTO> > > Grip.Api.Controllers.ClassController.Get←  
Class ( )
```

Get all classes

Returns

Classes

Definition at line 53 of file ClassController.cs.

6.10.3.3 GetClass() [2/2]

```
async Task<ActionResult<ClassDTO> > Grip.Api.Controllers.ClassController.GetClass (   
    int id )
```

Get a specific class

Parameters

<i>id</i>	Identifyer of the class
-----------	-------------------------

Returns

A single class

Definition at line 69 of file ClassController.cs.

6.10.3.4 GetClassOnDay()

```
async Task<ActionResult<IEnumerable<ClassDTO> > > Grip.Api.Controllers.ClassController.Get←  
ClassOnDay (   
    DateOnly date )
```

Gets all classes for the logged in user on the given day

Parameters

<i>date</i>	The date to get classes for
-------------	-----------------------------

Returns

Classes for the user on the given day

Definition at line 139 of file ClassController.cs.

6.10.3.5 PostClass()

```
async Task<ActionResult<ClassDTO> > Grip.Api.Controllers.ClassController.PostClass (
    [FromBody] CreateClassDTO @ class )
```

Creates a new class

Parameters

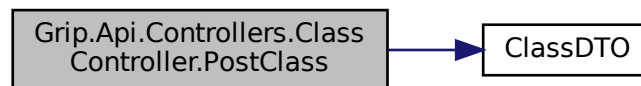
<i>class</i>	The class object
--------------	------------------

Returns

A newly created class

Definition at line 107 of file ClassController.cs.

Here is the call graph for this function:



6.10.3.6 PutClass()

```
async Task<IActionResult> Grip.Api.Controllers.ClassController.PutClass (
    int id,
    [FromBody] ClassDTO @ class )
```

Update a class

Parameters

<i>id</i>	Identifyer of the class
<i>class</i>	The class object

Definition at line 86 of file ClassController.cs.

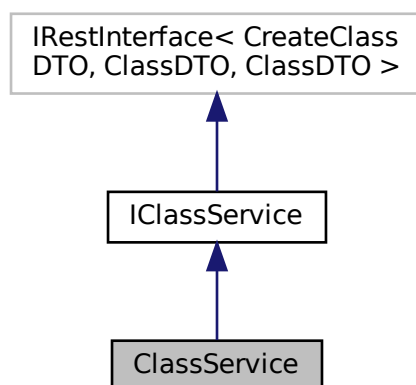
The documentation for this class was generated from the following file:

- [Server/Api/Controllers/ClassController.cs](#)

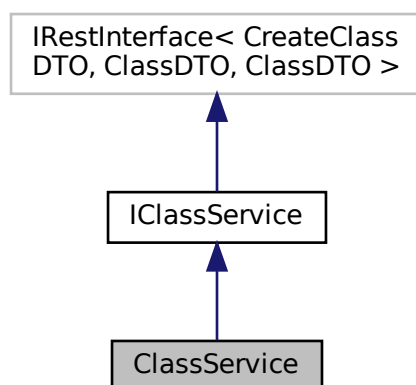
6.11 ClassService Class Reference

Service class for managing classes.

Inheritance diagram for ClassService:



Collaboration diagram for ClassService:



Public Member Functions

- [ClassService](#) ([ApplicationDbContext](#) context, IMapper mapper, UserManager< [User](#) > userManager)
Initializes a new instance of the [ClassService](#) class.
- async Task< [ClassDTO](#) > [Create](#) (CreateClassDTO dto)
Creates a new class.
- async Task [Delete](#) (int id)
Deletes a class by its ID.
- async Task< [ClassDTO](#) > [Get](#) (int id)
Retrieves a class by its ID.
- async Task< IEnumerable< [ClassDTO](#) > > [GetAll](#) ()
Retrieves all classes.
- async Task< IEnumerable< [ClassDTO](#) > > [GetClassesForUserOnDay](#) ([User](#) user, DateOnly date)
Retrieves classes for a user on a specific day.
- async Task [Update](#) ([ClassDTO](#) dto)
Updates a class.

6.11.1 Detailed Description

Service class for managing classes.

Definition at line 15 of file ClassService.cs.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 ClassService()

```
ClassService.ClassService (
    ApplicationDbContext context,
    IMapper mapper,
    UserManager< User > userManager )
```

Initializes a new instance of the [ClassService](#) class.

Parameters

<i>context</i>	The application database context.
<i>mapper</i>	The AutoMapper instance.
<i>userManager</i>	The user manager instance.

Definition at line 27 of file ClassService.cs.

6.11.3 Member Function Documentation

6.11.3.1 Create()

```
async Task<ClassDTO> ClassService.Create (
    CreateClassDTO dto )
```

Creates a new class.

Parameters

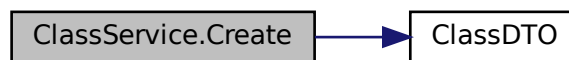
<i>dto</i>	The DTO for creating a class.
------------	-------------------------------

Returns

The created class DTO.

Definition at line 38 of file ClassService.cs.

Here is the call graph for this function:



6.11.3.2 Delete()

```
async Task ClassService.Delete (
    int id )
```

Deletes a class by its ID.

Parameters

<i>id</i>	The ID of the class to delete.
-----------	--------------------------------

Definition at line 57 of file ClassService.cs.

6.11.3.3 Get()

```
async Task<ClassDTO> ClassService.Get (
    int id )
```

Retrieves a class by its ID.

Parameters

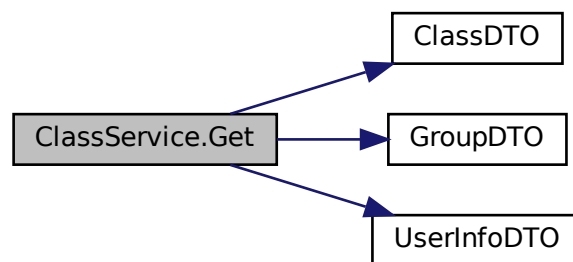
<i>id</i>	The ID of the class to retrieve.
-----------	----------------------------------

Returns

The class DTO.

Definition at line 73 of file ClassService.cs.

Here is the call graph for this function:

**6.11.3.4 GetAll()**

```
async Task<IEnumerable<ClassDTO> > ClassService.GetAll ( )
```

Retrieves all classes.

Returns

A list of class DTOs.

Definition at line 92 of file ClassService.cs.

Here is the call graph for this function:



6.11.3.5 GetClassesForUserOnDay()

```
async Task<IEnumerable<ClassDTO> > ClassService.GetClassesForUserOnDay (
    User user,
    DateOnly date )
```

Retrieves classes for a user on a specific day.

Parameters

<i>user</i>	The user.
<i>date</i>	The date.

Returns

A list of class DTOs.

Implements [IClassService](#).

Definition at line 108 of file ClassService.cs.

6.11.3.6 Update()

```
async Task ClassService.Update (
    ClassDTO dto )
```

Updates a class.

Parameters

<i>dto</i>	The updated class DTO.
------------	------------------------

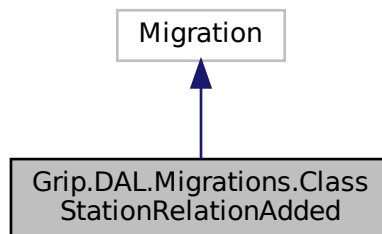
Definition at line 124 of file ClassService.cs.

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

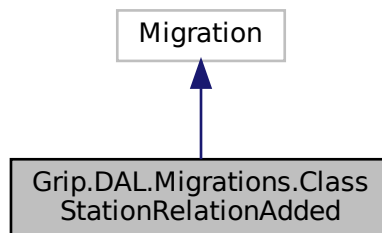
- Server/BI/Services/[ClassService.cs](#)

6.12 Grip.DAL.Migrations.ClassStationRelationAdded Class Reference

Inheritance diagram for Grip.DAL.Migrations.ClassStationRelationAdded:



Collaboration diagram for Grip.DAL.Migrations.ClassStationRelationAdded:



Protected Member Functions

- override void [Up](#) (MigrationBuilder migrationBuilder)
- override void [Down](#) (MigrationBuilder migrationBuilder)
- override void [BuildTargetModel](#) (ModelBuilder modelBuilder)

6.12.1 Detailed Description

Definition at line 8 of file 20230511180151_ClassStationRelationAdded.cs.

6.12.2 Member Function Documentation

6.12.2.1 BuildTargetModel()

```
override void Grip.DAL.Migrations.ClassStationRelationAdded.BuildTargetModel (
    modelBuilder modelBuilder ) [protected]
```

Definition at line 19 of file 20230511180151_ClassStationRelationAdded.Designer.cs.

6.12.2.2 Down()

```
override void Grip.DAL.Migrations.ClassStationRelationAdded.Down (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 35 of file 20230511180151_ClassStationRelationAdded.cs.

6.12.2.3 Up()

```
override void Grip.DAL.Migrations.ClassStationRelationAdded.Up (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 11 of file 20230511180151_ClassStationRelationAdded.cs.

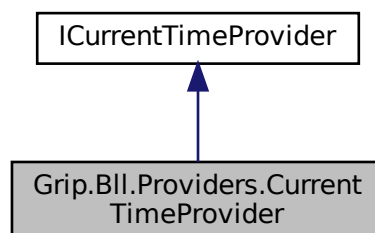
The documentation for this class was generated from the following files:

- [Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.cs](#)
- [Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.Designer.cs](#)

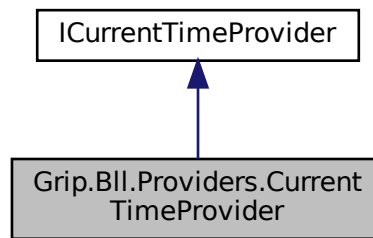
6.13 Grip.Bll.Providers.CurrentTimeProvider Class Reference

Class for abstracting the current time for testing purposes.

Inheritance diagram for Grip.Bll.Providers.CurrentTimeProvider:



Collaboration diagram for Grip.Bll.Providers.CurrentTimeProvider:



Properties

- `DateTime Now` [get]
Current time in UTC.

6.13.1 Detailed Description

Class for abstracting the current time for testing purposes.

Definition at line 11 of file `CurrentTimeProvider.cs`.

6.13.2 Property Documentation

6.13.2.1 Now

```
DateTime Grip.Bll.Providers.CurrentTimeProvider.Now [get]
```

Current time in UTC.

Definition at line 16 of file `CurrentTimeProvider.cs`.

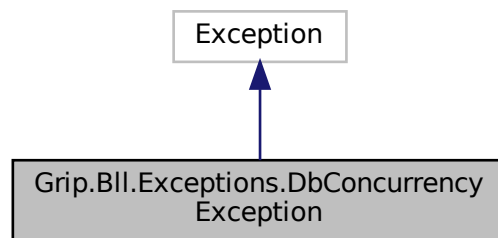
The documentation for this class was generated from the following file:

- `Server/Bll/Providers/CurrentTimeProvider.cs`

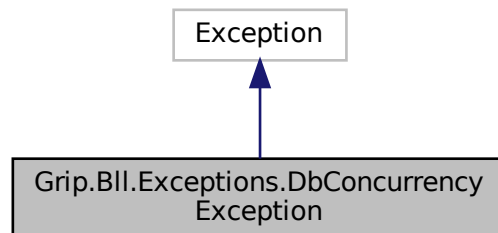
6.14 Grip.Bll.Exceptions.DbConcurrencyException Class Reference

Exception thrown when a db update failed due to concurrency: HTTP 409.

Inheritance diagram for Grip.Bll.Exceptions.DbConcurrencyException:



Collaboration diagram for Grip.Bll.Exceptions.DbConcurrencyException:



Public Member Functions

- [DbConcurrencyException](#) (string? message=null)
Creates an instance of the exception.

6.14.1 Detailed Description

Exception thrown when a db update failed due to concurrency: HTTP 409.

Definition at line 11 of file `DbConcurrencyException.cs`.

6.14.2 Constructor & Destructor Documentation

6.14.2.1 DbConcurrencyException()

```
Grip.Bll.Exceptions.DbConcurrencyException.DbConcurrencyException (
    string? message = null )
```

Creates an instance of the exception.

Parameters

<i>message</i>	Description of the cause of the exception
----------------	---

Definition at line 17 of file DbConcurrencyException.cs.

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

- Server/Bll/Exceptions/[DbConcurrencyException.cs](#)

6.15 DeChunkingMiddleware Class Reference

Middleware to de-chunk HTTP requests, because ucontroller doesn't support chunked responses

Public Member Functions

- [DeChunkingMiddleware](#) (RequestDelegate next, ILogger< [DeChunkingMiddleware](#) > logger)
Initializes a new instance of the [DeChunkingMiddleware](#) class.
- async Task [InvokeAsync](#) (HttpContext context)
Invokes the API key validation middleware.

6.15.1 Detailed Description

Middleware to de-chunk HTTP requests, because ucontroller doesn't support chunked responses

Definition at line 12 of file DeChunkingMiddleware.cs.

6.15.2 Constructor & Destructor Documentation

6.15.2.1 DeChunkingMiddleware()

```
DeChunkingMiddleware.DeChunkingMiddleware (
    RequestDelegate next,
    ILogger< DeChunkingMiddleware > logger )
```

Initializes a new instance of the [DeChunkingMiddleware](#) class.

Parameters

<i>next</i>	The next middleware delegate.
<i>logger</i>	The logger.

Definition at line 21 of file DeChunkingMiddleware.cs.

6.15.3 Member Function Documentation

6.15.3.1 InvokeAsync()

```
async Task DeChunkingMiddleware.InvokeAsync (
    HttpContext context )
```

Invokes the API key validation middleware.

Parameters

<i>context</i>	The HTTP context.
----------------	-------------------

Returns

A task representing the asynchronous middleware invocation.

Definition at line 32 of file DeChunkingMiddleware.cs.

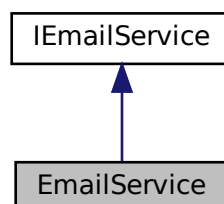
The documentation for this class was generated from the following file:

- Server/Api/Middleware/[DeChunkingMiddleware.cs](#)

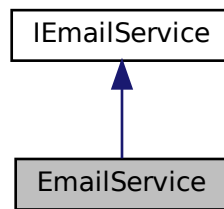
6.16 EmailService Class Reference

Service class for sending emails.

Inheritance diagram for EmailService:



Collaboration diagram for EmailService:



Public Member Functions

- [EmailService](#) (IConfiguration configuration, ILogger< [EmailService](#) > logger)
Initializes a new instance of the [EmailService](#) class.
- async Task [SendEmailAsync](#) (string to, string subject, string message)
Sends an email asynchronously to a single recipient.
- async Task [SendEmailAsync](#) (IEnumerable< string > to, string subject, string message)
Sends an email asynchronously to multiple recipients.

6.16.1 Detailed Description

Service class for sending emails.

Definition at line 8 of file EmailService.cs.

6.16.2 Constructor & Destructor Documentation

6.16.2.1 EmailService()

```
EmailService.EmailService (
    IConfiguration configuration,
    ILogger< EmailService > logger )
```

Initializes a new instance of the [EmailService](#) class.

Parameters

<i>configuration</i>	The configuration object for accessing SMTP settings.
<i>logger</i>	The logger object for logging.

Definition at line 18 of file EmailService.cs.

6.16.3 Member Function Documentation

6.16.3.1 SendEmailAsync() [1/2]

```
async Task EmailService.SendEmailAsync (
    IEnumerable< string > to,
    string subject,
    string message )
```

Sends an email asynchronously to multiple recipients.

Parameters

<i>to</i>	The recipient email addresses.
<i>subject</i>	The email subject.
<i>message</i>	The email message body.

Implements [Grip.Bll.Services.Interfaces.IEmailService](#).

Definition at line 41 of file EmailService.cs.

6.16.3.2 SendEmailAsync() [2/2]

```
async Task EmailService.SendEmailAsync (
    string to,
    string subject,
    string message )
```

Sends an email asynchronously to a single recipient.

Parameters

<i>to</i>	The recipient's email address.
<i>subject</i>	The email subject.
<i>message</i>	The email message body.

Implements [Grip.Bll.Services.Interfaces.IEmailService](#).

Definition at line 30 of file EmailService.cs.

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

- Server/Bll/Services/[EmailService.cs](#)

6.17 Grip.DAL.Model.Exempt Class Reference

Represents an exemption in the database.

Properties

- `int Id` [get, set]
Gets or sets the ID of the exemption.
- `virtual User IssuedBy = null!` [get, set]
Gets or sets the user who issued the exemption.
- `int IssuedById` [get, set]
Gets or sets the ID of the user who issued the exemption.
- `virtual User IssuedTo = null!` [get, set]
Gets or sets the user to whom the exemption is issued.
- `int IssuedToId` [get, set]
Gets or sets the ID of the user to whom the exemption is issued.
- `DateTime ValidFrom` [get, set]
Gets or sets the date and time from which the exemption is valid.
- `DateTime ValidTo` [get, set]
Gets or sets the date and time until which the exemption is valid.

6.17.1 Detailed Description

Represents an exemption in the database.

Definition at line 8 of file Exempt.cs.

6.17.2 Property Documentation

6.17.2.1 Id

```
int Grip.DAL.Model.Exempt.Id [get], [set]
```

Gets or sets the ID of the exemption.

Definition at line 13 of file Exempt.cs.

6.17.2.2 IssuedBy

```
virtual User Grip.DAL.Model.Exempt.IssuedBy = null! [get], [set]
```

Gets or sets the user who issued the exemption.

Definition at line 18 of file Exempt.cs.

6.17.2.3 IssuedById

```
int Grip.DAL.Model.Exempt.IssuedById [get], [set]
```

Gets or sets the ID of the user who issued the exemption.

Definition at line 23 of file Exempt.cs.

6.17.2.4 IssuedTo

```
virtual User Grip.DAL.Model.Exempt.IssuedTo = null! [get], [set]
```

Gets or sets the user to whom the exemption is issued.

Definition at line 28 of file Exempt.cs.

6.17.2.5 IssuedToId

```
int Grip.DAL.Model.Exempt.IssuedToId [get], [set]
```

Gets or sets the ID of the user to whom the exemption is issued.

Definition at line 33 of file Exempt.cs.

6.17.2.6 ValidFrom

```
DateTime Grip.DAL.Model.Exempt.ValidFrom [get], [set]
```

Gets or sets the date and time from which the exemption is valid.

Definition at line 38 of file Exempt.cs.

6.17.2.7 ValidTo

```
DateTime Grip.DAL.Model.Exempt.ValidTo [get], [set]
```

Gets or sets the date and time until which the exemption is valid.

Definition at line 43 of file Exempt.cs.

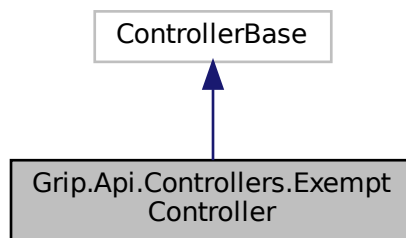
The documentation for this class was generated from the following file:

- Server/DAL/Model/[Exempt.cs](#)

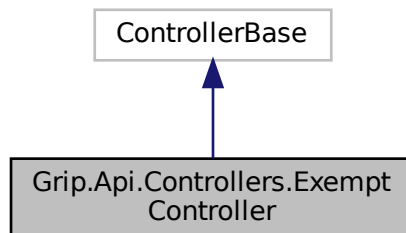
6.18 Grip.Api.Controllers.ExemptController Class Reference

Endpoint for managing exempt items

Inheritance diagram for Grip.Api.Controllers.ExemptController:



Collaboration diagram for Grip.Api.Controllers.ExemptController:



Public Member Functions

- [ExemptController](#) ([ApplicationDbContext](#) context, IMapper mapper, UserManager< [User](#) > userManager, RoleManager< [Role](#) > roleManager, [IExemptService](#) exemptService)
Constructor for the exempt controller
- async Task< ActionResult< IEnumerable< ExemptDTO > > > [GetExempt](#) ()
Retrieves a list of exempt items.
- async Task< ActionResult< ExemptDTO > > [GetExempt](#) (int id)
Get a specific exempt
- async Task< ActionResult< ExemptDTO > > [PostExempt](#) (CreateExemptDTO exempt)
Creates an exempt
- async Task< IActionResult > [DeleteExempt](#) (int id)
Deletes an exempt

6.18.1 Detailed Description

Endpoint for managing exempt items

Definition at line 19 of file ExemptController.cs.

6.18.2 Constructor & Destructor Documentation

6.18.2.1 ExemptController()

```
Grip.Api.Controllers.ExemptController.ExemptController (
    ApplicationDbContext context,
    IMapper mapper,
    UserManager< User > userManager,
    RoleManager< Role > roleManager,
    IExemptService exemptService )
```

Constructor for the exempt controller

Parameters

<i>context</i>	Database context
<i>mapper</i>	Auto mapper
<i>userManager</i>	User manager
<i>roleManager</i>	Role manager
<i>exemptService</i>	Exempt service

Definition at line 35 of file ExemptController.cs.

6.18.3 Member Function Documentation

6.18.3.1 DeleteExempt()

```
async Task<IActionResult> Grip.Api.Controllers.ExemptController.DeleteExempt (
    int id )
```

Deletes an exempt

Parameters

<i>id</i>	Id of exempt to delete
-----------	------------------------

Definition at line 117 of file ExemptController.cs.

6.18.3.2 GetExempt() [1/2]

```
async Task<ActionResult<IEnumerable<ExemptDTO> > > Grip.Api.Controllers.ExemptController.  
GetExempt ( )
```

Retrieves a list of exempt items.

Returns

An ActionResult containing the list of ExemptDTO items.

This function is accessible via HTTP GET request. The user must be authorized to access this endpoint. Returns 200 OK if the request is successful. Returns 401 Unauthorized if the user is not authorized to access the endpoint.

Definition at line 60 of file ExemptController.cs.

6.18.3.3 GetExempt() [2/2]

```
async Task<ActionResult<ExemptDTO> > Grip.Api.Controllers.ExemptController.GetExempt (   
    int id )
```

Get a specific exempt

Parameters

<i>id</i>	Identifyer of the exempt
-----------	--------------------------

Returns

A single exempt

Definition at line 80 of file ExemptController.cs.

6.18.3.4 PostExempt()

```
async Task<ActionResult<ExemptDTO> > Grip.Api.Controllers.ExemptController.PostExempt (   
    CreateExemptDTO exempt )
```

Creates an exempt

Parameters

<i>exempt</i>	Exempt to create
---------------	------------------

Returns

Created exempt

Definition at line 101 of file ExemptController.cs.

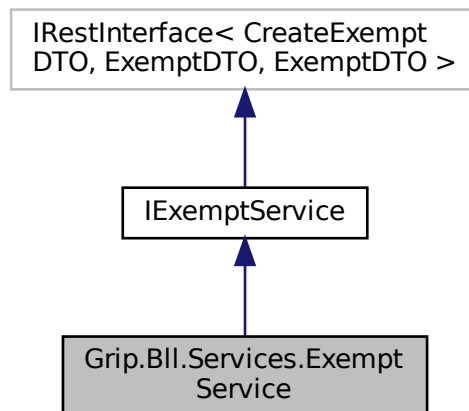
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[ExemptController.cs](#)

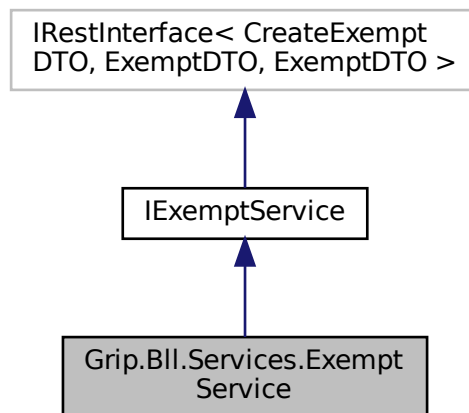
6.19 Grip.Bll.Services.ExemptService Class Reference

Service class for managing exempts.

Inheritance diagram for Grip.Bll.Services.ExemptService:



Collaboration diagram for Grip.Bll.Services.ExemptService:



Public Member Functions

- [ExemptService](#) (IMapper mapper, [ApplicationDbContext](#) context, UserManager< [User](#) > userManager)
Initializes a new instance of the [ExemptService](#) class.
- Task< ExemptDTO > [Create](#) (CreateExemptDTO dto)
Not used, because user check is required for authorization, only here for interface implementation
- async Task< ExemptDTO > [Create](#) (CreateExemptDTO dto, [User](#) teacher)
Creates a new exempt.
- async Task [Delete](#) (int id)
Deletes an exempt by its ID.
- async Task< ExemptDTO > [Get](#) (int id)
Gets an exempt by its ID (doesn't check if the user is authorized to read it).
- async Task< ExemptDTO > [GetByUser](#) (int id, [User](#) user)
Gets an exempt by its ID and checks if the user is authorized to read it.
- async Task< IEnumerable< ExemptDTO > > [GetAll](#) ()
Gets all exempts.
- async Task< IEnumerable< ExemptDTO > > [GetAllForUser](#) ([User](#) user)
Gets all exempts for a specific user.
- Task [Update](#) (ExemptDTO dto)
Updates an exempt (not implemented).

6.19.1 Detailed Description

Service class for managing exempts.

Definition at line 19 of file ExemptService.cs.

6.19.2 Constructor & Destructor Documentation

6.19.2.1 ExemptService()

```
Grip.Bll.Services.ExemptService.ExemptService (
    IMapper mapper,
    ApplicationDbContext context,
    UserManager< User > userManager )
```

Initializes a new instance of the [ExemptService](#) class.

Parameters

<i>mapper</i>	The AutoMapper instance.
<i>context</i>	The application database context.
<i>userManager</i>	The user manager instance.

Definition at line 31 of file ExemptService.cs.

6.19.3 Member Function Documentation

6.19.3.1 Create() [1/2]

```
Task<ExemptDTO> Grip.Bll.Services.ExemptService.Create (
    CreateExemptDTO dto )
```

Not used, because user check is required for authorization, only here for interface implementation

Definition at line 40 of file ExemptService.cs.

6.19.3.2 Create() [2/2]

```
async Task<ExemptDTO> Grip.Bll.Services.ExemptService.Create (
    CreateExemptDTO dto,
    User teacher )
```

Creates a new exempt.

Parameters

<i>dto</i>	The DTO for creating an exempt.
<i>teacher</i>	The teacher who issued the exempt.

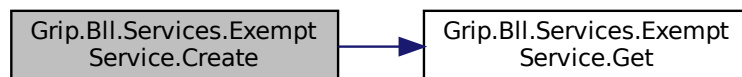
Returns

The created exempt [DTO](#).

Implements [IExemptService](#).

Definition at line 51 of file ExemptService.cs.

Here is the call graph for this function:

**6.19.3.3 Delete()**

```
async Task Grip.Bll.Services.ExemptService.Delete (
    int id )
```

Deletes an exempt by its ID.

Parameters

<i>id</i>	The ID of the exempt to delete.
-----------	---------------------------------

Definition at line 67 of file ExemptService.cs.

6.19.3.4 Get()

```
async Task<ExemptDTO> Grip.Bll.Services.ExemptService.Get (
    int id )
```

Gets an exempt by its ID (doesn't check if the user is authorized to read it).

Parameters

<i>id</i>	The ID of the exempt to retrieve.
-----------	-----------------------------------

Returns

The exempt [DTO](#).

Definition at line 85 of file ExemptService.cs.

6.19.3.5 GetAll()

```
async Task<IEnumerable<ExemptDTO> > Grip.Bll.Services.ExemptService.GetAll ( )
```

Gets all exempts.

Returns

A list of all exempt DTOs.

Definition at line 116 of file ExemptService.cs.

6.19.3.6 GetAllForUser()

```
async Task<IEnumerable<ExemptDTO> > Grip.Bll.Services.ExemptService.GetAllForUser (
    User user )
```

Gets all exempts for a specific user.

Parameters

<i>user</i>	The user for whom to retrieve the exempts.
-------------	--

Returns

A list of exempt DTOs for the user.

Definition at line 126 of file ExemptService.cs.

6.19.3.7 GetByUser()

```
async Task<ExemptDTO> Grip.Bll.Services.ExemptService.GetByUser (
    int id,
    User user )
```

Gets an exempt by its ID and checks if the user is authorized to read it.

Parameters

<i>id</i>	The ID of the exempt to retrieve.
<i>user</i>	The user making the request.

Returns

The exempt [DTO](#).

Definition at line 96 of file ExemptService.cs.

6.19.3.8 Update()

```
Task Grip.Bll.Services.ExemptService.Update (
    ExemptDTO dto )
```

Updates an exempt (not implemented).

Parameters

<i>dto</i>	The updated exempt DTO .
------------	--

Definition at line 139 of file ExemptService.cs.

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

- Server/Bll/Services/[ExemptService.cs](#)

6.20 Grip.DAL.Model.Group Class Reference

Represents a group in the database.

Properties

- int [Id](#) [get, set]
Gets or sets the ID of the group.
- ICollection< [User](#) > [Users](#) = null! [get, set]
Gets or sets the collection of users associated with the group.
- string [Name](#) = null! [get, set]
Gets or sets the name of the group.
- ICollection< [Class](#) > [Classes](#) = null! [get, set]
Gets or sets the collection of classes associated with the group.

6.20.1 Detailed Description

Represents a group in the database.

Definition at line 10 of file Group.cs.

6.20.2 Property Documentation

6.20.2.1 Classes

```
ICollection<Class> Grip.DAL.Model.Group.Classes = null! [get], [set]
```

Gets or sets the collection of classes associated with the group.

Definition at line 32 of file Group.cs.

6.20.2.2 Id

```
int Grip.DAL.Model.Group.Id [get], [set]
```

Gets or sets the ID of the group.

Definition at line 15 of file Group.cs.

6.20.2.3 Name

```
string Grip.DAL.Model.Group.Name = null! [get], [set]
```

Gets or sets the name of the group.

Definition at line 26 of file Group.cs.

6.20.2.4 Users

```
ICollection<User> Grip.DAL.Model.Group.Users = null! [get], [set]
```

Gets or sets the collection of users associated with the group.

Definition at line 21 of file Group.cs.

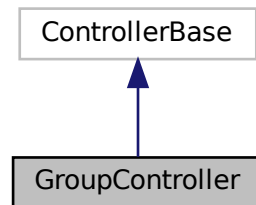
The documentation for this class was generated from the following file:

- Server/DAL/Model/[Group.cs](#)

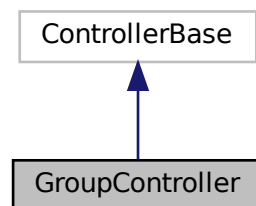
6.21 GroupController Class Reference

Represents a controller for managing groups.

Inheritance diagram for GroupController:



Collaboration diagram for GroupController:



Public Member Functions

- [GroupController](#) ([ApplicationDbContext](#) context, UserManager< [User](#) > userManager, IMapper mapper, [IGroupService](#) groupService)
Initializes a new instance of the [GroupController](#) class.
- async Task< ActionResult< IEnumerable< [GroupDTO](#) > > > [GetGroup](#) ()
Retrieves all groups.
- async Task< ActionResult< [GroupDTO](#) > > [GetGroup](#) (int id)
Retrieves a group by its ID.
- async Task< IActionResult > [PutGroup](#) (int id, [FromBody] [GroupDTO](#) @group)
Updates a group by its ID.
- async Task< IActionResult > [PostGroup](#) ([FromBody] [GroupDTO](#) @group)
Creates a new group.
- async Task< IActionResult > [DeleteGroup](#) (int id)
Deletes a group by its ID.
- async Task< IActionResult > [AddUserToGroup](#) (int groupId, int userId)
Adds a user to a group.
- async Task< IActionResult > [RemoveUserFromGroup](#) (int groupId, int userId)
Removes a user from a group.

6.21.1 Detailed Description

Represents a controller for managing groups.

Definition at line 19 of file GroupController.cs.

6.21.2 Constructor & Destructor Documentation

6.21.2.1 GroupController()

```
GroupController.GroupController (
    ApplicationDbContext context,
    UserManager< User > userManager,
    IMapper mapper,
    IGroupService groupService )
```

Initializes a new instance of the [GroupController](#) class.

Parameters

<i>context</i>	The application database context.
<i>userManager</i>	The user manager.
<i>mapper</i>	The mapper.
<i>groupService</i>	The group service.

Definition at line 33 of file GroupController.cs.

6.21.3 Member Function Documentation

6.21.3.1 AddUserToGroup()

```
async Task<ActionResult> GroupController.AddUserToGroup (
    int groupId,
    int userId )
```

Adds a user to a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user to add.

Returns

An ActionResult indicating the result of the operation.

This function is accessible via HTTP PATCH request. The user must be authorized with the "Admin" role to access this endpoint. Returns 200 OK if the operation is successful. Returns 404 Not Found if the group or user with the specified ID is not found. Returns 401 Unauthorized if the user is not authorized to access the endpoint. Returns 400 Bad Request if the ID parameters are invalid.

Definition at line 182 of file GroupController.cs.

6.21.3.2 DeleteGroup()

```
async Task<IActionResult> GroupController.DeleteGroup (
    int id )
```

Deletes a group by its ID.

Parameters

<i>id</i>	The ID of the group to delete.
-----------	--------------------------------

Returns

An IActionResult indicating the result of the delete operation.

This function is accessible via HTTP DELETE request. The user must be authorized with the "Admin" role to access this endpoint. Returns 204 No Content if the delete operation is successful. Returns 404 Not Found if the group with the specified ID is not found. Returns 401 Unauthorized if the user is not authorized to access the endpoint.

Definition at line 155 of file GroupController.cs.

6.21.3.3 GetGroup() [1/2]

```
async Task<ActionResult<IEnumerable<GroupDTO> > > GroupController.GetGroup ( )
```

Retrieves all groups.

Returns

An ActionResult containing the list of [GroupDTO](#) items.

This function is accessible via HTTP GET request. The user must be authorized to access this endpoint. Returns 200 OK if the request is successful. Returns 401 Unauthorized if the user is not authorized to access the endpoint.

Definition at line 55 of file GroupController.cs.

6.21.3.4 GetGroup() [2/2]

```
async Task<ActionResult<GroupDTO> > GroupController.GetGroup (
    int id )
```

Retrieves a group by its ID.

Parameters

<i>id</i>	The ID of the group to retrieve.
-----------	----------------------------------

Returns

An ActionResult containing the [GroupDTO](#) item.

This function is accessible via HTTP GET request. The user must be authorized to access this endpoint. Returns 200 OK if the request is successful. Returns 404 Not Found if the group with the specified ID is not found. Returns 401 Unauthorized if the user is not authorized to access the endpoint. Returns 400 Bad Request if the ID parameter is invalid.

Definition at line 79 of file GroupController.cs.

6.21.3.5 PostGroup()

```
async TaskGroupDTO @ group )
```

Creates a new group.

Parameters

<i>group</i>	The group to create.
--------------	----------------------

Returns

An ActionResult indicating the result of the create operation.

This function is accessible via HTTP POST request. The user must be authorized with the "Admin" role to access this endpoint. Returns 201 Created if the create operation is successful. Returns 400 Bad Request if the group object is malformed. Returns 401 Unauthorized if the user is not authorized to access the endpoint.

Definition at line 131 of file GroupController.cs.

6.21.3.6 PutGroup()

```
async Task<IActionResult> GroupController.PutGroup (
    int id,
    [FromBody] GroupDTO @ group )
```

Updates a group by its ID.

Parameters

<i>id</i>	The ID of the group to update.
<i>group</i>	The updated group data.

Returns

An IActionResult indicating the result of the update operation.

This function is accessible via HTTP PUT request. The user must be authorized with the "Admin" role to access this endpoint. Returns 204 No Content if the update operation is successful. Returns 400 Bad Request if the ID parameter is invalid or if the group object is malformed. Returns 404 Not Found if the group with the specified ID is not found. Returns 401 Unauthorized if the user is not authorized to access the endpoint

Definition at line 104 of file GroupController.cs.

6.21.3.7 RemoveUserFromGroup()

```
async Task<ActionResult> GroupController.RemoveUserFromGroup (
    int groupId,
    int userId )
```

Removes a user from a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user to remove.

Returns

An ActionResult indicating the result of the operation.

This function is accessible via HTTP PATCH request. The user must be authorized with the "Admin" role to access this endpoint. Returns 200 OK if the operation is successful. Returns 404 Not Found if the group or user with the specified ID is not found. Returns 401 Unauthorized if the user is not authorized to access the endpoint. Returns 400 Bad Request if the ID parameters are invalid.

Definition at line 209 of file GroupController.cs.

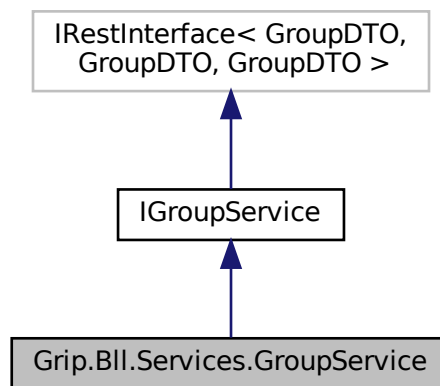
The documentation for this class was generated from the following file:

- [Server/Api/Controllers/GroupController.cs](#)

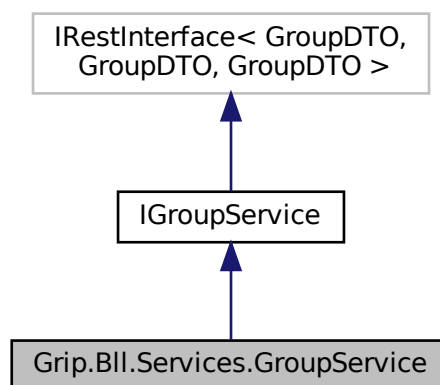
6.22 Grip.Bll.Services.GroupService Class Reference

Service class for managing groups.

Inheritance diagram for Grip.Bll.Services.GroupService:



Collaboration diagram for Grip.Bll.Services.GroupService:



Public Member Functions

- `GroupService` (IMapper mapper, UserManager< `User` > userManager, `ApplicationDbContext` context)
Initializes a new instance of the `GroupService` class.
- `async Task< GroupDTO > Create` (`GroupDTO` dto)

- Creates a new group.*
 - async Task [Delete](#) (int id)
- Deletes a group by its ID.*
 - async Task [AddUserToGroup](#) (int groupId, int userId)
- Adds a user to a group.*
 - async Task [RemoveUserFromGroup](#) (int groupId, int userId)
- Removes a user from a group.*
 - async Task< [GroupDTO](#) > [Get](#) (int id)
- Retrieves a group by its ID.*
 - async Task< IEnumerable< [GroupDTO](#) > > [GetAll](#) ()
- Retrieves all groups.*
 - async Task [Update](#) ([GroupDTO](#) dto)
- Updates a group.*
 - async Task< IEnumerable< [UserInfoDTO](#) > > [GetUsersInGroup](#) (int groupId)
- Retrieves all users in a group.*

6.22.1 Detailed Description

Service class for managing groups.

Definition at line 19 of file GroupService.cs.

6.22.2 Constructor & Destructor Documentation

6.22.2.1 GroupService()

```
Grip.Bll.Services.GroupService.GroupService (
    IMapper mapper,
    UserManager< User > userManager,
    ApplicationDbContext context )
```

Initializes a new instance of the [GroupService](#) class.

Parameters

<i>mapper</i>	The AutoMapper object for object mapping.
<i>userManager</i>	The UserManager object for user management.
<i>context</i>	The ApplicationDbContext object for accessing the database context.

Definition at line 31 of file GroupService.cs.

6.22.3 Member Function Documentation

6.22.3.1 AddUserToGroup()

```
async Task Grip.Bll.Services.GroupService.AddUserToGroup (
    int groupId,
    int userId )
```

Adds a user to a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user to add.

Implements [IGroupService](#).

Definition at line 70 of file GroupService.cs.

6.22.3.2 Create()

```
async Task<GroupDTO> Grip.Bll.Services.GroupService.Create (
    GroupDTO dto )
```

Creates a new group.

Parameters

<i>dto</i>	The GroupDTO object containing the group information.
------------	---

Returns

The created GroupDTO object.

Definition at line 42 of file GroupService.cs.

Here is the call graph for this function:



6.22.3.3 Delete()

```
async Task Grip.Bll.Services.GroupService.Delete (
    int id )
```

Deletes a group by its ID.

Parameters

<i>id</i>	The ID of the group to delete.
-----------	--------------------------------

Definition at line 54 of file GroupService.cs.

6.22.3.4 Get()

```
async Task<GroupDTO> Grip.Bll.Services.GroupService.Get (
    int id )
```

Retrieves a group by its ID.

Parameters

<i>id</i>	The ID of the group to retrieve.
-----------	----------------------------------

Returns

The GroupDTO object representing the retrieved group.

Definition at line 116 of file GroupService.cs.

Here is the call graph for this function:

**6.22.3.5 GetAll()**

```
async Task<IEnumerable<GroupDTO> > Grip.Bll.Services.GroupService.GetAll ( )
```

Retrieves all groups.

Returns

A collection of GroupDTO objects representing all groups.

Definition at line 131 of file GroupService.cs.

Here is the call graph for this function:

**6.22.3.6 GetUsersInGroup()**

```
async Task<IEnumerable<UserInfoDTO> > Grip.Bll.Services.GroupService.GetUsersInGroup (
    int groupId )
```

Retrieves all users in a group.

Parameters

<i>groupId</i>	The ID of the group.
----------------	----------------------

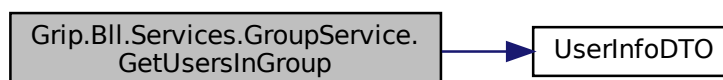
Returns

A collection of UserInfoDTO objects representing the users in the group.

Implements [IGroupService](#).

Definition at line 181 of file GroupService.cs.

Here is the call graph for this function:



6.22.3.7 RemoveUserFromGroup()

```
async Task Grip.Bll.Services.GroupService.RemoveUserFromGroup (
    int groupId,
    int userId )
```

Removes a user from a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user to remove.

Implements [IGroupService](#).

Definition at line 92 of file GroupService.cs.

6.22.3.8 Update()

```
async Task Grip.Bll.Services.GroupService.Update (
    GroupDTO dto )
```

Updates a group.

Parameters

<i>dto</i>	The GroupDTO object containing the updated group information.
------------	---

Definition at line 140 of file GroupService.cs.

Here is the call graph for this function:



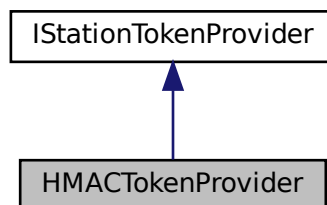
The documentation for this class was generated from the following file:

- [Server/Bll/Services/GroupService.cs](#)

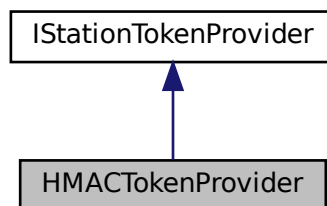
6.23 HMACTokenProvider Class Reference

Generates and validates HMAC tokens for stations.

Inheritance diagram for HMACTokenProvider:



Collaboration diagram for HMACTokenProvider:



Public Member Functions

- string [GenerateToken](#) (string key, string message)
Generates a token based on a key and a message
- bool [ValidateToken](#) (string key, string message, string token)
Validates a token based on a key and a message

6.23.1 Detailed Description

Generates and validates HMAC tokens for stations.

Definition at line 11 of file HMACTokenProvider.cs.

6.23.2 Member Function Documentation

6.23.2.1 GenerateToken()

```
string HMACTokenProvider.GenerateToken (
    string key,
    string message )
```

Generates a token based on a key and a message

Parameters

<i>key</i>	The key to use for generating the token.
<i>message</i>	The message to use for generating the token.

Returns

The generated token.

Implements [IStationTokenProvider](#).

Definition at line 19 of file HMACTokenProvider.cs.

6.23.2.2 ValidateToken()

```
bool HMACTokenProvider.ValidateToken (
    string key,
    string message,
    string token )
```

Validates a token based on a key and a message

Parameters

<i>key</i>	The key to use for validating the token.
<i>message</i>	The message to use for validating the token.
<i>token</i>	The token to validate.

Returns

True if the token is valid, false otherwise.

Implements [IStationTokenProvider](#).

Definition at line 38 of file HMACTokenProvider.cs.

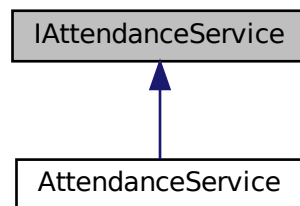
The documentation for this class was generated from the following file:

- [Server/BI/Providers/HMACTokenProvider.cs](#)

6.24 IAttendanceService Interface Reference

Interface for the attendance service

Inheritance diagram for IAttendanceService:



Public Member Functions

- Task [VerifyPhoneScan](#) (ActiveAttendanceDTO request, [User](#) user)
Register a new attendance by phone scan
- Task [VerifyPassiveScan](#) (PassiveAttendanceDTO request)
Verify the station scan
- Task< IEnumerable< [AttendanceDTO](#) > > [GetAttendanceForDay](#) ([User](#) user, DateOnly date)
Get all attendances for a user on a specific day

6.24.1 Detailed Description

Interface for the attendance service

Definition at line 11 of file IAttendanceService.cs.

6.24.2 Member Function Documentation

6.24.2.1 GetAttendanceForDay()

```

Task<IEnumerable<AttendanceDTO> > IAttendanceService.GetAttendanceForDay (
    User user,
    DateOnly date )
  
```

Get all attendances for a user on a specific day

Parameters

<i>user</i>	User used for querying
<i>date</i>	Date for querying

Returns

List of AttendanceDTO-s

Implemented in [AttendanceService](#).

6.24.2.2 VerifyPassiveScan()

```
Task IAttendanceService.VerifyPassiveScan (
    PassiveAttendanceDTO request )
```

Verify the station scan

Parameters

<i>request</i>	The request
----------------	-------------

Implemented in [AttendanceService](#).

6.24.2.3 VerifyPhoneScan()

```
Task IAttendanceService.VerifyPhoneScan (
    ActiveAttendanceDTO request,
    User user )
```

Register a new attendance by phone scan

Parameters

<i>request</i>	The request DTO
<i>user</i>	The user requesting verification

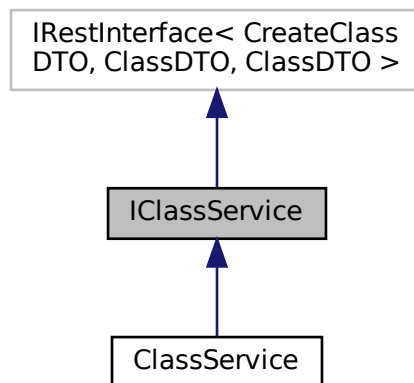
The documentation for this interface was generated from the following file:

- Server/BI/Services/Interfaces/[IAttendanceService.cs](#)

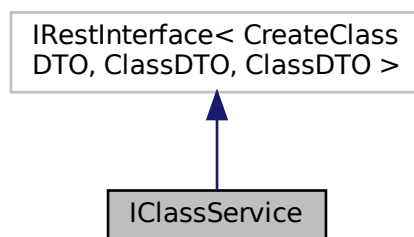
6.25 IClassService Interface Reference

Interface for the class service

Inheritance diagram for IClassService:



Collaboration diagram for IClassService:



Public Member Functions

- Task< IEnumerable< [ClassDTO](#) > > [GetClassesForUserOnDay](#) ([User](#) user, DateOnly date)
Retrieves classes for a user on a specific day.

6.25.1 Detailed Description

Interface for the class service

Definition at line 9 of file IClassService.cs.

6.25.2 Member Function Documentation

6.25.2.1 GetClassesForUserOnDay()

```
Task<IEnumerable<ClassDTO> > IClassService.GetClassesForUserOnDay (
    User user,
    DateOnly date )
```

Retrieves classes for a user on a specific day.

Parameters

<i>user</i>	The user.
<i>date</i>	The date.

Returns

A list of class DTOs.

Implemented in [ClassService](#).

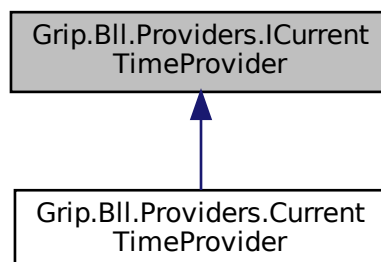
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IClassService.cs](#)

6.26 Grip.Bll.Providers.ICurrentTimeProvider Interface Reference

Interface for abstracting the current time for testing purposes.

Inheritance diagram for Grip.Bll.Providers.ICurrentTimeProvider:



Properties

- DateTime [Now](#) [get]
Current time in UTC.

6.26.1 Detailed Description

Interface for abstracting the current time for testing purposes.

Definition at line 11 of file ICurrentTimeProvider.cs.

6.26.2 Property Documentation

6.26.2.1 Now

```
DateTime Grip.Bll.Providers.ICurrentTimeProvider.Now [get]
```

Current time in UTC.

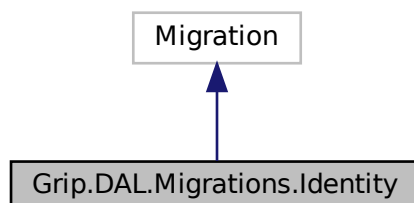
Definition at line 16 of file ICurrentTimeProvider.cs.

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

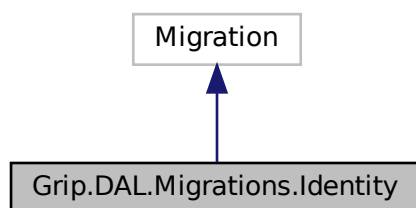
- [Server/Bll/Providers/ICurrentTimeProvider.cs](#)

6.27 Grip.DAL.Migrations.Identity Class Reference

Inheritance diagram for Grip.DAL.Migrations.Identity:



Collaboration diagram for Grip.DAL.Migrations.Identity:



Protected Member Functions

- override void [Up](#) (MigrationBuilder migrationBuilder)
- override void [Down](#) (MigrationBuilder migrationBuilder)
- override void [BuildTargetModel](#) (ModelBuilder modelBuilder)

6.27.1 Detailed Description

Definition at line 10 of file 20230317172813_Identity.cs.

6.27.2 Member Function Documentation

6.27.2.1 BuildTargetModel()

```
override void Grip.DAL.Migrations.Identity.BuildTargetModel (
    modelBuilder modelBuilder ) [protected]
```

Definition at line 19 of file 20230317172813_Identity.Designer.cs.

6.27.2.2 Down()

```
override void Grip.DAL.Migrations.Identity.Down (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 201 of file 20230317172813_Identity.cs.

6.27.2.3 Up()

```
override void Grip.DAL.Migrations.Identity.Up (
    MigrationBuilder migrationBuilder ) [protected]
```

Definition at line 13 of file 20230317172813_Identity.cs.

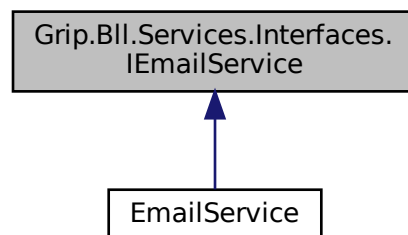
The documentation for this class was generated from the following files:

- [Server/DAL/Migrations/20230317172813_Identity.cs](#)
- [Server/DAL/Migrations/20230317172813_Identity.Designer.cs](#)

6.28 Grip.Bll.Services.Interfaces.IEmailService Interface Reference

Represents a service for sending emails.

Inheritance diagram for Grip.Bll.Services.Interfaces.IEmailService:



Public Member Functions

- Task [SendEmailAsync](#) (string to, string subject, string message)
Sends an email to a single recipient asynchronously.
- Task [SendEmailAsync](#) (IEnumerable< string > to, string subject, string message)
Sends an email to multiple recipients asynchronously.

6.28.1 Detailed Description

Represents a service for sending emails.

Definition at line 11 of file IEmailService.cs.

6.28.2 Member Function Documentation

6.28.2.1 SendEmailAsync() [1/2]

```
Task Grip.Bll.Services.Interfaces.IEmailService.SendEmailAsync (
    IEnumerable< string > to,
    string subject,
    string message )
```

Sends an email to multiple recipients asynchronously.

Parameters

<i>to</i>	The email addresses of the recipients.
<i>subject</i>	The subject of the email.
<i>message</i>	The body content of the email.

Returns

A task representing the asynchronous email sending operation.

Implemented in [EmailService](#).

6.28.2.2 SendEmailAsync() [2/2]

```
Task Grip.Bll.Services.Interfaces.IEmailService.SendEmailAsync (
    string to,
    string subject,
    string message )
```

Sends an email to a single recipient asynchronously.

Parameters

<i>to</i>	The email address of the recipient.
<i>subject</i>	The subject of the email.
<i>message</i>	The body content of the email.

Returns

A task representing the asynchronous email sending operation.

Implemented in [EmailService](#).

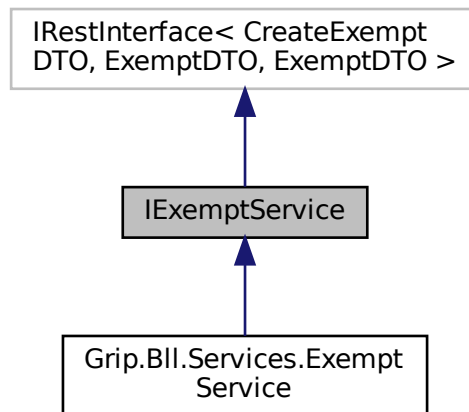
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IEmailService.cs](#)

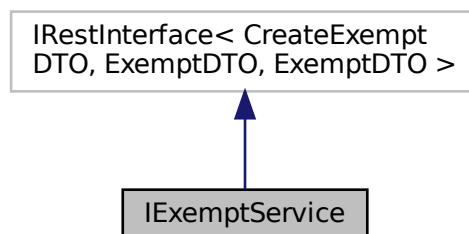
6.29 IExemptService Interface Reference

Represents an interface for the exempt service.

Inheritance diagram for IExemptService:



Collaboration diagram for IExemptService:



Public Member Functions

- `Task< ExemptDTO > Create (CreateExemptDTO dto, User teacher)`
Creates a new exempt with the specified data and assigns it to a teacher.

6.29.1 Detailed Description

Represents an interface for the exempt service.

Definition at line 8 of file `IExemptService.cs`.

6.29.2 Member Function Documentation

6.29.2.1 Create()

```
Task<ExemptDTO> IExemptService.Create (
    CreateExemptDTO dto,
    User teacher )
```

Creates a new exempt with the specified data and assigns it to a teacher.

Parameters

<i>dto</i>	The data for creating the exempt.
<i>teacher</i>	The teacher who issued the exempt.

Returns

A task representing the asynchronous creation of the exempt.

Implemented in [Grip.Bll.Services.ExemptService](#).

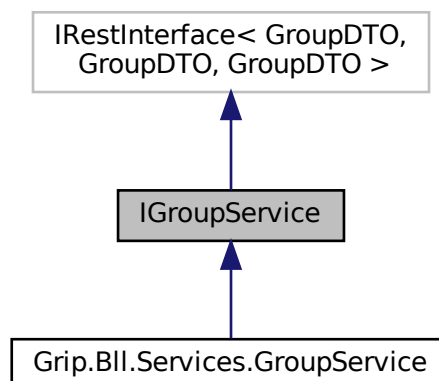
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IExemptService.cs](#)

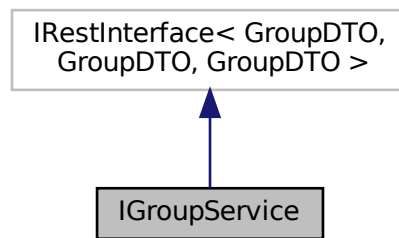
6.30 IGroupService Interface Reference

Interface representing the Group Service, which provides CRUD operations for groups as well as specific group operations.

Inheritance diagram for IGroupService:



Collaboration diagram for IGroupService:



Public Member Functions

- Task [AddUserToGroup](#) (int groupId, int userId)
Adds a user to a group.
- Task [RemoveUserFromGroup](#) (int groupId, int userId)
Removes a user from a group.
- Task< IEnumerable< [UserInfoDTO](#) > > [GetUsersInGroup](#) (int groupId)
Gets all users in a group.

6.30.1 Detailed Description

Interface representing the Group Service, which provides CRUD operations for groups as well as specific group operations.

Definition at line 8 of file `IGroupService.cs`.

6.30.2 Member Function Documentation

6.30.2.1 AddUserToGroup()

```
Task IGroupService.AddUserToGroup (
    int groupId,
    int userId )
```

Adds a user to a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user.

Returns

A task representing the asynchronous operation.

Implemented in [Grip.Bll.Services.GroupService](#).

6.30.2.2 GetUsersInGroup()

```
Task<IEnumerable<UserInfoDTO> > IGroupService.GetUsersInGroup (
    int groupId )
```

Gets all users in a group.

Parameters

<i>groupId</i>	The ID of the group.
----------------	----------------------

Returns

A task representing the asynchronous operation. The result is an IEnumerable of UserInfoDTO representing all users in the group.

Implemented in [Grip.Bll.Services.GroupService](#).

6.30.2.3 RemoveUserFromGroup()

```
Task IGroupService.RemoveUserFromGroup (
    int groupId,
    int userId )
```

Removes a user from a group.

Parameters

<i>groupId</i>	The ID of the group.
<i>userId</i>	The ID of the user.

Returns

A task representing the asynchronous operation.

Implemented in [Grip.Bll.Services.GroupService](#).

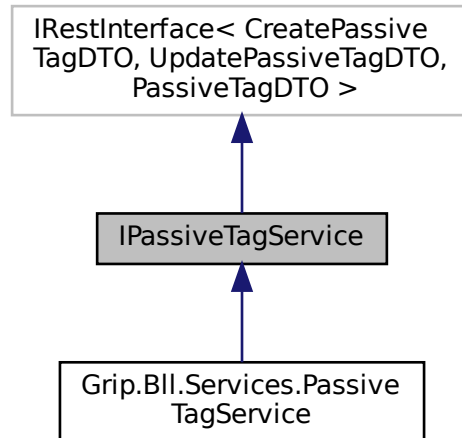
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IGroupService.cs](#)

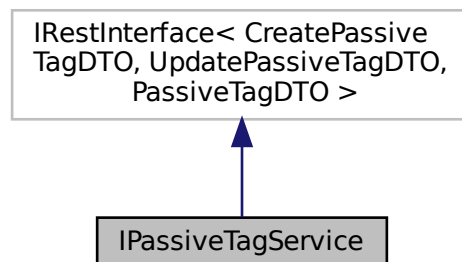
6.31 IPassiveTagService Interface Reference

Represents an interface for the passive tag service.

Inheritance diagram for IPassiveTagService:



Collaboration diagram for IPassiveTagService:



6.31.1 Detailed Description

Represents an interface for the passive tag service.

Definition at line 8 of file `IPassiveTagService.cs`.

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

- [Server/Bll/Services/Interfaces/IPassiveTagService.cs](#)

6.32 Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDTO, GetDTO > Interface Template Reference

Represents a generic REST interface for CRUD operations.

Public Member Functions

- Task< IEnumerable< GetDTO > > [GetAll](#) ()
Retrieves all resources.
- Task< GetDTO > [Get](#) (int id)
Retrieves a resource by its ID.
- Task< GetDTO > [Create](#) (CreatedDTO dto)
Creates a new resource.
- Task [Update](#) (UpdatedDTO dto)
Updates an existing resource.
- Task [Delete](#) (int id)
Deletes a resource by its ID.

6.32.1 Detailed Description

Represents a generic REST interface for CRUD operations.

Template Parameters

<i>CreatedDTO</i>	The DTO type for creating a new resource.
<i>UpdatedDTO</i>	The DTO type for updating an existing resource.
<i>GetDTO</i>	The DTO type for retrieving a resource.

Definition at line 14 of file IRestInterface.cs.

6.32.2 Member Function Documentation

6.32.2.1 Create()

```
Task<GetDTO> Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDTO, GetDTO >.Create  
(  
    CreatedDTO dto )
```

Creates a new resource.

Parameters

<i>dto</i>	The CreatedDTO object containing the resource information.
------------	--

Returns

The GetDTO representing the created resource.

6.32.2.2 Delete()

```
Task Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDTO, GetDTO >.Delete (
    int id )
```

Deletes a resource by its ID.

Parameters

<i>id</i>	The ID of the resource to delete.
-----------	-----------------------------------

6.32.2.3 Get()

```
Task<GetDTO> Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDTO, GetDTO >.Get
(
    int id )
```

Retrieves a resource by its ID.

Parameters

<i>id</i>	The ID of the resource to retrieve.
-----------	-------------------------------------

Returns

The GetDTO representing the retrieved resource.

6.32.2.4 GetAll()

```
Task<IEnumerable<GetDTO> > Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDT↵
DTO, GetDTO >.GetAll ( )
```

Retrieves all resources.

Returns

A collection of GetDTO representing all resources.

6.32.2.5 Update()

```
Task Grip.Bll.Services.Interfaces.IRestInterface< CreatedDTO, UpdatedDTO, GetDTO >.Update (
    UpdatedDTO dto )
```

Updates an existing resource.

Parameters

<i>dto</i>	The UpdatedDTO object containing the updated resource information.
------------	--

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

- Server/Bll/Services/Interfaces/[IRestInterface.cs](#)

6.33 Grip.Api.Hubs.IStationClient Interface Reference

This interface is used to define the methods that can be called from the station hub on the client

Public Member Functions

- Task [ReceiveScan](#) (StationScanDTO scanDTO)
This method is called when an attendance is registered

6.33.1 Detailed Description

This interface is used to define the methods that can be called from the station hub on the client

Definition at line 12 of file IStationClient.cs.

6.33.2 Member Function Documentation

6.33.2.1 ReceiveScan()

```
Task Grip.Api.Hubs.IStationClient.ReceiveScan (
    StationScanDTO scanDTO )
```

This method is called when an attendance is registered

Parameters

<i>scanDTO</i>	DTO containing information about the scan
----------------	---

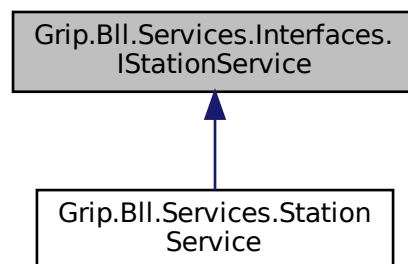
The documentation for this interface was generated from the following file:

- [Server/Api/Hubs/IStationClient.cs](#)

6.34 Grip.Bll.Services.Interfaces.IStationService Interface Reference

Represents an interface for the station service.

Inheritance diagram for Grip.Bll.Services.Interfaces.IStationService:



Public Member Functions

- `Task< StationSecretKeyDTO > GetSecretKey (int StationNumber)`
Gets the secret key for a station.

6.34.1 Detailed Description

Represents an interface for the station service.

Definition at line 12 of file `IStationService.cs`.

6.34.2 Member Function Documentation

6.34.2.1 GetSecretKey()

```
Task<StationSecretKeyDTO> Grip.Bll.Services.Interfaces.IStationService.GetSecretKey (
    int StationNumber )
```

Gets the secret key for a station.

Implemented in [Grip.Bll.Services.StationService](#).

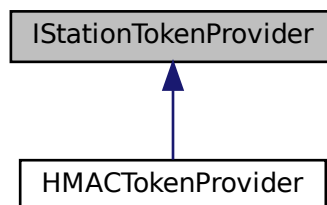
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IStationService.cs](#)

6.35 IStationTokenProvider Interface Reference

Interface for generating and validating tokens for stations.

Inheritance diagram for IStationTokenProvider:



Public Member Functions

- string [GenerateToken](#) (string key, string message)
Generates a token based on a key and a message
- bool [ValidateToken](#) (string key, string message, string token)
Validates a token based on a key and a message

6.35.1 Detailed Description

Interface for generating and validating tokens for stations.

Definition at line 11 of file IStationTokenProvider.cs.

6.35.2 Member Function Documentation

6.35.2.1 GenerateToken()

```
string IStationTokenProvider.GenerateToken (  
    string key,  
    string message )
```

Generates a token based on a key and a message

Parameters

<i>key</i>	The key to use for generating the token.
<i>message</i>	The message to use for generating the token.

Returns

The generated token.

Implemented in [HMACTokenProvider](#).

6.35.2.2 ValidateToken()

```
bool IStationTokenProvider.ValidateToken (
    string key,
    string message,
    string token )
```

Validates a token based on a key and a message

Parameters

<i>key</i>	The key to use for validating the token.
<i>message</i>	The message to use for validating the token.
<i>token</i>	The token to validate.

Returns

True if the token is valid, false otherwise.

Implemented in [HMACTokenProvider](#).

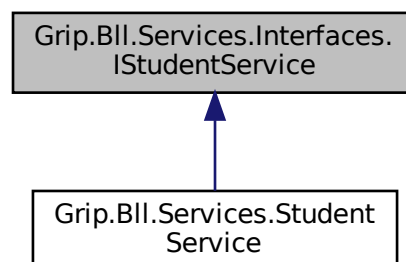
The documentation for this interface was generated from the following file:

- [Server/Bll/Providers/IStationTokenProvider.cs](#)

6.36 Grip.Bll.Services.Interfaces.IStudentService Interface Reference

Interface for the student service

Inheritance diagram for Grip.Bll.Services.Interfaces.IStudentService:



Public Member Functions

- Task< IEnumerable< [UserInfoDTO](#) > > [SearchStudentsAsync](#) (string? name, int? groupId)
Search students by name and group
- Task< StudentDetailDTO > [GetStudentDetailsAsync](#) (int id)
Get details of a student

6.36.1 Detailed Description

Interface for the student service

Definition at line 8 of file IStudentService.cs.

6.36.2 Member Function Documentation

6.36.2.1 GetStudentDetailsAsync()

```
Task<StudentDetailDTO> Grip.Bll.Services.Interfaces.IStudentService.GetStudentDetailsAsync (
    int id )
```

Get details of a student

Parameters

<i>id</i>	Id of the student
-----------	-------------------

Returns

Detail [DTO](#) of the student

Implemented in [Grip.Bll.Services.StudentService](#).

6.36.2.2 SearchStudentsAsync()

```
Task<IEnumerable<UserInfoDTO> > Grip.Bll.Services.Interfaces.IStudentService.SearchStudents↔
Async (
    string? name,
    int? groupId )
```

Search students by name and group

Parameters

<i>name</i>	String that the name of the student should contain, null or empty if this filter shouldn't apply
<i>group</i> ↔	Id of the group the students are part of, null if theus filter shouldn't apply
Generated by Doxygen	

Returns

IEnumerable containing description of users

Implemented in [Grip.Bll.Services.StudentService](#).

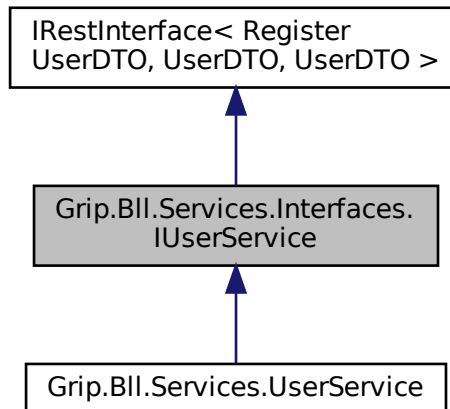
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IStudentService.cs](#)

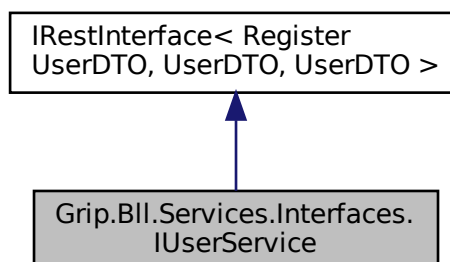
6.37 Grip.Bll.Services.Interfaces.IUserService Interface Reference

Represents an interface for the user service.

Inheritance diagram for Grip.Bll.Services.Interfaces.IUserService:



Collaboration diagram for Grip.Bll.Services.Interfaces.IUserService:



Public Member Functions

- Task< [LoginResultDTO](#) > [Login](#) ([LoginUserDTO](#) dto)
Logs in a user with the specified login credentials.
- Task [ConfirmEmail](#) ([ConfirmEmailDTO](#) dto)
Confirms the email of a user with the specified data.
- Task [ForgotPassword](#) ([ForgotPasswordDTO](#) dto)
Sends a forgot password email to a user with the specified data.
- Task [ResetPassword](#) ([ResetPasswordDTO](#) dto)
Resets the password of a user with the specified data.
- Task [AddRole](#) (int userId, string role)
Adds a role to a user with the specified ID.
- Task [RemoveRole](#) (int userId, string role)
Removes a role from a user with the specified ID.

6.37.1 Detailed Description

Represents an interface for the user service.

Definition at line 12 of file IUserService.cs.

6.37.2 Member Function Documentation

6.37.2.1 AddRole()

```
Task Grip.Bll.Services.Interfaces.IUserService.AddRole (
    int userId,
    string role )
```

Adds a role to a user with the specified ID.

Parameters

<i>userId</i>	The ID of the user.
<i>role</i>	The role to add.

Returns

A task representing the asynchronous role addition.

Implemented in [Grip.Bll.Services.UserService](#).

6.37.2.2 ConfirmEmail()

```
Task Grip.Bll.Services.Interfaces.IUserService.ConfirmEmail (
    ConfirmEmailDTO dto )
```

Confirms the email of a user with the specified data.

Parameters

<i>dto</i>	The data for confirming the email.
------------	------------------------------------

Returns

A task representing the asynchronous email confirmation.

Implemented in [Grip.Bll.Services.UserService](#).

6.37.2.3 ForgotPassword()

```
Task Grip.Bll.Services.Interfaces.IUserService.ForgotPassword (
    ForgotPasswordDTO dto )
```

Sends a forgot password email to a user with the specified data.

Parameters

<i>dto</i>	The data for sending the forgot password email.
------------	---

Returns

A task representing the asynchronous sending of the forgot password email.

Implemented in [Grip.Bll.Services.UserService](#).

6.37.2.4 Login()

```
Task<LoginResultDTO> Grip.Bll.Services.Interfaces.IUserService.Login (
    LoginUserDTO dto )
```

Logs in a user with the specified login credentials.

Parameters

<i>dto</i>	The login credentials.
------------	------------------------

Returns

A task representing the asynchronous login operation.

Implemented in [Grip.Bll.Services.UserService](#).

6.37.2.5 RemoveRole()

```
Task Grip.Bll.Services.Interfaces.IUserService.RemoveRole (
    int userId,
    string role )
```

Removes a role from a user with the specified ID.

Parameters

<i>userId</i>	The ID of the user.
<i>role</i>	The role to remove.

Returns

A task representing the asynchronous role removal.

Implemented in [Grip.Bll.Services.UserService](#).

6.37.2.6 ResetPassword()

```
Task Grip.Bll.Services.Interfaces.IUserService.ResetPassword (
    ResetPasswordDTO dto )
```

Resets the password of a user with the specified data.

Parameters

<i>dto</i>	The data for resetting the password.
------------	--------------------------------------

Returns

A task representing the asynchronous password reset.

Implemented in [Grip.Bll.Services.UserService](#).

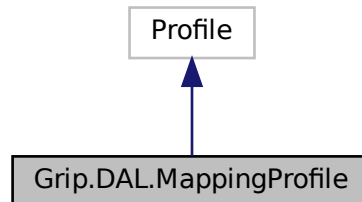
The documentation for this interface was generated from the following file:

- [Server/Bll/Services/Interfaces/IUserService.cs](#)

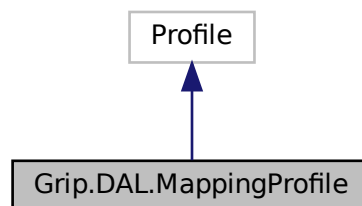
6.38 Grip.DAL.MappingProfile Class Reference

Mapping profile for AutoMapper.

Inheritance diagram for Grip.DAL.MappingProfile:



Collaboration diagram for Grip.DAL.MappingProfile:



Public Member Functions

- [MappingProfile](#) ()
Initializes a new instance of the [MappingProfile](#) class.

6.38.1 Detailed Description

Mapping profile for AutoMapper.

Definition at line 10 of file `MappingProfile.cs`.

6.38.2 Constructor & Destructor Documentation

6.38.2.1 MappingProfile()

```
Grip.DAL.MappingProfile.MappingProfile ( )
```

Initializes a new instance of the [MappingProfile](#) class.

Definition at line 15 of file MappingProfile.cs.

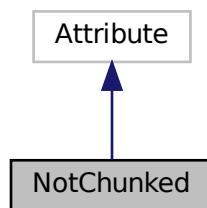
The documentation for this class was generated from the following file:

- [Server/DAL/MappingProfile.cs](#)

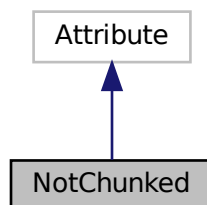
6.39 NotChunked Class Reference

Attribute for marking a controller method write response in a not chunked manner.

Inheritance diagram for NotChunked:



Collaboration diagram for NotChunked:



6.39.1 Detailed Description

Attribute for marking a controller method write response in a not chunked manner.

Definition at line 11 of file NotChunked.cs.

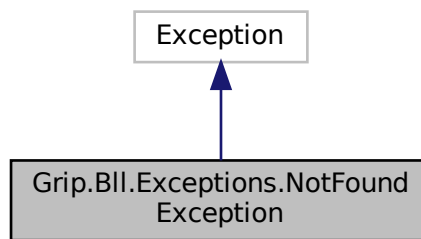
The documentation for this class was generated from the following file:

- [Server/Api/Middleware/NotChunked.cs](#)

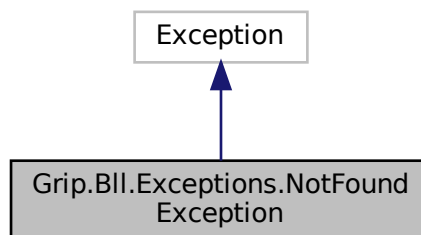
6.40 Grip.Bll.Exceptions.NotFoundException Class Reference

Exception thrown when a resource is not found: HTTP 404.

Inheritance diagram for Grip.Bll.Exceptions.NotFoundException:



Collaboration diagram for Grip.Bll.Exceptions.NotFoundException:



Public Member Functions

- [NotFoundException](#) (string? message=null)
Creates an instance of the exception.

6.40.1 Detailed Description

Exception thrown when a resource is not found: HTTP 404.

Definition at line 11 of file NotFoundException.cs.

6.40.2 Constructor & Destructor Documentation

6.40.2.1 NotFoundException()

```
Grip.Bll.Exceptions.NotFoundException.NotFoundException (
    string? message = null )
```

Creates an instance of the exception.

Parameters

<i>message</i>	Description of the cause of the exception
----------------	---

Definition at line 17 of file NotFoundException.cs.

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

- Server/Bll/Exceptions/[NotFoundException.cs](#)

6.41 Grip.DAL.Model.PassiveTag Class Reference

Represents a passive tag in the database.

Properties

- int [Id](#) [get, set]
Gets or sets the ID of the passive tag.
- Int64 [SerialNumber](#) [get, set]
Gets or sets the serial number of the passive tag.
- [User](#) [User](#) = null! [get, set]
Gets or sets the user associated with the passive tag.

6.41.1 Detailed Description

Represents a passive tag in the database.

Definition at line 8 of file PassiveTag.cs.

6.41.2 Property Documentation

6.41.2.1 Id

```
int Grip.DAL.Model.PassiveTag.Id [get], [set]
```

Gets or sets the ID of the passive tag.

Definition at line 13 of file PassiveTag.cs.

6.41.2.2 SerialNumber

```
Int64 Grip.DAL.Model.PassiveTag.SerialNumber [get], [set]
```

Gets or sets the serial number of the passive tag.

Definition at line 18 of file PassiveTag.cs.

6.41.2.3 User

```
User Grip.DAL.Model.PassiveTag.User = null! [get], [set]
```

Gets or sets the user associated with the passive tag.

Definition at line 23 of file PassiveTag.cs.

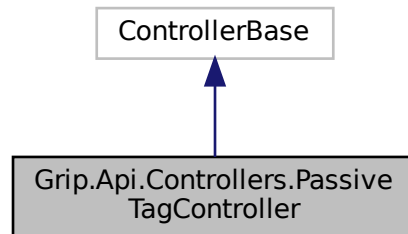
The documentation for this class was generated from the following file:

- [Server/DAL/Model/PassiveTag.cs](#)

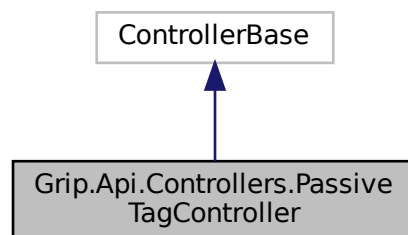
6.42 Grip.Api.Controllers.PassiveTagController Class Reference

API controller for managing passive tags.

Inheritance diagram for Grip.Api.Controllers.PassiveTagController:



Collaboration diagram for Grip.Api.Controllers.PassiveTagController:



Public Member Functions

- [PassiveTagController](#) ([ApplicationDbContext](#) context, [IMapper](#) mapper, [IPassiveTagService](#) passiveTagService)
 - Initializes a new instance of the [PassiveTagController](#) class.*
- [async Task< ActionResult< IEnumerable< \[PassiveTagDTO\]\(#\) > > > GetPassiveTags \(\)](#)
 - Retrieves all passive tags.*
- [async Task< ActionResult< \[PassiveTagDTO\]\(#\) > > GetPassiveTag \(int id\)](#)
 - Retrieves a specific passive tag by its ID.*
- [async Task< IActionResult > PutPassiveTag \(int id, \[FromBody\] \[UpdatePassiveTagDTO\]\(#\) passiveTag\)](#)
 - Updates a specific passive tag.*
- [async Task< ActionResult< \[PassiveTagDTO\]\(#\) > > PostPassiveTag \(\[FromBody\] \[CreatePassiveTagDTO\]\(#\) passiveTag\)](#)
 - Creates a new passive tag.*
- [async Task< IActionResult > DeletePassiveTag \(int id\)](#)
 - Deletes a specific passive tag.*

6.42.1 Detailed Description

API controller for managing passive tags.

Definition at line 22 of file PassiveTagController.cs.

6.42.2 Constructor & Destructor Documentation

6.42.2.1 PassiveTagController()

```
Grip.Api.Controllers.PassiveTagController.PassiveTagController (
    ApplicationDbContext context,
    IMapper mapper,
    IPassiveTagService passiveTagService )
```

Initializes a new instance of the [PassiveTagController](#) class.

Parameters

<i>context</i>	The application database context.
<i>mapper</i>	The mapper for DTO mapping.
<i>passiveTagService</i>	The service for managing passive tags.

Definition at line 34 of file PassiveTagController.cs.

6.42.3 Member Function Documentation

6.42.3.1 DeletePassiveTag()

```
async Task<IActionResult> Grip.Api.Controllers.PassiveTagController.DeletePassiveTag (
    int id )
```

Deletes a specific passive tag.

Parameters

<i>id</i>	The ID of the passive tag to delete.
-----------	--------------------------------------

Returns

An IActionResult indicating the result of the operation.

This function is accessible via HTTP DELETE request. The user must be authorized with the "Admin" role to access this endpoint. Returns 400 Bad Request if the ID parameter is invalid. Returns 204 No Content if the operation is successful. Returns 404 Not Found if the passive tag with the specified ID is not found.

Definition at line 151 of file PassiveTagController.cs.

6.42.3.2 GetPassiveTag()

```
async Task<ActionResult<PassiveTagDTO> > Grip.Api.Controllers.PassiveTagController.Get↵  
PassiveTag (   
    int id )
```

Retrieves a specific passive tag by its ID.

Parameters

<i>id</i>	The ID of the passive tag to retrieve.
-----------	--

Returns

An ActionResult representing the passive tag.

This function is accessible via HTTP GET request. The user must be authorized with the "Admin" role to access this endpoint. Returns 400 Bad Request if the ID parameter is invalid. Returns 404 Not Found if the passive tag with the specified ID is not found. Returns 200 OK if the operation is successful.

Definition at line 75 of file PassiveTagController.cs.

6.42.3.3 GetPassiveTags()

```
async Task<ActionResult<IEnumerable<PassiveTagDTO> > > Grip.Api.Controllers.PassiveTag↵  
Controller.GetPassiveTags ( )
```

Retrieves all passive tags.

Returns

A collection of [PassiveTagDTO](#) representing the passive tags.

This function is accessible via HTTP GET request. The user must be authorized with the "Admin" role to access this endpoint. Returns 200 OK if the operation is successful.

Definition at line 53 of file PassiveTagController.cs.

6.42.3.4 PostPassiveTag()

```
async Task<ActionResult<PassiveTagDTO> > Grip.Api.Controllers.PassiveTagController.Post↵  
PassiveTag (   
    [FromBody] CreatePassiveTagDTO passiveTag )
```

Creates a new passive tag.

Parameters

<i>passiveTag</i>	The passive tag data to create.
-------------------	---------------------------------

Returns

An ActionResult representing the created passive tag.

This function is accessible via HTTP POST request. The user must be authorized with the "Admin" role to access this endpoint. Returns 400 Bad Request if the passive tag data is invalid. Returns 201 Created if the passive tag is created successfully. Returns 404 Not Found if the referenced entities in the passive tag data are not found.

Definition at line 127 of file PassiveTagController.cs.

6.42.3.5 PutPassiveTag()

```
async Task<ActionResult> Grip.Api.Controllers.PassiveTagController.PutPassiveTag (
    int id,
    [FromBody] UpdatePassiveTagDTO passiveTag )
```

Updates a specific passive tag.

Parameters

<i>id</i>	The ID of the passive tag to update.
<i>passiveTag</i>	The updated passive tag data.

Returns

An IActionResult indicating the result of the operation.

This function is accessible via HTTP PUT request. The user must be authorized with the "Admin" role to access this endpoint. Returns 400 Bad Request if the ID parameter or the passive tag data is invalid. Returns 404 Not Found if the passive tag with the specified ID is not found. Returns 204 No Content if the operation is successful.

Definition at line 98 of file PassiveTagController.cs.

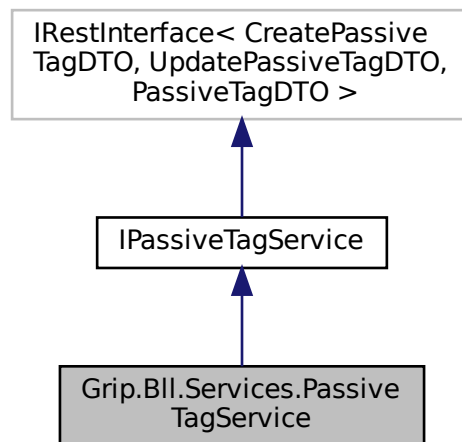
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[PassiveTagController.cs](#)

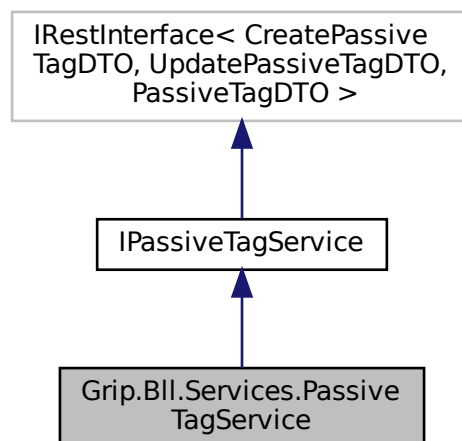
6.43 Grip.Bll.Services.PassiveTagService Class Reference

Service class for managing passive tags.

Inheritance diagram for Grip.Bll.Services.PassiveTagService:



Collaboration diagram for Grip.Bll.Services.PassiveTagService:



Public Member Functions

- `PassiveTagService` (IMapper mapper, [ApplicationDbContext](#) context, UserManager< [User](#) > userManager)
Initializes a new instance of the [PassiveTagService](#) class.
- `async Task< PassiveTagDTO > Create` (`CreatePassiveTagDTO` dto)
Creates a new passive tag.

- async Task [Delete](#) (int id)
Deletes a passive tag by its ID.
- async Task< [PassiveTagDTO](#) > [Get](#) (int id)
Retrieves a passive tag by its ID.
- async Task< IEnumerable< [PassiveTagDTO](#) > > [GetAll](#) ()
Retrieves all passive tags.
- async Task [Update](#) ([UpdatePassiveTagDTO](#) dto)
Updates a passive tag.

6.43.1 Detailed Description

Service class for managing passive tags.

Definition at line 19 of file PassiveTagService.cs.

6.43.2 Constructor & Destructor Documentation

6.43.2.1 PassiveTagService()

```
Grip.Bll.Services.PassiveTagService.PassiveTagService (
    IMapper mapper,
    ApplicationDbContext context,
    UserManager< User > userManager )
```

Initializes a new instance of the [PassiveTagService](#) class.

Parameters

<i>mapper</i>	The AutoMapper object for object mapping.
<i>context</i>	The ApplicationDbContext object for accessing the database context.
<i>userManager</i>	The UserManager object for user management.

Definition at line 32 of file PassiveTagService.cs.

6.43.3 Member Function Documentation

6.43.3.1 Create()

```
async Task<PassiveTagDTO> Grip.Bll.Services.PassiveTagService.Create (
    CreatePassiveTagDTO dto )
```

Creates a new passive tag.

Parameters

<i>dto</i>	The CreatePassiveTagDTO object containing the passive tag information.
------------	--

Returns

The created PassiveTagDTO object.

Definition at line 44 of file PassiveTagService.cs.

Here is the call graph for this function:

**6.43.3.2 Delete()**

```
async Task Grip.Bll.Services.PassiveTagService.Delete (
    int id )
```

Deletes a passive tag by its ID.

Parameters

<i>id</i>	The ID of the passive tag to delete.
-----------	--------------------------------------

Definition at line 66 of file PassiveTagService.cs.

6.43.3.3 Get()

```
async Task<PassiveTagDTO> Grip.Bll.Services.PassiveTagService.Get (
    int id )
```

Retrieves a passive tag by its ID.

Parameters

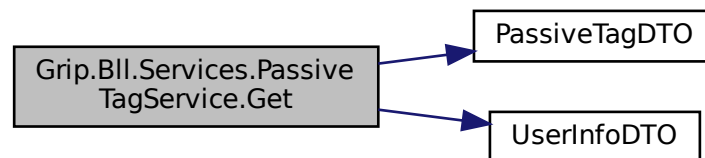
<i>id</i>	The ID of the passive tag to retrieve.
-----------	--

Returns

The PassiveTagDTO object representing the retrieved passive tag.

Definition at line 83 of file PassiveTagService.cs.

Here is the call graph for this function:

**6.43.3.4 GetAll()**

```
async Task<IEnumerable<PassiveTagDTO> > Grip.Bll.Services.PassiveTagService.GetAll ( )
```

Retrieves all passive tags.

Returns

A collection of PassiveTagDTO objects representing all passive tags.

Definition at line 98 of file PassiveTagService.cs.

Here is the call graph for this function:

**6.43.3.5 Update()**

```
async Task Grip.Bll.Services.PassiveTagService.Update (
    UpdatePassiveTagDTO dto )
```

Updates a passive tag.

Parameters

<i>dto</i>	The UpdatePassiveTagDTO object containing the updated passive tag information.
------------	--

Definition at line 111 of file PassiveTagService.cs.

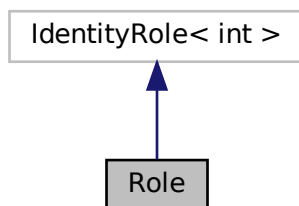
The documentation for this class was generated from the following file:

- Server/Bll/Services/[PassiveTagService.cs](#)

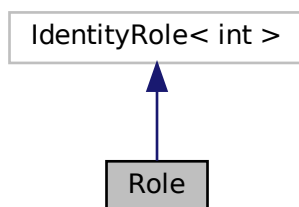
6.44 Role Class Reference

Represents a role in the database.

Inheritance diagram for Role:



Collaboration diagram for Role:



Static Public Attributes

- `const string Admin = "Admin"`
Constant string representing the admin role.
- `const string Teacher = "Teacher"`
Constant string representing the teacher role.
- `const string Student = "Student"`
Constant string representing the student role.
- `const string Doorman = "Doorman"`
Constant string representing the doorman role.

6.44.1 Detailed Description

Represents a role in the database.

Definition at line 8 of file Role.cs.

6.44.2 Member Data Documentation

6.44.2.1 Admin

```
const string Role.Admin = "Admin" [static]
```

Constant string representing the admin role.

Definition at line 13 of file Role.cs.

6.44.2.2 Doorman

```
const string Role.Doorman = "Doorman" [static]
```

Constant string representing the doorman role.

Definition at line 25 of file Role.cs.

6.44.2.3 Student

```
const string Role.Student = "Student" [static]
```

Constant string representing the student role.

Definition at line 21 of file Role.cs.

6.44.2.4 Teacher

```
const string Role.Teacher = "Teacher" [static]
```

Constant string representing the teacher role.

Definition at line 17 of file Role.cs.

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

- Server/DAL/Model/[Role.cs](#)

6.45 Grip.DAL.Model.Station Class Reference

Represents a station in the database.

Properties

- int [Id](#) [get, set]
Gets or sets the ID of the station.
- int [StationNumber](#) [get, set]
Gets or sets the station number.
- string? [Name](#) [get, set]
Gets or sets the name of the station. (E.g., Room 103)
- string [SecretKey](#) = null! [get, set]
Gets or sets the secret key used to generate tokens for this station.
- ICollection< [Attendance](#) > [Attendances](#) = null! [get, set]
Gets or sets the collection of attendances associated with this station.
- ICollection< [Class](#) > [Classes](#) = null! [get, set]
Gets or sets the collection of classes held at this station.

6.45.1 Detailed Description

Represents a station in the database.

Definition at line 15 of file Station.cs.

6.45.2 Property Documentation

6.45.2.1 Attendances

```
ICollection<Attendance> Grip.DAL.Model.Station.Attendances = null! [get], [set]
```

Gets or sets the collection of attendances associated with this station.

Definition at line 40 of file Station.cs.

6.45.2.2 Classes

```
ICollection<Class> Grip.DAL.Model.Station.Classes = null! [get], [set]
```

Gets or sets the collection of classes held at this station.

Definition at line 45 of file Station.cs.

6.45.2.3 Id

```
int Grip.DAL.Model.Station.Id [get], [set]
```

Gets or sets the ID of the station.

Definition at line 20 of file Station.cs.

6.45.2.4 Name

```
string? Grip.DAL.Model.Station.Name [get], [set]
```

Gets or sets the name of the station. (E.g., Room 103)

Definition at line 30 of file Station.cs.

6.45.2.5 SecretKey

```
string Grip.DAL.Model.Station.SecretKey = null! [get], [set]
```

Gets or sets the secret key used to generate tokens for this station.

Definition at line 35 of file Station.cs.

6.45.2.6 StationNumber

```
int Grip.DAL.Model.Station.StationNumber [get], [set]
```

Gets or sets the station number.

Definition at line 25 of file Station.cs.

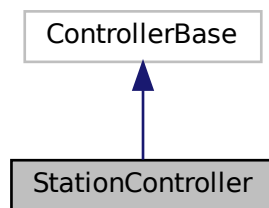
The documentation for this class was generated from the following file:

- Server/DAL/Model/[Station.cs](#)

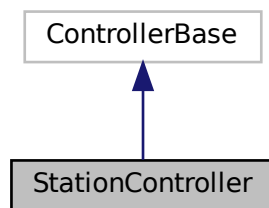
6.46 StationController Class Reference

Represents a controller for managing stations.

Inheritance diagram for StationController:



Collaboration diagram for StationController:



Public Member Functions

- [StationController](#) (ILogger< [StationController](#) > logger, [IStationService](#) stationService)
Initializes a new instance of the [StationController](#) class.
- async Task< ActionResult< [StationSecretKeyDTO](#) > > [GetKey](#) ([FromRoute] int StationNumber)
Gets the secret key for a specific station.

6.46.1 Detailed Description

Represents a controller for managing stations.

Definition at line 15 of file `StationController.cs`.

6.46.2 Constructor & Destructor Documentation

6.46.2.1 StationController()

```
StationController.StationController (
    ILogger< StationController > logger,
    IStationService stationService )
```

Initializes a new instance of the [StationController](#) class.

Parameters

<i>logger</i>	The logger instance used for logging.
<i>stationService</i>	The service for managing stations.

Definition at line 25 of file StationController.cs.

6.46.3 Member Function Documentation

6.46.3.1 GetKey()

```
async Task<ActionResult<StationSecretKeyDTO> > StationController.GetKey (
    [FromRoute] int StationNumber )
```

Gets the secret key for a specific station.

Parameters

<i>StationNumber</i>	The number of the station.
----------------------	----------------------------

Returns

An `ActionResult<T>` representing the result of the operation.

This function is accessible via HTTP GET request. It requires the station number to be provided as a route parameter. The response HTTP message is not chunked. The API key is validated using the [ValidateApiKey](#) middleware. Returns 200 OK if the operation is successful. Returns 400 Bad Request if the station number is invalid. Returns 401 Unauthorized if the API key is not valid.

Definition at line 51 of file StationController.cs.

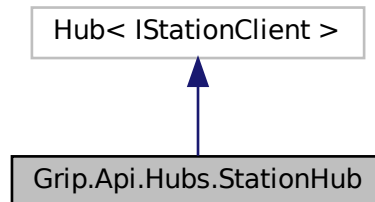
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[StationController.cs](#)

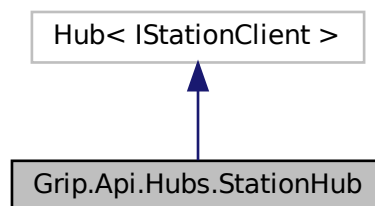
6.47 Grip.Api.Hubs.StationHub Class Reference

SignalR Hub for station related events

Inheritance diagram for Grip.Api.Hubs.StationHub:



Collaboration diagram for Grip.Api.Hubs.StationHub:



Public Member Functions

- async Task [SelectStation](#) (int stationNumber)
Clients subscribe to station events through this method

6.47.1 Detailed Description

SignalR Hub for station related events

Definition at line 10 of file StationHub.cs.

6.47.2 Member Function Documentation

6.47.2.1 SelectStation()

```
async Task Grip.Api.Hubs.StationHub.SelectStation (
    int stationNumber )
```

Clients subscribe to station events through this method

Parameters

<i>stationNumber</i>	Id of the station
----------------------	-------------------

Definition at line 16 of file StationHub.cs.

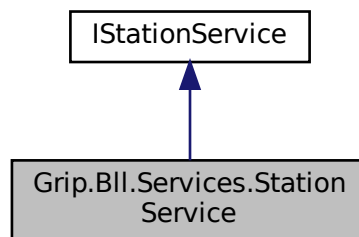
The documentation for this class was generated from the following file:

- Server/Api/Hubs/[StationHub.cs](#)

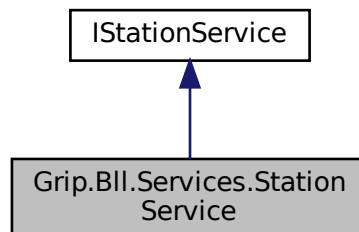
6.48 Grip.Bll.Services.StationService Class Reference

Provides methods for managing stations.

Inheritance diagram for Grip.Bll.Services.StationService:



Collaboration diagram for Grip.Bll.Services.StationService:



Public Member Functions

- [StationService](#) (ILogger< [StationService](#) > logger, IConfiguration configuration, [ApplicationDbContext](#) context)
Initializes a new instance of the [StationService](#) class with specified dependencies.
- async Task< [StationSecretKeyDTO](#) > [GetSecretKey](#) (int StationNumber)
Retrieves the secret key for the station with specified station number.

6.48.1 Detailed Description

Provides methods for managing stations.

Definition at line 12 of file StationService.cs.

6.48.2 Constructor & Destructor Documentation

6.48.2.1 StationService()

```
Grip.Bll.Services.StationService.StationService (
    ILogger< StationService > logger,
    IConfiguration configuration,
    ApplicationDbContext context )
```

Initializes a new instance of the [StationService](#) class with specified dependencies.

Parameters

<i>logger</i>	Logger object for logging errors and warnings.
<i>configuration</i>	Configuration object for accessing app settings.
<i>context</i>	Database context object for accessing station data.

Definition at line 24 of file StationService.cs.

6.48.3 Member Function Documentation

6.48.3.1 GetSecretKey()

```
async Task<StationSecretKeyDTO> Grip.Bll.Services.StationService.GetSecretKey (
    int StationNumber )
```

Retrieves the secret key for the station with specified station number.

Parameters

<i>StationNumber</i>	The station number for which to retrieve the secret key.
----------------------	--

Returns

A `StationSecretKeyDTO` object containing the secret key for the specified station.

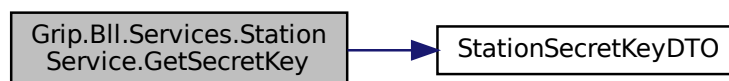
Exceptions

<i>BadRequestException</i>	Thrown when station does not exist and configuration flag for creating station on key request is set to false.
----------------------------	--

Implements [Grip.Bll.Services.Interfaces.IStationService](#).

Definition at line 37 of file `StationService.cs`.

Here is the call graph for this function:



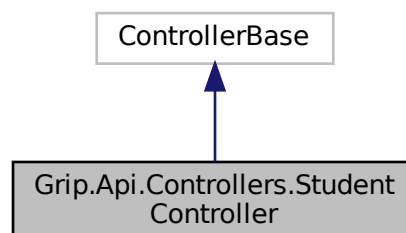
The documentation for this class was generated from the following file:

- `Server/Bll/Services/StationService.cs`

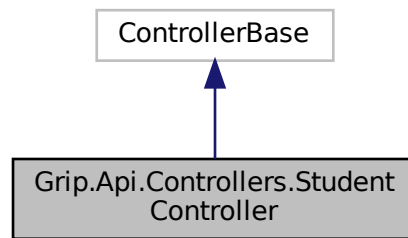
6.49 Grip.Api.Controllers.StudentController Class Reference

Controller for managing students.

Inheritance diagram for `Grip.Api.Controllers.StudentController`:



Collaboration diagram for Grip.Api.Controllers.StudentController:



Public Member Functions

- [StudentController](#) ([IStudentService](#) studentService)
Constructor for the student controller.
- `async Task< ActionResult< IEnumerable< UserInfoDTO > > > GetStudents ([FromQuery] string? name, [FromQuery] int? groupId)`
Returns students matching the search criteria.
- `async Task< ActionResult< StudentDetailDTO > > GetStudentDetails (int id)`
Gets the details of a student, including their abs.

6.49.1 Detailed Description

Controller for managing students.

Definition at line 17 of file StudentController.cs.

6.49.2 Constructor & Destructor Documentation

6.49.2.1 StudentController()

```
Grip.Api.Controllers.StudentController.StudentController (
    IStudentService studentService )
```

Constructor for the student controller.

Parameters

<i>studentService</i>	Student service
-----------------------	-----------------

Definition at line 25 of file StudentController.cs.

6.49.3 Member Function Documentation

6.49.3.1 GetStudentDetails()

```
async Task<ActionResult<StudentDetailDTO> > Grip.Api.Controllers.StudentController.Get↵  
StudentDetails (   
    int id )
```

Gets the details of a student, including their abs.

Parameters

<i>id</i>	
-----------	--

Returns

Definition at line 53 of file StudentController.cs.

6.49.3.2 GetStudents()

```
async Task<ActionResult<IEnumerable<UserInfoDTO> > > Grip.Api.Controllers.StudentController.↵  
GetStudents (   
    [FromQuery] string? name,   
    [FromQuery] int? groupId )
```

Returns students matching the search criteria.

Parameters

<i>name</i>	The string the students name should contain
<i>group↵ Id</i>	Id of the student

Returns

List of the users matchin the criteria

Definition at line 39 of file StudentController.cs.

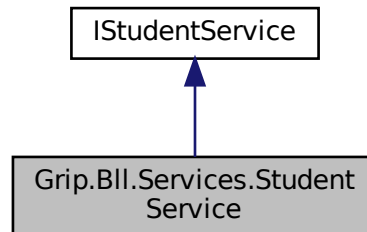
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[StudentController.cs](#)

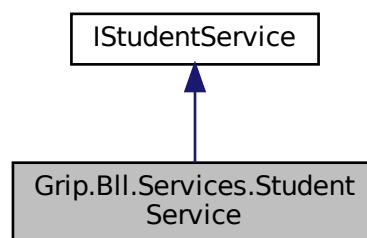
6.50 Grip.Bll.Services.StudentService Class Reference

Service for managing students

Inheritance diagram for Grip.Bll.Services.StudentService:



Collaboration diagram for Grip.Bll.Services.StudentService:



Public Member Functions

- [StudentService](#) ([ApplicationDbContext](#) context, [IMapper](#) mapper, [ICurrentTimeProvider](#) currentTimeProvider)
Constructor for the student service
- `async Task< IEnumerable< UserInfoDTO > >` [SearchStudentsAsync](#) (string? name, int? groupId)
Search students by name and group
- `async Task< StudentDetailDTO >` [GetStudentDetailsAsync](#) (int id)
Get details of a student

6.50.1 Detailed Description

Service for managing students

Definition at line 19 of file `StudentService.cs`.

6.50.2 Constructor & Destructor Documentation

6.50.2.1 StudentService()

```
Grip.Bll.Services.StudentService.StudentService (
    ApplicationDbContext context,
    IMapper mapper,
    ICurrentTimeProvider currentTimeProvider )
```

Constructor for the student service

Parameters

<i>context</i>	Database context
<i>mapper</i>	Auto mapper
<i>currentTimeProvider</i>	Current time provider

Definition at line 31 of file StudentService.cs.

6.50.3 Member Function Documentation

6.50.3.1 GetStudentDetailsAsync()

```
async Task<StudentDetailDTO> Grip.Bll.Services.StudentService.GetStudentDetailsAsync (
    int id )
```

Get details of a student

Parameters

<i>id</i>	Id of the student
-----------	-------------------

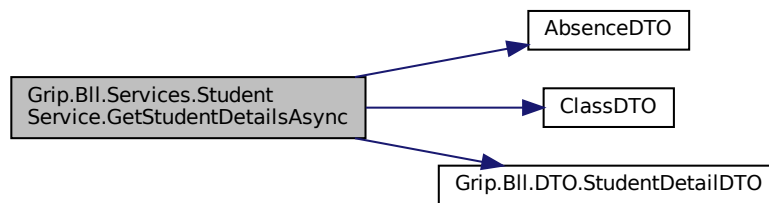
Returns

Detail [DTO](#) of the student

Implements [Grip.Bll.Services.Interfaces.IStudentService](#).

Definition at line 62 of file StudentService.cs.

Here is the call graph for this function:



6.50.3.2 SearchStudentsAsync()

```

async Task<IEnumerable<UserInfoDTO> > Grip.Bll.Services.StudentService.SearchStudentsAsync (
    string? name,
    int? groupId )
  
```

Search students by name and group

Parameters

<i>name</i>	String that the name of the student should contain, null or empty if this filter shouldn't apply
<i>groupId</i>	Id of the group the students are part of, null if the filter shouldn't apply

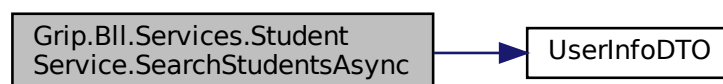
Returns

`IEnumerable` containing description of users

Implements [Grip.Bll.Services.Interfaces.IStudentService](#).

Definition at line 43 of file `StudentService.cs`.

Here is the call graph for this function:



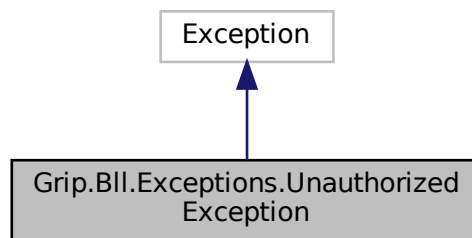
The documentation for this class was generated from the following file:

- `Server/Bll/Services/StudentService.cs`

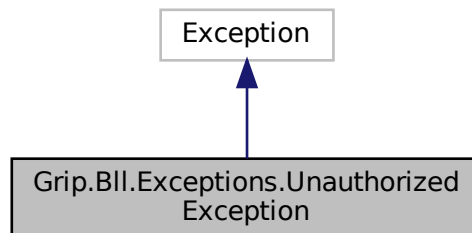
6.51 Grip.Bll.Exceptions.UnauthorizedException Class Reference

Exception thrown when a request needs authorization: HTTP 401.

Inheritance diagram for Grip.Bll.Exceptions.UnauthorizedException:



Collaboration diagram for Grip.Bll.Exceptions.UnauthorizedException:



Public Member Functions

- [UnauthorizedException](#) (string? message=null)
Creates an instance of the exception.

6.51.1 Detailed Description

Exception thrown when a request needs authorization: HTTP 401.

Definition at line 11 of file `UnauthorizedException.cs`.

6.51.2 Constructor & Destructor Documentation

6.51.2.1 UnauthorizedException()

```
Grip.Bl1.Exceptions.UnauthorizedException.UnauthorizedException (
    string? message = null )
```

Creates an instance of the exception.

Parameters

<i>message</i>	Description of the cause of the exception
----------------	---

Definition at line 17 of file UnauthorizedException.cs.

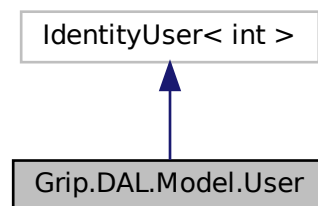
The documentation for this class was generated from the following file:

- [Server/Bl1/Exceptions/UnauthorizedException.cs](#)

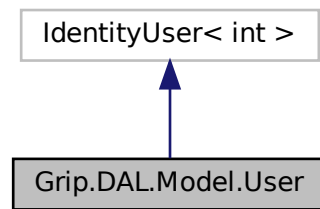
6.52 Grip.DAL.Model.User Class Reference

Represents a user in the database.

Inheritance diagram for Grip.DAL.Model.User:



Collaboration diagram for Grip.DAL.Model.User:



Properties

- `ICollection< PassiveTag > PassiveTags = null!` [get, set]
Gets or sets the collection of passive tags associated with the user.
- `ICollection< Attendance > Attendances = null!` [get, set]
Gets or sets the collection of attendances associated with the user.
- `ICollection< Group > Groups = null!` [get, set]
Gets or sets the collection of groups the user belongs to.
- `ICollection< Exempt > Exemptions = null!` [get, set]
Gets or sets the collection of exemptions associated with the user.
- `ICollection< Exempt > IssuedExemptions = null!` [get, set]
Gets or sets the collection of exemptions issued by the user.

6.52.1 Detailed Description

Represents a user in the database.

Definition at line 8 of file User.cs.

6.52.2 Property Documentation

6.52.2.1 Attendances

```
ICollection<Attendance> Grip.DAL.Model.User.Attendances = null! [get], [set]
```

Gets or sets the collection of attendances associated with the user.

Definition at line 18 of file User.cs.

6.52.2.2 Exemptions

```
ICollection<Exempt> Grip.DAL.Model.User.Exemptions = null! [get], [set]
```

Gets or sets the collection of exemptions associated with the user.

Definition at line 28 of file User.cs.

6.52.2.3 Groups

```
ICollection<Group> Grip.DAL.Model.User.Groups = null! [get], [set]
```

Gets or sets the collection of groups the user belongs to.

Definition at line 23 of file User.cs.

6.52.2.4 IssuedExemptions

```
ICollection<Exempt> Grip.DAL.Model.User.IssuedExemptions = null! [get], [set]
```

Gets or sets the collection of exemptions issued by the user.

Definition at line 33 of file User.cs.

6.52.2.5 PassiveTags

```
ICollection<PassiveTag> Grip.DAL.Model.User.PassiveTags = null! [get], [set]
```

Gets or sets the collection of passive tags associated with the user.

Definition at line 13 of file User.cs.

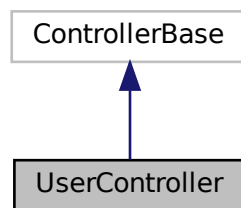
The documentation for this class was generated from the following file:

- [Server/DAL/Model/User.cs](#)

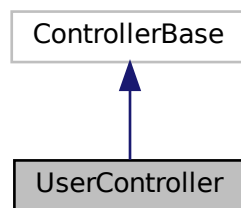
6.53 UserController Class Reference

API endpoints for managing user accounts.

Inheritance diagram for UserController:



Collaboration diagram for UserController:



Public Member Functions

- **UserController** (ILogger< [User](#) > logger, [ApplicationDbContext](#) context, UserManager< [User](#) > userManager, SignInManager< [User](#) > signInManager, RoleManager< [Role](#) > roleManager, IMapper mapper, [IEmailService](#) emailService, [IUserService](#) userService)
Constructor for the user controller.
- ActionResult< IEnumerable< [UserDTO](#) > > **Get** ()
Gets a list of all users.
- async Task< ActionResult< [UserDTO](#) > > **Get** (int id)
Gets a specific user by ID.
- async Task< ActionResult< [UserDTO](#) > > **Register** ([FromBody] [RegisterUserDTO](#) user)
Registers a new user.
- async Task< ActionResult< [LoginResultDTO](#) > > **Login** ([FromBody] [LoginUserDTO](#) user)
Authenticates a user and generates a login token.
- async Task< ActionResult > **ConfirmEmail** ([FromBody] [ConfirmEmailDTO](#) confirmEmailDTO)

- Confirms a user's email address.*
 - async Task< ActionResult > [ForgotPassword](#) ([FromBody] [ForgotPasswordDTO](#) forgotPassword)
- Sends a password reset email to the user.*
 - async Task< ActionResult > [ResetPassword](#) ([FromBody] [ResetPasswordDTO](#) resetPassword)
- Resets the user's password.*
 - async Task< ActionResult > [Delete](#) (int id)
- Deletes a user.*
 - async Task< ActionResult > [Update](#) (int id, [FromBody] [UserDTO](#) user)
- Updates a user's information.*
 - async Task< ActionResult > [AddRole](#) (int userId, string roleId)
- Add role to user*
 - async Task< ActionResult > [RemoveRole](#) (int userId, string roleId)
- Add role to user*

6.53.1 Detailed Description

API endpoints for managing user accounts.

Definition at line 20 of file UserController.cs.

6.53.2 Constructor & Destructor Documentation

6.53.2.1 UserController()

```
UserController.UserController (
    ILogger< User > logger,
    ApplicationDbContext context,
    UserManager< User > userManager,
    SignInManager< User > signInManager,
    RoleManager< Role > roleManager,
    IMapper mapper,
    IEmailService emailService,
    IUserService userService )
```

Constructor for the user controller.

Parameters

<i>logger</i>	Logger for logging
<i>context</i>	Database context
<i>userManager</i>	User manager
<i>signInManager</i>	Sign in manager
<i>roleManager</i>	Role manager
<i>mapper</i>	Auto mapper
<i>emailService</i>	Email service
<i>userService</i>	User service

Definition at line 43 of file UserController.cs.

6.53.3 Member Function Documentation

6.53.3.1 AddRole()

```
async Task<ActionResult> UserController.AddRole (
    int userId,
    string roleId )
```

Add role to user

Parameters

<i>userId</i>	User id
<i>roleId</i>	Role id

Definition at line 222 of file UserController.cs.

6.53.3.2 ConfirmEmail()

```
async Task<ActionResult> UserController.ConfirmEmail (
    [FromBody] ConfirmEmailDTO confirmEmailDTO )
```

Confirms a user's email address.

Parameters

<i>confirmEmailDTO</i>	The email confirmation data.
------------------------	------------------------------

Returns

Ok if the email is confirmed successfully.

Definition at line 135 of file UserController.cs.

6.53.3.3 Delete()

```
async Task<ActionResult> UserController.Delete (
    int id )
```

Deletes a user.

Parameters

<i>id</i>	The ID of the user to delete.
-----------	-------------------------------

Returns

Ok if the user is deleted successfully.

Definition at line 183 of file UserController.cs.

6.53.3.4 ForgotPassword()

```
async Task<ActionResult> UserController.ForgotPassword (
    [FromBody] ForgotPasswordDTO forgotPassword )
```

Sends a password reset email to the user.

Parameters

<i>forgotPassword</i>	The forgot password data.
-----------------------	---------------------------

Returns

Ok if the password reset email is sent successfully.

Definition at line 151 of file UserController.cs.

6.53.3.5 Get() [1/2]

```
ActionResult<IEnumerable<UserDTO> > UserController.Get ( )
```

Gets a list of all users.

Returns

The list of users.

Definition at line 63 of file UserController.cs.

Here is the call graph for this function:



6.53.3.6 Get() [2/2]

```
async Task<ActionResult<UserDTO> > UserController.Get (
    int id )
```

Gets a specific user by ID.

Parameters

<i>id</i>	The ID of the user.
-----------	---------------------

Returns

The user.

Definition at line 78 of file UserController.cs.

6.53.3.7 Login()

```
async Task<ActionResult<LoginResultDTO> > UserController.Login (
    [FromBody] LoginUserDTO user )
```

Authenticates a user and generates a login token.

Parameters

<i>user</i>	The user login data.
-------------	----------------------

Returns

The login result containing the authentication token.

Definition at line 118 of file UserController.cs.

6.53.3.8 Register()

```
async Task<ActionResult<UserDTO> > UserController.Register (
    [FromBody] RegisterUserDTO user )
```

Registers a new user.

Parameters

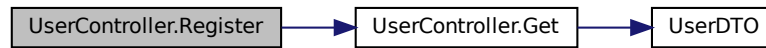
<i>user</i>	The user registration data.
-------------	-----------------------------

Returns

The created user.

Definition at line 102 of file UserController.cs.

Here is the call graph for this function:

**6.53.3.9 RemoveRole()**

```
async Task<ActionResult> UserController.RemoveRole (
    int userId,
    string roleId )
```

Add role to user

Parameters

<i>userId</i>	User id
<i>roleId</i>	Role id

Definition at line 239 of file UserController.cs.

6.53.3.10 ResetPassword()

```
async Task<ActionResult> UserController.ResetPassword (
    [FromBody] ResetPasswordDTO resetPassword )
```

Resets the user's password.

Parameters

<i>resetPassword</i>	The reset password data.
----------------------	--------------------------

Returns

Ok if the password is reset successfully.

Definition at line 167 of file UserController.cs.

6.53.3.11 Update()

```
async Task<ActionResult> UserController.Update (
    int id,
    [FromBody] UserDTO user )
```

Updates a user's information.

Parameters

<i>id</i>	The ID of the user to update.
<i>user</i>	The updated user data.

Returns

No content if the user is updated successfully.

Definition at line 202 of file UserController.cs.

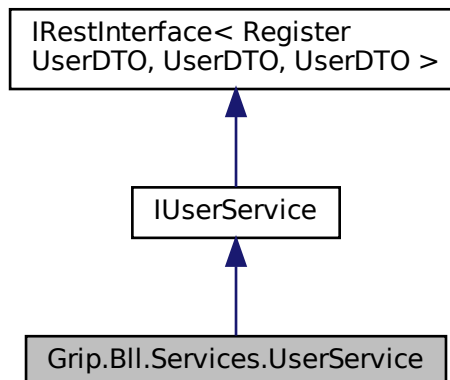
The documentation for this class was generated from the following file:

- Server/Api/Controllers/[UserController.cs](#)

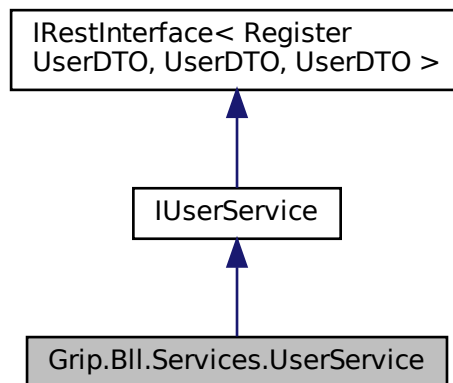
6.54 Grip.Bll.Services.UserService Class Reference

Represents a service for managing users.

Inheritance diagram for Grip.Bll.Services.UserService:



Collaboration diagram for Grip.Bll.Services.UserService:



Public Member Functions

- [UserService](#) (ILogger< [UserService](#) > logger, [ApplicationDbContext](#) context, UserManager< [User](#) > userManager, SignInManager< [User](#) > signInManager, RoleManager< [Role](#) > roleManager, IMapper mapper, [IEmailService](#) emailService)
Initializes a new instance of the [UserService](#) class.
- async Task [AddRole](#) (int userId, string role)
Adds a role to a user.
- async Task [ConfirmEmail](#) ([ConfirmEmailDTO](#) confirmEmailDTO)
Confirms the email for a user.
- async Task< [UserDTO](#) > [Create](#) ([RegisterUserDTO](#) user)
Creates a new user.
- async Task [Delete](#) (int id)
Deletes a user.
- async Task [ForgotPassword](#) ([ForgotPasswordDTO](#) forgotPassword)
Sends a password reset email to the user.
- async Task< [UserDTO](#) > [Get](#) (int id)
Retrieves a user by ID.
- async Task< IEnumerable< [UserDTO](#) > > [GetAll](#) ()
Retrieves all users.
- async Task< [LoginResultDTO](#) > [Login](#) ([LoginUserDTO](#) login)
Logs in a user with the provided credentials.
- async Task [RemoveRole](#) (int userId, string roleId)
Removes a role from a user.
- async Task [ResetPassword](#) ([ResetPasswordDTO](#) dto)
Resets the password for a user.
- async Task [Update](#) ([UserDTO](#) user)
Updates a user.

6.54.1 Detailed Description

Represents a service for managing users.

Definition at line 15 of file UserService.cs.

6.54.2 Constructor & Destructor Documentation

6.54.2.1 UserService()

```
Grip.Bll.Services.UserService.UserService (
    ILogger< UserService > logger,
    ApplicationDbContext context,
    UserManager< User > userManager,
    SignInManager< User > signInManager,
    RoleManager< Role > roleManager,
    IMapper mapper,
    IEmailService emailService )
```

Initializes a new instance of the [UserService](#) class.

Parameters

<i>logger</i>	The logger.
<i>context</i>	The application database context.
<i>userManager</i>	The user manager.
<i>signInManager</i>	The sign-in manager.
<i>roleManager</i>	The role manager.
<i>mapper</i>	The AutoMapper instance.
<i>emailService</i>	The email service.

Definition at line 36 of file UserService.cs.

6.54.3 Member Function Documentation

6.54.3.1 AddRole()

```
async Task Grip.Bll.Services.UserService.AddRole (
    int userId,
    string role )
```

Adds a role to a user.

Parameters

<i>userId</i>	The ID of the user.
<i>role</i>	The role to add.

Returns

A task representing the asynchronous operation.

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 52 of file UserService.cs.

6.54.3.2 ConfirmEmail()

```
async Task Grip.Bll.Services.UserService.ConfirmEmail (
    ConfirmEmailDTO confirmEmailDTO )
```

Confirms the email for a user.

Parameters

<i>confirmEmailDTO</i>	The confirm email DTO .
------------------------	---

Returns

A task representing the asynchronous operation.

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 66 of file UserService.cs.

6.54.3.3 Create()

```
async Task<UserDTO> Grip.Bll.Services.UserService.Create (
    RegisterUserDTO user )
```

Creates a new user.

Parameters

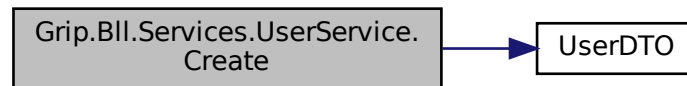
<i>user</i>	The register user DTO .
-------------	---

Returns

The created user [DTO](#).

Definition at line 100 of file UserService.cs.

Here is the call graph for this function:

**6.54.3.4 Delete()**

```
async Task Grip.Bll.Services.UserService.Delete (
    int id )
```

Deletes a user.

Parameters

<i>id</i>	The ID of the user.
-----------	---------------------

Returns

A task representing the asynchronous operation.

Definition at line 124 of file UserService.cs.

6.54.3.5 ForgotPassword()

```
async Task Grip.Bll.Services.UserService.ForgotPassword (
    ForgotPasswordDTO forgotPassword )
```

Sends a password reset email to the user.

Parameters

<i>forgotPassword</i>	The forgot password DTO .
-----------------------	---

Returns

A task representing the asynchronous operation.

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 148 of file UserService.cs.

6.54.3.6 Get()

```
async Task<UserDTO> Grip.Bll.Services.UserService.Get (
    int id )
```

Retrieves a user by ID.

Parameters

<i>id</i>	The ID of the user.
-----------	---------------------

Returns

The user [DTO](#).

Definition at line 167 of file UserService.cs.

Here is the call graph for this function:

**6.54.3.7 GetAll()**

```
async Task<IEnumerable<UserDTO> > Grip.Bll.Services.UserService.GetAll ( )
```

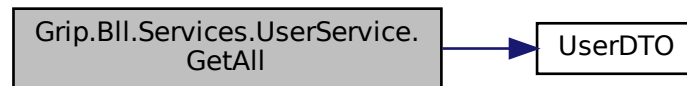
Retrieves all users.

Returns

A collection of user DTOs.

Definition at line 181 of file UserService.cs.

Here is the call graph for this function:

**6.54.3.8 Login()**

```
async Task<LoginResultDTO> Grip.Bll.Services.UserService.Login (  
    LoginUserDTO login )
```

Logs in a user with the provided credentials.

Parameters

<i>login</i>	The login user DTO .
--------------	--------------------------------------

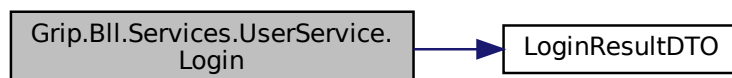
Returns

The login result [DTO](#).

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 191 of file UserService.cs.

Here is the call graph for this function:



6.54.3.9 RemoveRole()

```
async Task Grip.Bll.Services.UserService.RemoveRole (
    int userId,
    string roleId )
```

Removes a role from a user.

Parameters

<i>userId</i>	The ID of the user.
<i>roleId</i>	The ID of the role.

Returns

A task representing the asynchronous operation.

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 223 of file UserService.cs.

6.54.3.10 ResetPassword()

```
async Task Grip.Bll.Services.UserService.ResetPassword (
    ResetPasswordDTO dto )
```

Resets the password for a user.

Parameters

<i>dto</i>	The reset password DTO .
------------	--

Returns

A task representing the asynchronous operation.

Implements [Grip.Bll.Services.Interfaces.IUserService](#).

Definition at line 246 of file UserService.cs.

6.54.3.11 Update()

```
async Task Grip.Bll.Services.UserService.Update (
    UserDTO user )
```

Updates a user.

Parameters

<i>user</i>	The user DTO .
-------------	--------------------------------

Returns

A task representing the asynchronous operation.

Definition at line 269 of file UserService.cs.

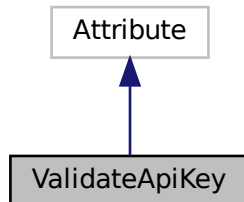
The documentation for this class was generated from the following file:

- Server/Bll/Services/[UserService.cs](#)

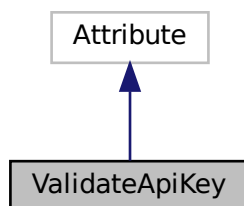
6.55 ValidateApiKey Class Reference

Attribute for marking a controller method as requiring a valid API key.

Inheritance diagram for ValidateApiKey:



Collaboration diagram for ValidateApiKey:



6.55.1 Detailed Description

Attribute for marking a controller method as requiring a valid API key.

Definition at line 11 of file `ValidateApiKey.cs`.

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

- `Server/Api/Middleware/ValidateApiKey.cs`

Chapter 7

File Documentation

7.1 Server/Api/Controllers/AttendanceController.cs File Reference

Classes

- class [AttendanceController](#)
Controller for handling attendance

7.2 Server/Api/Controllers/ClassController.cs File Reference

Classes

- class [Grip.Api.Controllers.ClassController](#)
Endpoint for managing classes

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Controllers](#)

7.3 Server/Api/Controllers/ExemptController.cs File Reference

Classes

- class [Grip.Api.Controllers.ExemptController](#)
Endpoint for managing exempt items

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Controllers](#)

7.4 Server/Api/Controllers/GroupController.cs File Reference

Classes

- class [GroupController](#)
Represents a controller for managing groups.

7.5 Server/Api/Controllers/PassiveTagController.cs File Reference

Classes

- class [Grip.Api.Controllers.PassiveTagController](#)
API controller for managing passive tags.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Controllers](#)

7.6 Server/Api/Controllers/StationController.cs File Reference

Classes

- class [StationController](#)
Represents a controller for managing stations.

7.7 Server/Api/Controllers/StudentController.cs File Reference

Classes

- class [Grip.Api.Controllers.StudentController](#)
Controller for managing students.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Controllers](#)

7.8 Server/Api/Controllers/UserController.cs File Reference

Classes

- class [UserController](#)
API endpoints for managing user accounts.

7.9 Server/Api/Hubs/IStationClient.cs File Reference

Classes

- interface [Grip.Api.Hubs.IStationClient](#)
This interface is used to define the methods that can be called from the station hub on the client

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Hubs](#)

7.10 Server/Api/Hubs/StationHub.cs File Reference

Classes

- class [Grip.Api.Hubs.StationHub](#)
SignalR Hub for station related events

Namespaces

- namespace [Grip](#)
- namespace [Grip.Api](#)
- namespace [Grip.Api.Hubs](#)

7.11 Server/Api/Middleware/ApiKeyValidationMiddleware.cs File Reference

Classes

- class [ApiKeyValidationMiddleware](#)
Validates the api key in the header of the request.

7.12 Server/Api/Middleware/DeChunkingMiddleware.cs File Reference

Classes

- class [DeChunkingMiddleware](#)

Middleware to de-chunk HTTP requests, because ucontroller doesn't support chunked responses

7.13 Server/Api/Middleware/NotChunked.cs File Reference

Classes

- class [NotChunked](#)

Attribute for marking a controller method write response in a not chunked manner.

7.14 Server/Api/Middleware/ValidateApiKey.cs File Reference

Classes

- class [ValidateApiKey](#)

Attribute for marking a controller method as requiring a valid API key.

7.15 Server/Bll/DTO/Attendance/ActiveAttendanceDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.DTO](#)

Functions

- record [Grip.Bll.DTO.ActiveAttendanceDTO](#) ([Required][RegularExpression(@"^\d*_\d*_\d*\$")] string Message, [Required] string Token)

[DTO](#) for the active attendance registration

7.16 Server/Bll/DTO/Attendance/AttendanceDTO.cs File Reference

Functions

- record [AttendanceDTO](#) ([ClassDTO](#) Class, bool HasExempt, bool Present, DateTime? AuthenticationTime)

A record that represents a users attendance at a class, or lack thereof

7.16.1 Function Documentation

7.16.1.1 AttendanceDTO()

```
record AttendanceDTO (
    ClassDTO Class,
    bool HasExempt,
    bool Present,
    DateTime? AuthenticationTime )
```

A record that represents a users attendance at a class, or lack thereof

Parameters

<i>Class</i>	ClassDTO of the class in question
<i>Present</i>	Whether the user was present at the class
<i>HasExempt</i>	Whether the user has an exempt for the class
<i>AuthenticationTime</i>	Scan time if Present is true, null otherwise

7.17 Server/BII/DTO/Attendance/PassiveAttendanceDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.BII](#)
- namespace [Grip.BII.DTO](#)

Functions

- record [Grip.BII.DTO.PassiveAttendanceDTO](#) ([Required] int StationId, [Required] Int64 SerialNumber)
DTO for the passive attendance registration

7.18 Server/BII/DTO/Class/ClassDTO.cs File Reference

Functions

- record [ClassDTO](#) (int Id, string Name, DateTime StartDateTime, [UserInfoDTO](#) Teacher, [GroupDTO](#) Group)
DTO for the class

7.18.1 Function Documentation

7.18.1.1 ClassDTO()

```
record ClassDTO (
    int Id,
    string Name,
    DateTime StartDateTime,
    UserInfoDTO Teacher,
    GroupDTO Group )
```

DTO for the class

Parameters

<i>Id</i>	Id of the class
<i>Name</i>	Name of the class
<i>StartDateTime</i>	Start of the class
<i>Teacher</i>	Informtaion about the teacher
<i>Group</i>	Infromation about the group

7.19 Server/Bll/DTO/Class/CreateClassDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.DTO](#)

Variables

- record [Grip.Bll.DTO.CreateClassDTO](#)
Represents the data transfer object for creating a class.

7.20 Server/Bll/DTO/Exempt/CreateExemptDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.DTO](#)

Variables

- record [Grip.Bll.DTO.CreateExemptDTO](#)
Represents the data transfer object for creating an exemption.

7.21 Server/BII/DTO/Exempt/ExemptDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.BII](#)
- namespace [Grip.BII.DTO](#)

Variables

- record [Grip.BII.DTO.ExemptDTO](#)
Represents a data transfer object (DTO) for an exemption.

7.22 Server/BII/DTO/Group/GroupDTO.cs File Reference

Functions

- record [GroupDTO](#) ([Required] int Id, [Required] string Name)
Represents the data transfer object for a group.

7.22.1 Function Documentation

7.22.1.1 GroupDTO()

```
record GroupDTO (  
    [Required] int Id,  
    [Required] string Name )
```

Represents the data transfer object for a group.

Parameters

<i>Id</i>	Id of the group
<i>Name</i>	Name of the group

7.23 Server/BII/DTO/PassiveTag/CreatePassiveTagDTO.cs File Reference

Functions

- record [CreatePassiveTagDTO](#) ([Required] Int64 SerialNumber, [Required] int UserId)
DTO for creating a passive tag

7.23.1 Function Documentation

7.23.1.1 CreatePassiveTagDTO()

```
record CreatePassiveTagDTO (
    [Required] Int64 SerialNumber,
    [Required] int UserId )
```

DTO for creating a passive tag

Parameters

<i>SerialNumber</i>	Serial number of the passive tag
<i>UserId</i>	Id of the owner of the tag

7.24 Server/Bll/DTO/PassiveTag/PassiveTagDTO.cs File Reference

Functions

- record [PassiveTagDTO](#) (int Id, Int64 SerialNumber, [UserInfoDTO](#) User)
DTO for a passive tag

7.24.1 Function Documentation

7.24.1.1 PassiveTagDTO()

```
record PassiveTagDTO (
    int Id,
    Int64 SerialNumber,
    UserInfoDTO User )
```

DTO for a passive tag

Parameters

<i>Id</i>	Id of the tag
<i>SerialNumber</i>	Serial number of the tag
<i>User</i>	Information of the owner

Returns

7.25 Server/BII/DTO/PassiveTag/UpdatePassiveTagDTO.cs File Reference

Functions

- record [UpdatePassiveTagDTO](#) ([Required] int Id, [Required] Int64 SerialNumber, [Required] int UserId)
DTO for updating a passive tag

7.25.1 Function Documentation

7.25.1.1 UpdatePassiveTagDTO()

```
record UpdatePassiveTagDTO (  
    [Required] int Id,  
    [Required] Int64 SerialNumber,  
    [Required] int UserId )
```

DTO for updating a passive tag

Parameters

<i>Id</i>	Id of the tag
<i>SerialNumber</i>	Serial number of the tag
<i>UserId</i>	Id of the owner

7.26 Server/BII/DTO/Station/StationScanDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.BII](#)
- namespace [Grip.BII.DTO](#)

Variables

- record [Grip.BII.DTO.StationScanDTO](#)
Represents the data transfer object for a station scan.

7.27 Server/BII/DTO/Station/StationSecretKeyDTO.cs File Reference

Functions

- record [StationSecretKeyDTO](#) (string SecretKey)
DTO for the station secret key

7.27.1 Function Documentation

7.27.1.1 StationSecretKeyDTO()

```
record StationSecretKeyDTO (  
    string SecretKey )
```

DTO for the station secret key

Parameters

<i>SecretKey</i>	The secret key for the station
------------------	--------------------------------

7.28 Server/BII/DTO/Student/AbsenceDTO.cs File Reference

Functions

- record [AbsenceDTO](#) ([ClassDTO](#) Class, bool HasExempt)
Represents the data transfer object for a student absence.

7.28.1 Function Documentation

7.28.1.1 AbsenceDTO()

```
record AbsenceDTO (  
    ClassDTO Class,  
    bool HasExempt )
```

Represents the data transfer object for a student absence.

Parameters

<i>Class</i>	Info about the class the student was absent from
<i>HasExempt</i>	Whether the student had an exemption that was valid at the time

7.29 Server/BII/DTO/Student/StudentAttendanceDTO.cs File Reference

Functions

- record [StudentAttendanceDTO](#) ([UserInfoDTO](#) User, IEnumerable< [AttendanceDTO](#) > Attendances)
Represents the data transfer object for a student attendance.

7.29.1 Function Documentation

7.29.1.1 StudentAttendanceDTO()

```
record StudentAttendanceDTO (
    UserInfoDTO User,
    IEnumerable< AttendanceDTO > Attendances )
```

Represents the data transfer object for a student attendance.

Parameters

<i>User</i>	Info about the user
<i>Attendances</i>	List of attendances

7.30 Server/BII/DTO/Student/StudentDetailDTO.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.BII](#)
- namespace [Grip.BII.DTO](#)

Functions

- record [Grip.BII.DTO.StudentDetailDTO](#) ()
Represents the data transfer object for a student detail.

7.31 Server/BII/DTO/User/ConfirmEmailDTO.cs File Reference

Functions

- record [ConfirmEmailDTO](#) ([Required][EmailAddress] string Email, [Required] string Token, [Required][RegularExpression([Grip.Utils.Consts.UserPasswordRegex](#))] string Password)
DTO for confirming the email address of a user

7.31.1 Function Documentation

7.31.1.1 ConfirmEmailDTO()

```
record ConfirmEmailDTO (
    [EmailAddress] string Email,
    [Required] string Token,
    [RegularExpression(Grip.Utils.Consts.UserPasswordRegex)] string Password )
```

DTO for confirming the email address of a user

Parameters

<i>Email</i>	Email of the user
<i>Token</i>	Token sent to the email previously
<i>Password</i>	Password to be set for the user

7.32 Server/Bll/DTO/User/ForgotPasswordDTO.cs File Reference

Functions

- record [ForgotPasswordDTO](#) ([Required][EmailAddress] string Email)
DTO for the forgot password request

7.32.1 Function Documentation

7.32.1.1 ForgotPasswordDTO()

```
record ForgotPasswordDTO (
    [EmailAddress] string Email )
```

DTO for the forgot password request

Parameters

<i>Email</i>	Email to send the recovery link to
--------------	------------------------------------

7.33 Server/BII/DTO/User/LoginResultDTO.cs File Reference

Functions

- record [LoginResultDTO](#) (string UserName, string Email, string[] Roles)
DTO for the login result

7.33.1 Function Documentation

7.33.1.1 LoginResultDTO()

```
record LoginResultDTO (  
    string UserName,  
    string Email,  
    string[] Roles )
```

DTO for the login result

Parameters

<i>UserName</i>	Username of the user
<i>Email</i>	Email address of the user
<i>Roles</i>	List of roles of the user

Creates an empty instance of the [LoginResultDTO](#) class.

Definition at line 9 of file LoginResultDTO.cs.

7.34 Server/BII/DTO/User/LoginUserDTO.cs File Reference

Functions

- record [LoginUserDTO](#) ([Required][EmailAddress] string Email, [Required] string Password)
DTO for the login request

7.34.1 Function Documentation

7.34.1.1 LoginUserDTO()

```
record LoginUserDTO (  
    [EmailAddress] string Email,  
    [Required] string Password )
```

DTO for the login request

Parameters

<i>Email</i>	Email of the user
<i>Password</i>	Password of the user

Returns

7.35 Server/BII/DTO/User/RegisterUserDTO.cs File Reference

Functions

- record [RegisterUserDTO](#) ([Required][EmailAddress] string Email, [Required][RegularExpression(Grip.Utils.Consts.UserNameRegex, ErrorMessage="Invalid name")] string Name)

Represents the data transfer object for registering a user.

7.35.1 Function Documentation

7.35.1.1 RegisterUserDTO()

```
record RegisterUserDTO (  
    [EmailAddress] string Email,  
    [RegularExpression(Grip.Utils.Consts.UserNameRegex, ErrorMessage="Invalid name")]  
    string Name )
```

Represents the data transfer object for registering a user.

Parameters

<i>Email</i>	Eamil address of the user
<i>Name</i>	name of the user

Returns

7.36 Server/BII/DTO/User/ResetPasswordDTO.cs File Reference

Functions

- record [ResetPasswordDTO](#) ([Required][EmailAddress] string Email, [Required] string Token, [Required][RegularExpression(Grip.Utils.Consts.UserPasswordRegex)] string Password)

DTO for the reset password request

7.36.1 Function Documentation

7.36.1.1 ResetPasswordDTO()

```
record ResetPasswordDTO (
    [EmailAddress] string Email,
    [Required] string Token,
    [RegularExpression(Grip.Utls.Consts.UserPasswordRegex)] string Password )
```

DTO for the reset password request

Parameters

<i>Email</i>	Email of the user
<i>Token</i>	Token sent to the user
<i>Password</i>	New password to be set

Returns

7.37 Server/Bll/DTO/User/UserDTO.cs File Reference

Functions

- record [UserDTO](#) ([Required] int Id, [Required][EmailAddress] string Email, [Required][RegularExpression(Grip.Utls.Consts.UserNameRegex)] string UserName, bool EmailConfirmed)
DTO for the user

7.37.1 Function Documentation

7.37.1.1 UserDTO()

```
record UserDTO (
    [Required] int Id,
    [EmailAddress] string Email,
    [RegularExpression(Grip.Utls.Consts.UserNameRegex)] string UserName,
    bool EmailConfirmed )
```

DTO for the user

Parameters

<i>Id</i>	Id of the user
<i>Email</i>	Email of the user
<i>UserName</i>	Username of the user
<i>EmailConfirmed</i>	Wether the user verified their email address

Returns

Creates an empty user DTO

Definition at line 17 of file UserDTO.cs.

7.38 Server/BII/DTO/User/UserInfoDTO.cs File Reference

Functions

- record [UserInfoDTO](#) (int Id, string UserName)

Descriptive information about a user

7.38.1 Function Documentation

7.38.1.1 UserInfoDTO()

```
record UserInfoDTO (  
    int Id,  
    string UserName )
```

Descriptive information about a user

Parameters

<i>Id</i>	Id of the user
<i>UserName</i>	Username of the user

Returns

7.39 Server/Bll/Exceptions/BadRequestException.cs File Reference

Classes

- class [Grip.Bll.Exceptions.BadRequestException](#)
Exception thrown when a request is invalid: HTTP 400.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Exceptions](#)

7.40 Server/Bll/Exceptions/DbConcurrencyException.cs File Reference

Classes

- class [Grip.Bll.Exceptions.DbConcurrencyException](#)
Exception thrown when a db update failed due to concurrency: HTTP 409.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Exceptions](#)

7.41 Server/Bll/Exceptions/NotFoundException.cs File Reference

Classes

- class [Grip.Bll.Exceptions.NotFoundException](#)
Exception thrown when a resource is not found: HTTP 404.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Exceptions](#)

7.42 Server/Bll/Exceptions/UnauthorizedException.cs File Reference

Classes

- class [Grip.Bll.Exceptions.UnauthorizedException](#)
Exception thrown when a request needs authorization: HTTP 401.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Exceptions](#)

7.43 Server/Bll/Providers/CurrentTimeProvider.cs File Reference

Classes

- class [Grip.Bll.Providers.CurrentTimeProvider](#)
Class for abstracting the current time for testing purposes.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Providers](#)

7.44 Server/Bll/Providers/HMACTokenProvider.cs File Reference

Classes

- class [HMACTokenProvider](#)
Generates and validates HMAC tokens for stations.

7.45 Server/Bll/Providers/ICurrentTimeProvider.cs File Reference

Classes

- interface [Grip.Bll.Providers.ICurrentTimeProvider](#)
Interface for abstracting the current time for testing purposes.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Providers](#)

7.46 Server/Bll/Providers/IStationTokenProvider.cs File Reference

Classes

- interface [IStationTokenProvider](#)
Interface for generating and validating tokens for stations.

7.47 Server/Bll/Services/AttendanceService.cs File Reference

Classes

- class [AttendanceService](#)
Provides functionality related to attendance management.

7.48 Server/Bll/Services/ClassService.cs File Reference

Classes

- class [ClassService](#)
Service class for managing classes.

7.49 Server/Bll/Services/EmailService.cs File Reference

Classes

- class [EmailService](#)
Service class for sending emails.

7.50 Server/Bll/Services/ExemptService.cs File Reference

Classes

- class [Grip.Bll.Services.ExemptService](#)
Service class for managing exempts.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.51 Server/Bll/Services/GroupService.cs File Reference

Classes

- class [Grip.Bll.Services.GroupService](#)
Service class for managing groups.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.52 Server/Bll/Services/Interfaces/IAttendanceService.cs File Reference

Classes

- interface [IAttendanceService](#)
Interface for the attendance service

7.53 Server/Bll/Services/Interfaces/IClassService.cs File Reference

Classes

- interface [IClassService](#)
Interface for the class service

7.54 Server/Bll/Services/Interfaces/EmailService.cs File Reference

Classes

- interface [Grip.Bll.Services.Interfaces.IEmailService](#)
Represents a service for sending emails.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)
- namespace [Grip.Bll.Services.Interfaces](#)

7.55 Server/Bll/Services/Interfaces/IExemptService.cs File Reference

Classes

- interface [IExemptService](#)
Represents an interface for the exempt service.

7.56 Server/Bll/Services/Interfaces/IGroupService.cs File Reference

Classes

- interface [IGroupService](#)
Interface representing the Group Service, which provides CRUD operations for groups as well as specific group operations.

7.57 Server/Bll/Services/Interfaces/IPassiveTagService.cs File Reference

Classes

- interface [IPassiveTagService](#)
Represents an interface for the passive tag service.

7.58 Server/Bll/Services/Interfaces/IRestInterface.cs File Reference

Classes

- interface [Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >](#)
Represents a generic REST interface for CRUD operations.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)
- namespace [Grip.Bll.Services.Interfaces](#)

7.59 Server/Bll/Services/Interfaces/IStationService.cs File Reference

Classes

- interface [Grip.Bll.Services.Interfaces.IStationService](#)
Represents an interface for the station service.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)
- namespace [Grip.Bll.Services.Interfaces](#)

7.60 Server/Bll/Services/Interfaces/IStudentService.cs File Reference

Classes

- interface [Grip.Bll.Services.Interfaces.IStudentService](#)
Interface for the student service

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)
- namespace [Grip.Bll.Services.Interfaces](#)

7.61 Server/Bll/Services/Interfaces/IUserService.cs File Reference

Classes

- interface [Grip.Bll.Services.Interfaces.IUserService](#)
Represents an interface for the user service.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)
- namespace [Grip.Bll.Services.Interfaces](#)

7.62 Server/Bll/Services/PassiveTagService.cs File Reference

Classes

- class [Grip.Bll.Services.PassiveTagService](#)
Service class for managing passive tags.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.63 Server/Bll/Services/StationService.cs File Reference

Classes

- class [Grip.Bll.Services.StationService](#)
Provides methods for managing stations.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.64 Server/Bll/Services/StudentService.cs File Reference

Classes

- class [Grip.Bll.Services.StudentService](#)
Service for managing students

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.65 Server/Bll/Services/UserService.cs File Reference

Classes

- class [Grip.Bll.Services.UserService](#)
Represents a service for managing users.

Namespaces

- namespace [Grip](#)
- namespace [Grip.Bll](#)
- namespace [Grip.Bll.Services](#)

7.66 Server/Bll/Utils/Consts.cs File Reference

Classes

- class [Grip.Utils.Consts](#)
Contains constants

Namespaces

- namespace [Grip](#)
- namespace [Grip.Utils](#)

7.67 Server/DAL/ApplicationDbContext.cs File Reference

Classes

- class [ApplicationDbContext](#)
Represents the application's database context.

7.68 Server/DAL/MappingProfile.cs File Reference

Classes

- class [Grip.DAL.MappingProfile](#)
Mapping profile for AutoMapper.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)

7.69 Server/DAL/Migrations/20230317172813_Identity.cs File Reference

Classes

- class [Grip.DAL.Migrations.Identity](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.70 Server/DAL/Migrations/20230317172813_Identity.Designer.cs File Reference

Classes

- class [Grip.DAL.Migrations.Identity](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.71 Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.cs File Reference

Classes

- class [Grip.DAL.Migrations.AddClassesAndAttendance](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.72 Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.Designer.cs File Reference

Classes

- class [Grip.DAL.Migrations.AddClassesAndAttendance](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.73 Server/DAL/Migrations/20230501161544_AddExempts.cs File Reference

Classes

- class [Grip.DAL.Migrations.AddExempts](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.74 Server/DAL/Migrations/20230501161544_AddExempts.Designer.cs File Reference

Classes

- class [Grip.DAL.Migrations.AddExempts](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.75 Server/DAL/Migrations/20230511180151_ClassStationRelation← Added.cs File Reference

Classes

- class [Grip.DAL.Migrations.ClassStationRelationAdded](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.76 Server/DAL/Migrations/20230511180151_ClassStationRelation← Added.Designer.cs File Reference

Classes

- class [Grip.DAL.Migrations.ClassStationRelationAdded](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.77 Server/DAL/Migrations/ApplicationDbContextModelSnapshot.cs File Reference

Classes

- class [Grip.DAL.Migrations.ApplicationDbContextModelSnapshot](#)

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Migrations](#)

7.78 Server/DAL/Model/Attendance.cs File Reference

Classes

- class [Grip.DAL.Model.Attendance](#)
Represents the attendance information of a user at a station.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

7.79 Server/DAL/Model/Class.cs File Reference

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

Variables

- record [Grip.DAL.Model.Class](#)
Represents a class in the database.

7.80 Server/DAL/Model/Exempt.cs File Reference

Classes

- class [Grip.DAL.Model.Exempt](#)
Represents an exemption in the database.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

7.81 Server/DAL/Model/Group.cs File Reference

Classes

- class [Grip.DAL.Model.Group](#)
Represents a group in the database.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

7.82 Server/DAL/Model/PassiveTag.cs File Reference

Classes

- class [Grip.DAL.Model.PassiveTag](#)
Represents a passive tag in the database.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

7.83 Server/DAL/Model/Role.cs File Reference

Classes

- class [Role](#)
Represents a role in the database.

7.84 Server/DAL/Model/Station.cs File Reference

Classes

- class [Grip.DAL.Model.Station](#)
Represents a station in the database.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

7.85 Server/DAL/Model/User.cs File Reference

Classes

- class [Grip.DAL.Model.User](#)
Represents a user in the database.

Namespaces

- namespace [Grip](#)
- namespace [Grip.DAL](#)
- namespace [Grip.DAL.Model](#)

Index

- AbsenceDTO
 - AbsenceDTO.cs, [162](#)
- AbsenceDTO.cs
 - AbsenceDTO, [162](#)
- ActiveAttendanceDTO
 - Grip.Bll.DTO, [12](#)
- ActiveAuthentication
 - AttendanceController, [30](#)
- AddRole
 - Grip.Bll.Services.Interfaces.IUserService, [101](#)
 - Grip.Bll.Services.UserService, [145](#)
 - UserController, [138](#)
- AddUserToGroup
 - Grip.Bll.Services.GroupService, [72](#)
 - GroupController, [67](#)
 - IGroupService, [90](#)
- Admin
 - Role, [118](#)
- ApiKeyValidationMiddleware, [20](#)
 - ApiKeyValidationMiddleware, [21](#)
 - InvokeAsync, [21](#)
- ApplicationDbContext, [22](#)
 - ApplicationDbContext, [23](#)
 - Attendances, [24](#)
 - Classes, [24](#)
 - Exempts, [24](#)
 - Groups, [25](#)
 - OnConfiguring, [23](#)
 - OnModelCreating, [24](#)
 - PassiveTags, [25](#)
 - Stations, [25](#)
- AttendanceController, [28](#)
 - ActiveAuthentication, [30](#)
 - AttendanceController, [29](#)
 - GetAttendanceForDay, [30](#)
 - PassiveAuthentication, [30](#)
- AttendanceDTO
 - AttendanceDTO.cs, [157](#)
- AttendanceDTO.cs
 - AttendanceDTO, [157](#)
- Attendances
 - ApplicationDbContext, [24](#)
 - Grip.DAL.Model.Station, [119](#)
 - Grip.DAL.Model.User, [134](#)
- AttendanceService, [31](#)
 - AttendanceService, [32](#)
 - GetAttendanceForDay, [32](#)
 - VerifyPassiveScan, [33](#)
 - VerifyPhoneScan, [33](#)
- BadRequestException
 - Grip.Bll.Exceptions.BadRequestException, [35](#)
- BuildModel
 - Grip.DAL.Migrations.ApplicationDbContextModelSnapshot, [26](#)
- BuildTargetModel
 - Grip.DAL.Migrations.AddClassesAndAttendance, [18](#)
 - Grip.DAL.Migrations.AddExempts, [19](#)
 - Grip.DAL.Migrations.ClassStationRelationAdded, [46](#)
 - Grip.DAL.Migrations.Identity, [85](#)
- ClassController
 - Grip.Api.Controllers.ClassController, [37](#)
- ClassDTO
 - ClassDTO.cs, [157](#)
- ClassDTO.cs
 - ClassDTO, [157](#)
- Classes
 - ApplicationDbContext, [24](#)
 - Grip.DAL.Model.Group, [65](#)
 - Grip.DAL.Model.Station, [119](#)
- ClassService, [40](#)
 - ClassService, [41](#)
 - Create, [41](#)
 - Delete, [42](#)
 - Get, [42](#)
 - GetAll, [44](#)
 - GetClassesForUserOnDay, [44](#)
 - Update, [45](#)
- ConfirmEmail
 - Grip.Bll.Services.Interfaces.IUserService, [101](#)
 - Grip.Bll.Services.UserService, [146](#)
 - UserController, [138](#)
- ConfirmEmailDTO
 - ConfirmEmailDTO.cs, [164](#)
- ConfirmEmailDTO.cs
 - ConfirmEmailDTO, [164](#)
- Create
 - ClassService, [41](#)
 - Grip.Bll.Services.ExemptService, [61](#)
 - Grip.Bll.Services.GroupService, [73](#)
 - Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >, [93](#)
 - Grip.Bll.Services.PassiveTagService, [114](#)
 - Grip.Bll.Services.UserService, [146](#)
 - IExemptService, [89](#)
- CreatePassiveTagDTO
 - CreatePassiveTagDTO.cs, [160](#)

- CreatePassiveTagDTO.cs
 - CreatePassiveTagDTO, [160](#)
- DbConcurrencyException
 - Grip.Bll.Exceptions.DbConcurrencyException, [50](#)
- DeChunkingMiddleware, [50](#)
 - DeChunkingMiddleware, [50](#)
 - InvokeAsync, [51](#)
- Delete
 - ClassService, [42](#)
 - Grip.Bll.Services.ExemptService, [62](#)
 - Grip.Bll.Services.GroupService, [73](#)
 - Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >, [94](#)
 - Grip.Bll.Services.PassiveTagService, [115](#)
 - Grip.Bll.Services.UserService, [147](#)
 - UserController, [138](#)
- DeleteClass
 - Grip.Api.Controllers.ClassController, [37](#)
- DeleteExempt
 - Grip.Api.Controllers.ExemptController, [57](#)
- DeleteGroup
 - GroupController, [68](#)
- DeletePassiveTag
 - Grip.Api.Controllers.PassiveTagController, [110](#)
- Doorman
 - Role, [118](#)
- Down
 - Grip.DAL.Migrations.AddClassesAndAttendance, [18](#)
 - Grip.DAL.Migrations.AddExempts, [20](#)
 - Grip.DAL.Migrations.ClassStationRelationAdded, [47](#)
 - Grip.DAL.Migrations.Identity, [85](#)
- EmailService, [51](#)
 - EmailService, [52](#)
 - SendEmailAsync, [53](#)
- ExemptController
 - Grip.Api.Controllers.ExemptController, [57](#)
- Exemptions
 - Grip.DAL.Model.User, [134](#)
- Exempts
 - ApplicationDbContext, [24](#)
- ExemptService
 - Grip.Bll.Services.ExemptService, [61](#)
- ForgotPassword
 - Grip.Bll.Services.Interfaces.IUserService, [102](#)
 - Grip.Bll.Services.UserService, [147](#)
 - UserController, [140](#)
- ForgotPasswordDTO
 - ForgotPasswordDTO.cs, [164](#)
- ForgotPasswordDTO.cs
 - ForgotPasswordDTO, [164](#)
- GenerateToken
 - HMACTokenProvider, [79](#)
 - IStationTokenProvider, [97](#)
- Get
 - ClassService, [42](#)
 - Grip.Bll.Services.ExemptService, [62](#)
 - Grip.Bll.Services.GroupService, [75](#)
 - Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >, [94](#)
 - Grip.Bll.Services.PassiveTagService, [115](#)
 - Grip.Bll.Services.UserService, [148](#)
 - UserController, [140](#)
- GetAll
 - ClassService, [44](#)
 - Grip.Bll.Services.ExemptService, [63](#)
 - Grip.Bll.Services.GroupService, [75](#)
 - Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >, [94](#)
 - Grip.Bll.Services.PassiveTagService, [116](#)
 - Grip.Bll.Services.UserService, [148](#)
- GetAllForUser
 - Grip.Bll.Services.ExemptService, [63](#)
- GetAttendanceForDay
 - AttendanceController, [30](#)
 - AttendanceService, [32](#)
 - IAttendanceService, [80](#)
- GetByUser
 - Grip.Bll.Services.ExemptService, [63](#)
- GetClass
 - Grip.Api.Controllers.ClassController, [37](#), [38](#)
- GetClassesForUserOnDay
 - ClassService, [44](#)
 - IClassService, [83](#)
- GetClassOnDay
 - Grip.Api.Controllers.ClassController, [38](#)
- GetExempt
 - Grip.Api.Controllers.ExemptController, [58](#)
- GetGroup
 - GroupController, [68](#)
- GetKey
 - StationController, [122](#)
- GetPassiveTag
 - Grip.Api.Controllers.PassiveTagController, [111](#)
- GetPassiveTags
 - Grip.Api.Controllers.PassiveTagController, [111](#)
- GetSecretKey
 - Grip.Bll.Services.Interfaces.IStationService, [96](#)
 - Grip.Bll.Services.StationService, [125](#)
- GetStudentDetails
 - Grip.Api.Controllers.StudentController, [128](#)
- GetStudentDetailsAsync
 - Grip.Bll.Services.Interfaces.IStudentService, [99](#)
 - Grip.Bll.Services.StudentService, [130](#)
- GetStudents
 - Grip.Api.Controllers.StudentController, [128](#)
- GetUsersInGroup
 - Grip.Bll.Services.GroupService, [76](#)
 - IGroupService, [91](#)
- Grip, [11](#)
- Grip.Api, [11](#)
- Grip.Api.Controllers, [11](#)

- Grip.Api.Controllers.ClassController, 36
 - ClassController, 37
 - DeleteClass, 37
 - GetClass, 37, 38
 - GetClassOnDay, 38
 - PostClass, 39
 - PutClass, 39
- Grip.Api.Controllers.ExemptController, 56
 - DeleteExempt, 57
 - ExemptController, 57
 - GetExempt, 58
 - PostExempt, 58
- Grip.Api.Controllers.PassiveTagController, 109
 - DeletePassiveTag, 110
 - GetPassiveTag, 111
 - GetPassiveTags, 111
 - PassiveTagController, 110
 - PostPassiveTag, 111
 - PutPassiveTag, 112
- Grip.Api.Controllers.StudentController, 126
 - GetStudentDetails, 128
 - GetStudents, 128
 - StudentController, 127
- Grip.Api.Hubs, 12
- Grip.Api.Hubs.IStationClient, 95
 - ReceiveScan, 95
- Grip.Api.Hubs.StationHub, 123
 - SelectStation, 123
- Grip.Bll, 12
- Grip.Bll.DTO, 12
 - ActiveAttendanceDTO, 12
 - PassiveAttendanceDTO, 13
 - StudentDetailDTO, 13
- Grip.Bll.Exceptions, 13
- Grip.Bll.Exceptions.BadRequestException, 34
 - BadRequestException, 35
- Grip.Bll.Exceptions.DbConcurrencyException, 49
 - DbConcurrencyException, 50
- Grip.Bll.Exceptions.NotFoundException, 106
 - NotFoundException, 107
- Grip.Bll.Exceptions.UnauthorizedException, 132
 - UnauthorizedException, 133
- Grip.Bll.Providers, 14
- Grip.Bll.Providers.CurrentTimeProvider, 47
 - Now, 48
- Grip.Bll.Providers.ICurrentTimeProvider, 83
 - Now, 84
- Grip.Bll.Services, 14
- Grip.Bll.Services.ExemptService, 59
 - Create, 61
 - Delete, 62
 - ExemptService, 61
 - Get, 62
 - GetAll, 63
 - GetAllForUser, 63
 - GetByUser, 63
 - Update, 64
- Grip.Bll.Services.GroupService, 71
 - AddUserToGroup, 72
 - Create, 73
 - Delete, 73
 - Get, 75
 - GetAll, 75
 - GetUsersInGroup, 76
 - GroupService, 72
 - RemoveUserFromGroup, 76
 - Update, 77
- Grip.Bll.Services.Interfaces, 14
- Grip.Bll.Services.Interfaces.IEmailService, 86
 - SendEmailAsync, 86, 87
- Grip.Bll.Services.Interfaces.IRestInterface< Create-
eDTO, UpdatedDTO, GetDTO >, 93
 - Create, 93
 - Delete, 94
 - Get, 94
 - GetAll, 94
 - Update, 94
- Grip.Bll.Services.Interfaces.IStationService, 96
 - GetSecretKey, 96
- Grip.Bll.Services.Interfaces.IStudentService, 98
 - GetStudentDetailsAsync, 99
 - SearchStudentsAsync, 99
- Grip.Bll.Services.Interfaces.IUserService, 100
 - AddRole, 101
 - ConfirmEmail, 101
 - ForgotPassword, 102
 - Login, 102
 - RemoveRole, 103
 - ResetPassword, 103
- Grip.Bll.Services.PassiveTagService, 112
 - Create, 114
 - Delete, 115
 - Get, 115
 - GetAll, 116
 - PassiveTagService, 114
 - Update, 116
- Grip.Bll.Services.StationService, 124
 - GetSecretKey, 125
 - StationService, 125
- Grip.Bll.Services.StudentService, 129
 - GetStudentDetailsAsync, 130
 - SearchStudentsAsync, 131
 - StudentService, 130
- Grip.Bll.Services.UserService, 143
 - AddRole, 145
 - ConfirmEmail, 146
 - Create, 146
 - Delete, 147
 - ForgotPassword, 147
 - Get, 148
 - GetAll, 148
 - Login, 149
 - RemoveRole, 149
 - ResetPassword, 150
 - Update, 150
 - UserService, 145

- Grip.DAL, 15
- Grip.DAL.MappingProfile, 104
 - MappingProfile, 104
- Grip.DAL.Migrations, 15
- Grip.DAL.Migrations.AddClassesAndAttendance, 17
 - BuildTargetModel, 18
 - Down, 18
 - Up, 18
- Grip.DAL.Migrations.AddExempts, 19
 - BuildTargetModel, 19
 - Down, 20
 - Up, 20
- Grip.DAL.Migrations.ApplicationDbContextModelSnapshot, GroupDTO.cs
 - 26
 - BuildModel, 26
- Grip.DAL.Migrations.ClassStationRelationAdded, 46
 - BuildTargetModel, 46
 - Down, 47
 - Up, 47
- Grip.DAL.Migrations.Identity, 84
 - BuildTargetModel, 85
 - Down, 85
 - Up, 85
- Grip.DAL.Model, 15
- Grip.DAL.Model.Attendance, 27
 - Id, 27
 - Station, 27
 - Time, 28
 - User, 28
- Grip.DAL.Model.Exempt, 54
 - Id, 54
 - IssuedBy, 54
 - IssuedById, 54
 - IssuedTo, 55
 - IssuedTold, 55
 - ValidFrom, 55
 - ValidTo, 55
- Grip.DAL.Model.Group, 64
 - Classes, 65
 - Id, 65
 - Name, 65
 - Users, 65
- Grip.DAL.Model.PassiveTag, 107
 - Id, 108
 - SerialNumber, 108
 - User, 108
- Grip.DAL.Model.Station, 119
 - Attendances, 119
 - Classes, 119
 - Id, 120
 - Name, 120
 - SecretKey, 120
 - StationNumber, 120
- Grip.DAL.Model.User, 133
 - Attendances, 134
 - Exemptions, 134
 - Groups, 135
 - IssuedExemptions, 135
 - PassiveTags, 135
- Grip.Utilis, 15
- GroupController, 66
 - AddUserToGroup, 67
 - DeleteGroup, 68
 - GetGroup, 68
 - GroupController, 67
 - PostGroup, 69
 - PutGroup, 69
 - RemoveUserFromGroup, 70
- GroupDTO
 - GroupDTO.cs, 159
 - GroupDTO, 159
- Groups
 - ApplicationDbContext, 25
 - Grip.DAL.Model.User, 135
- GroupService
 - Grip.Bll.Services.GroupService, 72
- HMACTokenProvider, 78
 - GenerateToken, 79
 - ValidateToken, 79
- IAttendanceService, 80
 - GetAttendanceForDay, 80
 - VerifyPassiveScan, 81
 - VerifyPhoneScan, 81
- IClassService, 81
 - GetClassesForUserOnDay, 83
- Id
 - Grip.DAL.Model.Attendance, 27
 - Grip.DAL.Model.Exempt, 54
 - Grip.DAL.Model.Group, 65
 - Grip.DAL.Model.PassiveTag, 108
 - Grip.DAL.Model.Station, 120
- IExemptService, 88
 - Create, 89
- IGroupService, 89
 - AddUserToGroup, 90
 - GetUsersInGroup, 91
 - RemoveUserFromGroup, 91
- InvokeAsync
 - ApiKeyValidationMiddleware, 21
 - DeChunkingMiddleware, 51
- IPassiveTagService, 92
- IssuedBy
 - Grip.DAL.Model.Exempt, 54
- IssuedById
 - Grip.DAL.Model.Exempt, 54
- IssuedExemptions
 - Grip.DAL.Model.User, 135
- IssuedTo
 - Grip.DAL.Model.Exempt, 55
- IssuedTold
 - Grip.DAL.Model.Exempt, 55
- IStationTokenProvider, 97
 - GenerateToken, 97
 - ValidateToken, 98

- Login
 - Grip.Bll.Services.Interfaces.IUserService, 102
 - Grip.Bll.Services.UserService, 149
 - UserController, 141
- LoginResultDTO
 - LoginResultDTO.cs, 165
- LoginResultDTO.cs
 - LoginResultDTO, 165
- LoginUserDTO
 - LoginUserDTO.cs, 165
- LoginUserDTO.cs
 - LoginUserDTO, 165
- MappingProfile
 - Grip.DAL.MappingProfile, 104
- Name
 - Grip.DAL.Model.Group, 65
 - Grip.DAL.Model.Station, 120
- NotChunked, 105
- NotFoundException
 - Grip.Bll.Exceptions.NotFoundException, 107
- Now
 - Grip.Bll.Providers.CurrentTimeProvider, 48
 - Grip.Bll.Providers.ICurrentTimeProvider, 84
- OnConfiguring
 - ApplicationDbContext, 23
- OnModelCreating
 - ApplicationDbContext, 24
- PassiveAttendanceDTO
 - Grip.Bll.DTO, 13
- PassiveAuthentication
 - AttendanceController, 30
- PassiveTagController
 - Grip.Api.Controllers.PassiveTagController, 110
- PassiveTagDTO
 - PassiveTagDTO.cs, 160
- PassiveTagDTO.cs
 - PassiveTagDTO, 160
- PassiveTags
 - ApplicationDbContext, 25
 - Grip.DAL.Model.User, 135
- PassiveTagService
 - Grip.Bll.Services.PassiveTagService, 114
- PostClass
 - Grip.Api.Controllers.ClassController, 39
- PostExempt
 - Grip.Api.Controllers.ExemptController, 58
- PostGroup
 - GroupController, 69
- PostPassiveTag
 - Grip.Api.Controllers.PassiveTagController, 111
- PutClass
 - Grip.Api.Controllers.ClassController, 39
- PutGroup
 - GroupController, 69
- PutPassiveTag
 - Grip.Api.Controllers.PassiveTagController, 112
- ReceiveScan
 - Grip.Api.Hubs.IStationClient, 95
- Register
 - UserController, 141
- RegisterUserDTO
 - RegisterUserDTO.cs, 166
- RegisterUserDTO.cs
 - RegisterUserDTO, 166
- RemoveRole
 - Grip.Bll.Services.Interfaces.IUserService, 103
 - Grip.Bll.Services.UserService, 149
 - UserController, 142
- RemoveUserFromGroup
 - Grip.Bll.Services.GroupService, 76
 - GroupController, 70
 - IGroupService, 91
- ResetPassword
 - Grip.Bll.Services.Interfaces.IUserService, 103
 - Grip.Bll.Services.UserService, 150
 - UserController, 142
- ResetPasswordDTO
 - ResetPasswordDTO.cs, 167
- ResetPasswordDTO.cs
 - ResetPasswordDTO, 167
- Role, 117
 - Admin, 118
 - Doorman, 118
 - Student, 118
 - Teacher, 118
- SearchStudentsAsync
 - Grip.Bll.Services.Interfaces.IStudentService, 99
 - Grip.Bll.Services.StudentService, 131
- SecretKey
 - Grip.DAL.Model.Station, 120
- SelectStation
 - Grip.Api.Hubs.StationHub, 123
- SendEmailAsync
 - EmailService, 53
 - Grip.Bll.Services.Interfaces.IEmailService, 86, 87
- SerialNumber
 - Grip.DAL.Model.PassiveTag, 108
- Server/Api/Controllers/AttendanceController.cs, 153
- Server/Api/Controllers/ClassController.cs, 153
- Server/Api/Controllers/ExemptController.cs, 153
- Server/Api/Controllers/GroupController.cs, 154
- Server/Api/Controllers/PassiveTagController.cs, 154
- Server/Api/Controllers/StationController.cs, 154
- Server/Api/Controllers/StudentController.cs, 154
- Server/Api/Controllers/UserController.cs, 155
- Server/Api/Hubs/IStationClient.cs, 155
- Server/Api/Hubs/StationHub.cs, 155
- Server/Api/Middleware/ApiKeyValidationMiddleware.cs, 155
- Server/Api/Middleware/DeChunkingMiddleware.cs, 156
- Server/Api/Middleware/NotChunked.cs, 156
- Server/Api/Middleware/ValidateApiKey.cs, 156

- Server/Bll/DTO/Attendance/ActiveAttendanceDTO.cs, 156
- Server/Bll/DTO/Attendance/AttendanceDTO.cs, 156
- Server/Bll/DTO/Attendance/PassiveAttendanceDTO.cs, 157
- Server/Bll/DTO/Class/ClassDTO.cs, 157
- Server/Bll/DTO/Class/CreateClassDTO.cs, 158
- Server/Bll/DTO/Exempt/CreateExemptDTO.cs, 158
- Server/Bll/DTO/Exempt/ExemptDTO.cs, 159
- Server/Bll/DTO/Group/GroupDTO.cs, 159
- Server/Bll/DTO/PassiveTag/CreatePassiveTagDTO.cs, 159
- Server/Bll/DTO/PassiveTag/PassiveTagDTO.cs, 160
- Server/Bll/DTO/PassiveTag/UpdatePassiveTagDTO.cs, 161
- Server/Bll/DTO/Station/StationScanDTO.cs, 161
- Server/Bll/DTO/Station/StationSecretKeyDTO.cs, 162
- Server/Bll/DTO/Student/AbsenceDTO.cs, 162
- Server/Bll/DTO/Student/StudentAttendanceDTO.cs, 163
- Server/Bll/DTO/Student/StudentDetailDTO.cs, 163
- Server/Bll/DTO/User/ConfirmEmailDTO.cs, 163
- Server/Bll/DTO/User/ForgotPasswordDTO.cs, 164
- Server/Bll/DTO/User/LoginResultDTO.cs, 165
- Server/Bll/DTO/User/LoginUserDTO.cs, 165
- Server/Bll/DTO/User/RegisterUserDTO.cs, 166
- Server/Bll/DTO/User/ResetPasswordDTO.cs, 166
- Server/Bll/DTO/User/UserDTO.cs, 167
- Server/Bll/DTO/User/UserInfoDTO.cs, 168
- Server/Bll/Exceptions/BadRequestException.cs, 169
- Server/Bll/Exceptions/DbConcurrencyException.cs, 169
- Server/Bll/Exceptions/NotFoundException.cs, 169
- Server/Bll/Exceptions/UnauthorizedException.cs, 170
- Server/Bll/Providers/CurrentTimeProvider.cs, 170
- Server/Bll/Providers/HMACTokenProvider.cs, 170
- Server/Bll/Providers/ICurrentTimeProvider.cs, 170
- Server/Bll/Providers/IStationTokenProvider.cs, 171
- Server/Bll/Services/AttendanceService.cs, 171
- Server/Bll/Services/ClassService.cs, 171
- Server/Bll/Services/EmailService.cs, 171
- Server/Bll/Services/ExemptService.cs, 171
- Server/Bll/Services/GroupService.cs, 172
- Server/Bll/Services/Interfaces/IAttendanceService.cs, 172
- Server/Bll/Services/Interfaces/IClassService.cs, 172
- Server/Bll/Services/Interfaces/IEmailService.cs, 172
- Server/Bll/Services/Interfaces/IExemptService.cs, 173
- Server/Bll/Services/Interfaces/IGroupService.cs, 173
- Server/Bll/Services/Interfaces/IPassiveTagService.cs, 173
- Server/Bll/Services/Interfaces/IRestInterface.cs, 173
- Server/Bll/Services/Interfaces/IStationService.cs, 173
- Server/Bll/Services/Interfaces/IStudentService.cs, 174
- Server/Bll/Services/Interfaces/IUserService.cs, 174
- Server/Bll/Services/PassiveTagService.cs, 174
- Server/Bll/Services/StationService.cs, 175
- Server/Bll/Services/StudentService.cs, 175
- Server/Bll/Services/UserService.cs, 175
- Server/Bll/Utils/Consts.cs, 176
- Server/DAL/ApplicationDbContext.cs, 176
- Server/DAL/MappingProfile.cs, 176
- Server/DAL/Migrations/20230317172813_Identity.cs, 176
- Server/DAL/Migrations/20230317172813_Identity.Designer.cs, 177
- Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.cs, 177
- Server/DAL/Migrations/20230411164235_AddClassesAndAttendance.Des, 177
- Server/DAL/Migrations/20230501161544_AddExempts.cs, 177
- Server/DAL/Migrations/20230501161544_AddExempts.Designer.cs, 178
- Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.cs, 178
- Server/DAL/Migrations/20230511180151_ClassStationRelationAdded.Des, 178
- Server/DAL/Migrations/ApplicationDbContextModelSnapshot.cs, 179
- Server/DAL/Model/Attendance.cs, 179
- Server/DAL/Model/Class.cs, 179
- Server/DAL/Model/Exempt.cs, 179
- Server/DAL/Model/Group.cs, 180
- Server/DAL/Model/PassiveTag.cs, 180
- Server/DAL/Model/Role.cs, 180
- Server/DAL/Model/Station.cs, 180
- Server/DAL/Model/User.cs, 181
- Station
 - Grip.DAL.Model.Attendance, 27
- StationController, 121
 - GetKey, 122
 - StationController, 122
- StationNumber
 - Grip.DAL.Model.Station, 120
- Stations
 - ApplicationDbContext, 25
- StationSecretKeyDTO
 - StationSecretKeyDTO.cs, 162
- StationSecretKeyDTO.cs
 - StationSecretKeyDTO, 162
- StationService
 - Grip.Bll.Services.StationService, 125
- Student
 - Role, 118
- StudentAttendanceDTO
 - StudentAttendanceDTO.cs, 163
- StudentAttendanceDTO.cs
 - StudentAttendanceDTO, 163
- StudentController
 - Grip.Api.Controllers.StudentController, 127
- StudentDetailDTO
 - Grip.Bll.DTO, 13
- StudentService
 - Grip.Bll.Services.StudentService, 130
- Teacher
 - Role, 118
- Time

- Grip.DAL.Model.Attendance, [28](#)
- UnauthorizedException
 - Grip.Bll.Exceptions.UnauthorizedException, [133](#)
- Up
 - Grip.DAL.Migrations.AddClassesAndAttendance, [18](#)
 - Grip.DAL.Migrations.AddExempts, [20](#)
 - Grip.DAL.Migrations.ClassStationRelationAdded, [47](#)
 - Grip.DAL.Migrations.Identity, [85](#)
- Update
 - ClassService, [45](#)
 - Grip.Bll.Services.ExemptService, [64](#)
 - Grip.Bll.Services.GroupService, [77](#)
 - Grip.Bll.Services.Interfaces.IRestInterface< CreateDTO, UpdateDTO, GetDTO >, [94](#)
 - Grip.Bll.Services.PassiveTagService, [116](#)
 - Grip.Bll.Services.UserService, [150](#)
 - UserController, [143](#)
- UpdatePassiveTagDTO
 - UpdatePassiveTagDTO.cs, [161](#)
- UpdatePassiveTagDTO.cs
 - UpdatePassiveTagDTO, [161](#)
- User
 - Grip.DAL.Model.Attendance, [28](#)
 - Grip.DAL.Model.PassiveTag, [108](#)
- UserController, [136](#)
 - AddRole, [138](#)
 - ConfirmEmail, [138](#)
 - Delete, [138](#)
 - ForgotPassword, [140](#)
 - Get, [140](#)
 - Login, [141](#)
 - Register, [141](#)
 - RemoveRole, [142](#)
 - ResetPassword, [142](#)
 - Update, [143](#)
 - UserController, [137](#)
- UserDTO
 - UserDTO.cs, [167](#)
- UserDTO.cs
 - UserDTO, [167](#)
- UserInfoDTO
 - UserInfoDTO.cs, [168](#)
- UserInfoDTO.cs
 - UserInfoDTO, [168](#)
- Users
 - Grip.DAL.Model.Group, [65](#)
- UserService
 - Grip.Bll.Services.UserService, [145](#)
- ValidateApiKey, [151](#)
- ValidateToken
 - HMACTokenProvider, [79](#)
 - IStationTokenProvider, [98](#)
- ValidFrom
 - Grip.DAL.Model.Exempt, [55](#)
- ValidTo
 - Grip.DAL.Model.Exempt, [55](#)
- VerifyPassiveScan
 - AttendanceService, [33](#)
 - IAttendanceService, [81](#)
- VerifyPhoneScan
 - AttendanceService, [33](#)
 - IAttendanceService, [81](#)