Namespace AttendanceSystem.Api

Classes

[AuthenticationService](#)

[FunctionExecutorHostBuilderExtensions](#)

Extension methods to enable registration of the custom [IFunctionExecutor⧉](#) implementation generated for the current worker.

[Roles](#)

[Services](#)

[SwaggerSetup](#)

[WorkerHostBuilderFunctionMetadataProviderExtension](#)

Extension methods to enable registration of the custom [IFunctionMetadataProvider⧉](#) implementation generated for the current worker.

Class AuthenticationService

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

```
public class AuthenticationService
```

Inheritance

[object](#) ← AuthenticationService

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

AuthenticationService(ILogger<AuthenticationService>, JsonWebTokenHandler)

```
public AuthenticationService(ILogger<AuthenticationService> logger,
JsonWebTokenHandler jwtHandler)
```

Parameters

`logger` [ILogger](#)<[AuthenticationService](#)>

`jwtHandler` [JsonWebTokenHandler](#)

Methods

GetRoles(TokenValidationResult)

```
public List<string> GetRoles(TokenValidationResult jwt)
```

Parameters

jwt [TokenValidationResult↗]

Returns

[List↗]<[string↗]>

IsAuthenticated(TokenValidationResult, IEnumerable<string>)

```
public bool IsAuthenticated(TokenValidationResult jwt, IEnumerable<string>
requiredRoles)
```

Parameters

jwt [TokenValidationResult↗]

requiredRoles [IEnumerable↗]<[string↗]>

Returns

[bool↗]

Class FunctionExecutorHostBuilder Extensions

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

Extension methods to enable registration of the custom [IFunctionExecutor⧉](#) implementation generated for the current worker.

```
public static class FunctionExecutorHostBuilderExtensions
```

Inheritance

[object⧉](#) ← FunctionExecutorHostBuilderExtensions

Inherited Members

[object.Equals(object)⧉](#) , [object.Equals(object, object)⧉](#) , [object.GetHashCode()⧉](#) , [object.GetType()⧉](#) , [object.MemberwiseClone()⧉](#) , [object.ReferenceEquals(object, object)⧉](#) , [object.ToString()⧉](#)

Methods

ConfigureGeneratedFunctionExecutor(IHostBuilder)

Configures an optimized function executor to the invocation pipeline.

```
public static IHostBuilder ConfigureGeneratedFunctionExecutor(this
IHostBuilder builder)
```

Parameters

`builder` [IHostBuilder⧉](#)

Returns

[IHostBuilder⧉](#)

Class Roles

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

```
public static class Roles
```

Inheritance

[object](#) ← Roles

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Fields

Admin

```
public const string Admin = "role.admin"
```

Field Value

[string](#)

AllowAll

```
public static string[] AllowAll
```

Field Value

[string](#)[]

AllowElevated

```
public static string[] AllowElevated
```

Field Value

[string](#)[]

Student

```
public const string Student = "role.student"
```

Field Value

[string](#)

Teacher

```
public const string Teacher = "role.teacher"
```

Field Value

[string](#)

Class Services

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

```
public static class Services
```

Inheritance

[object](#) ← Services

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Methods

AddServices(IServiceCollection)

```
public static IServiceCollection AddServices(this IServiceCollection sp)
```

Parameters

sp [IServiceCollection](#)

Returns

[IServiceCollection](#)

Class SwaggerSetup

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

```
public static class SwaggerSetup
```

Inheritance

[object](#) ← SwaggerSetup

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Methods

AddSwagger(IServiceCollection)

```
public static IServiceCollection AddSwagger(this IServiceCollection services)
```

Parameters

services [IServiceCollection](#)

Returns

[IServiceCollection](#)

Class WorkerHostBuilderFunctionMetadata ProviderExtension

Namespace: [AttendanceSystem](#).[Api](#)

Assembly: AttendanceSystem.Api.dll

Extension methods to enable registration of the custom [IFunctionMetadataProvider⧉](#) implementation generated for the current worker.

```
public static class WorkerHostBuilderFunctionMetadataProviderExtension
```

Inheritance

[object⧉](#) ← WorkerHostBuilderFunctionMetadataProviderExtension

Inherited Members

[object.Equals(object)⧉](#) , [object.Equals(object, object)⧉](#) , [object.GetHashCode()⧉](#) , [object.GetType()⧉](#) , [object.MemberwiseClone()⧉](#) , [object.ReferenceEquals(object, object)⧉](#) , [object.ToString()⧉](#)

Methods

ConfigureGeneratedFunctionMetadataProvider(IHost Builder)

Adds the GeneratedFunctionMetadataProvider to the service collection. During initialization, the worker will return generated function metadata instead of relying on the Azure Functions host for function indexing.

```
public static IHostBuilder ConfigureGeneratedFunctionMetadataProvider(this
IHostBuilder builder)
```

Parameters

`builder` [IHostBuilder⧉](#)

Returns

[IHostBuilder](#)

Namespace AttendanceSystem.Api. Contracts

Classes

AttendanceRecordDto

CreateCourseContract

CreateSessionContract

CreateUserContract

EnrollUserContract

ExtendedAttendanceRecord

GetAttendanceListContract

SessionDTO

UpdateAttendanceContractItem

Enums

UserType

Class AttendanceRecordDto

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public record AttendanceRecordDto : IEquatable<AttendanceRecordDto>
```

Inheritance

[object](#) ← AttendanceRecordDto

Implements

[IEquatable](#)<[AttendanceRecordDto](#)>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Status

```
public AttendanceSubmission Status { get; set; }
```

Property Value

[AttendanceSubmission](#)

StudentId

```
public string StudentId { get; set; }
```

Property Value

[string](#)

StudentName

```
public string StudentName { get; set; }
```

Property Value

[string](#)

TeacherStatus

```
public AttendanceSubmission TeacherStatus { get; set; }
```

Property Value

[AttendanceSubmission](#)

Class CreateCourseContract

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class CreateCourseContract
```

Inheritance

[object](#) ← CreateCourseContract

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

DepartmentId

```
public required string DepartmentId { get; set; }
```

Property Value

[string](#)

Id

```
public required string Id { get; set; }
```

Property Value

[string](#)

Name

```csharp
public required string Name { get; set; }
```

Property Value

[string](#)⧉

TeacherIds

```csharp
public required List<string> TeacherIds { get; set; }
```

Property Value

[List](#)⧉<[string](#)⧉>

Class CreateSessionContract

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class CreateSessionContract
```

Inheritance

[object](#) ← CreateSessionContract

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

EndDate

```
public required DateTime EndDate { get; set; }
```

Property Value

[DateTime](#)

Participants

```
public required List<string> Participants { get; set; }
```

Property Value

[List](#)<[string](#)>

StartDate

```
public required DateTime StartDate { get; set; }
```

Property Value

[DateTime](⧉)

Class CreateUserContract

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class CreateUserContract
```

Inheritance

[object](#) ← CreateUserContract

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Email

```
public required string Email { get; set; }
```

Property Value

[string](#)

Id

```
public required string Id { get; set; }
```

Property Value

[string](#)

Name

```csharp
public required string Name { get; set; }
```

Property Value

[string](#)⧉

Type

```csharp
public required UserType Type { get; set; }
```

Property Value

[UserType](#)

Class EnrollUserContract

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class EnrollUserContract
```

Inheritance

[object](#) ← EnrollUserContract

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

UserId

```
public required string UserId { get; set; }
```

Property Value

[string](#)

Class ExtendedAttendanceRecord

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public record ExtendedAttendanceRecord : AttendanceRecord,
IEquatable<AttendanceRecord>, IEquatable<ExtendedAttendanceRecord>
```

Inheritance

[object](#) ← [AttendanceRecord](#) ← ExtendedAttendanceRecord

Implements

[IEquatable](#)<[AttendanceRecord](#)>, [IEquatable](#)<[ExtendedAttendanceRecord](#)>

Inherited Members

[AttendanceRecord.SessionId](#) , [AttendanceRecord.StudentId](#) ,
[AttendanceRecord.StudentSubmission](#) , [AttendanceRecord.TeacherSubmission](#) ,
[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

ExtendedAttendanceRecord()

```
public ExtendedAttendanceRecord()
```

ExtendedAttendanceRecord(AttendanceRecord)

```
public ExtendedAttendanceRecord(AttendanceRecord attendanceRecord)
```

Parameters

attendanceRecord [AttendanceRecord](#)

Properties

StudentName

```
public string StudentName { get; set; }
```

Property Value

[string](#)

Class GetAttendanceListContract

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class GetAttendanceListContract
```

Inheritance

[object](#) ← GetAttendanceListContract

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

Course

```
public required Course Course { get; set; }
```

Property Value

[Course](#)

EndTime

```
public required DateTime EndTime { get; set; }
```

Property Value

[DateTime](#)

Id

```
public Guid Id { get; init; }
```

Property Value

[Guid⌝](#)

Register

```
public required ICollection<ExtendedAttendanceRecord> Register { get; set; }
```

Property Value

[ICollection⌝](#)<[ExtendedAttendanceRecord](#)>

StartTime

```
public required DateTime StartTime { get; set; }
```

Property Value

[DateTime⌝](#)

Class SessionDTO

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public record SessionDTO : IEquatable<SessionDTO>
```

Inheritance

[object](#) ← SessionDTO

Implements

[IEquatable](#)<[SessionDTO](#)>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Attendance

```
public AttendanceRecordDto? Attendance { get; set; }
```

Property Value

[AttendanceRecordDto](#)

CourseName

```
public required string CourseName { get; set; }
```

Property Value

[string](#)

EndDate

```
public required DateTime EndDate { get; set; }
```

Property Value

[DateTime](#)⤢

Id

```
public required Guid Id { get; set; }
```

Property Value

[Guid](#)⤢

StartDate

```
public required DateTime StartDate { get; set; }
```

Property Value

[DateTime](#)⤢

Class UpdateAttendanceContractItem

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
public class UpdateAttendanceContractItem
```

Inheritance

[object](#) ← UpdateAttendanceContractItem

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

Kind

```
[JsonConverter(typeof(JsonStringEnumConverter<AttendanceKind>))]
public required AttendanceKind Kind { get; init; }
```

Property Value

[AttendanceKind](#)

UserId

```
public string? UserId { get; init; }
```

Property Value

[string](#)

Enum UserType

Namespace: [AttendanceSystem](#).[Api](#).[Contracts](#)

Assembly: AttendanceSystem.Api.dll

```
[JsonConverter(typeof(JsonStringEnumConverter<UserType>))]
public enum UserType
```

Fields

Administrator = 2

Student = 0

Teacher = 1

Namespace AttendanceSystem.Api. Controllers

Classes

[BaseController](#)

[CoursesController](#)

[FeedController](#)

[SessionsController](#)

[SwaggerController](#)
   Represents the controller that serves the swagger documents

[UsersController](#)

Class BaseController

Namespace: AttendanceSystem.Api.Controllers

Assembly: AttendanceSystem.Api.dll

```
public class BaseController
```

Inheritance

object ← BaseController

Derived

CoursesController, FeedController, SessionsController, UsersController

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() ,
object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) ,
object.ToString()

Constructors

BaseController(AuthenticationService, UserService)

```
public BaseController(AuthenticationService authenticationService,
UserService userService)
```

Parameters

authenticationService AuthenticationService

userService UserService

Fields

_userService

```
protected readonly UserService _userService
```

Field Value

UserService

Methods

AssertAuthentication(FunctionContext, IEnumerable<string>)

```
protected Task AssertAuthentication(FunctionContext context,
IEnumerable<string> allowedRoles)
```

Parameters

`context` [FunctionContext⧉](FunctionContext)

`allowedRoles` [IEnumerable⧉](IEnumerable)<[string⧉](string)>

Returns

[Task⧉](Task)

GetUser(FunctionContext)

```
protected Task<User> GetUser(FunctionContext context)
```

Parameters

`context` [FunctionContext⧉](FunctionContext)

Returns

[Task⧉](Task)<User>

GetUserId(FunctionContext)

```
protected string GetUserId(FunctionContext context)
```

Parameters

context FunctionContext↗

Returns

string↗

GetUserRoles(FunctionContext)

```
protected List<string> GetUserRoles(FunctionContext context)
```

Parameters

context FunctionContext↗

Returns

List↗<string↗>

Class CoursesController

Namespace: AttendanceSystem.Api.Controllers

Assembly: AttendanceSystem.Api.dll

```
public class CoursesController : BaseController
```

Inheritance

object ⮐ ← BaseController ← CoursesController

Inherited Members

BaseController._userService , BaseController.GetUserId(FunctionContext) ,
BaseController.GetUserRoles(FunctionContext) , BaseController.GetUser(FunctionContext) ,
BaseController.AssertAuthentication(FunctionContext, IEnumerable<string>) ,
object.Equals(object)⮐ , object.Equals(object, object)⮐ , object.GetHashCode()⮐ ,
object.GetType()⮐ , object.MemberwiseClone()⮐ , object.ReferenceEquals(object, object)⮐ ,
object.ToString()⮐

Constructors

CoursesController(ILogger<CoursesController>, CourseService, AuthenticationService, UserService)

```
public CoursesController(ILogger<CoursesController> logger, CourseService
courseService, AuthenticationService authenticationService, UserService userService)
```

Parameters

`logger` ILogger⮐ <CoursesController>

`courseService` CourseService

`authenticationService` AuthenticationService

`userService` UserService

Methods

ConfigureCourse(HttpRequest, FunctionContext, string, CourseAlteration)

```
[Function("CoursesController-ConfigureCourse")]
public Task<IActionResult> ConfigureCourse(HttpRequest req, FunctionContext ctx,
string courseId, CourseAlteration alteration)
```

Parameters

req HttpRequest⊠

ctx FunctionContext⊠

courseId string⊠

alteration CourseAlteration

Returns

Task⊠<IActionResult⊠>

CreateNewCourse(HttpRequest, FunctionContext, CreateCourseContract)

```
[Function("CoursesController-CreateNewCourse")]
public Task<IActionResult> CreateNewCourse(HttpRequest req, FunctionContext ctx,
CreateCourseContract contract)
```

Parameters

req HttpRequest⊠

ctx FunctionContext⊠

contract CreateCourseContract

Returns

Task⬈<IActionResult⬈>

DeleteCourse(HttpRequest, FunctionContext, string)

```
[Function("CoursesController-DeleteCourse")]
public Task<IActionResult> DeleteCourse(HttpRequest req, FunctionContext ctx,
string courseId)
```

Parameters

req HttpRequest⬈

ctx FunctionContext⬈

courseId string⬈

Returns

Task⬈<IActionResult⬈>

EnrollUser(HttpRequest, FunctionContext, string, EnrollUserContract)

```
[Function("CoursesController-EnrollUser")]
public Task<IActionResult> EnrollUser(HttpRequest req, FunctionContext ctx, string
courseId, EnrollUserContract contract)
```

Parameters

req HttpRequest⬈

ctx FunctionContext⬈

courseId string⬈

contract EnrollUserContract

Returns

Task⧉<IActionResult⧉>

GetAllCourses(HttpRequest, FunctionContext)

```
[Function("CoursesController-GetAllCourses")]
public Task<IActionResult> GetAllCourses(HttpRequest req, FunctionContext ctx)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

Returns

Task⧉<IActionResult⧉>

GetCourse(HttpRequest, FunctionContext, string)

```
[Function("CoursesController-GetCourse")]
public Task<IActionResult> GetCourse(HttpRequest req, FunctionContext ctx,
string courseId)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

courseId string⧉

Returns

Task⧉<IActionResult⧉>

Class FeedController

Namespace: AttendanceSystem.Api.Controllers

Assembly: AttendanceSystem.Api.dll

```
public class FeedController : BaseController
```

Inheritance

object ← BaseController ← FeedController

Inherited Members

BaseController._userService , BaseController.GetUserId(FunctionContext) ,
BaseController.GetUserRoles(FunctionContext) , BaseController.GetUser(FunctionContext) ,
BaseController.AssertAuthentication(FunctionContext, IEnumerable<string>) ,
object.Equals(object) , object.Equals(object, object) , object.GetHashCode() ,
object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) ,
object.ToString()

Constructors

FeedController(ILogger<SessionsController>, AttendanceService, AuthenticationService, UserService)

```
public FeedController(ILogger<SessionsController> logger, AttendanceService
attendanceService, AuthenticationService authenticationService,
UserService userService)
```

Parameters

`logger` ILogger<SessionsController>

`attendanceService` AttendanceService

`authenticationService` AuthenticationService

`userService` UserService

Methods

GetUpcomingSessions(HttpRequest, FunctionContext)

Gets the upcoming session for the logged-in user

```
[Function("FeedController-GetUpcomingSessions")]
[ProducesResponseType<List<SessionDTO>>(200)]
public Task<IActionResult> GetUpcomingSessions(HttpRequest req, FunctionContext ctx)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

Returns

Task⧉<IActionResult⧉>

The upcoming session which the user needs to attend.

Class SessionsController

Namespace: [AttendanceSystem](#).[Api](#).[Controllers](#)

Assembly: AttendanceSystem.Api.dll

```
public class SessionsController : BaseController
```

Inheritance

[object](#) ← [BaseController](#) ← SessionsController

Inherited Members

[BaseController._userService](#) , [BaseController.GetUserId(FunctionContext)](#) , [BaseController.GetUserRoles(FunctionContext)](#) , [BaseController.GetUser(FunctionContext)](#) , [BaseController.AssertAuthentication(FunctionContext, IEnumerable<string>)](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

SessionsController(ILogger<SessionsController>, AttendanceService, AuthenticationService, UserService, CourseService)

```
public SessionsController(ILogger<SessionsController> logger, AttendanceService attendanceService, AuthenticationService authenticationService, UserService userService, CourseService courseService)
```

Parameters

`logger` [ILogger](#)<[SessionsController](#)>

`attendanceService` [AttendanceService](#)

`authenticationService` [AuthenticationService](#)

`userService` [UserService](#)

courseService CourseService

Methods

CreateNewSession(HttpRequest, FunctionContext, string, CreateSessionContract)

```
[Function("SessionsController-CreateNewSession")]
public Task<IActionResult> CreateNewSession(HttpRequest req, FunctionContext ctx,
string courseId, CreateSessionContract contract)
```

Parameters

req HttpRequest⬀

ctx FunctionContext⬀

courseId string⬀

contract CreateSessionContract

Returns

Task⬀<IActionResult⬀>

DeleteSession(HttpRequest, FunctionContext, string, Guid)

```
[Function("SessionsController-DeleteSession")]
public Task<IActionResult> DeleteSession(HttpRequest req, FunctionContext ctx,
string courseId, Guid sessionId)
```

Parameters

req HttpRequest⬀

ctx FunctionContext⬀

courseId string↗

sessionId Guid↗

Returns

Task↗<IActionResult↗>

GetAllSessions(HttpRequest, FunctionContext, string)

```
[Function("SessionsController-GetAllSessions")]
public Task<IActionResult> GetAllSessions(HttpRequest req, FunctionContext ctx,
string courseId)
```

Parameters

req HttpRequest↗

ctx FunctionContext↗

courseId string↗

Returns

Task↗<IActionResult↗>

GetSessionInfo(HttpRequest, FunctionContext, string, Guid)

Gets the information for a specific session.

```
[Function("SessionsController-GetSessionInfo")]
[ProducesResponseType<GetAttendanceListContract>(200)]
public Task<IActionResult> GetSessionInfo(HttpRequest req, FunctionContext ctx,
string courseId, Guid sessionId)
```

Parameters

req HttpRequest↗

The http request of the function.

ctx [FunctionContext⧉](#)

The context of the functions.

courseId [string⧉](#)

The id of the course.

sessionId [Guid⧉](#)

The id of the session.

Returns

[Task⧉](#)<[IActionResult⧉](#)>

The session information.

Remarks

When the user is a teacher, the register is available. When the user is a student, only their own attendance is available.

SetAttendance(HttpRequest, FunctionContext, string, Guid, List<UpdateAttendanceContractItem>)

```
[Function("SessionsController-SetAttendance")]
public Task<IActionResult> SetAttendance(HttpRequest req, FunctionContext ctx,
string courseId, Guid sessionId, List<UpdateAttendanceContractItem> contract)
```

Parameters

req [HttpRequest⧉](#)

ctx [FunctionContext⧉](#)

courseId [string⧉](#)

sessionId [Guid⧉](#)

contract [List⧉](#)<[UpdateAttendanceContractItem](#)>

Returns

Task <IActionResult>

Class SwaggerController

Namespace: [AttendanceSystem](#).[Api](#).[Controllers](#)

Assembly: AttendanceSystem.Api.dll

Represents the controller that serves the swagger documents

```
public class SwaggerController
```

Inheritance

[object](#) ← SwaggerController

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

SwaggerController(ILogger<SwaggerController>, ISwashBuckleClient)

Represents the controller that serves the swagger documents

```
public SwaggerController(ILogger<SwaggerController> logger,
ISwashBuckleClient swashBuckleClient)
```

Parameters

`logger` [ILogger](#)<[SwaggerController](#)>

`swashBuckleClient` ISwashBuckleClient

Methods

SwaggerJson(HttpRequestData)

```
[SwaggerIgnore]
[Function("Swagger-Json")]
public Task<HttpResponseData> SwaggerJson(HttpRequestData req)
```

Parameters

req [HttpRequestData](#)⊠

Returns

[Task](#)⊠<[HttpResponseData](#)⊠>

SwaggerOAuth2Redirect(HttpRequestData)

This is only needed for OAuth2 client. This redirecting document is normally served as a static content. Functions don't provide this out of the box, so we serve it here. Don't forget to set OAuth2RedirectPath configuration option to reflect this route.

```
[SwaggerIgnore]
[Function("Swagger-OAuth2Redirect")]
public Task<HttpResponseData> SwaggerOAuth2Redirect(HttpRequestData req)
```

Parameters

req [HttpRequestData](#)⊠

The request given by the azure functions

Returns

[Task](#)⊠<[HttpResponseData](#)⊠>

An OAuth2 Redirect Response

SwaggerUi(HttpRequestData)

```
[SwaggerIgnore]
[Function("Swagger-Ui")]
```

```
  public Task<HttpResponseData> SwaggerUi(HttpRequestData req)
```

Parameters

req [HttpRequestData](#)⬀

Returns

[Task](#)⬀<[HttpResponseData](#)⬀>

Class UsersController

Namespace: [AttendanceSystem](#).[Api](#).[Controllers](#)

Assembly: AttendanceSystem.Api.dll

```
public class UsersController : BaseController
```

Inheritance

[object](#) ← [BaseController](#) ← UsersController

Inherited Members

[BaseController._userService](#) , [BaseController.GetUserId(FunctionContext)](#) ,
[BaseController.GetUserRoles(FunctionContext)](#) , [BaseController.GetUser(FunctionContext)](#) ,
[BaseController.AssertAuthentication(FunctionContext, IEnumerable<string>)](#) ,
[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

UsersController(ILogger<UsersController>, UserService, AuthenticationService)

```
public UsersController(ILogger<UsersController> logger, UserService userService,
AuthenticationService authenticationService)
```

Parameters

`logger` [ILogger](#)<[UsersController](#)>

`userService` [UserService](#)

`authenticationService` [AuthenticationService](#)

Methods

ConfigureUser(HttpRequest, FunctionContext, string, UserAlteration)

```
[Function("UsersController-ConfigureUser")]
public Task<IActionResult> ConfigureUser(HttpRequest req, FunctionContext ctx,
string userId, UserAlteration alteration)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

userId string⧉

alteration UserAlteration

Returns

Task⧉<IActionResult⧉>

CreateUser(HttpRequest, FunctionContext, CreateUserContract)

```
[Function("UsersController-CreateUser")]
public Task<IActionResult> CreateUser(HttpRequest req, FunctionContext ctx,
CreateUserContract contract)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

contract CreateUserContract

Returns

Task⧉<IActionResult⧉>

DeleteUser(HttpRequest, FunctionContext, string)

```
[Function("UsersController-DeleteUser")]
public Task<IActionResult> DeleteUser(HttpRequest req, FunctionContext ctx,
string userId)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

userId string⧉

Returns

Task⧉<IActionResult⧉>

GetAllUsers(HttpRequest, FunctionContext)

```
[Function("UsersController-GetAllUsers")]
public Task<IActionResult> GetAllUsers(HttpRequest req, FunctionContext ctx)
```

Parameters

req HttpRequest⧉

ctx FunctionContext⧉

Returns

Task⧉<IActionResult⧉>

GetUser(HttpRequest, FunctionContext, string)

```
[Function("UsersController-GetUser")]
public Task<IActionResult> GetUser(HttpRequest req, FunctionContext ctx,
string userId)
```

Parameters

`req` [HttpRequest↗](#)

`ctx` [FunctionContext↗](#)

`userId` [string↗](#)

Returns

[Task↗](#)<[IActionResult↗](#)>

Namespace AttendanceSystem.Api. Middleware

Classes

[AuthenticationHandler](#)

[ExceptionToErrorCodeHandler](#)

Class AuthenticationHandler

Namespace: [AttendanceSystem](#).[Api](#).[Middleware](#)

Assembly: AttendanceSystem.Api.dll

```
public class AuthenticationHandler : IFunctionsWorkerMiddleware
```

Inheritance

[object](#) ← AuthenticationHandler

Implements

[IFunctionsWorkerMiddleware](#)

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

AuthenticationHandler(ILogger<AuthenticationHandler>, JsonWebTokenHandler)

```
public AuthenticationHandler(ILogger<AuthenticationHandler> logger,
JsonWebTokenHandler jwtHandler)
```

Parameters

`logger` [ILogger](#)<[AuthenticationHandler](#)>

`jwtHandler` [JsonWebTokenHandler](#)

Methods

Invoke(FunctionContext, FunctionExecutionDelegate)

Invokes the middleware.

```
public Task Invoke(FunctionContext context, FunctionExecutionDelegate next)
```

Parameters

`context` [FunctionContext](#)

The [FunctionContext](#) for the current invocation.

`next` [FunctionExecutionDelegate](#)

The next middleware in the pipeline.

Returns

[Task](#)

A [Task](#) that represents the asynchronous invocation.

Class ExceptionToErrorCodeHandler

Namespace: AttendanceSystem.Api.Middleware

Assembly: AttendanceSystem.Api.dll

```
public class ExceptionToErrorCodeHandler : IFunctionsWorkerMiddleware
```

Inheritance

object ← ExceptionToErrorCodeHandler

Implements

IFunctionsWorkerMiddleware

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() ,
object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) ,
object.ToString()

Constructors

ExceptionToErrorCodeHandler(ILogger<ExceptionToError CodeHandler>)

```
public ExceptionToErrorCodeHandler(ILogger<ExceptionToErrorCodeHandler> logger)
```

Parameters

logger ILogger<ExceptionToErrorCodeHandler>

Methods

Invoke(FunctionContext, FunctionExecutionDelegate)

Invokes the middleware.

```
public Task Invoke(FunctionContext context, FunctionExecutionDelegate next)
```

Parameters

`context` [FunctionContext](#)⧉

    The [FunctionContext](#)⧉ for the current invocation.

`next` [FunctionExecutionDelegate](#)⧉

    The next middleware in the pipeline.

Returns

[Task](#)⧉

    A [Task](#)⧉ that represents the asynchronous invocation.

Namespace AttendanceSystem.Data

Classes

CoursesContext

Class CoursesContext

Namespace: [AttendanceSystem](#).[Data](#)

Assembly: AttendanceSystem.Data.dll

```
public class CoursesContext : DbContext, IInfrastructure<IServiceProvider>,
IDbContextDependencies, IDbSetCache, IDbContextPoolable, IResettableService,
IDisposable, IAsyncDisposable
```

Inheritance

[object](#) ← [DbContext](#) ← CoursesContext

Implements

[IInfrastructure](#)<[IServiceProvider](#)>, [IDbContextDependencies](#), [IDbSetCache](#),
[IDbContextPoolable](#), [IResettableService](#), [IDisposable](#), [IAsyncDisposable](#)

Inherited Members

[DbContext.Set<TEntity>()](#) , [DbContext.Set<TEntity>(string)](#) ,
[DbContext.OnConfiguring(DbContextOptionsBuilder)](#) ,
[DbContext.ConfigureConventions(ModelConfigurationBuilder)](#) ,
[DbContext.SaveChanges()](#) , [DbContext.SaveChanges(bool)](#) ,
[DbContext.SaveChangesAsync(CancellationToken)](#) ,
[DbContext.SaveChangesAsync(bool, CancellationToken)](#) , [DbContext.Dispose()](#) ,
[DbContext.DisposeAsync()](#) , [DbContext.Entry<TEntity>(TEntity)](#) ,
[DbContext.Entry(object)](#) , [DbContext.Add<TEntity>(TEntity)](#) ,
[DbContext.AddAsync<TEntity>(TEntity, CancellationToken)](#) ,
[DbContext.Attach<TEntity>(TEntity)](#) , [DbContext.Update<TEntity>(TEntity)](#) ,
[DbContext.Remove<TEntity>(TEntity)](#) , [DbContext.Add(object)](#) ,
[DbContext.AddAsync(object, CancellationToken)](#) , [DbContext.Attach(object)](#) ,
[DbContext.Update(object)](#) , [DbContext.Remove(object)](#) ,
[DbContext.AddRange(params object[])](#) , [DbContext.AddRangeAsync(params object[])](#) ,
[DbContext.AttachRange(params object[])](#) , [DbContext.UpdateRange(params object[])](#) ,
[DbContext.RemoveRange(params object[])](#) ,
[DbContext.AddRange(IEnumerable<object>)](#) ,
[DbContext.AddRangeAsync(IEnumerable<object>, CancellationToken)](#) ,
[DbContext.AttachRange(IEnumerable<object>)](#) ,
[DbContext.UpdateRange(IEnumerable<object>)](#) ,
[DbContext.RemoveRange(IEnumerable<object>)](#) ,
[DbContext.Find(Type, params object[])](#) , [DbContext.FindAsync(Type, params object[])](#) ,

[DbContext.FindAsync(Type, object[], CancellationToken)](#) ,
[DbContext.Find<TEntity>(params object[])](#) ,
[DbContext.FindAsync<TEntity>(params object[])](#) ,
[DbContext.FindAsync<TEntity>(object[], CancellationToken)](#) ,
[DbContext.FromExpression<TResult>(Expression<Func<IQueryable<TResult>>>)](#) ,
[DbContext.Database](#) , [DbContext.ChangeTracker](#) , [DbContext.Model](#) ,
[DbContext.ContextId](#) , [DbContext.SavingChanges](#) , [DbContext.SavedChanges](#) ,
[DbContext.SaveChangesFailed](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) ,
[object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) ,
[object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

CoursesContext(DbContextOptions<CoursesContext>)

```
public CoursesContext(DbContextOptions<CoursesContext> options)
```

Parameters

options [DbContextOptions](#)<[CoursesContext](#)>

Properties

Courses

```
public DbSet<Course> Courses { get; set; }
```

Property Value

[DbSet](#)<[Course](#)>

Departments

```
public DbSet<Department> Departments { get; set; }
```

Property Value

[DbSet⧉](#)<[Department](#)>

Sessions

```
public DbSet<Session> Sessions { get; set; }
```

Property Value

[DbSet⧉](#)<[Session](#)>

Students

```
public DbSet<Student> Students { get; set; }
```

Property Value

[DbSet⧉](#)<[Student](#)>

Users

```
public DbSet<User> Users { get; set; }
```

Property Value

[DbSet⧉](#)<[User](#)>

Methods

OnModelCreating(ModelBuilder)

Override this method to further configure the model that was discovered by convention from the entity types exposed in [DbSet<TEntity>⧉](#) properties on your derived context. The

resulting model may be cached and re-used for subsequent instances of your derived context.

```
protected override void OnModelCreating(ModelBuilder modelBuilder)
```

Parameters

modelBuilder ModelBuilder⬚

The builder being used to construct the model for this context. Databases (and other extensions) typically define extension methods on this object that allow you to configure aspects of the model that are specific to a given database.

Remarks

If a model is explicitly set on the options for this context (via UseModel(IModel)⬚) then this method will not be run. However, it will still run when creating a compiled model.

See Modeling entity types and relationships⬚ for more information and examples.

Namespace AttendanceSystem.Data.Migrations

Classes

[Initial](#)

A base class inherited by each EF Core migration.

Class Initial

Namespace: AttendanceSystem.Data.Migrations

Assembly: AttendanceSystem.Data.dll

A base class inherited by each EF Core migration.

```
[DbContext(typeof(CoursesContext))]
[Migration("20250305112651_Initial")]
public class Initial : Migration
```

Inheritance

object ← Migration ← Initial

Inherited Members

Migration.InitialDatabase , Migration.TargetModel , Migration.UpOperations ,
Migration.DownOperations , Migration.ActiveProvider , object.Equals(object) ,
object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

See Database migrations for more information and examples.

Methods

BuildTargetModel(ModelBuilder)

Implemented to build the TargetModel.

```
protected override void BuildTargetModel(ModelBuilder modelBuilder)
```

Parameters

`modelBuilder` ModelBuilder

The ModelBuilder to use to build the model.

Remarks

See [Database migrations⧉](#) for more information and examples.

Down(MigrationBuilder)

Builds the operations that will migrate the database 'down'.

```
protected override void Down(MigrationBuilder migrationBuilder)
```

Parameters

migrationBuilder [MigrationBuilder⧉](#)

The [MigrationBuilder⧉](#) that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by this migration so that it returns to the state that it was in before this migration was applied.

This method must be overridden in each class that inherits from [Migration⧉](#) if both 'up' and 'down' migrations are to be supported. If it is not overridden, then calling it will throw and it will not be possible to migrate in the 'down' direction.

See [Database migrations⧉](#) for more information and examples.

Up(MigrationBuilder)

Builds the operations that will migrate the database 'up'.

```
protected override void Up(MigrationBuilder migrationBuilder)
```

Parameters

migrationBuilder [MigrationBuilder⧉](#)

The [MigrationBuilder⧉](#) that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by the previous migration so that it is up-to-date with regard to this migration.

This method must be overridden in each class that inherits from [Migration↗](#).

See [Database migrations↗](#) for more information and examples.

Namespace AttendanceSystem.Domain. Model

Classes

[Administrator](#)

[AttendanceRecord](#)

[AttendanceSubmission](#)

[Course](#)

[Department](#)

[Session](#)

[Student](#)

[Teacher](#)

[User](#)

Enums

[AttendanceKind](#)

Class Administrator

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public class Administrator : User
```

Inheritance

[object](#) ← [User](#) ← Administrator

Inherited Members

[User.Id](#) , [User.Name](#) , [User.Email](#) , [User.Courses](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Enum AttendanceKind

Namespace: AttendanceSystem.Domain.Model

Assembly: AttendanceSystem.Domain.Model.dll

```
[JsonConverter(typeof(JsonStringEnumConverter))]
public enum AttendanceKind
```

Fields

Absent = 2

Late = 3

LeftEarly = 5

Present = 1

Sick = 4

Unknown = 0

Class AttendanceRecord

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public record AttendanceRecord : IEquatable<AttendanceRecord>
```

Inheritance

[object](#) ← AttendanceRecord

Implements

[IEquatable](#)<[AttendanceRecord](#)>

Derived

[ExtendedAttendanceRecord](#)

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

SessionId

```
public required Guid SessionId { get; set; }
```

Property Value

[Guid](#)

StudentId

```
public required string StudentId { get; set; }
```

Property Value

[string↗](#)

StudentSubmission

```
public required AttendanceSubmission StudentSubmission { get; set; }
```

Property Value

[AttendanceSubmission](#)

TeacherSubmission

```
public required AttendanceSubmission TeacherSubmission { get; set; }
```

Property Value

[AttendanceSubmission](#)

Class AttendanceSubmission

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
[Owned]
public record AttendanceSubmission : IEquatable<AttendanceSubmission>
```

Inheritance

[object](#) ← AttendanceSubmission

Implements

[IEquatable](#)<[AttendanceSubmission](#)>

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

AttendanceSubmission()

```
public AttendanceSubmission()
```

Properties

Attendance

```
public required AttendanceKind Attendance { get; set; }
```

Property Value

[AttendanceKind](#)

IsSubmitted

```
[NotMapped]
public bool IsSubmitted { get; }
```

Property Value

[bool](⧉)

Submitted

```
public DateTime? Submitted { get; }
```

Property Value

[DateTime](⧉)?

Class Course

Namespace: [AttendanceSystem](.).[Domain](.).[Model]
Assembly: AttendanceSystem.Domain.Model.dll

```
public class Course
```

Inheritance

[object](#) ← Course

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

Course(string, string, string, List<string>)

```
public Course(string id, string name, string department, List<string> teachers)
```

Parameters

id [string](#)

name [string](#)

department [string](#)

teachers [List](#)<[string](#)>

Properties

Department

```
public string Department { get; }
```

Property Value

[string](#)

Id

```
public string Id { get; }
```

Property Value

[string](#)

Name

```
public string Name { get; }
```

Property Value

[string](#)

Sessions

```
public List<Session> Sessions { get; }
```

Property Value

[List](#)<[Session](#)>

Students

```
public List<string> Students { get; }
```

Property Value

[List](#) <[string](#) >

Teachers

```
public List<string> Teachers { get; }
```

Property Value

[List](#) <[string](#) >

Class Department

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```csharp
public class Department
```

Inheritance

[object](#) ← Department

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

Name

```csharp
public string Name { get; set; }
```

Property Value

[string](#)

Class Session

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public class Session
```

Inheritance

[object](#) ← Session

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

Session(Course, List<string>, DateTime, DateTime)

```
public Session(Course course, List<string> students, DateTime startTime,
DateTime endTime)
```

Parameters

course [Course](#)

students [List](#)<[string](#)>

startTime [DateTime](#)

endTime [DateTime](#)

Properties

Course

```
public required Course Course { get; set; }
```

Property Value

[Course](#)

EndTime

```
public required DateTime EndTime { get; set; }
```

Property Value

[DateTime⧉](#)

Id

```
public Guid Id { get; init; }
```

Property Value

[Guid⧉](#)

Register

```
public ICollection<AttendanceRecord> Register { get; }
```

Property Value

[ICollection⧉](#)<[AttendanceRecord](#)>

StartTime

```
public required DateTime StartTime { get; set; }
```

Property Value

[DateTime⤢](#)

Methods

SetStudentAttendance(string, AttendanceKind)

```
public void SetStudentAttendance(string studentId, AttendanceKind attendance)
```

Parameters

studentId [string⤢](#)

attendance [AttendanceKind](#)

SetTeacherApproval(string, AttendanceKind)

```
public void SetTeacherApproval(string studentId, AttendanceKind attendance)
```

Parameters

studentId [string⤢](#)

attendance [AttendanceKind](#)

Class Student

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public class Student : User
```

Inheritance

[object](#) ← [User](#) ← Student

Inherited Members

[User.Id](#) , [User.Name](#) , [User.Email](#) , [User.Courses](#) , [object.Equals(object)](#) ,
[object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Class Teacher

Namespace: <u>AttendanceSystem</u>.<u>Domain</u>.<u>Model</u>

Assembly: AttendanceSystem.Domain.Model.dll

```
public class Teacher : User
```

Inheritance

<u>object</u>⬓ ← <u>User</u> ← Teacher

Inherited Members

<u>User.Id</u> , <u>User.Name</u> , <u>User.Email</u> , <u>User.Courses</u> , <u>object.Equals(object)</u>⬓ ,
<u>object.Equals(object, object)</u>⬓ , <u>object.GetHashCode()</u>⬓ , <u>object.GetType()</u>⬓ ,
<u>object.MemberwiseClone()</u>⬓ , <u>object.ReferenceEquals(object, object)</u>⬓ , <u>object.ToString()</u>⬓

Class User

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public class User
```

Inheritance

[object](#) ← User

Derived

[Administrator](#), [Student](#), [Teacher](#)

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

Courses

```
public List<Course> Courses { get; set; }
```

Property Value

[List](#)<[Course](#)>

Email

```
public string Email { get; set; }
```

Property Value

[string](#)

Id

Gets or sets the internal ID managed by the external identity provider.

```
public string Id { get; init; }
```

Property Value

[string](#)

Remarks

We want to store this because otherwise, we cannot identify the user based on the information provided by the identity provider.

Name

```
public string Name { get; set; }
```

Property Value

[string](#)

Namespace AttendanceSystem.Domain. Model.Exceptions

Classes

[EntityNotFoundException](#)

Class EntityNotFoundException

Namespace: [AttendanceSystem](#).[Domain](#).[Model](#).[Exceptions](#)

Assembly: AttendanceSystem.Domain.Model.dll

```
public class EntityNotFoundException : Exception, ISerializable
```

Inheritance

[object](#) ← [Exception](#) ← EntityNotFoundException

Implements

[ISerializable](#)

Inherited Members

[Exception.GetBaseException()](#) , [Exception.GetType()](#) , [Exception.ToString()](#) ,
[Exception.Data](#) , [Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) ,
[Exception.Message](#) , [Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) ,
[Exception.SerializeObjectState](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) ,
[object.GetHashCode()](#) , [object.MemberwiseClone()](#) ,
[object.ReferenceEquals(object, object)](#)

Constructors

EntityNotFoundException(string)

```
public EntityNotFoundException(string message)
```

Parameters

`message` [string](#)

Namespace AttendanceSystem.Domain. Services

Classes

[AttendanceService](#)

[CourseService](#)

[UserService](#)

Class AttendanceService

Namespace: [AttendanceSystem](#).[Domain](#).[Services](#)

Assembly: AttendanceSystem.Domain.Services.dll

```
public class AttendanceService
```

Inheritance

[object](#) ← AttendanceService

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

AttendanceService(CoursesContext)

```
public AttendanceService(CoursesContext context)
```

Parameters

context [CoursesContext](#)

Methods

CreateSession(string, DateTime, DateTime, List<string>)

```
public Task<Session> CreateSession(string courseId, DateTime startdate, DateTime
enddate, List<string> students)
```

Parameters

courseId string⧉

startdate DateTime⧉

enddate DateTime⧉

students List⧉<string⧉>

Returns

Task⧉<Session>

DeleteSession(Guid)

```
public Task DeleteSession(Guid sessionId)
```

Parameters

sessionId Guid⧉

Returns

Task⧉

GetSession(Guid)

```
public Task<Session> GetSession(Guid sessionId)
```

Parameters

sessionId Guid⧉

Returns

Task⧉<Session>

GetSessionStatus(Guid, string)

```
public Task<AttendanceRecord?> GetSessionStatus(Guid sessionId, string userId)
```

Parameters

sessionId [Guid⌁](#)

userId [string⌁](#)

Returns

[Task⌁](#)<[AttendanceRecord](#)>

GetSessionStatuses(string, List<Guid>)

```
public Task<Dictionary<Guid, AttendanceRecord>> GetSessionStatuses(string userId,
List<Guid> toList)
```

Parameters

userId [string⌁](#)

toList [List⌁](#)<[Guid⌁](#)>

Returns

[Task⌁](#)<[Dictionary⌁](#)<[Guid⌁](#), [AttendanceRecord](#)>>

GetSessionWithRegister(Guid)

```
public Task<Session> GetSessionWithRegister(Guid sessionId)
```

Parameters

sessionId [Guid⌁](#)

Returns

Task☑<Session>

GetSessions(string)

```
public Task<List<Session>> GetSessions(string courseId)
```

Parameters

courseId string☑

Returns

Task☑<List☑<Session>>

GetUpcomingSessionsForUser(string)

```
public Task<List<Session>> GetUpcomingSessionsForUser(string userId)
```

Parameters

userId string☑

Returns

Task☑<List☑<Session>>

SetStudentAttendance(Guid, string, AttendanceKind)

```
public Task SetStudentAttendance(Guid sessionId, string studentId,
AttendanceKind kind)
```

Parameters

sessionId Guid☑

studentId string☑

kind AttendanceKind

Returns

Task⤢

SetTeacherApproval(Guid, string, AttendanceKind)

```
public Task SetTeacherApproval(Guid sessionId, string studentId,
AttendanceKind kind)
```

Parameters

sessionId Guid⤢

studentId string⤢

kind AttendanceKind

Returns

Task⤢

Class CourseService

Namespace: <u>AttendanceSystem</u>.<u>Domain</u>.<u>Services</u>

Assembly: AttendanceSystem.Domain.Services.dll

```
public class CourseService
```

Inheritance

<u>object</u> ← CourseService

Inherited Members

<u>object.Equals(object)</u> , <u>object.Equals(object, object)</u> , <u>object.GetHashCode()</u> ,
<u>object.GetType()</u> , <u>object.MemberwiseClone()</u> , <u>object.ReferenceEquals(object, object)</u> ,
<u>object.ToString()</u>

Constructors

CourseService(CoursesContext)

```
public CourseService(CoursesContext context)
```

Parameters

context <u>CoursesContext</u>

Methods

ConfigureCourse(string, CourseAlteration)

```
public Task<Course> ConfigureCourse(string courseId, CourseAlteration alteration)
```

Parameters

courseId <u>string</u>

alteration [CourseAlteration](#)

Returns

[Task](#)<[Course](#)>

CourseExists(string)

```
public Task<bool> CourseExists(string courseId)
```

Parameters

courseId [string](#)

Returns

[Task](#)<[bool](#)>

CreateNewCourse(string, string, string, List<string>)

```
public Task<Course> CreateNewCourse(string id, string name, string departmentId,
List<string> teacherIds)
```

Parameters

id [string](#)

name [string](#)

departmentId [string](#)

teacherIds [List](#)<[string](#)>

Returns

[Task](#)<[Course](#)>

DeleteCourse(string)

```
public Task DeleteCourse(string courseId)
```

Parameters

courseId [string](#)

Returns

[Task](#)

EnrollUser(string, string)

```
public Task EnrollUser(string courseId, string userId)
```

Parameters

courseId [string](#)

userId [string](#)

Returns

[Task](#)

GetAllCourses()

```
public Task<List<Course>> GetAllCourses()
```

Returns

[Task](#)<[List](#)<[Course](#)>>

GetAllCourses(string)

```
public Task<List<Course>> GetAllCourses(string userId)
```

Parameters

courseId **string**⧉

Returns

[Task]⧉<[List]⧉<[Course]>>

GetCourse(string)

```
public Task<Course> GetCourse(string courseId)
```

Parameters

courseId **string**⧉

Returns

[Task]⧉<[Course]>

UserCanAccessCourse(string, string)

```
public Task<bool> UserCanAccessCourse(string courseId, string userId)
```

Parameters

courseId **string**⧉

userId **string**⧉

Returns

[Task]⧉<[bool]⧉>

Class UserService

Namespace: [AttendanceSystem](#).[Domain](#).[Services](#)

Assembly: AttendanceSystem.Domain.Services.dll

```
public class UserService
```

Inheritance

[object](#) ← UserService

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

UserService(CoursesContext)

```
public UserService(CoursesContext context)
```

Parameters

context [CoursesContext](#)

Methods

CreateAdministrator(string, string, string)

```
public Task<User> CreateAdministrator(string id, string name, string email)
```

Parameters

id [string](#)

name string↗

email string↗

Returns

Task↗<User>

CreateStudent(string, string, string)

```
public Task<User> CreateStudent(string id, string name, string email)
```

Parameters

id string↗

name string↗

email string↗

Returns

Task↗<User>

CreateTeacher(string, string, string)

```
public Task<User> CreateTeacher(string id, string name, string email)
```

Parameters

id string↗

name string↗

email string↗

Returns

Task↗<User>

DeleteUser(string)

```
public Task DeleteUser(string userId)
```

Parameters

userId string⧉

Returns

Task⧉

EditUser(string, UserAlteration)

```
public Task<User> EditUser(string userId, UserAlteration alteration)
```

Parameters

userId string⧉

alteration UserAlteration

Returns

Task⧉<User>

GetAllStudents()

```
public Task<List<User>> GetAllStudents()
```

Returns

Task⧉<List⧉<User>>

GetUser(string)

```
public Task<User> GetUser(string userId)
```

Parameters

userId string↗

Returns

Task↗<User>

GetUsers(List<string>)

```
public Task<List<User>> GetUsers(List<string> users)
```

Parameters

users List↗<string↗>

Returns

Task↗<List↗<User>>

Namespace AttendanceSystem.Domain. Services.Alterations

Classes

CourseAlteration

UserAlteration

Class CourseAlteration

Namespace: [AttendanceSystem](#).[Domain](#).[Services](#).[Alterations](#)

Assembly: AttendanceSystem.Domain.Services.dll

```
public class CourseAlteration
```

Inheritance

[object](#) ← CourseAlteration

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Properties

Department

```
public string? Department { get; set; }
```

Property Value

[string](#)

Name

```
public string? Name { get; set; }
```

Property Value

[string](#)

Class UserAlteration

Namespace: [AttendanceSystem](#).[Domain](#).[Services](#).[Alterations](#)

Assembly: AttendanceSystem.Domain.Services.dll

```
public class UserAlteration
```

Inheritance

[object](#) ← UserAlteration

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Properties

Email

```
public string? Email { get; set; }
```

Property Value

[string](#)

Name

```
public string? Name { get; set; }
```

Property Value

[string](#)

Namespace AttendanceSystem.Domain. Services.Tools

Classes

MockDataGenerator

Class MockDataGenerator

Namespace: [AttendanceSystem](#).[Domain](#).[Services](#).[Tools](#)

Assembly: AttendanceSystem.Domain.Services.dll

```
public class MockDataGenerator
```

Inheritance

[object](#) ← MockDataGenerator

Inherited Members

[object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) ,
[object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) ,
[object.ToString()](#)

Constructors

MockDataGenerator(CoursesContext)

```
public MockDataGenerator(CoursesContext context)
```

Parameters

context [CoursesContext](#)

Methods

GenerateRealData()

```
public Task GenerateRealData()
```

Returns

[Task](#)