**OutOfOfficeApp**

ASP.NET Core REST Web API

**Github**: <https://github.com/ElizabethGorbunova/OutOfOfficeApp>

Yelyzaveta Horbunova

Contents

[**OutOfOfficeApp** 1](#_Toc171879748)

[1. Introduction 3](#_Toc171879749)

[2. Database diagram with description 4](#_Toc171879750)

[3. The script of creation of database tables 6](#_Toc171879751)

[4. Screenshots from MSSQL Server Management Studio 10](#_Toc171879752)

# Introduction

OutOfOfficeApp is Web API application, which can be eventually used in browser, mobile applications or each application where all communication between client and server is provided by HTTP protocol. Application allows to read, create or modify data from the server according to REST architecture. The database structure is configured in MSSQL Management Studio with a help of ORM - Entity Framework Core.

Additional application features and applied packages:

-database MSSQL based on C# classes (Microsoft.EntityFrameworkCore.SqlServer);

-validation of input models (FluentValidation.AspNetCore);

-automatic mapping for models in and out (AutoMapper.Extensions.Microsoft.DependencyInjection);

- using of built-in container for dependency injections;

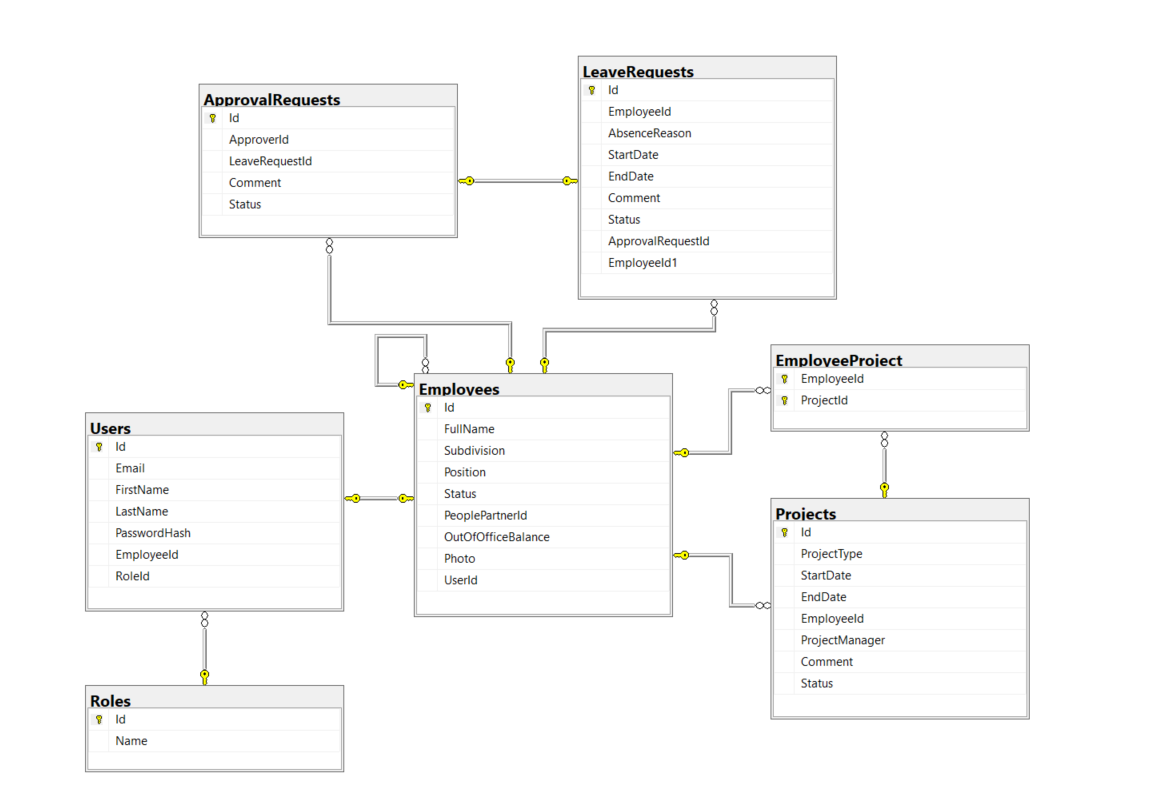
-authentication of users using JWT tokens (Microsoft.AspNetCore.Authentication.JwtBearer);

-mechanism of authorization for performing actions based on user roles;

- protect user passwords with hashes;

-implementation of custom middleware for catching and handling exceptions;

# Database diagram with description:



**Employees table:**

PK: Id

FK: PeoplePartnerId

FK: UserId

**ApprovalRequests table:**

PK: Id

FK: ApproverId

FK: LeaveRequestId

**LeaveRequests table:**

PK: Id

FK: EmployeeId

**Projects table:**

PK: Id

FK: EmployeeId

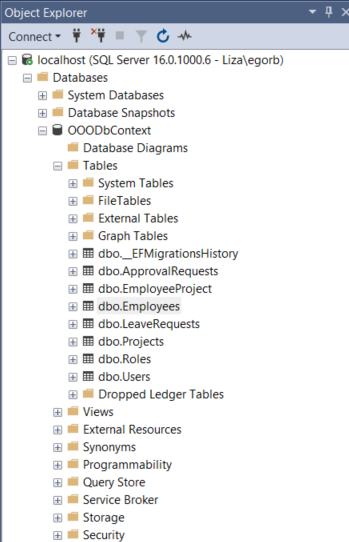
**Users table:**

PK: Id

FK: RoleId

**Role table:**

PK: Id



# The script of creation of database tables

protected override void Up(MigrationBuilder migrationBuilder)

{

migrationBuilder.CreateTable(

name: "Roles",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Name = table.Column<string>(type: "nvarchar(max)", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Roles", x => x.Id);

});

migrationBuilder.CreateTable(

name: "Users",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

Email = table.Column<string>(type: "nvarchar(max)", nullable: true),

FirstName = table.Column<string>(type: "nvarchar(max)", nullable: true),

LastName = table.Column<string>(type: "nvarchar(max)", nullable: true),

PasswordHash = table.Column<string>(type: "nvarchar(max)", nullable: true),

EmployeeId = table.Column<int>(type: "int", nullable: true),

RoleId = table.Column<int>(type: "int", nullable: true)

},

constraints: table =>

{

table.PrimaryKey("PK\_Users", x => x.Id);

table.ForeignKey(

name: "FK\_Users\_Roles\_RoleId",

column: x => x.RoleId,

principalTable: "Roles",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateTable(

name: "Employees",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

FullName = table.Column<string>(type: "nvarchar(max)", nullable: false),

Subdivision = table.Column<int>(type: "int", nullable: false),

Position = table.Column<int>(type: "int", nullable: false),

Status = table.Column<int>(type: "int", nullable: false),

PeoplePartnerId = table.Column<int>(type: "int", nullable: true),

OutOfOfficeBalance = table.Column<float>(type: "real", nullable: false),

Photo = table.Column<string>(type: "nvarchar(max)", nullable: true),

UserId = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Employees", x => x.Id);

table.ForeignKey(

name: "FK\_Employees\_Employees\_PeoplePartnerId",

column: x => x.PeoplePartnerId,

principalTable: "Employees",

principalColumn: "Id",

onDelete: ReferentialAction.Restrict);

table.ForeignKey(

name: "FK\_Employees\_Users\_UserId",

column: x => x.UserId,

principalTable: "Users",

principalColumn: "Id",

onDelete: ReferentialAction.Restrict);

});

migrationBuilder.CreateTable(

name: "LeaveRequests",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

EmployeeId = table.Column<int>(type: "int", nullable: false),

AbsenceReason = table.Column<int>(type: "int", nullable: false),

StartDate = table.Column<DateTime>(type: "datetime2", nullable: false),

EndDate = table.Column<DateTime>(type: "datetime2", nullable: false),

Comment = table.Column<string>(type: "nvarchar(max)", nullable: true),

Status = table.Column<int>(type: "int", nullable: false),

ApprovalRequestId = table.Column<int>(type: "int", nullable: true),

EmployeeId1 = table.Column<int>(type: "int", nullable: true)

},

constraints: table =>

{

table.PrimaryKey("PK\_LeaveRequests", x => x.Id);

table.ForeignKey(

name: "FK\_LeaveRequests\_Employees\_EmployeeId",

column: x => x.EmployeeId,

principalTable: "Employees",

principalColumn: "Id",

onDelete: ReferentialAction.Restrict);

table.ForeignKey(

name: "FK\_LeaveRequests\_Employees\_EmployeeId1",

column: x => x.EmployeeId1,

principalTable: "Employees",

principalColumn: "Id");

});

migrationBuilder.CreateTable(

name: "Projects",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

ProjectType = table.Column<int>(type: "int", nullable: false),

StartDate = table.Column<DateTime>(type: "datetime2", nullable: false),

EndDate = table.Column<DateTime>(type: "datetime2", nullable: false),

EmployeeId = table.Column<int>(type: "int", nullable: true),

ProjectManager = table.Column<int>(type: "int", nullable: false),

Comment = table.Column<string>(type: "nvarchar(max)", nullable: true),

Status = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_Projects", x => x.Id);

table.ForeignKey(

name: "FK\_Projects\_Employees\_EmployeeId",

column: x => x.EmployeeId,

principalTable: "Employees",

principalColumn: "Id");

});

migrationBuilder.CreateTable(

name: "ApprovalRequests",

columns: table => new

{

Id = table.Column<int>(type: "int", nullable: false)

.Annotation("SqlServer:Identity", "1, 1"),

ApproverId = table.Column<int>(type: "int", nullable: true),

LeaveRequestId = table.Column<int>(type: "int", nullable: true),

Comment = table.Column<string>(type: "nvarchar(max)", nullable: true),

Status = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_ApprovalRequests", x => x.Id);

table.ForeignKey(

name: "FK\_ApprovalRequests\_Employees\_ApproverId",

column: x => x.ApproverId,

principalTable: "Employees",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

table.ForeignKey(

name: "FK\_ApprovalRequests\_LeaveRequests\_LeaveRequestId",

column: x => x.LeaveRequestId,

principalTable: "LeaveRequests",

principalColumn: "Id",

onDelete: ReferentialAction.Restrict);

});

migrationBuilder.CreateTable(

name: "EmployeeProject",

columns: table => new

{

EmployeeId = table.Column<int>(type: "int", nullable: false),

ProjectId = table.Column<int>(type: "int", nullable: false)

},

constraints: table =>

{

table.PrimaryKey("PK\_EmployeeProject", x => new { x.ProjectId, x.EmployeeId });

table.ForeignKey(

name: "FK\_EmployeeProject\_Employees\_EmployeeId",

column: x => x.EmployeeId,

principalTable: "Employees",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

table.ForeignKey(

name: "FK\_EmployeeProject\_Projects\_ProjectId",

column: x => x.ProjectId,

principalTable: "Projects",

principalColumn: "Id",

onDelete: ReferentialAction.Cascade);

});

migrationBuilder.CreateIndex(

name: "IX\_ApprovalRequests\_ApproverId",

table: "ApprovalRequests",

column: "ApproverId");

migrationBuilder.CreateIndex(

name: "IX\_ApprovalRequests\_LeaveRequestId",

table: "ApprovalRequests",

column: "LeaveRequestId",

unique: true);

migrationBuilder.CreateIndex(

name: "IX\_EmployeeProject\_EmployeeId",

table: "EmployeeProject",

column: "EmployeeId");

migrationBuilder.CreateIndex(

name: "IX\_Employees\_PeoplePartnerId",

table: "Employees",

column: "PeoplePartnerId");

migrationBuilder.CreateIndex(

name: "IX\_Employees\_UserId",

table: "Employees",

column: "UserId",

unique: true);

migrationBuilder.CreateIndex(

name: "IX\_LeaveRequests\_EmployeeId",

table: "LeaveRequests",

column: "EmployeeId");

migrationBuilder.CreateIndex(

name: "IX\_LeaveRequests\_EmployeeId1",

table: "LeaveRequests",

column: "EmployeeId1");

migrationBuilder.CreateIndex(

name: "IX\_Projects\_EmployeeId",

table: "Projects",

column: "EmployeeId");

migrationBuilder.CreateIndex(

name: "IX\_Users\_RoleId",

table: "Users",

column: "RoleId");

}

protected override void Down(MigrationBuilder migrationBuilder)

{

migrationBuilder.DropTable(

name: "ApprovalRequests");

migrationBuilder.DropTable(

name: "EmployeeProject");

migrationBuilder.DropTable(

name: "LeaveRequests");

migrationBuilder.DropTable(

name: "Projects");

migrationBuilder.DropTable(

name: "Employees");

migrationBuilder.DropTable(

name: "Users");

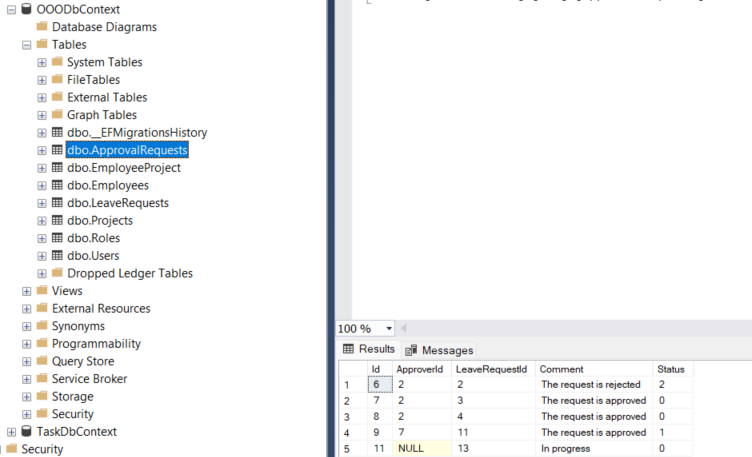
migrationBuilder.DropTable(

name: "Roles");

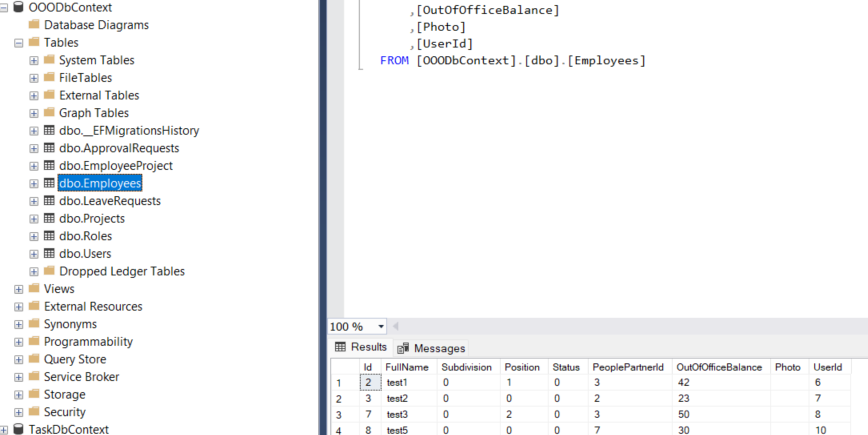
}

# Screenshots from MSSQL Server Management Studio

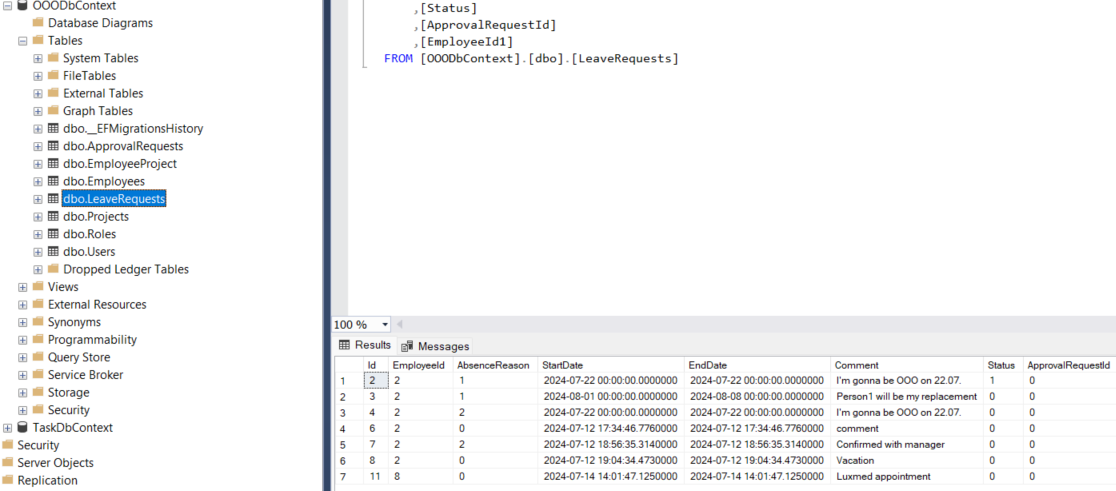
**dbo.ApprovalRequests:**



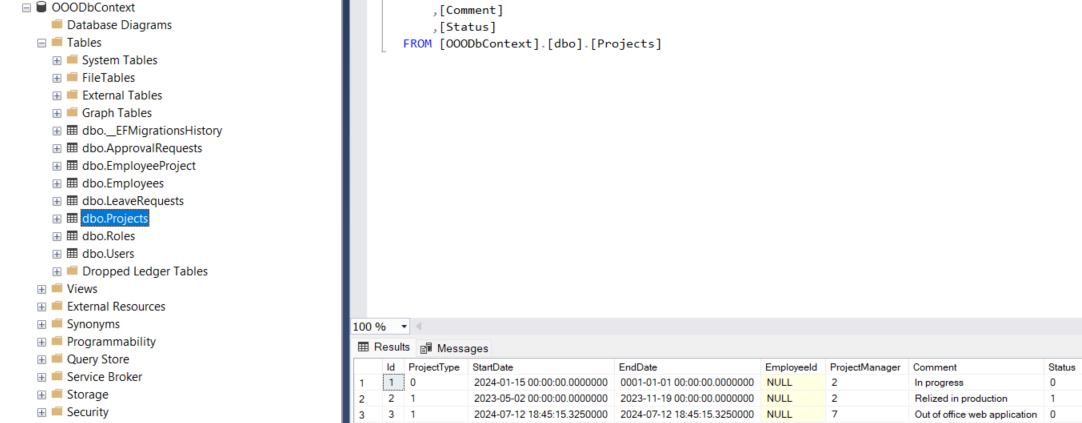
**dbo.Employees:**



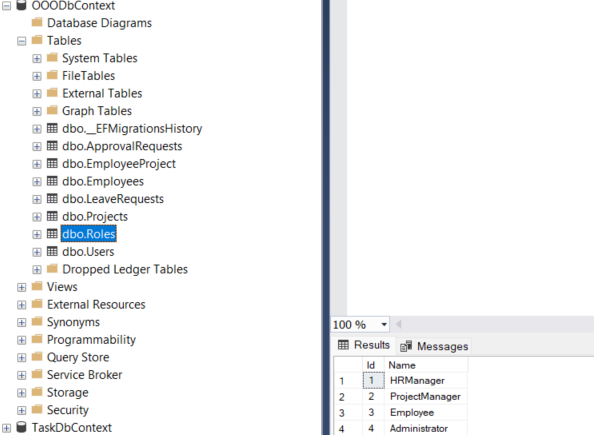
**Dbo.LeaveRequests**



**dbo.Projects**



**dbo.Roles**



**dbo.Users**

