

**Міністерство освіти і науки України**  
**Національний технічний університет України «Київський політехнічний**  
**інститут імені Ігоря Сікорського" Факультет інформатики та**  
**обчислювальної техніки**

**Кафедра інформатики та програмної інженерії**

**Звіт**

з лабораторної роботи № 4 з дисципліни  
«Сучасні технології розробки WEB-застосувань на платформі Microsoft.NET»

**“Імплементація REST API”**

**Виконав(ла)**

*ІП-15 Костін В.А.*

(шифр, прізвище, ім'я, по батькові)

**Перевірів**

*Бардін В.*

(прізвище, ім'я, по батькові)

Київ 2023

## Лабораторна робота 4

### Імплементація REST API

#### Завдання

1. Використовуючи архітектуру розроблену в попередній роботі. Імплементувати REST веб-API, використавши N Layered архітектуру.
2. Покрити модульними та інтеграційними тестами основні компоненти рішення.
3. Експортувати розроблене API до Postman.

Доменна область:

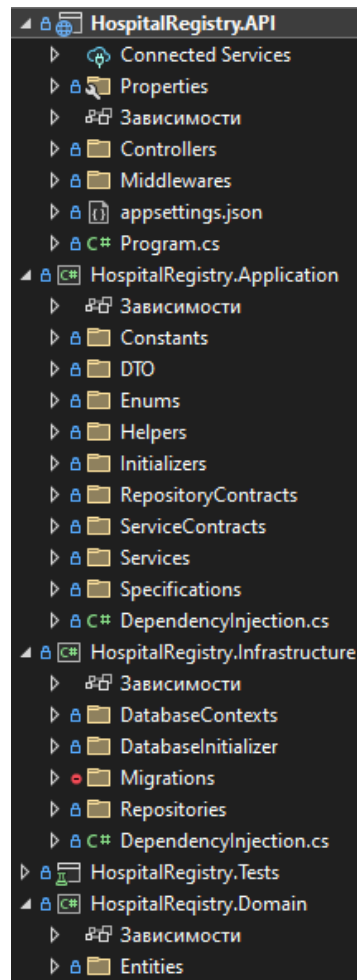
9	Реєстратура лікарні. Запис до лікарів на прийом	<p>1. Реєстратура надає дані стосовно наявності лікарів та розкладу прийому хворих.</p> <p>2. Хворим можливо записатись на прийом до лікаря, якщо є вільний час у розкладі лікаря.</p> <p>3. В реєстратурі ведуться картки відвідування хворими лікарні, в які записується час відвідання лікаря, діагноз та лікар, що його поставив.</p> <p><b>Функціональні вимоги:</b></p> <p>1. Ведення картотеки лікарні;</p> <p>2. Керування прийомами хворих у лікарів.</p>
---	---	--

Код проекту можна побачити за посиланням:

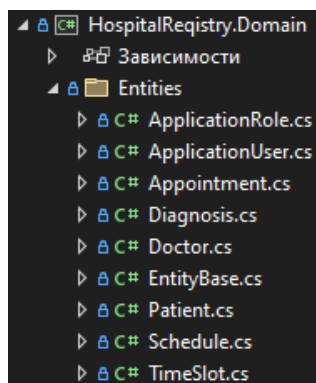
<https://github.com/VadimKostinProg/HospitalRegistryWebApp>

## 1. Структура проекту

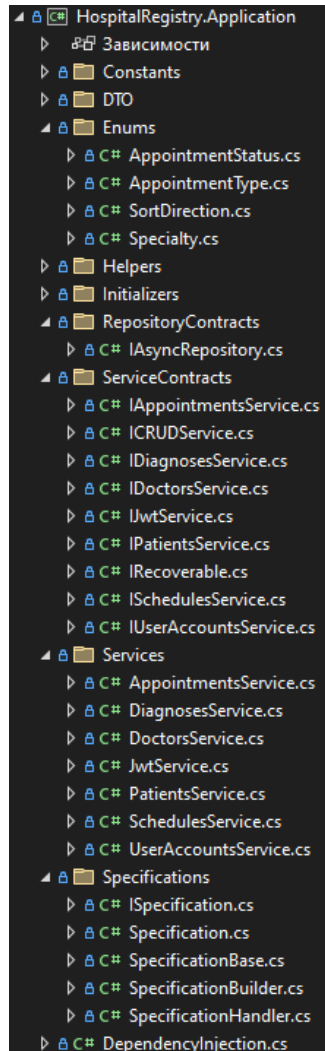
Для реалізації N-Layered архітектури було використано паттерн Clean Architecture. Структура застосунку має наступний вигляд:



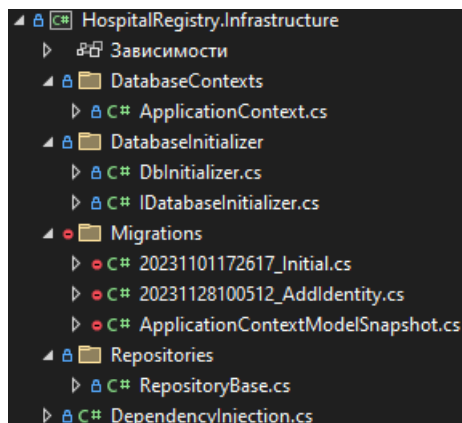
Domain Layer: містить імплементацію сутностей предметної області.



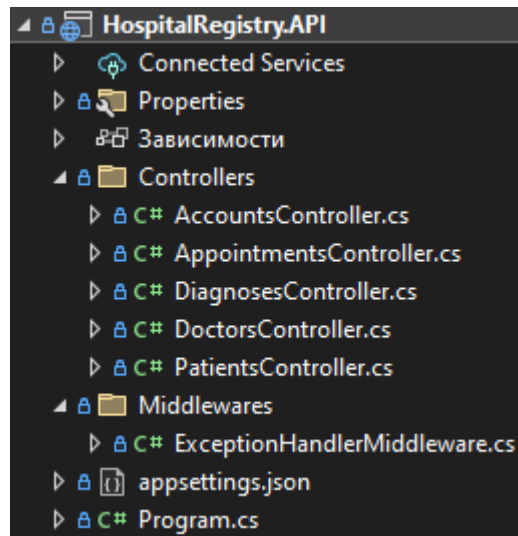
Application Layer: містить імплементацію інтерфейсу репозиторія, інтерфейси та імплементації сервісів, DTO, реалізація Specification Pattern.



Infrastructure Layer: містить імплементацію репозиторію, контекст EF Core бази даних, ініціалізатор та міграції до БД.



Presentation Layer: містить імплементацію контролерів, middlewares та загальну конфігурацію проекту.



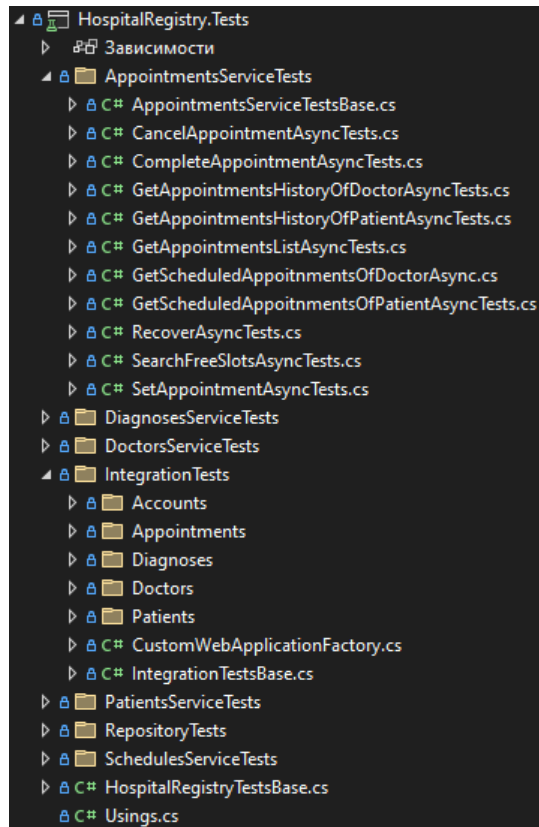
Далі наведено документацію API у вигляді Swagger:

Accounts		^
GET	/api/v1/Accounts	▼
POST	/api/v1/Accounts	▼
POST	/api/v1/Accounts/login	▼
POST	/api/v1/Accounts/register	▼
GET	/api/v1/Accounts/logout	▼
DELETE	/api/v1/Accounts/{id}	▼
Appointments		^
GET	/api/v1/Appointments	▼
POST	/api/v1/Appointments	▼
PUT	/api/v1/Appointments	▼
GET	/api/v1/Appointments/{id}	▼
DELETE	/api/v1/Appointments/{id}	▼
GET	/api/v1/Appointments/free-slots	▼
POST	/api/v1/Appointments/{id}/recover	▼
DELETE	/api/v1/Appointments/canceled	▼

Diagnoses		^
GET	/api/v1/Diagnoses	▼
POST	/api/v1/Diagnoses	▼
PUT	/api/v1/Diagnoses	▼
GET	/api/v1/Diagnoses/{id}	▼
DELETE	/api/v1/Diagnoses/{id}	▼
GET	/api/v1/Diagnoses/deleted	▼
POST	/api/v1/Diagnoses/{id}/recover	▼
Doctors		^
GET	/api/v1/Doctors	▼
POST	/api/v1/Doctors	▼
PUT	/api/v1/Doctors	▼
GET	/api/v1/Doctors/{id}	▼
DELETE	/api/v1/Doctors/{id}	▼
GET	/api/v1/Doctors/deleted	▼
POST	/api/v1/Doctors/{id}/recover	▼
GET	/api/v1/Doctors/{id}/schedule	▼
POST	/api/v1/Doctors/schedule	▼
GET	/api/v1/Doctors/{id}/appointments/history	▼
GET	/api/v1/Doctors/{id}/appointments/scheduled	▼
Patients		^
GET	/api/v1/Patients	▼
POST	/api/v1/Patients	▼
PUT	/api/v1/Patients	▼
GET	/api/v1/Patients/{id}	▼
DELETE	/api/v1/Patients/{id}	▼
GET	/api/v1/Patients/deleted	▼
POST	/api/v1/Patients/{id}/recover	▼
GET	/api/v1/Patients/{id}/appointments/history	▼
GET	/api/v1/Patients/{id}/appointments/scheduled	▼

## 2. Тестування

В проєкті присутні як модульні, так і інтеграційні тести. Для модульних тестів кожного методу створюється окремий клас. Так само і для інтеграційних методів кожного ендпоінту.








Результат тестування:

✓ HospitalRegistry.Tests (223)	20,6 с
✓ HospitalRegistry.Tests.AppointmentsSe...	1,2 с
✓ HospitalRegistry.Tests.DiagnosesServic...	566 мс
✓ HospitalRegistry.Tests.DoctorsServiceTe...	544 мс
✓ HospitalRegistry.Tests.IntegrationTests....	2,4 с
✓ HospitalRegistry.Tests.IntegrationTests....	380 мс
✓ HospitalRegistry.Tests.IntegrationTests....	4,5 с
✓ HospitalRegistry.Tests.IntegrationTests....	3,4 с
✓ HospitalRegistry.Tests.IntegrationTests....	2,5 с
✓ HospitalRegistry.Tests.PatientsServiceTe...	149 мс
✓ HospitalRegistry.Tests.RepositoryTests (...)	4,9 с
✓ HospitalRegistry.Tests.SchedulesService...	96 мс

Результат покриття тестами:

Total	84%	191/1226
HospitalRegistry.Application <ul style="list-style-type: none"> <li>HospitalRegistry.Application                 <ul style="list-style-type: none"> <li>Specifications: 97% (3/90)</li> <li>DTO: 91% (24/264)</li> <li>Services: 81% (152/801)                     <ul style="list-style-type: none"> <li>JwtService: 100% (0/16)</li> <li>SchedulesService: 97% (2/78)</li> <li>DiagnosesService: 96% (4/96)</li> <li>AppointmentsService: 90% (26/252)</li> <li>PatientsService: 83% (18/108)</li> <li>DoctorsService: 81% (23/122)</li> <li>UserAccountsService: 39% (79/129)</li> </ul> </li> </ul> </li> <li>HospitalRegistry.Infrastructure                 <ul style="list-style-type: none"> <li>HospitalRegistry.Infrastructure.Repositories: 83% (12/71)</li> <li>RepositoryBase: 83% (12/71)</li> </ul> </li> </ul>	85%	179/1155

- ✓ HospitalRegistry
  - ✓  Diagnoses
    - GET GET - Diagnoses List
    - GET GET - Diagnoses By Id
    - GET GET - Deleted Diagnoses List
    - POST POST - Create Diagnoses
    - PUT PUT - Update Diagnoses
    - DEL DELETE - Delete Diagnosis
    - POST Post Recover Diagnosis
  - >  Doctors
  - >  Patients
  - >  Appointments
- ✓  Accounts
  - POST POST - Login Admin
  - POST POST - Login Doctor
  - POST POST - Login Patient
  - POST POST - Register Doctor
  - POST POST - Register Patient
  - GET GET - Accounts List
  - GET GET - Logout
  - POST POST - Create Account
  - DEL DELETE - Delete Account



## Запит на вхід у систему в якості адміна:

### Запит на список усіх наявних записів:

GET

▼

{{base\_url}}/appointments?pageSize=2&pageNumber=1

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Type

Bearer Token ▼

ⓘ

Heads up! These parameters hold sensitive data. To keep this data secure, we've redacted some values.
 [variables](#)

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token

{{admin-token}}

Body

Cookies (1)

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

JSON ▼

⌄

```

1  {
2    "list": [
3      {
4        "id": "f0a281cd-10b1-4f21-a55f-1ee92ed1e10f",
5        "dateAndTime": "2023-12-11T12:00:00",
6        "doctor": {
7          "id": "29dc783b-b167-4d4e-8a13-sb2e3ec27f85",
8          "name": "DoctorName1",
9          "surname": "DoctorSurname1",
10         "patronymic": "DoctorPatronymic1",
11         "dateOfBirth": "30.11.2004",
12         "specialty": "Dermatologist",
13         "email": "testDoctorEmail1@gmail.com",
14         "phoneNumber": "+380959999999"
15       },
16       "patient": {
17         "id": "ee210537-0cc1-489e-a1dc-dd62c78868dd",
18         "name": "PatientName1",
19         "surname": "PatientSurname1",
20         "patronymic": "PatientPatronymic1",
21         "dateOfBirth": "28.11.2003",
22         "email": "patientEmail1@gmail.com",
23         "phoneNumber": "+380959999999"
24       },
25       "appointmentType": "HealthVisit",
26       "diagnosis": "Allergy",
27       "extraClinicalData": "test clinica data",
28       "status": "Completed",
29       "conclusion": "test conclusion"
30     ]
31   }

```

## Запит на створення нового доктору в системі:

The screenshot shows a REST client interface with a POST request to `{{base_url}}/doctors`. The request body is a JSON object with the following fields: `name`, `surname`, `patronymic`, `dateOfBirth`, `specialty`, `email`, and `phoneNumber`. The response body is a JSON object with an `id` field and the same data as the request body.

```
POST {{base_url}}/doctors

{
  "name": "DoctorName7",
  "surname": "DoctorSurname7",
  "patronymic": "DoctorPatronymic7",
  "dateOfBirth": "2001-01-01",
  "specialty": "Dermatologist",
  "email": "testDoctorEmail7@gmail.com",
  "phoneNumber": "+380959999999"
}
```

Body Cookies (1) Headers (4) Test Results

Pretty Raw Preview Visualize JSON

```
{
  "id": "5f8c223b-4472-44b8-a1e9-4537ff380a5a",
  "name": "DoctorName7",
  "surname": "DoctorSurname7",
  "patronymic": "DoctorPatronymic7",
  "dateOfBirth": "01.01.2001",
  "specialty": "Dermatologist",
  "email": "testDoctorEmail7@gmail.com",
  "phoneNumber": "+380959999999"
}
```

## Запит оновлення назви діагнозу:

The screenshot shows a REST client interface with a PUT request to `{{base_url}}/diagnoses`. The request body is a JSON object with `id` and `name` fields. The response body is a JSON object with the same `id` and `name` fields.

```
PUT {{base_url}}/diagnoses

{
  "id": "d8601429-b9c2-45e8-9b24-9fed90a9901d",
  "name": "new diagnoses"
}
```

Body Cookies (1) Headers (4) Test Results

Pretty Raw Preview Visualize JSON

```
{
  "id": "d8601429-b9c2-45e8-9b24-9fed90a9901d",
  "name": "new diagnoses"
}
```

# Запит на історію записів пацієнта:

GET

{{(base\_url)}}/patients/ee210537-0cc1-489e-a1dc-dd62c78860dd/appointments/history

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Type

Bearer Token

Heads up! These parameters hold sensitive data. To keep this data secure, we've replaced them with variables.

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token

{{(patient-token)}}

Body

Cookies (1)

Headers (4)

Test Results

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

{},"list": [{"id": "2476af9d-6e1b-43df-896a-6ce0f6e7c914", "dateAndTime": "2023-12-11T14:00:00", "doctor": {"id": "29dc783b-b167-4d4e-8a13-eb2e3ec27f85", "name": "DoctorName1", "surname": "DoctorSurname1", "patronymic": "DoctorPatronymic1", "dateOfBirth": "30.11.2004", "specialty": "Dermatologist", "email": "testDoctorEmail1@gmail.com", "phoneNumber": "+380959999999"}, "patient": {"id": "ee210537-0cc1-489e-a1dc-dd62c78860dd", "name": "PatientName1", "surname": "PatientSurname1", "patronymic": "PatientPatronymic1", "dateOfBirth": "28.11.2003", "email": "patientEmail1@gmail.com", "phoneNumber": "+380959999999"}, "appointmentType": "HealthVisit", "diagnosis": "Allergy", "extraClinicalData": "test clinica data", "status": "Completed", "conclusion": "test conclusion"}]