

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота № 2

з дисципліни «Сучасні технології розробки WEB-застосувань на платформі
Microsoft.NET»

Тема: «Проектування REST веб-API»

Виконав:

студент групи IA-13

Тарасовець Максим

Дата здачі _____

Захищено з балом _____

Перевірила:

ст. вик. кафедри ІІІ

Крамар Ю. М.

Київ, 2023

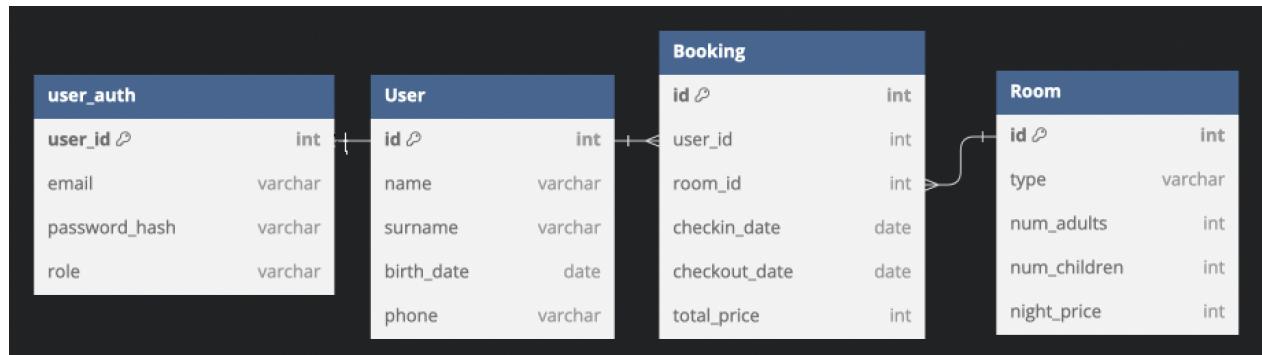
Завдання

1. З дотриманням вимог REST-у спроектувати веб-API для обраної(згідно варіанту) доменної області, використовуючи методологію C4 для створення діаграмами архітектури системи.
2. Створити ER-діаграму для DAL (Data Access Layer), яка відображатиме структуру бази даних веб-API.
3. Оформити спроектоване рішення у вигляді звіту до лабораторної роботи.

Варіант

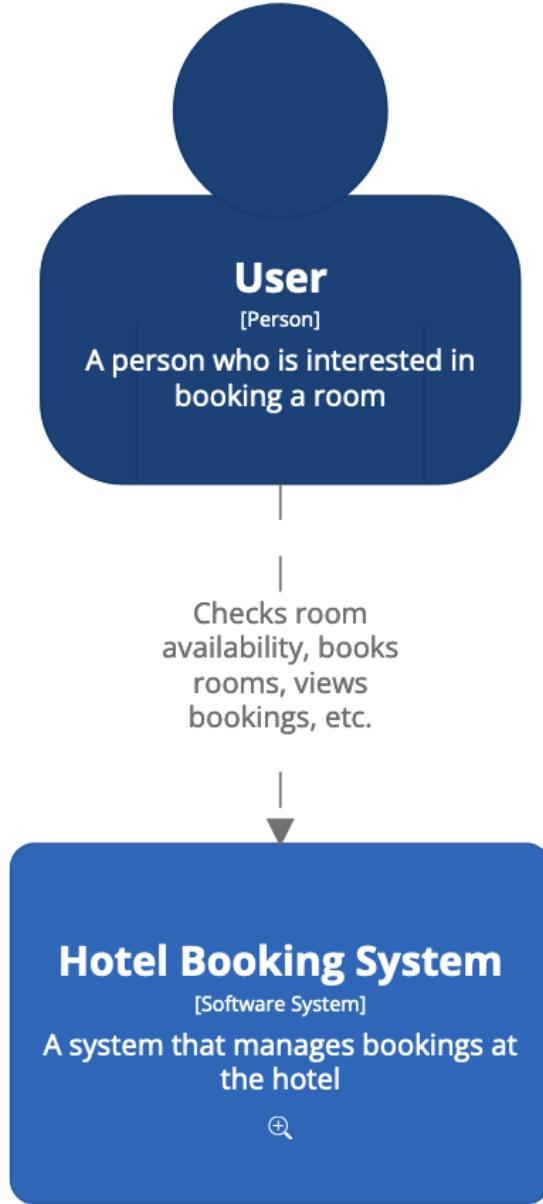
8	Готель. Сервіс бронювання номерів	<p>1. Готель надає послуги розміщення туристів у номерах.</p> <p>2. Номера мають певну категорію, від якої залежить ціна.</p> <p>3. Номери можна безпосередньо здати у використання на певну кількість діб, отримавши плату, чи попередньо забронювати та згодом перевести їх у стан зданих.</p> <p>Функціональні вимоги:</p> <ol style="list-style-type: none">1. Бронювання номерів;2. Отримання інформації про наявність вільних номерів на певний термін.
---	-----------------------------------	---

ER Diagram



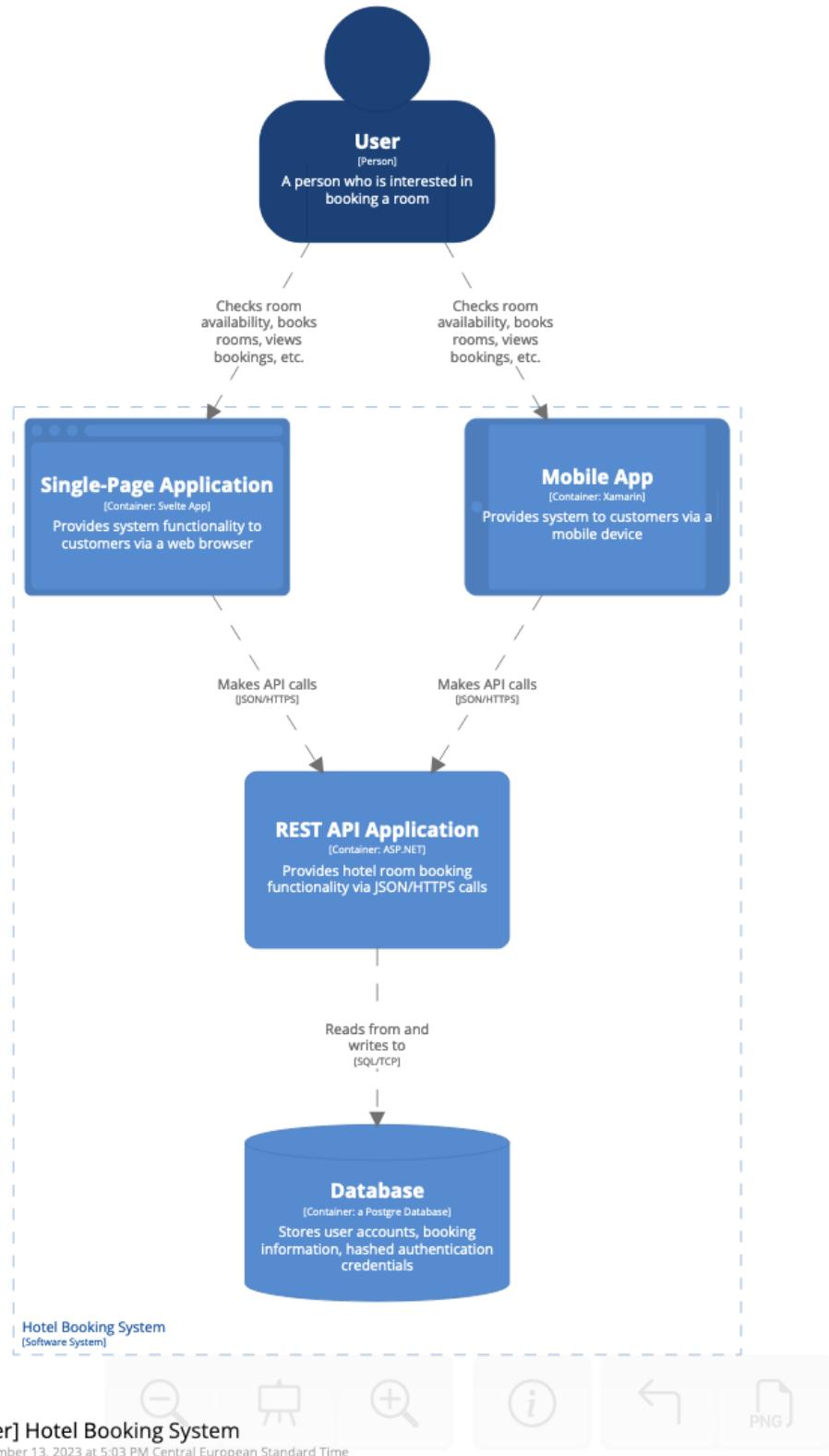
C4 Diagram

Level 1 System Context Diagram:



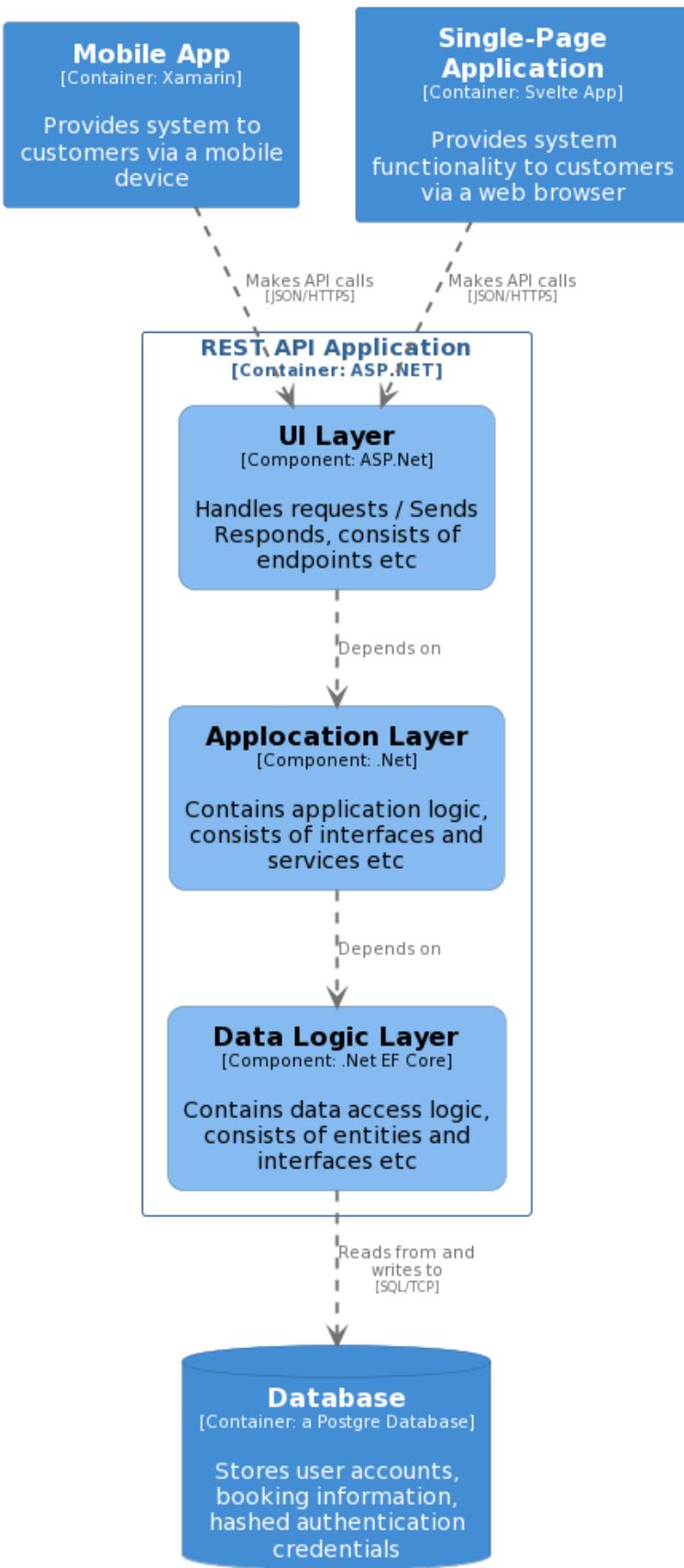
[System Context] Hotel Booking System
Monday, November 13, 2023 at 4:59 PM Central European Standard Time

Level 2 Container:



Level 3 Component Level:

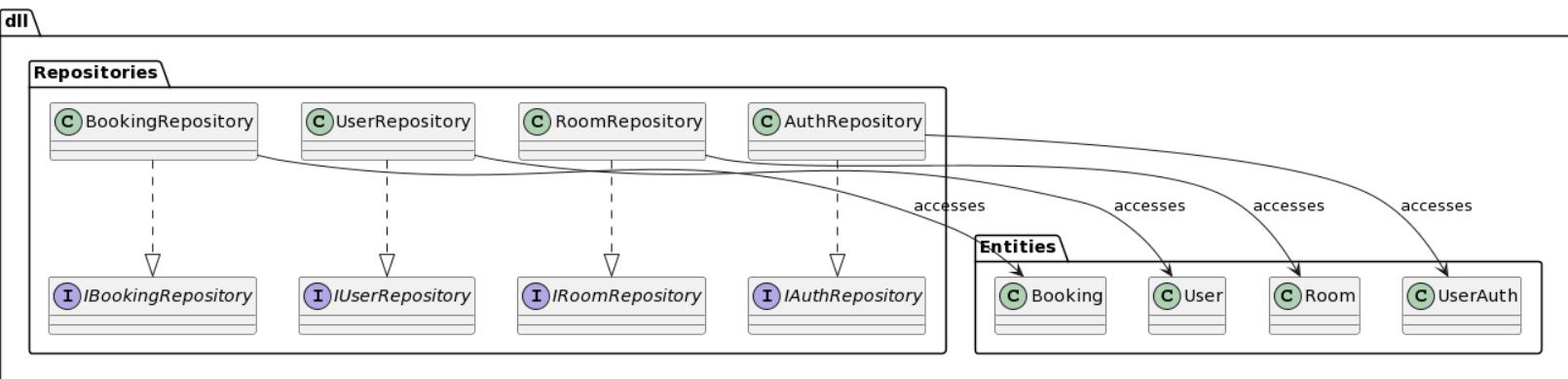
Hotel Booking System - REST API Application - Components



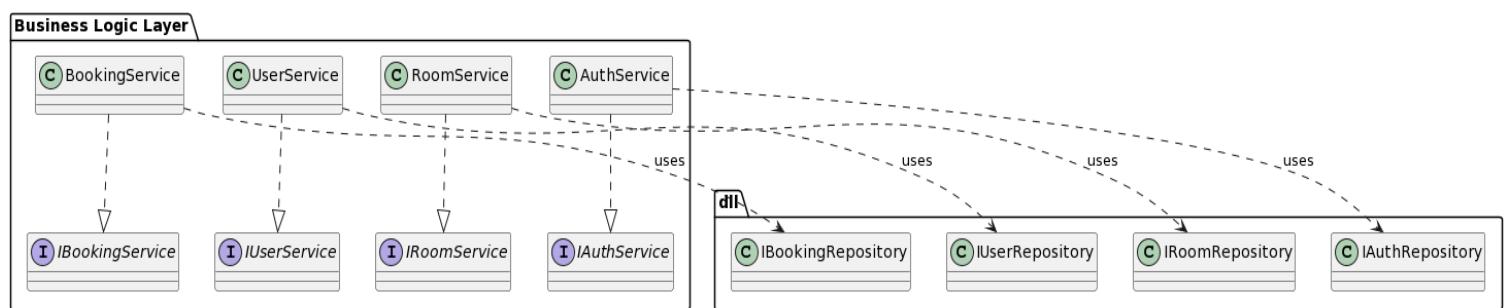
Level 4 Code:

Divided into 4 parts (dll , pl , bll)

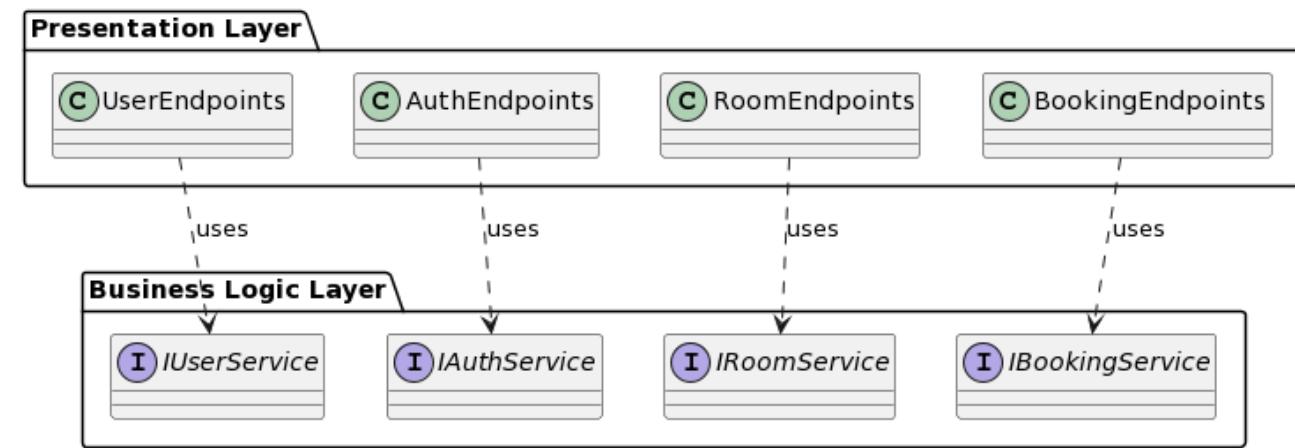
Data Logic Layer:



Business Logic Layer:



Presentation Layer:



URIs scheme:

AUTH

POST api/auth/login { email, password }

POST api/auth/register { email, password, name, surname, birthday_date, phone}

PATCH /api/auth/change-password { currentPassword, newPassword } #auth required

ROOMS

POST api/rooms/{id} { type, num_adults, num_children, night_price } #admin role required

POST api/rooms/{id} { type, num_adults, num_children, night_price } #admin role required

GET api/rooms/search?[checkindate, checkoutdate, num_children, num_adults ,type]

GET api/room/{id}

DELETE api/room/{id} #admin role required

USERS

GET api/users/{id} #admin role required

GET api/users #admin role required

POST /api/users { email, password, name, surname, birthday_date, phone } #admin role required

PATCH api/users/{id} { email, password, name, surname, birthday_date, phone } #admin role required

DELETE api/users/{id} #admin role required

USERS/me (disabled for admins)

GET api/users/me #auth required

DELETE api/users/me/ #auth required

PATCH api/users/me {name,surname,birthday_date,phone} #auth required

USERS/me/bookings (disabled for admins)

GET api/users/me/bookings #auth required

GET api/users/me/bookings/{id} #auth required

POST api/users/me/bookings {room_id, chekin_date, chekout_date} #auth required

PATCH api/users/me/booking {room_id, chekin_date, chekout_date} #auth required

DELETE api/users/me/booking/{id} {room_id, chekin_date, chekout_date} #auth required

USERS/bookings (admins only)

GET /api/users/bookings #admin role required

GET /api/users/bookings/search? #admin role required

GET /api/users/bookings/{id} #admin role required

POST /api/users/bookings {userId, roomId, checkInDate, checkOutDate} #admin role required

PATCH /api/users/bookings/{id} {roomId, checkInDate, checkOutDate} #admin role required

DELETE /api/users/bookings/{id} #admin role required

Висновок

У цій лр я спроектував рест апі згідно з варіантом розробив с4 діаграму та ер діаграму.