

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

**ІКНІ  
Кафедра ПЗ**



**ЗВІТ**

до лабораторної роботи № 5

**на тему:** *“Проектування системи за допомогою методології UML.”*

**з дисципліни:** *“Архітектура та проектування ПЗ”*

**Лектор:**

проф. каф. ПЗ

Фоменко А. В.

**Виконали:**

студ. групи ПЗ-41

Проців О. М.

Бабілля О. О.

Малєєв А. Д

**Прийняв:**

асис. каф. ПЗ

Шкраб Р.Р.

«\_\_\_» \_\_\_\_\_ 2024р.

$\Sigma$  = \_\_\_\_\_

Львів – 2024

Тема роботи: Проектування системи за допомогою методології UML.

Мета роботи: Вивчення основних принципів методології UML, створення  
Ознайомитися з технологією побудови UML-діаграм та побудувати їх.

### **Завдання**

1. Постановка завдання. Необхідно описати постановку задачі виходячи з індивідуального завдання із зазначенням основних вимог до майбутньої системи та переліком її функціональності, повинна бути проведена структурна класифікація системи з метою визначення системних сутностей і їх стосунки між собою.
2. Представлення Use Case для системи з вказівкою акторів і вузлів. При виконанні даного етапу повинні бути побудовані діаграми Use Case, діаграма компонентів, діаграма кооперації, а також повинні бути подані відповідні опису акторів, вузлів і компонентів системи.
3. Діаграма послідовності, діаграма внутрішньої структури та діаграма основних класів системи.
4. Діаграму пакетів основних програмних сервісів системи і діаграма розгортання.
5. Обґрунтування вибору архітектури системи (тонкий або товстий клієнт), обґрунтування вибору середовища розробки, СУБД і операційної системи.
6. Прототип інтерфейсу. Повинні бути розроблені основні інтерфейси системи з урахуванням вимог до інтерфейсів. У звіті повинні бути представлені зображення інтерфейсів, які можна розробити в графічному або HTML редакторі або з використанням будь-якої середовища розробки програмного забезпечення. Основною вимогою до прототипу інтерфейсу є надання можливості представити як буде виглядати система в майбутньому.

### **Хід роботи**

#### **1. Концепція системи**

##### **1.1. Опис проблемної області**

У сучасному суспільстві бібліотеки залишаються важливими осередками знань та культурного розвитку, однак для підтримки актуальності та забезпечення ефективного доступу до ресурсів потрібні сучасні технологічні рішення. Серед типових труднощів у бібліотечній системі — незручний процес пошуку потрібної книги, відсутність повної інформації про книги, відсутність гнучкої системи фільтрації та резервування. Також проблемою може бути недостатня інтеграція з системами обліку користувачів і штрафів. Система управління бібліотечним каталогом вирішує ці питання, забезпечуючи користувачам зручний доступ до

колекцій бібліотеки, простий пошук і фільтрацію книг, можливість бронювання, а також сучасну звітність для адміністраторів.

## **1.2. Постановка задачі**

Для підвищення зручності користування бібліотечним каталогом, система управління бібліотечним каталогом повинна включати функціонал для створення та управління базою книг, можливість фільтрації за різними категоріями, інтеграцію з модулями обліку оренди та повернень. Система має надавати користувачам можливість створювати облікові записи, зберігати історію орендованих книг, залишати відгуки, а також отримувати рекомендації на основі попередніх оренд. Адміністратори та бібліотекарі повинні мати окремий інтерфейс для управління каталогом, орендами та запитами користувачів

## **1.3. Опис основного функціоналу системи**

Система "Управління бібліотечним каталогом" забезпечує зручний каталог для пошуку та фільтрації книг, модуль для оренди та повернення, а також функціонал для залишення відгуків та отримання персоналізованих рекомендацій. Вона включає адміністрування каталогу та управління орендами, а також дозволяє користувачам звертатися до підтримки через спеціальний комунікаційний модуль.

## **1.4. Структурна класифікація системи**

Система складатиметься з наступних модулів:

1. Каталог книг
2. Модуль оренди та управління оплатою
3. Панель адміністратора
4. Аналітика популярності книг
5. Панель бібліотекаря для управління та редагування інформації про книги

## **1.5. Аналіз складності компонентів системи**

Система потребує розробки ефективної бази даних для зберігання інформації про товари та користувачів, підтримки транзакцій та обробки замовлень. Особливу увагу слід приділити реалізації модуля рекомендацій та фільтрації. Система буде побудована мовою програмування Python з використанням django rest framework для роботи з базою даних і управління залежностями між компонентами.

# **2. Архітектура системи**

## **2.1. Use Case діаграма**

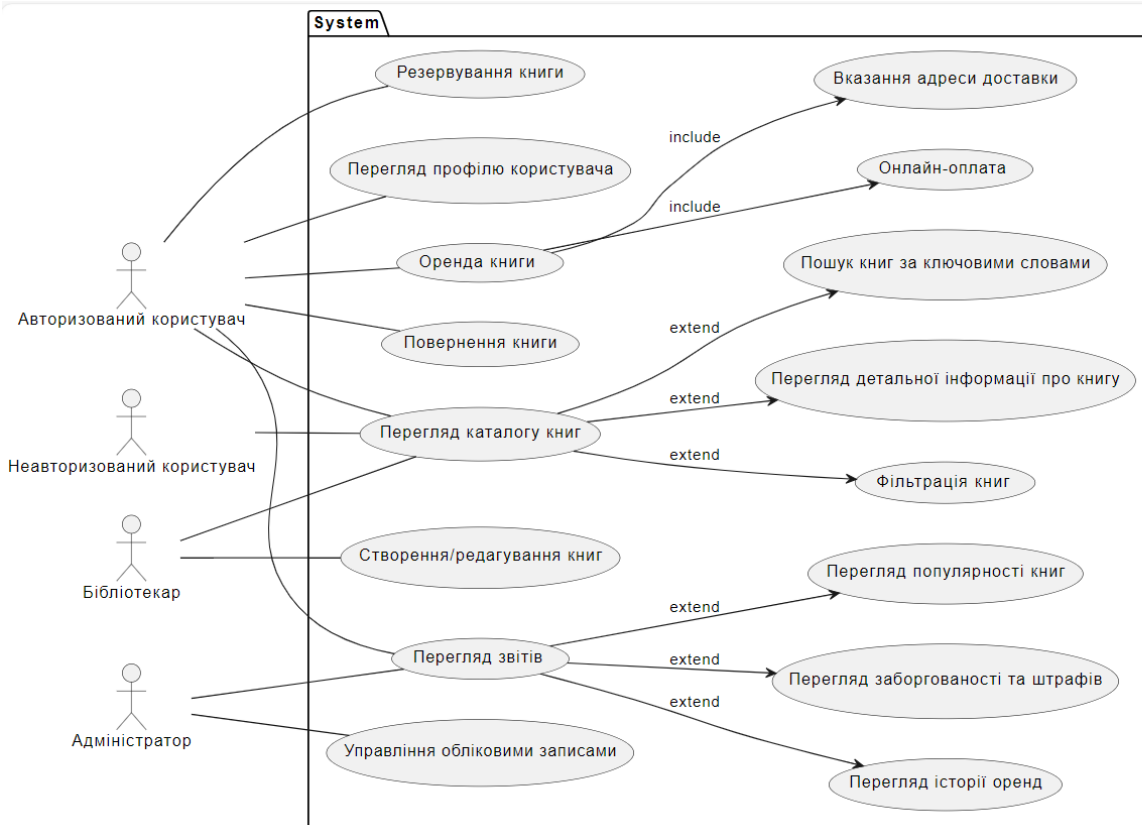


Рис. 1. Use Case діаграма

## 2.2. Діаграма послідовності

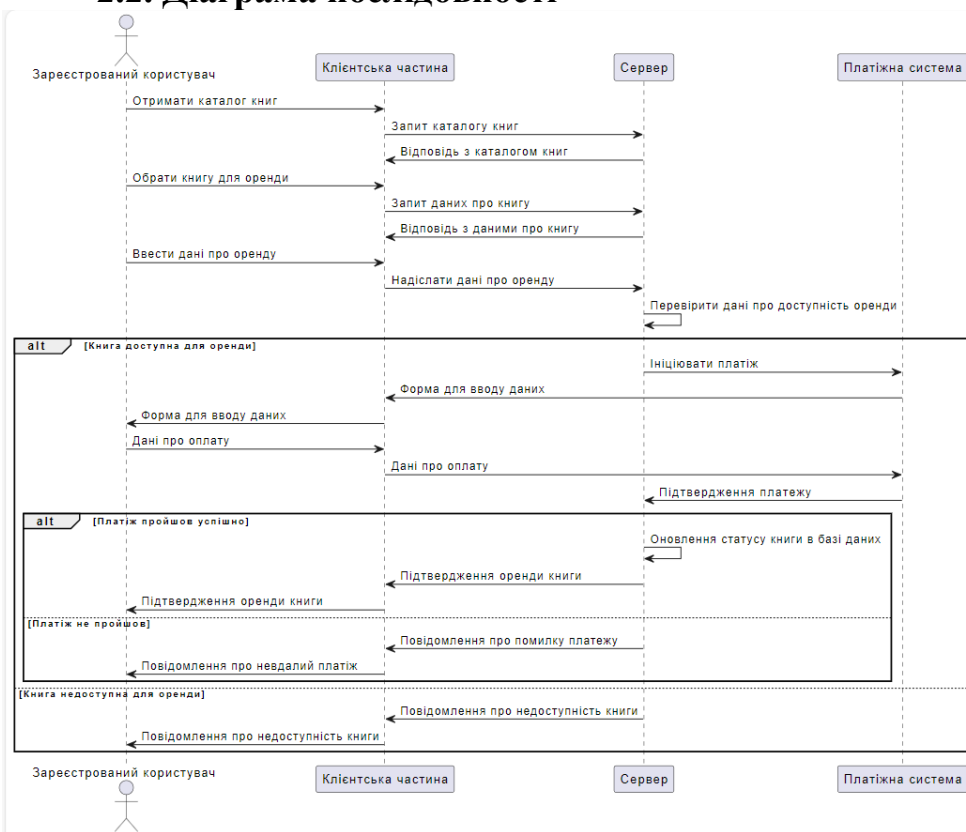


Рис. 2. Діаграма послідовності

### 2.3. Діаграма кооперації

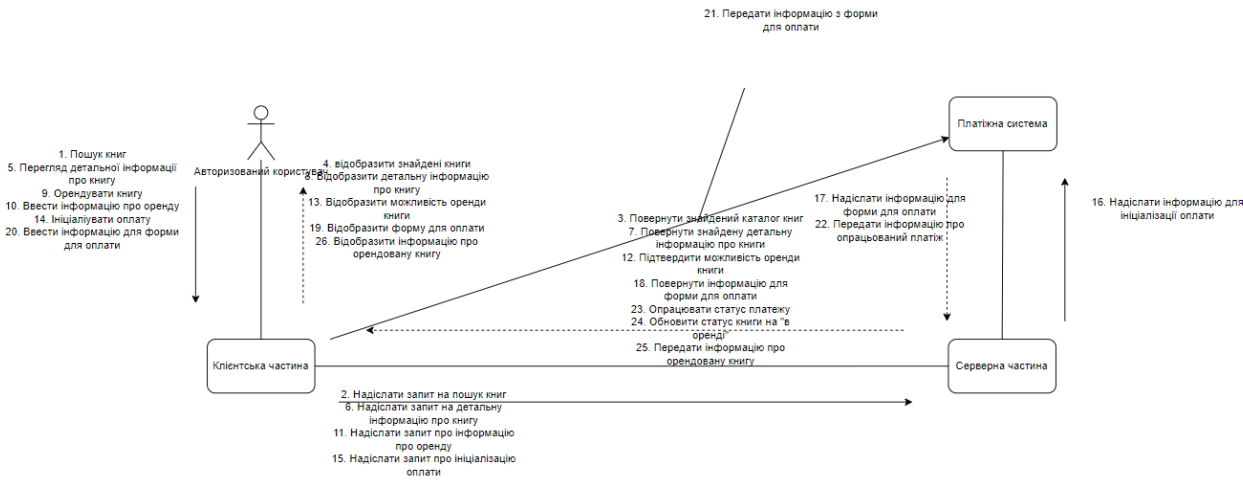


Рис. 3. Діаграма кооперації

### 2.4. Діаграма діяльності

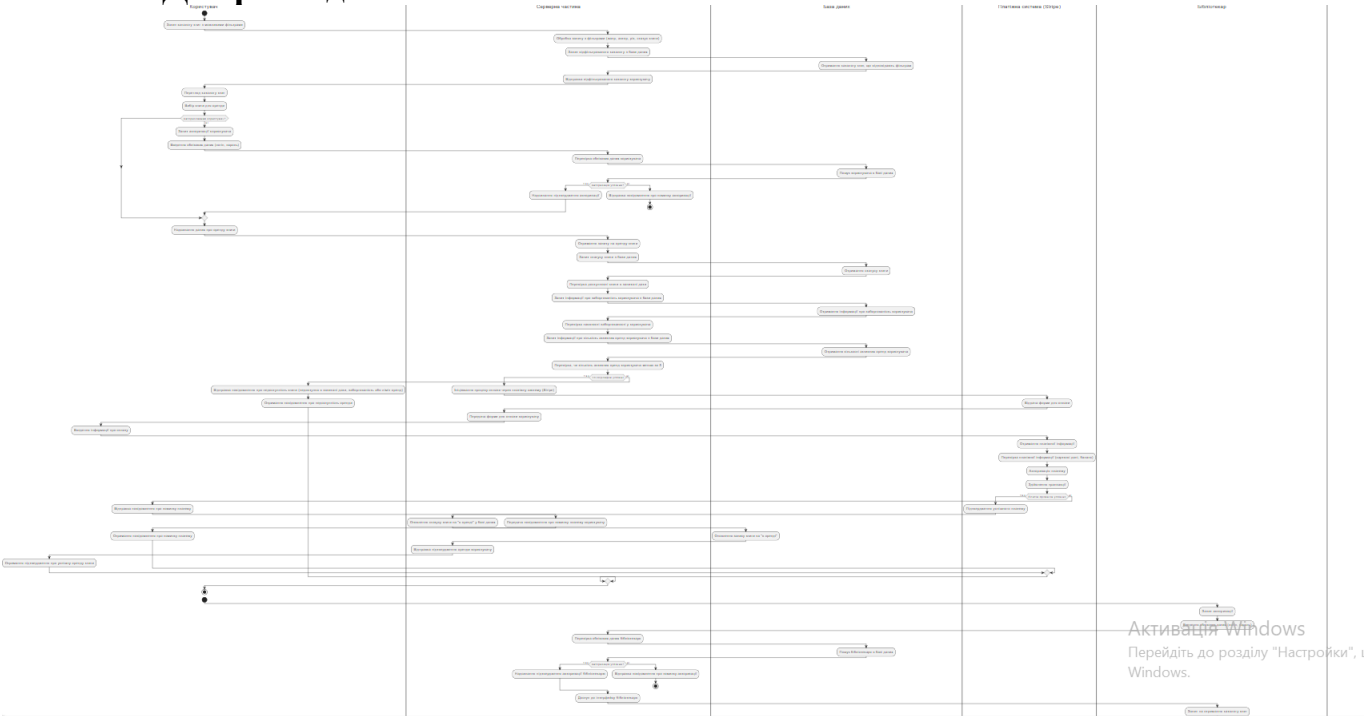


Рис. 4. Діаграма діяльності

### 2.5. Діаграма основних класів системи

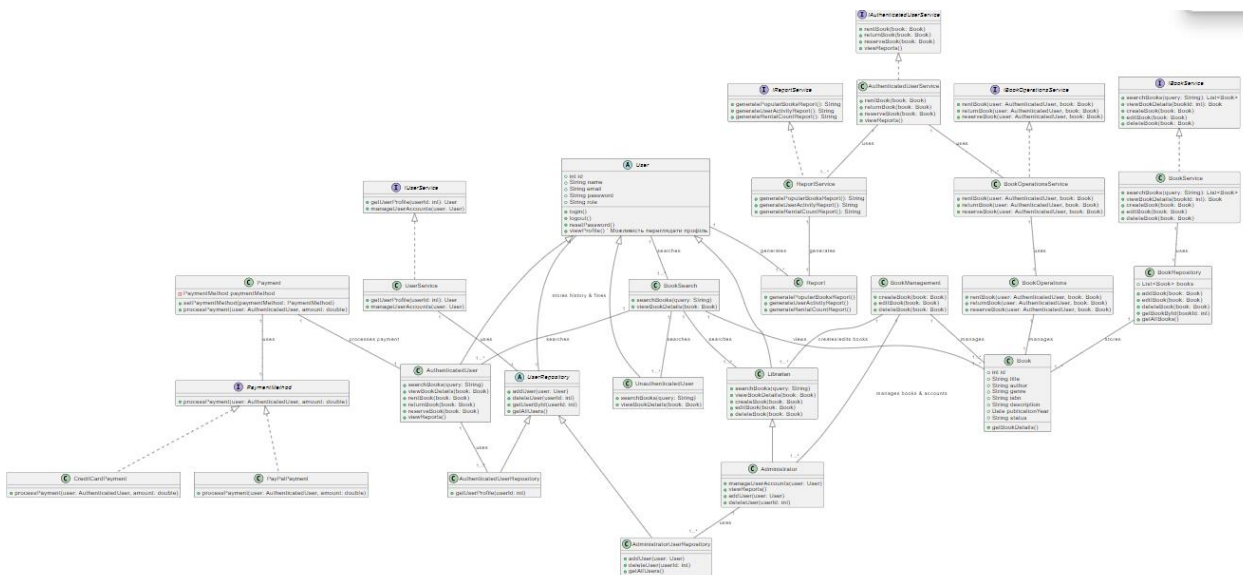


Рис. 5. Діаграма класів

## 2.6. Діаграма об'єктів

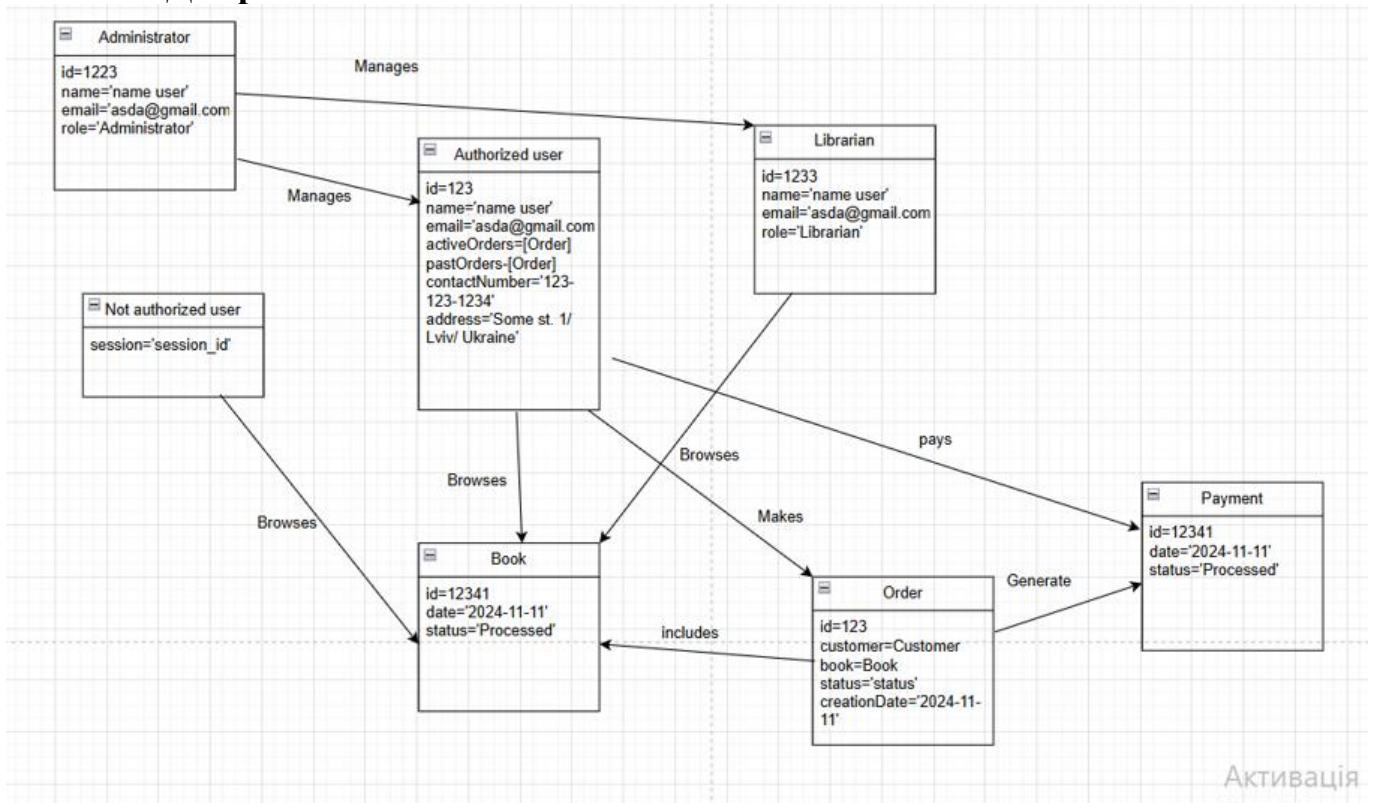


Рис. 6. Діаграма об'єктів

## 2.7. Діаграма станів

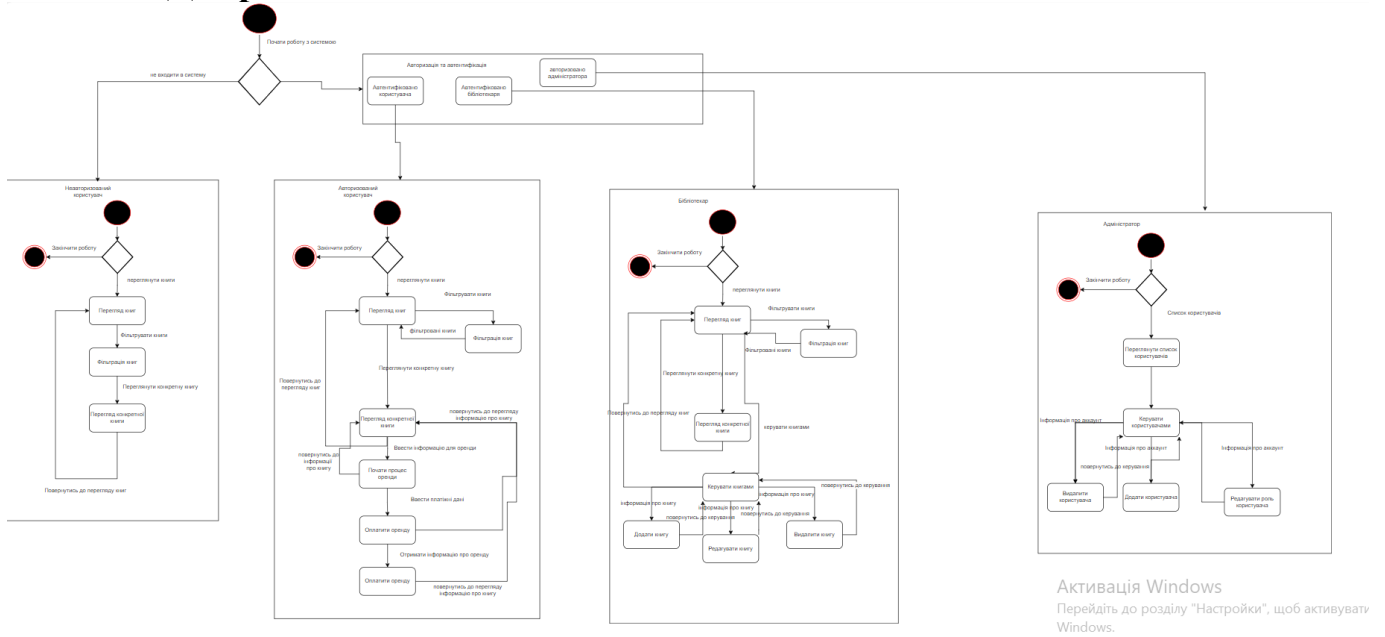


Рис. 7. Діаграма станів

## 2.8. Діаграма пакетів

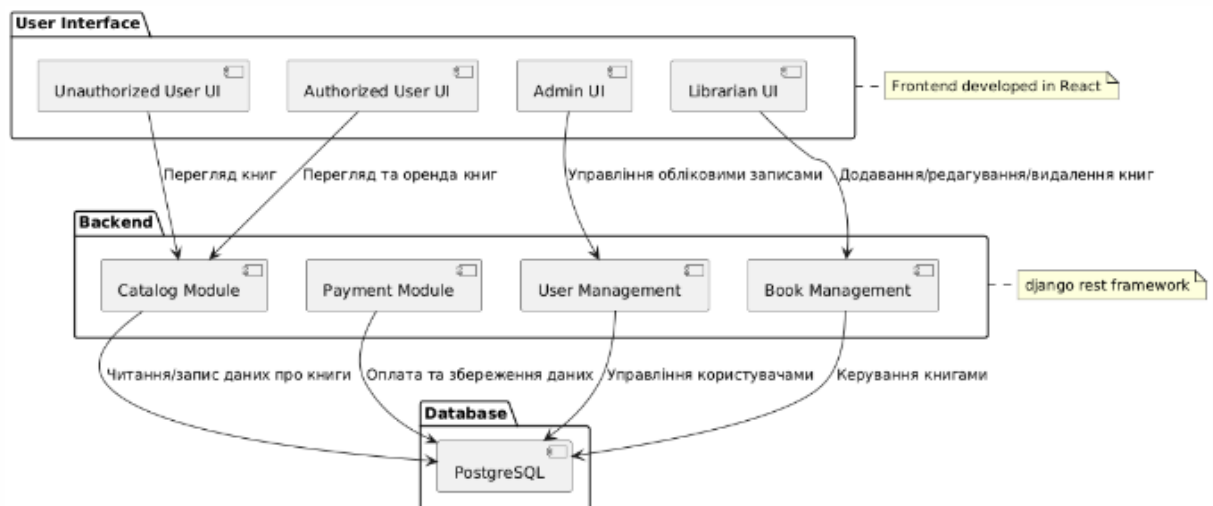


Рис. 8. Діаграма пакетів

## 2.9. Діаграма компонентів

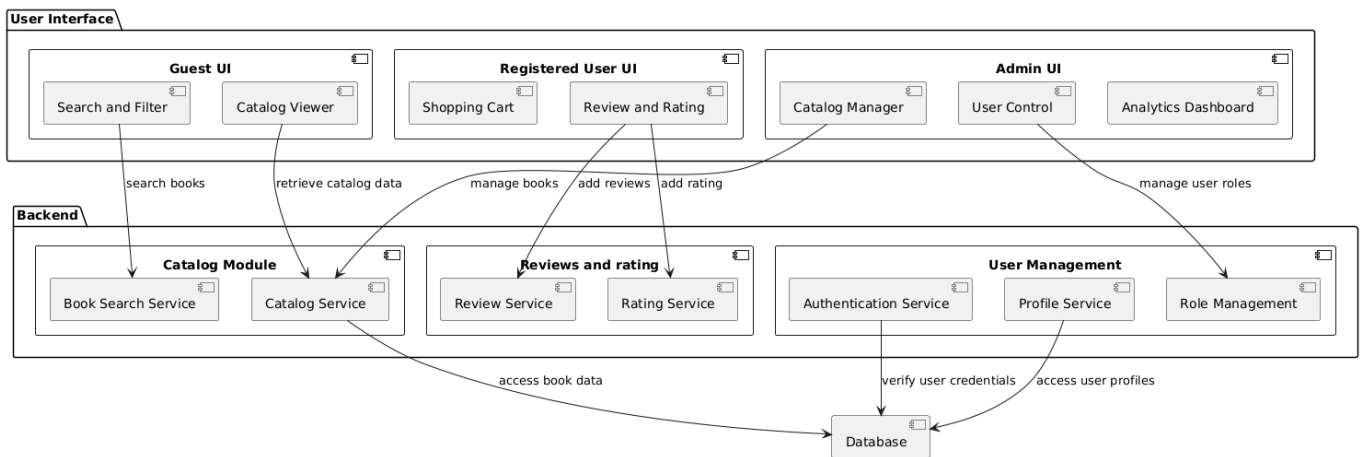


Рис. 9. Діаграма компонентів

## 2.10. Діаграма внутрішньої структури

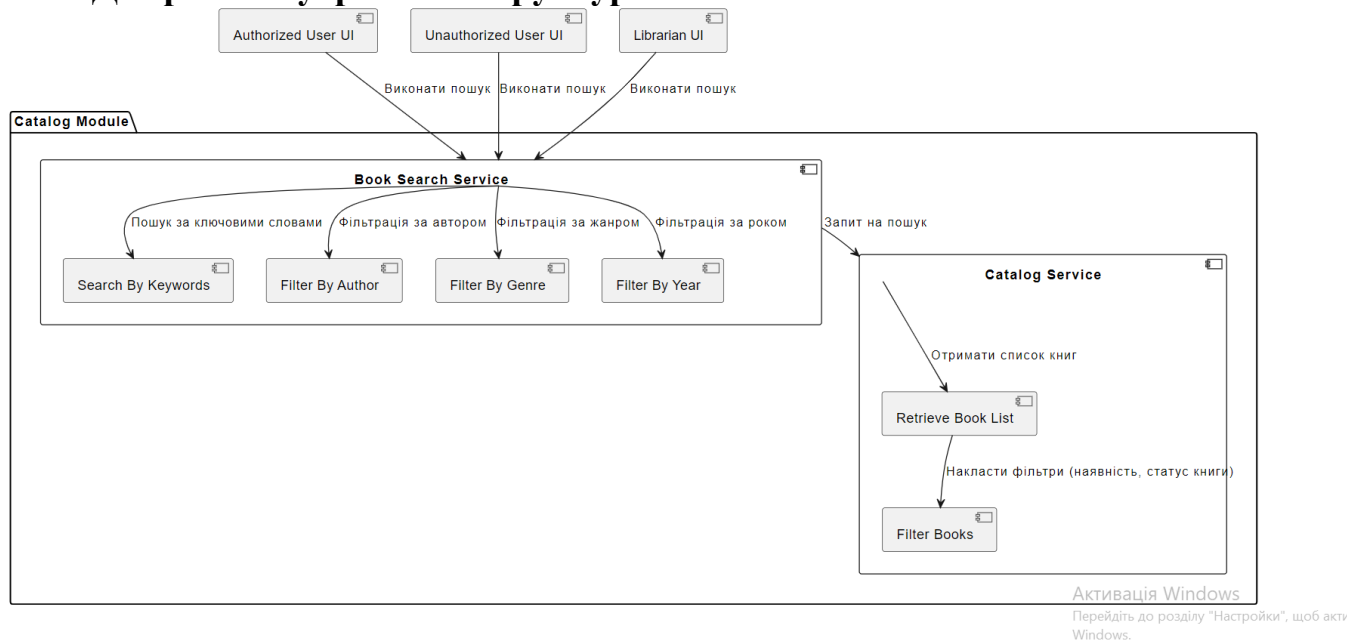


Рис. 10. Діаграма внутрішньої структури

## 2.11. Діаграма розгортання

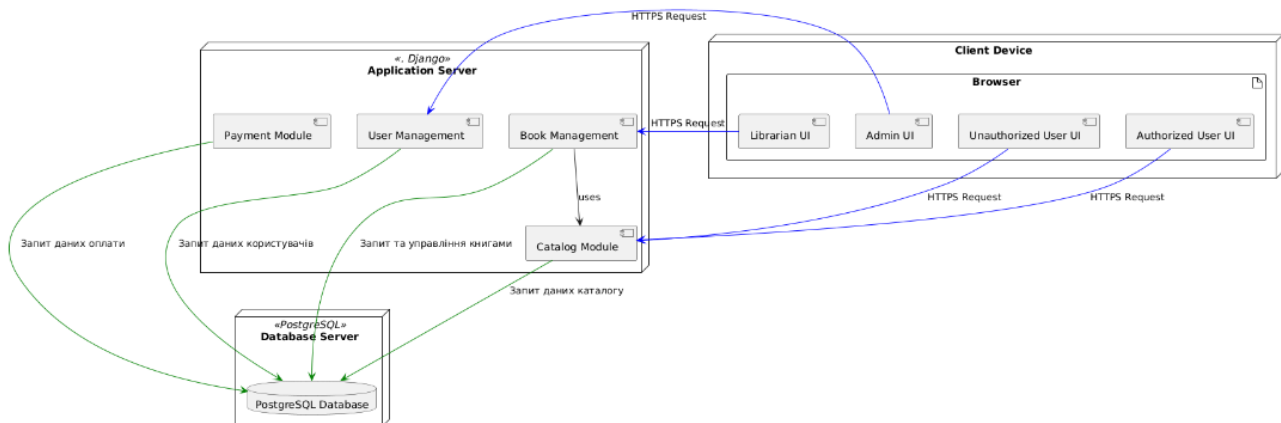


Рис. 11. Діаграма розгортання

## 2.12. Вибір архітектури системи та обґрунтування технічних рішень.



Клієнт-серверна архітектура з фронтендом як клієнтом і бекенд-сервісами як сервером через низку переваг. Така архітектура дозволяє розділити відповідальність між інтерфейсом користувача та обробкою даних, що робить систему більш масштабованою та легшою в обслуговуванні. Фронтенд відповідає за взаємодію з користувачем, а бекенд виконує обробку даних і бізнес-логіку, забезпечуючи централізоване управління ресурсами та захист даних. Це також дозволяє незалежний розвиток та оновлення фронтенду і бекенду, що сприяє більшій гнучкості в розробці.

### 2.13. Обґрунтування вибору середовища розробки, СУБД і операційної системи.

Для бекенду було обрано Django завдяки його надійності та швидкій розробці завдяки вбудованим інструментам і зручному ORM. На фронтенді використано React через його гнучкість у створенні інтерактивних SPA-додатків та широку підтримку JavaScript-екосистеми. PostgreSQL обрано як стабільну та ефективну СУБД з відкритим кодом для роботи з великими обсягами даних.

## 3. Архітектура системи

### 3.1. Опис інтерфейсу користувача (mockups)

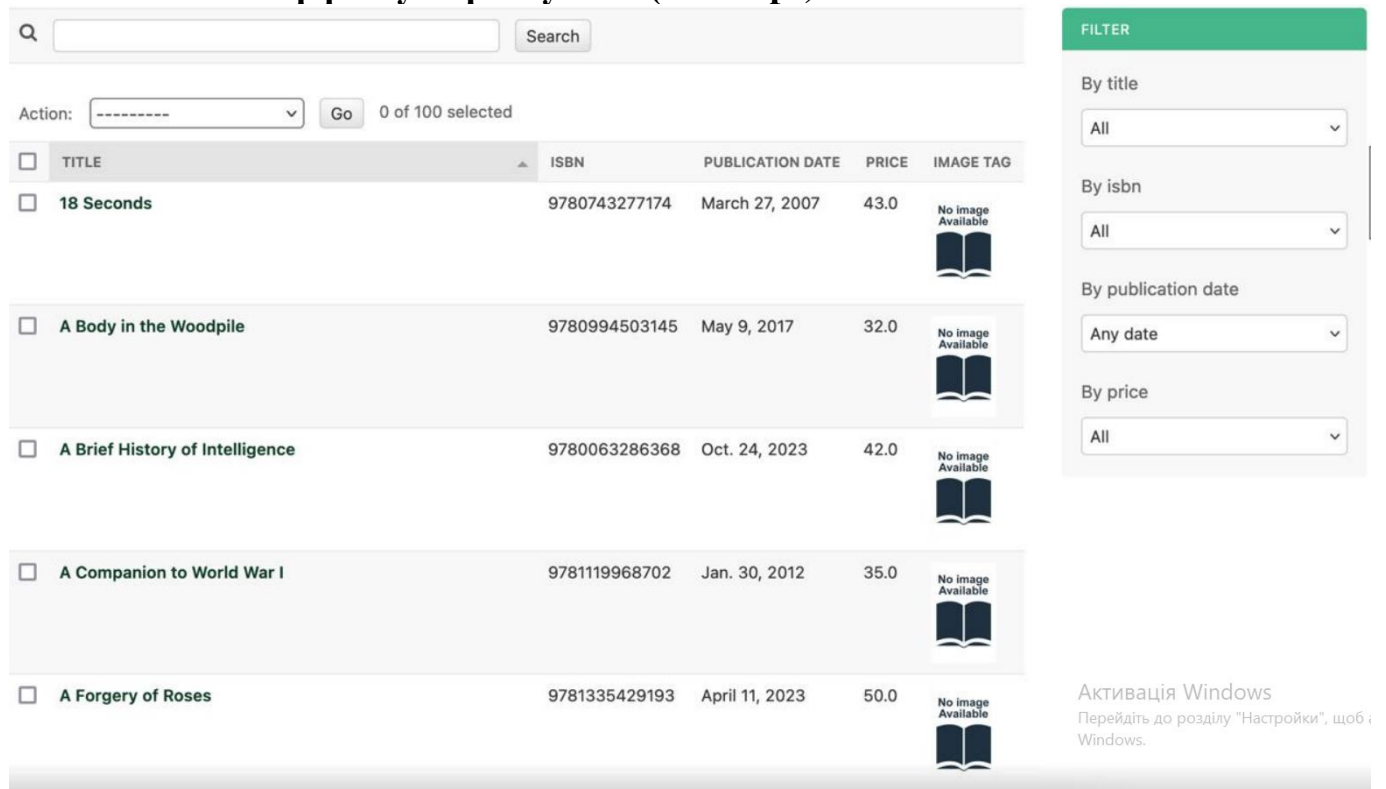


Рис. 12. Головне вікно (каталог книг)

**18 Seconds**

Title: 18 Seconds

ISBN: 9780743277174

Publication date: 2007-03-27 Today

Note: You are 2 hours ahead of server time.

Price: 43.0

Author:

Hold down "Control" or "Command" on a Mac, to select more than one.

Publisher: Simon and Schuster

Cover: Currently: default\_book\_image.jpg ☐ Clear  
 Change: No file selected.

About:

Page count: 386

Rating: 3.0

Genres:

Hold down "Control" or "Command" on a Mac, to select more than one.

Рис. 13. Детальна інформація про книгу (вигляд бібліотекара)

### 3.2. Основні компоненти інтерфейсу користувача – короткий огляд екранів і елементів GUI

1. Навігація до каталогу (зелені кнопки для переходу по сторінці):

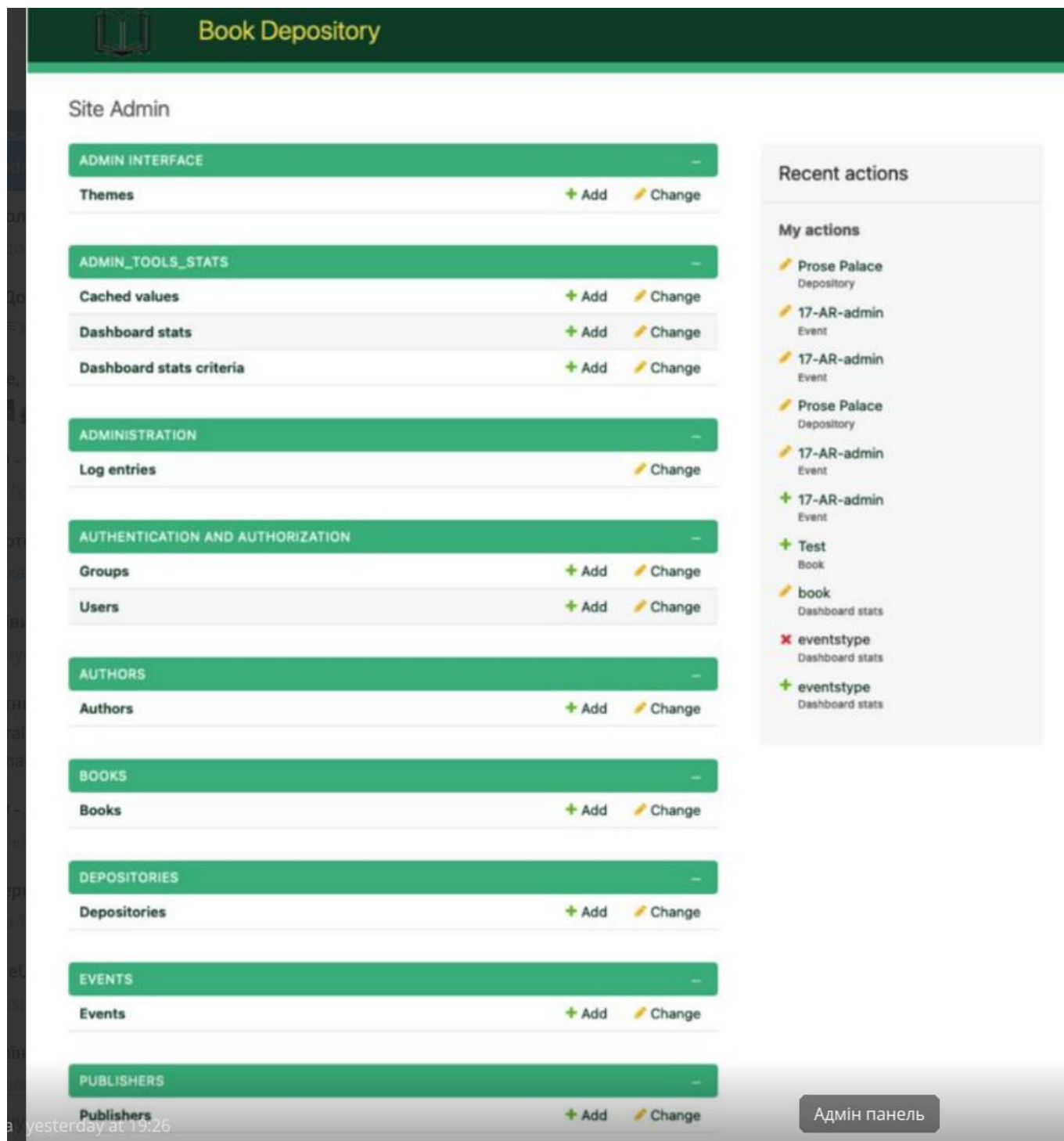


Рис. 14 Навігація з сторони адміністратора

2. Фільтрація книг (обведено червоним):

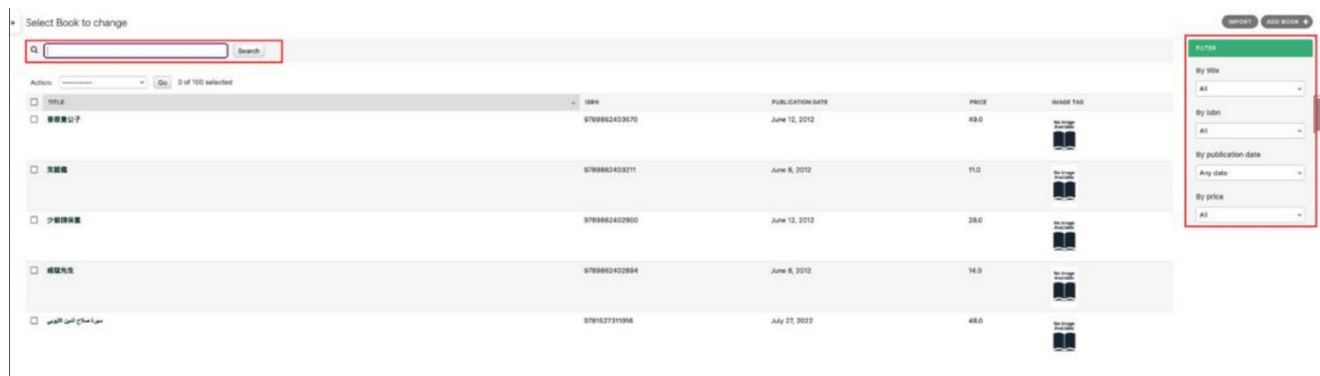


Рис. 15 Фільтрація

### 3. Панель адміністратора для адміністрування прав доступу:

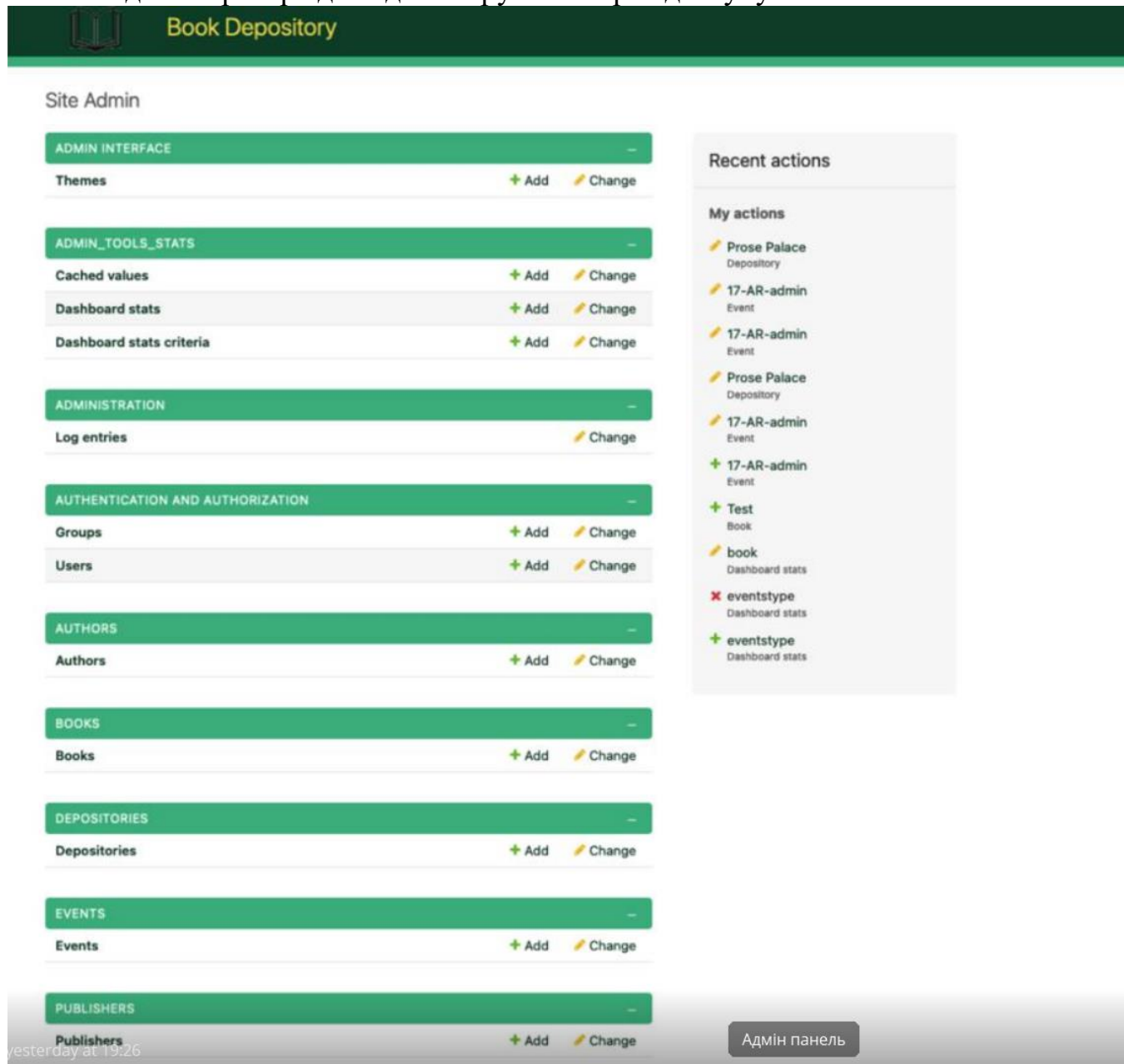


Рис. 16 Панель адміністратора

#### 4. Додавання книги бібліотекарем

Home > Books > Books > Add Book

**Add Book**

Title:

ISBN:

Publication date:  Today

Note: You are 3 hours ahead of server time.

Price:

Author: 

존 스튜어트  
Tahā Husayn  
سرهنگ نصرالله  
أبو المجدلين  
מאיר גולדברג  
Камиль Фламмерион  
Ирина Агапеева

Hold down "Control", or "Command" on a Mac, to select more than one.

Publisher:

Cover:  No file selected.

About:

Page count:

Rating:

Genres: 

science-fiction  
romance  
nonfiction  
mystery  
history  
fiction  
biography

Hold down "Control", or "Command" on a Mac, to select more than one.

Активация  
Перейдіть до р  
Windows.

Рис. 17 Сторінка додавання книги

#### Висновки

У результаті виконання лабораторної роботи було розроблено UML-діаграми до модуля пропозицій житла: use-case діаграма, діаграми послідовності, кооперації, активності, класів, об'єктів, станів, внутрішньої структури, компонентів та розгортання. Діаграми описують модуль з різних точок зору і допомагають візуалізувати компоненти системи.

Окрім цього було спроектовано та частково реалізовано інтерфейс користувача.