

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО”**

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**До захисту допущено:**

**Завідувач кафедри**

Сергій СТИРЕНКО

(підпис)

“ ” 2024 р.

**Дипломний проєкт**

**на здобуття ступеня бакалавра**

**за освітньо-професійною програмою “Інженерія програмного**

**забезпечення комп’ютерних систем”**

**спеціальності 121 “Інженерія програмного забезпечення”**

на тему: Сайт для продажу квитків до кінотеатру

Виконав : студент 4 курсу, групи ІП-04  
(шифр групи)

Гакавий Олександр Олександрович

(прізвище, ім'я, по батькові)

(підпис)

Керівник ас. Череватенко О.В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант (нормоконтроль) доцент, к.т.н. Волокита А. М.

(назва розділу)

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Рецензент доцент, к.т.н., Коган А. В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цьому дипломному  
проєкті немає запозичень з праць інших  
авторів без відповідних посилань.

Студент \_\_\_\_\_

(підпис)

Київ – 2024 р.

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО”**

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Рівень вищої освіти – перший (бакалавр)

Освітньо-професійна програма

“Інженерія програмного забезпечення комп’ютерних систем”

спеціальності 121 “Інженерія програмного забезпечення”

**ЗАТВЕРДЖУЮ**  
**Завідувач кафедри**  
Сергій СТИРЕНКО

\_\_\_\_\_  
(підпис)

“ \_ ” \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ**

на бакалаврський дипломний проєкт студента

Гакавого Олександра Олександровича

1. Тема проєкту Сайт для продажу квитків до кінотеатру  
керівник проєкту Череватенко Олексій Володимирович, ас.,  
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)  
затверджені наказом по університету від 27 травня 2024 року №2112-с
2. Термін здачі студентом закінченого проєкту 8 червня 2024 р.
3. Вихідні дані до проєкту технічне завдання, UI/UX дизайн, програмний код, база даних.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які розробляються)  
Розділ 1. Визначення вимог до програмного забезпечення на основі існуючих рішень.  
Розділ 2. Моделювання та конструювання програмного забезпечення.

Розділ 3. Деталі розробки системи.

Розділ 4. Представлення розробленої системи.

5. Перелік графічного матеріалу (з точним позначенням обов'язкових креслень) діаграма різновиду використання (структурна схема), діаграма даних (функціональна схема) (діаграма класів), алгоритм дій програмного забезпечення (принципова схема).

6. Консультанта проєкту, з вказівкою розділів проєкту, які до них вносяться

| Розділ        | Консультант    | Підпис, дата   |                  |
|---------------|----------------|----------------|------------------|
|               |                | Завдання видав | Завдання прийняв |
| Нормоконтроль | Волокита А. М. |                |                  |
|               |                |                |                  |

7. Дата видачі завдання «9» жовтня 2024 р.

Календарний план

| № п/п | Найменування етапів дипломного проєкту                     | Терміни виконання етапів проєкту | Примітки |
|-------|------------------------------------------------------------|----------------------------------|----------|
| 1.    | <i>Затвердження теми проєкту</i>                           | <i>6.11.2023-20.11.2023</i>      |          |
| 2.    | <i>Вивчення та аналіз завдання</i>                         | <i>20.11.2023-18.12.2023</i>     |          |
| 3.    | <i>Аналіз існуючих рішень з тематики роботи</i>            | <i>18.12.2023-08.01.2024</i>     |          |
| 4.    | <i>Розробка архітектури та загальної структури системи</i> | <i>08.01.2024-5.02.2024</i>      |          |
| 5.    | <i>Розробка дизайну застосунку</i>                         | <i>5.02.2024-04.03.2024</i>      |          |
| 6.    | <i>Програмна реалізація системи</i>                        | <i>04.03.2024-29.04.2024</i>     |          |
| 7.    | <i>Перевірка та налагодження системи</i>                   | <i>29.04.2024-13.05.2024</i>     |          |
| 6.    | <i>Оформлення пояснювальної записки</i>                    | <i>13.05.2024-27.05.2024</i>     |          |
| 7.    | <i>Захист програмного продукту</i>                         | <i>3.06.2024</i>                 |          |
| 8.    | <i>Передзахист</i>                                         | <i>7.06.2024</i>                 |          |
| 9.    | <i>Захист</i>                                              | <i>20.06.2024</i>                |          |

Студент-дипломник \_\_\_\_\_ Олександр ГАКАВИЙ  
(підпис)

Керівник проєкту \_\_\_\_\_ Олексій ЧЕРЕВАТЕНКО  
(підпис)

## АНОТАЦІЯ

Пояснювальна записка дослідницького проекту складається з чотирьох розділів, містить 44 рисунків, 8 таблиць, 3 додатків, 28 джерел та 88 сторінок.

Метою даного проекту є розробка конкурентоспроможного веб-сайту для кінотеатру Cinemax, який забезпечить зручний і ефективний спосіб купівлі квитків, перегляду розкладу сеансів та отримання інформації про фільми.

Для досягнення поставленої мети було проведено аналіз існуючих рішень на ринку веб-сайтів для продажу квитків, визначено функціональні та нефункціональні вимоги до програмного забезпечення, а також розроблено дизайн і архітектуру веб-сайту з використанням сучасних технологій веб-розробки.

Створено прототип веб-сайту, який відповідає визначеним вимогам, має інтуїтивно зрозумілий інтерфейс, забезпечує високу швидкість роботи та безпеку даних. Проведене тестування підтвердило зручність користування сайтом та його відповідність очікуванням користувачів.

Рекомендується впровадити розроблений веб-сайт для кінотеатру Cinemax з метою підвищення ефективності роботи кінотеатру, покращення якості обслуговування відвідувачів та збільшення продажів квитків. Впровадження даного веб-сайту дозволить зменшити навантаження на каси кінотеатру та забезпечити зручність для користувачів при купівлі квитків онлайн.

Ключові слова: веб-сайт, веб-застосунок, кінотеатр, купівля квитків онлайн, Next, React, Postgres, TypeScript.

## **ANNOTATION**

The explanatory note of the diploma project consists of four sections, contains 44 figures, 8 tables, 3 appendices, 28 sources and 88 pages.

The objective of this project is to develop a competitive website for the Cinemax cinema, providing a convenient and efficient way to purchase tickets, view showtimes, and obtain information about movies.

To achieve the goal, an analysis of existing solutions in the ticket sales website market was conducted, functional and non-functional requirements for the software were defined, and the design and architecture of the website were developed using modern web development technologies.

A prototype of the website was created that meets the defined requirements, has an intuitive interface, ensures high speed performance and data security. Testing confirmed the convenience of using the site and its compliance with user expectations.

It is recommended to implement the developed website for Cinemax cinema to improve the efficiency of the cinema, enhance the quality of customer service, and increase ticket sales. Implementing this website will reduce the load on the cinema's ticket offices and provide convenience for users when purchasing tickets online.

Keywords: website, web application, cinema, ticket purchase online, Next, React, Postgres, TypeScript.

| Справк | Формат | Значення            | Найменування             | Кіл.<br>листів | №<br>акземплія | Додаток |
|--------|--------|---------------------|--------------------------|----------------|----------------|---------|
|        |        |                     | Документація загальна    |                |                |         |
|        |        |                     | Знову розроблена         |                |                |         |
|        | A4     | ІАЛЦ.467200.002 ТЗ  | Сайт для продажу квитків | 4              |                |         |
|        |        |                     | до кінотеатру            |                |                |         |
|        |        |                     | Технічне завдання        |                |                |         |
|        | A4     | ІАЛЦ.467200.003 ПЗ  | Сайт для продажу квитків | 88             |                |         |
|        |        |                     | квитків до кінотеатру    |                |                |         |
|        |        |                     | Пояснювальна записка     |                |                |         |
|        | A4     | ІАЛЦ.467200.004 Д1  | Сайт для продажу квитків | 1              |                |         |
|        |        |                     | квитків до кінотеатру    |                |                |         |
|        |        |                     | Діаграма різновиду       |                |                |         |
|        |        |                     | використання             |                |                |         |
|        |        |                     | (Структурна схема)       |                |                |         |
|        | A4     | ІАЛЦ.4672008.005 Д2 | Сайт для продажу квитків | 1              |                |         |
|        |        |                     | квитків до кінотеатру    |                |                |         |
|        |        |                     | Діаграма даних           |                |                |         |
|        |        |                     | (Функціональна схема)    |                |                |         |
|        | A4     | ІАЛЦ.467200.006 ДЗ  | Сайт для продажу квитків | 1              |                |         |
|        |        |                     | квитків до кінотеатру    |                |                |         |
|        |        |                     | Алгоритм дій програмного |                |                |         |
|        |        |                     | забезпечення             |                |                |         |
|        |        |                     | (Принципова схема)       |                |                |         |
|        | A4     | ІАЛЦ.467200.007 Д4  | Сайт для продажу квитків | 39             |                |         |
|        |        |                     | квитків до кінотеатру    |                |                |         |
|        |        |                     | Текст програмного коду   |                |                |         |
|        |        |                     |                          |                |                |         |
|        |        |                     |                          |                |                |         |

|        |      |                      |      |      |                                                           |  |                                                   |       |
|--------|------|----------------------|------|------|-----------------------------------------------------------|--|---------------------------------------------------|-------|
|        |      |                      |      |      | <b>ІАЛЦ.467200.001 ОА</b>                                 |  |                                                   |       |
| Зм     | Лист | № докум.             | Підп | Дата |                                                           |  |                                                   |       |
| Розроб |      | Гакавий О. О.        |      |      | Сайт для продажу<br>квитків до кінотеатру<br>Опис альбому |  | Літ.                                              | Аркуш |
| Перев  |      | Череватенко<br>О. В. |      |      |                                                           |  | 1                                                 | 1     |
|        |      |                      |      |      |                                                           |  | <b>КПІ ім. Ігоря Сікорського,<br/>ФІОТ, ІП-04</b> |       |

**ТЕХНІЧНЕ ЗАВДАННЯ  
ДО ДИПЛОМНОГО ПРОЄКТУ**

на тему: «Сайт для продажу квитків до кінотеатру»

Київ – 2024

## ЗМІСТ

|                                           |   |
|-------------------------------------------|---|
| НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ..... | 2 |
| ПІДСТАВИ ДЛЯ РОЗРОБКИ.....                | 2 |
| МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ.....         | 2 |
| ДЖЕРЕЛА РОЗРОБКИ.....                     | 3 |
| ТЕХНІЧНІ ВИМОГИ.....                      | 3 |
| Вимоги до розробленого продукту.....      | 3 |
| Вимоги до програмного забезпечення.....   | 4 |
| Вимоги до апаратної частини.....          | 4 |
| ЕТАПИ РОЗРОБКИ.....                       | 4 |

|           |                   |          |        |      |                                                                    |  |  |  |                                           |       |         |   |  |
|-----------|-------------------|----------|--------|------|--------------------------------------------------------------------|--|--|--|-------------------------------------------|-------|---------|---|--|
|           |                   |          |        |      | ІАЛЦ.467200.002 ТЗ                                                 |  |  |  |                                           |       |         |   |  |
|           |                   |          |        |      |                                                                    |  |  |  |                                           |       |         |   |  |
| Зм.       | Арк.              | № докум. | Підпис | Дата | <b>Сайт для продажу квитків до кінотеатру</b><br>Технічне завдання |  |  |  | Літ.                                      | Аркуш | Аркушів |   |  |
| Розробив  | Гакавий О. О.     |          |        |      |                                                                    |  |  |  |                                           |       | 1       | 4 |  |
| Перевірив | Череватенко О. В. |          |        |      |                                                                    |  |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |   |  |
| Реценз.   | Коган А. В.       |          |        |      |                                                                    |  |  |  |                                           |       |         |   |  |
| Н. Контр. | Волокита А. М.    |          |        |      |                                                                    |  |  |  |                                           |       |         |   |  |
| Затвердив |                   |          |        |      |                                                                    |  |  |  |                                           |       |         |   |  |

# 1 НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ

Дане технічне завдання поширюється на розробку веб-сайту для кінотеатру Cinemax, що забезпечить зручний та ефективний спосіб купівлі квитків, перегляду розкладу сеансів та отримання інформації про фільми. Система також включає подальшу підтримку та вдосконалення розробленого веб-сайту.

Областю застосування цієї системи є обслуговування клієнтів кінотеатру Cinemax, надання інформації про фільми, розклади сеансів, а також забезпечення можливості онлайн-бронювання та купівлі квитків. Веб-сайт призначений для підвищення ефективності роботи кінотеатру, покращення якості обслуговування відвідувачів та збільшення продажів квитків. Він також може бути використаний для маркетингових та рекламних кампаній, збору та аналізу даних про користувачів, та для впровадження програм лояльності для постійних клієнтів.

## 2 ПІДСТАВИ ДЛЯ РОЗРОБКИ

Підставою для розробки даної системи є завдання для виконання роботи кваліфікаційно-освітнього рівня «бакалавр інженерії програмного забезпечення», який був затверджений факультетом “Інформатики та обчислювальної техніки” кафедрою обчислювальної техніки Національного технічного Університету України «Київський Політехнічний інститут ім. Ігоря Сікорського».

## 3 МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ

Метою є підвищення задоволеності відвідувачів кінотеатру за рахунок зручного та інтуїтивно зрозумілого інтерфейсу, а також покращення ефективності роботи кінотеатру через оптимізацію процесів продажу квитків та управління сеансами за рахунок розробки веб-сайту для кінотеатру Cinemax,

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.002 ТЗ | Арк. |
|     |      |          |        |      |                    | 2    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

який дозволить користувачам легко переглядати розклад сеансів, бронювати та купувати квитки онлайн, а також отримувати вичерпну інформацію про фільми

## 4 ДЖЕРЕЛА РОЗРОБКИ

Джерелом розробки даного дипломного проекту є офіційні документації, публікації та статті в мережі Інтернет на дану тему, науково-технічна література.

## 5 ТЕХНІЧНІ ВИМОГИ

### 5.1. Вимоги до розробленого продукту

Розроблений веб-сайт для кінотеатру Cinemax має відповідати таким вимогам:

- Проста та інтуїтивно зрозуміла навігація для користувачів.
- Надання актуальної інформації про розклад сеансів.
- Можливість бронювання та купівлі квитків онлайн з вибором місць на інтерактивній схемі зали.
- Інтеграція з популярними платіжними системами для забезпечення онлайн-оплати.
- Надання детальної інформації про фільми, включаючи описи, трейлери, відгуки та рейтинги.
- Зручна панель адміністрування для керування розкладом та фільмами.
- Зрозуміла документація для користувачів та адміністраторів.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.002 ТЗ | Арк. |
|     |      |          |        |      |                    | 3    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## 5.2. Вимоги до програмного забезпечення

- ОС Windows, Mac чи Linux.
- Мобільні пристрої на OS Android або IOS.
- Сучасний браузер (Chrome, Firefox, Safari, Edge) для роботи з веб-сайтом.

## 5.3. Вимоги до апаратної частини

- ЦП не менше ніж Intel® Core (TM) i3-2100T.
- RAM не менше ніж 4 ГБ.

## 6 ЕТАПИ РОЗРОБКИ

| Назва етапів виконання                                     | Термін виконання             |
|------------------------------------------------------------|------------------------------|
| <i>Затвердження теми проєкту</i>                           | <i>6.11.2023-20.11.2023</i>  |
| <i>Вивчення та аналіз завдання</i>                         | <i>20.11.2023-18.12.2023</i> |
| <i>Аналіз існуючих рішень з тематики роботи</i>            | <i>18.12.2023-08.01.2024</i> |
| <i>Розробка архітектури та загальної структури системи</i> | <i>08.01.2024-5.02.2024</i>  |
| <i>Розробка дизайну застосунку</i>                         | <i>5.02.2024-04.03.2024</i>  |
| <i>Програмна реалізація системи</i>                        | <i>04.03.2024-29.04.2024</i> |
| <i>Перевірка та налагодження системи</i>                   | <i>29.04.2024-13.05.2024</i> |
| <i>Оформлення пояснювальної записки</i>                    | <i>13.05.2024-27.05.2024</i> |

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.002 ТЗ | Арк. |
|     |      |          |        |      |                    | 4    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

**ПОЯСНЮВАЛЬНА ЗАПИСКА  
ДО ДИПЛОМНОГО ПРОЄКТУ**

на тему: «Сайт для продажу квитків до кінотеатру»

Київ – 2024

## ЗМІСТ

|                                                                                          |    |
|------------------------------------------------------------------------------------------|----|
| ПЕРЕЛІК СКОРОЧЕНЬ.....                                                                   | 4  |
| ВСТУП.....                                                                               | 5  |
| РОЗДІЛ 1. ВИЗНАЧЕННЯ ВИМОГ ДО ПРОГРАМНОГО<br>ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ ІСНУЮЧИХ РІШЕНЬ..... | 6  |
| 1.1. Загальні положення та опис предметної області.....                                  | 6  |
| 1.2. Огляд існуючих рішень.....                                                          | 7  |
| 1.2.1. Fandango.....                                                                     | 8  |
| 1.2.2. Movietickets.com.....                                                             | 8  |
| 1.2.3. BookMyShow .....                                                                  | 9  |
| 1.3. Порівняння функціональних можливостей.....                                          | 9  |
| 1.4. Визначення вимог до веб-сайту.....                                                  | 12 |
| 1.4.1. Функціональні вимоги.....                                                         | 12 |
| 1.4.2. Нефункціональні вимоги.....                                                       | 13 |
| ВИСНОВОК ДО РОЗДІЛУ 1.....                                                               | 15 |
| РОЗДІЛ 2. МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ<br>ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....                  | 16 |
| 2.1. Огляд технологій.....                                                               | 16 |
| 2.2. Архітектура системи.....                                                            | 19 |
| 2.2.1 Загальний опис.....                                                                | 19 |
| 2.2.2 Переваги на недоліки класичного підходу клієнт-сервер .....                        | 21 |
| 2.2.3 Переваги на недоліки використання Next.js.....                                     | 22 |
| 2.2.4 Обґрунтування вибору Next.js для проєкту.....                                      | 23 |
| 2.2.5 Визначення стратегії розгортання.....                                              | 24 |
| 2.3. Проектування бази даних.....                                                        | 25 |
| 2.3.1 Обґрунтування вибору Postgres.....                                                 | 25 |
| 2.3.2 Використання ORM Prisma.....                                                       | 26 |
| 2.3.3 Структура бази даних.....                                                          | 27 |

|           |      |                  |        |      |                                                                   |                                           |       |         |
|-----------|------|------------------|--------|------|-------------------------------------------------------------------|-------------------------------------------|-------|---------|
|           |      |                  |        |      | ІАЛЦ.467200.003 ПЗ                                                |                                           |       |         |
| Зм.       | Арк. | № докум.         | Підпис | Дата |                                                                   |                                           |       |         |
| Розробив  |      | Гакавий О. О.    |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Пояснювальна записка | Літ.                                      | Аркуш | Аркушів |
| Перевірів |      | Череватенко О.В. |        |      |                                                                   |                                           | 1     | 88      |
| Реценз.   |      | Коган А. В.      |        |      |                                                                   | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. |      | Волокита А. М.   |        |      |                                                                   |                                           |       |         |
| Затвердив |      |                  |        |      |                                                                   |                                           |       |         |

|                                                      |    |
|------------------------------------------------------|----|
| 2.4. Проектування користувацького інтерфейсу.....    | 28 |
| 2.4.1 Шапка сайту.....                               | 28 |
| 2.4.2. Головна сторінка.....                         | 29 |
| 2.4.3. Детальна сторінка фільму.....                 | 30 |
| 2.4.4. Сторінка оплати.....                          | 32 |
| 2.4.5. Сторінка про кінотеатр.....                   | 32 |
| 2.4.6. Сторінка зворотнього зв'язку.....             | 33 |
| 2.4.7. Особистий кабінет користувача.....            | 34 |
| ВИСНОВОК ДО РОЗДІЛУ 2.....                           | 36 |
| РОЗДІЛ 3. ДЕТАЛІ РОЗРОБКИ СИСТЕМИ.....               | 37 |
| 3.1. Структура проєкту.....                          | 37 |
| 3.2. Опис бази даних.....                            | 41 |
| 3.2.1. Таблиця Movie.....                            | 41 |
| 3.2.2. Таблиця Showtime.....                         | 42 |
| 3.2.3. Таблиця User.....                             | 43 |
| 3.2.4. Таблиця Review.....                           | 43 |
| 3.2.5. Таблиця Booking.....                          | 44 |
| 3.3. Розробка серверної частини.....                 | 45 |
| 3.4. Розробка клієнської частини.....                | 50 |
| 3.4.1. Сторінка з фільмами.....                      | 50 |
| 3.4.2. Детальна сторінка фільма.....                 | 54 |
| 3.4.3. Сторінка оплати.....                          | 60 |
| 3.4.4. Сторінка особистого кабінета користувача..... | 61 |
| 3.5. SEO-оптимізація.....                            | 62 |
| 3.5.1. Структуровані дані.....                       | 63 |
| 3.5.2. Заголовки та метадані.....                    | 63 |
| 3.5.3. Внутрішні посилання.....                      | 64 |
| 3.5.4. Оптимізація зображень.....                    | 65 |
| 3.5.5. Адаптивний дизайн.....                        | 66 |

|                                                  |    |
|--------------------------------------------------|----|
| ВИСНОВОК ДО РОЗДІЛУ 3.....                       | 68 |
| РОЗДІЛ 4. ПРЕДСТАВЛЕННЯ РОЗРОБЛЕНОЇ СИСТЕМИ..... | 69 |
| 4.1.Розгортання системи.....                     | 69 |
| 4.2.Демонстрація системи.....                    | 72 |
| ВИСНОВОК ДО РОЗДІЛУ 4.....                       | 82 |
| ВИСНОВКИ.....                                    | 83 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....                  | 85 |
| ДОДАТОК 1.....                                   |    |
| ДОДАТОК 2.....                                   |    |
| ДОДАТОК 3.....                                   |    |
| ДОДАТОК 4.....                                   |    |

## ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

|     |                                                                                                                                                                                                                                                         |
|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| API | (Application Programming Interface) Інтерфейс програмування застосунків                                                                                                                                                                                 |
| DB  | (Database) База даних                                                                                                                                                                                                                                   |
| SQL | (Structured Query Language) мова програмування, яка                                                                                                                                                                                                     |
| ORM | використовується для управління та маніпулювання даними в реляційних базах даних.<br>(Object-Relational Mapping) техніка програмування, яка дозволяє розробникам взаємодіяти з базою даних за допомогою об'єктів у коді, замість написання SQL-запитів. |
| SEO | (Search engine optimization) процес оптимізації веб-сайту для покращення його видимості в органічних результатах пошукових систем                                                                                                                       |
| URL | (Uniform Resource Locator) адреса, яка вказує на ресурс в Інтернеті                                                                                                                                                                                     |

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 4    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## ВСТУП

У сучасному світі кіноіндустрія є однією з найпопулярніших форм розваг, яка приваблює мільйони глядачів по всьому світу. Разом із розвитком технологій змінюються і способи взаємодії глядачів із кінотеатрами. Традиційні методи купівлі квитків у касах поступово відходять на другий план, поступаючись місцем електронним сервісам. Це зумовлено прагненням споживачів до зручності та економії часу, а також широкими можливостями, які надає Інтернет.

Кінотеатр Cinemax, прагнучи відповідати сучасним вимогам і покращити обслуговування своїх клієнтів, має на меті створення зручного та функціонального веб-сайту для продажу квитків. Цей проект спрямований на забезпечення користувачам можливості швидко та легко обирати фільми, бронювати місця в залі, здійснювати оплату онлайн і отримувати електронні квитки.

Розробка веб-сайту для кінотеатру Cinemax включає кілька ключових етапів: аналіз існуючих рішень на ринку, визначення функціональних та нефункціональних вимог, проектування інтерфейсу, реалізація програмного забезпечення та тестування готового продукту.

У вступній частині пояснювальної записки висвітлено актуальність обраної теми, мету та основні завдання дослідження. Подальші розділи містять детальний опис етапів розробки веб-сайту, технічні аспекти реалізації проекту, а також аналіз отриманих результатів і висновки щодо ефективності створеного продукту.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 5    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

# РОЗДІЛ 1. ВИЗНАЧЕННЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ ІСНУЮЧИХ РІШЕНЬ

## 1.1. Загальні положення та опис предметної області

Завдання: Аналіз ринку веб-сайтів для продажу квитків, визначення найкращих практик, визначення функціональних та нефункціональних вимог до програмного забезпечення.

Очікувані результати: Створення конкурентоспроможного веб-сайту для кінотеатру Cinemax, який задовольнить потреби користувачів та підвищить ефективність роботи кінотеатру.

Кінотеатр Cinemax - сучасний кінотеатр, який прагне забезпечити своїх відвідувачів якісним сервісом та зручними умовами для перегляду фільмів. Основними вимогами до веб-сайту кінотеатру є надання інформації про розклад сеансів, можливість онлайн-бронювання та купівлі квитків, а також забезпечення зручного користувацького досвіду.

Ключові аспекти предметної області:

- Користувачі: Відвідувачі кінотеатру, які шукають інформацію про сеанси, бажають придбати квитки онлайн, залишити відгуки про фільми.
- Адміністратори: Працівники кінотеатру, які керують списком фільмів та розкладом сеансів, цінами на квитки.
- Фільми: Опис, трейлери, відгуки, рейтинги фільмів, що показуються в кінотеатрі.
- Сеанси: Час і дата показу фільмів, доступні місця у залі.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 6    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Квитки: Можливість вибору місць, оплата онлайн.

Додатково, важливо врахувати технічні аспекти розробки веб-сайту, такі як забезпечення високої продуктивності і швидкості завантаження, підтримка різних пристроїв та екранів, а також дотримання стандартів безпеки для захисту персональних даних користувачів. Інтеграція з популярними платіжними системами та можливість зручного адміністрування сайту з боку працівників кінотеатру також є критично важливими аспектами, які забезпечать ефективну роботу веб-сайту та задоволення потреб усіх користувачів.

## 1.2. Огляд існуючих рішень

У сучасних умовах швидкого розвитку технологій ринок веб-сайтів для продажу квитків до кінотеатрів пропонує безліч рішень, які надають користувачам широкий спектр послуг.

Аналіз цих рішень допоможе визначити найкращі практики і уникнути можливих недоліків при розробці веб-сайту для кінотеатру Cinemax. Це дозволить створити веб-сайт, який буде ефективно задовольняти потреби користувачів та підвищить рівень обслуговування у кінотеатрі.

Дослідження популярних платформ також допоможе визначити ключові функціональні та нефункціональні вимоги до майбутнього сайту. Це включає забезпечення безпеки даних користувачів, надійність системи бронювання та зручність користування.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 7    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

### 1.2.1. Fandango

Fandango є одним із провідних веб-сайтів для продажу кінотеатральних квитків у США. Основні функціональні можливості включають:

- Розклад сеансів: Користувачі можуть переглядати розклад кіносеансів у різних кінотеатрах, обираючи зручне місце та час.
- Інтерактивний вибір місць: Система дозволяє вибирати місця на інтерактивній схемі зали.
- Оплата онлайн: Інтеграція з різними платіжними системами, такими як кредитні картки та PayPal.
- Рецензії та рейтинги: Можливість перегляду трейлерів, читання рецензій та оцінок фільмів.
- Мобільний додаток: Доступність усіх функцій через мобільний додаток для iOS та Android.

### 1.2.2. Movietickets.com

Movietickets.com пропонує аналогічний набір функцій, забезпечуючи користувачам зручний і ефективний спосіб купівлі квитків онлайн:

- Пошук фільмів та кінотеатрів: Можливість швидко знаходити фільми та найближчі кінотеатри за допомогою зручного інтерфейсу.
- Купівля квитків: Онлайн-покупка з можливістю вибору місць у залі.
- Інформація про фільми: Опис фільмів, трейлери, рецензії та рейтинги.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 8    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Акції та знижки: Спеціальні пропозиції для постійних користувачів.
- Підтримка різних платформ: Адаптований дизайн для ПК, планшетів та смартфонів.

### 1.2.3. BookMyShow

BookMyShow, популярний в Індії, надає додаткові можливості:

- Продаж квитків на інші події: Окрім кіно, квитки на концерти, спортивні заходи та театральні вистави.
- Персоналізовані рекомендації: Система рекомендацій на основі історії переглядів і покупок користувача.
- Соціальні функції: Можливість ділитися інформацією про куплені квитки у соціальних мережах.
- Підтримка кількох мов: Інтерфейс доступний на різних мовах для зручності користувачів.

## 1.3. Порівняння функціональних можливостей

Для кінотеатрів, що не є частиною великих мереж, такі як Cinemax, можна виділити кілька ключових функціональних можливостей, які важливо врахувати при розробці веб-сайту:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 9    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Таблиця 1.1 — Порівняння функціональних можливостей сайтів для продажу квитків до кінотеатру

| Критерій              | Fandango                                           | Movietickets.com                                     | BookMyShow                                         |
|-----------------------|----------------------------------------------------|------------------------------------------------------|----------------------------------------------------|
| Розклад сеансів       | Перегляд розкладу кіносеансів у різних кінотеатрах | Перегляд розкладу кіносеансів у різних кінотеатрах   | Перегляд розкладу кіносеансів у різних кінотеатрах |
| Вибір місць           | Інтерактивний вибір місць                          | Інтерактивний вибір місць                            | Інтерактивний вибір місць                          |
| Оплата онлайн         | Кредитні картки, PayPal                            | Кредитні картки, PayPal                              | Кредитні картки, цифрові гаманці, Інтернет-банкінг |
| Інформація про фільми | Опис, трейлери, рецензії, рейтинги                 | Опис, трейлери, рецензії, рейтинги                   | Опис, трейлери, рецензії, рейтинги                 |
| Мобільний додаток     | iOS, Android                                       | Немає, але адаптивний дизайн для мобільних пристроїв | iOS, Android                                       |
| Програми лояльності   | Спеціальні пропозиції                              | Спеціальні пропозиції                                | Програми лояльності, акції та знижки               |

Продовження таблиці 1.1

| Критерій                     | Fandango                                                 | Movietickets.com                                         | BookMyShow                                                                 |
|------------------------------|----------------------------------------------------------|----------------------------------------------------------|----------------------------------------------------------------------------|
| Персоналізовані рекомендації | Немає                                                    | Немає                                                    | Є, на основі історії переглядів і покупок користувача                      |
| Продаж квитків на інші події | Немає                                                    | Немає                                                    | Є, на концерти, спортивні заходи, театральні вистави                       |
| Підтримка кількох мов        | Немає                                                    | Немає                                                    | Є, інтерфейс доступний на різних мовах                                     |
| Соціальні функції            | Немає                                                    | Немає                                                    | Є, можливість ділитися інформацією про куплені квитки у соціальних мережах |
| Адміністративні функції      | Панель адміністрування для керування розкладом та цінами | Панель адміністрування для керування розкладом та цінами | Панель адміністрування для керування розкладом, цінами та акціями          |

Кінець таблиці 1.1

| Критерій                     | Fandango                         | Movietickets.com                 | BookMyShow                       |
|------------------------------|----------------------------------|----------------------------------|----------------------------------|
| Безпека                      | SSL-сертифікати,<br>захист даних | SSL-сертифікати,<br>захист даних | SSL-сертифікати,<br>захист даних |
| Швидкість<br>завантаження    | Висока                           | Висока                           | Висока                           |
| Підтримка<br>різних платформ | ПК, мобільні<br>пристрої         | ПК, мобільні<br>пристрої         | ПК, мобільні<br>пристрої         |

Ця таблиця порівнює основні функціональні можливості трьох популярних сервісів для продажу квитків, що допоможе краще зрозуміти їхні особливості, переваги та недоліки. Це, в свою чергу, сприятиме визначенню найкращих практик при розробці веб-сайту для кінотеатру Cinemax.

## 1.4. Визначення вимог до веб-сайту

### 1.4.1. Функціональні вимоги

Розклад і бронювання:

- Відображення актуального розкладу сеансів.
- Можливість вибору та бронювання місць на інтерактивній схемі зали.

Оплата:

- Інтеграція з популярними платіжними системами (кредитні картки, PayPal, інші локальні платіжні методи).
- Забезпечення безпечних транзакцій з використанням SSL-сертифікатів.

Інформація про фільми:

- Опис, трейлери, відгуки та рейтинги фільмів.
- Можливість залишати відгуки після перегляду.

Користувацький інтерфейс:

- Інтуїтивно зрозумілий та адаптивний дизайн.
- Простота та логічність навігації.

Адміністративні функції:

- Панель адміністратора для керування розкладом сеансів, фільмами цінами на квитки.
- Генерація звітів про продаж квитків.

#### 1.4.2. Нефункціональні вимоги

Продуктивність:

- Швидке завантаження сторінок.
- Висока пропускна здатність для обробки великої кількості одночасних запитів.

Надійність:

- Висока доступність та мінімальні простої.

Масштабованість:

- Можливість розширення функціоналу без значних змін в архітектурі сайту.
- Підтримка збільшення кількості користувачів та обсягу даних.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 3    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Зручність обслуговування:

- Легка модифікація та оновлення системи.
- Документованість коду та зрозуміла структура проекту.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 4    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## ВИСНОВОК ДО РОЗДІЛУ 1

Аналіз існуючих рішень показує, що успішний веб-сайт для продажу квитків до кінотеатру Сінемах повинен забезпечувати зручний і інтуїтивно зрозумілий інтерфейс, широкий функціонал для користувачів та адміністраторів, високу швидкість роботи та безпеку даних. Впровадження зазначених вимог дозволить створити конкурентоспроможний продукт, який задовольнить потреби сучасних користувачів та підвищить ефективність роботи кінотеатру. У наступних розділах буде детально описано етапи розробки веб-сайту, його архітектуру, дизайн та технічні аспекти реалізації.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 5    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## РОЗДІЛ 2. МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.

### 2.1. Огляд технологій

Для реалізації веб-сайту для продажу квитків до кінотеатру Cinemax були обрані наступні технології:

Frontend:

- HTML: Використовується для структурування веб-сторінок.
- CSS: Використовується для стилізації та оформлення веб-сторінок.
- JavaScript: Основна мова програмування для взаємодії з користувачем та динамічної роботи веб-сторінок.
- TypeScript: Надбудова над JavaScript, яка додає статичну типізацію, покращуючи надійність коду.
- React: Бібліотека для побудови користувацьких інтерфейсів, що дозволяє створювати компоненти та керувати станом додатку.
- Next.js: Фреймворк для React, що забезпечує серверний рендеринг, генерацію статичних сторінок та створення API, що спрощує розробку повнофункціональних додатків без необхідності написання окремого серверу.
- Tailwind CSS: Утилітарний CSS фреймворк для швидкого та гнучкого створення сучасних користувацьких інтерфейсів.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 6    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

#### Backend:

- Next.js Server Components: Використовуючи Server Components, ви можете забезпечити швидку та ефективну обробку запитів на сервері, що сприяє покращенню продуктивності вашого веб-сайту та зменшенню часу відповіді сервера.

#### DB:

- PostgreSQL: Реляційна база даних, що забезпечує високу продуктивність, масштабованість та надійність зберігання даних.
- Prisma: ORM (Object-Relational Mapping) для роботи з базами даних, що забезпечує зручний інтерфейс для взаємодії з базою даних та генерацію типів для TypeScript.

#### Аутентифікація:

- Auth0: Сервіс для управління автентифікацією та авторизацією, що забезпечує безпеку та зручність у використанні.

#### Контроль версій:

- Git: Система контролю версій, що дозволяє відслідковувати зміни у коді та працювати в команді.
- GitHub: Платформа для розміщення репозиторіїв та співпраці над кодом.

#### Інструменти розробки:

- Visual Studio Code: Популярний редактор коду з розширеннями, що підтримують різні мови програмування та інструменти розробки.
- Figma: Інструмент для проєктування користувацьких інтерфейсів, що дозволяє створювати макети та прототипи веб-сторінок, забезпечуючи зручну колаборацію між дизайнерами та розробниками.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 7    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Хостинг та деплоймент:

- Vercel: Платформа для розгортання Next.js додатків, що забезпечує швидкий та зручний деплоймент з інтеграцією з GitHub, автоматичним масштабуванням та іншими можливостями для продуктивного хостингу.

Обрані технології спрямовані на створення сучасного та продуктивного веб-сайту для кінотеатру Cinemax, який забезпечить зручність у використанні для користувачів та ефективність у розробці для команди. Давайте розглянемо кожен з них більш детально:

У фронтенді HTML, CSS та JavaScript використовуються для створення структури, оформлення та динамічної роботи веб-сторінок, а TypeScript допомагає підвищити надійність коду за рахунок статичної типізації. Бібліотека React дозволяє ефективно організувати код за допомогою компонентів, а Next.js надає потужні можливості, такі як серверний рендеринг та генерація статичних сторінок, що полегшує розробку та оптимізацію веб-додатків.

Додатково, використання Tailwind CSS дозволяє швидко та гнучко створювати сучасний та привабливий дизайн інтерфейсу, зменшуючи кількість написаного CSS коду.

У бекенді, Next.js Server Components забезпечують ефективну обробку запитів на сервері, що сприяє покращенню продуктивності та зменшенню часу відповіді сервера. Це дозволяє створювати високопродуктивні веб-додатки з гнучким серверним API.

Для зберігання та роботи з даними обрано PostgreSQL як надійну та продуктивну реляційну базу даних. Prisma, як ORM, дозволяє зручно взаємодіяти з базою даних та автоматично генерує типи для TypeScript, полегшуючи розробку та підтримку коду.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 8    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Аутентифікація та авторизація користувачів здійснюється за допомогою Auth0, що забезпечує безпеку та зручність у використанні. Контроль версій коду здійснюється за допомогою системи Git та платформи GitHub, що дозволяє ефективно вести розробку та співпрацювати над кодом у команді.

У відділі інструментів розробки використовуються популярний редактор коду Visual Studio Code для зручної роботи над проектом та Figma для проєктування та прототипування користувацьких інтерфейсів, що забезпечує ефективну колаборацію між дизайнерами та розробниками.

Завдяки хостингу та деплоюменту на платформі Vercel, веб-сайт буде легко розгортатися та масштабуватися, забезпечуючи швидкий та надійний доступ до нього для користувачів.

Використання цих технологій дозволить створити сучасний, продуктивний та зручний у використанні веб-сайт для кінотеатру Cinemax, який відповідатиме усім вимогам сучасного ринку та користувачів.

## 2.2. Архітектура системи

### 2.2.1 Загальний опис

Архітектура додатків на основі Next.js має кілька ключових особливостей. Перш за все, Next.js дозволяє розробникам працювати як з фронтендом, так і з бекендом на одній платформі. Це спрощує процес розробки та підтримки додатків, оскільки весь код можна писати в одній і тій же технології.

Новий підхід до розробки веб-додатків - React Server Components - дозволяє розбивати інтерфейс на компоненти, які рендеряться на сервері. Це покращує швидкодію та SEO-оптимізацію додатків, а також забезпечує кращий контроль над ресурсами.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 9    |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Що стосується взаємодії з базою даних, Next.js може легко інтегруватися з різними базами даних за допомогою ORM, такого як Prisma. Це дозволяє отримати доступ до даних у фронтенді та бекенді, а також реалізувати складні логічні операції з даними.

Next.js також дозволяє легко інтегрувати різні сервіси для аутентифікації та авторизації користувачів. Наприклад, Auth0 - це популярний сервіс для управління ідентифікацією та забезпечення безпеки користувачів, який можна легко інтегрувати з Next.js додатками.

Нарешті, Next.js дозволяє легко інтегрувати платіжні системи для здійснення транзакцій через веб-додаток. Це забезпечує зручну та безпечну оплату для користувачів. Загалом, архітектура Next.js додатків дозволяє розробникам створювати сучасні, швидкі та безпечні додатки, які забезпечують якісний користувацький досвід.

Архітектура веб-сайту буде базуватися на моделі єдиної платформи, яка включає як фронтенд, так і бекенд.

Основні компоненти системи:

Інтерфейс користувача (Frontend):

- Відображення інтерфейсу користувача.
- Взаємодія з серверними функціями через React Server Components.
- Обробка введених користувачем даних.

Серверна логіка (Backend):

- Обробка запитів від клієнта через React Server Components.
- Взаємодія з базою даних.
- Логіка аутентифікації та авторизації через Auth0.
- Проведення транзакцій через платіжні системи.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 20   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

База даних (DB):

- Зберігання інформації про фільми, сеанси, користувачів, бронювання.

### 2.2.2 Переваги на недоліки класичного підходу клієнт-сервер

Таблиця 2.1 — Переваги та недоліки класичного підходу до розділення клієнта та сервера

| Переваги                                                                                                                                                 | Недоліки                                                                                                                                   |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| Чітке відокремлення фронтенду та бекенду, що спрощує розробку та робить код більш організованим та керованим.                                            | Потреба у додатковому часі та зусиллях на встановлення, конфігурацію та підтримку двох окремих середовищ (фронтенду та бекенду).           |
| Більша контрольованість та гнучкість у виборі архітектури та технологій, що дозволяє розробникам вибирати найкращі інструменти для конкретного завдання. | Потенційна неузгодженість архітектури та інструментів між клієнтом та сервером, що може призвести до проблем зі сумісністю та інтеграцією. |
| Чітке визначення обов'язків між клієнтською та серверною частиною додатку, що сприяє більшій модульності та масштабованості.                             | Потенційна складність у взаємодії та синхронізації роботи клієнтської та серверної частини додатку, зокрема при роботі в команді.          |

### Кінець таблиці 2.1

| Переваги                                                                                                                                | Недоліки                                                                                                                          |
|-----------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Більша гнучкість у розподілі навантаження між клієнтом та сервером, що може призвести до оптимізації продуктивності та масштабованості. | Потреба у встановленні, налаштуванні та підтримці двох окремих серверів та середовищ для фронтенду та бекенду.                    |
| Більша стабільність та підтримка, оскільки кожна частина системи може бути розроблена, тестована та підтримувана окремо.                | Потенційна складність у взаємодії та синхронізації роботи клієнтської та серверної частини додатку, зокрема при роботі в команді. |

Цей підхід може бути корисним для великих або складних проєктів, де важлива гнучкість і контроль над архітектурою, але водночас може вимагати більших зусиль на його налагодження та підтримку

### 2.2.3 Переваги на недоліки використання Next.js

Таблиця 2.2 — Переваги та недоліки використання Next.js

| Переваги                                                                                      | Недоліки                                                                     |
|-----------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| Розробка як фронтенду, так і бекенду на одній платформі спрощує процес розробки та підтримки. | Потребує додаткового часу на оволодіння новими концепціями та інструментами. |

Кінець таблиці 2.2

| Переваги                                                                                                         | Недоліки                                                                                          |
|------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| Вбудовані засоби для оптимізації швидкості завантаження сторінок, що полегшує створення продуктивних додатків.   | Можливі обмеження з погляду розширюваності або специфічних функцій, які не підтримуються.         |
| Простий та зрозумілий API дозволяє швидко розробляти нові функції та вносити зміни.                              | Залежність від фреймворку може створювати проблеми з міграцією або підтримкою у майбутньому.      |
| Підтримка як серверного, так і клієнтського рендерінгу дозволяє ефективно керувати процесом рендерінгу сторінок. | Деякі функції можуть вимагати додаткової настройки або специфічних умов для правильної роботи.    |
| Зручний інструмент для швидкої розробки та тестування нових ідей без великих затрат часу.                        | Велика швидкість розробки може призвести до необдуманих рішень або недоліків у структурі проекту. |

Ці переваги та недоліки варто розглядати при виборі Next.js для конкретного проєкту, а також при оцінці його придатності та прийнятті рішення щодо архітектури та інструментів розробки.

#### 2.2.4 Обґрунтування вибору Next.js для проєкту

Використання Next.js для цього проєкту має ряд переваг, які роблять його привабливим вибором. По-перше, Next.js дозволяє розробляти як

фронтенд, так і бекенд на одній платформі, що спрощує процес розробки та підтримки проєкту. Це означає, що розробники можуть працювати з однією технологією і не потребується інтеграція різних фреймворків або бібліотек.

По-друге, Next.js має вбудовані засоби для оптимізації швидкості завантаження сторінок, такі як серверний рендерінг і статична генерація. Це дозволяє створювати ефективні додатки з високою продуктивністю, що є важливим аспектом для веб-сайту з продажу квитків до кінотеатру, де швидкість і реактивність важливі для користувачів.

Крім того, простий API Next.js спрощує розробку та масштабування проєкту. Розробники можуть швидко створювати нові функції та вносити зміни без необхідності вивчати нові технології або інтегрувати додаткові бібліотеки. За рахунок цього, розробка стає більш продуктивною і ефективною, що важливо для успішного впровадження та підтримки веб-сайту.

### 2.2.5 Визначення стратегії розгортання

Для розгортання нашого сайту для продажу квитків до кінотеатру, розробленого з використанням Next.js та бази даних PostgreSQL, ми вирішили скористатися платформою Vercel.

Vercel - це відома платформа для розгортання веб-додатків, яка надає простий та швидкий спосіб розгортання додатків. Однією з його переваг є інтеграція з Next.js, що дозволяє автоматизувати процес розгортання та підтримки Next.js додатків.

Vercel забезпечує швидке розгортання без необхідності налаштування серверів або інфраструктури, що робить його ідеальним вибором для нашого

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 24   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

проекту. Крім того, Vercel пропонує безкоштовний тарифний план для невеликих проектів, що дозволяє нам економити витрати на інфраструктуру.

Ще одна перевага Vercel - це простота масштабування. Ми можемо легко масштабувати наш сайт в разі зростання трафіку або потреб користувачів, просто збільшуючи ресурси через інтерфейс керування Vercel.

Отже, обираючи Vercel для розгортання нашого сайту, ми отримаємо швидке, просте та надійне рішення, що дозволить нам ефективно розвивати та підтримувати наш веб-сайт для продажу квитків до кінотеатру.

## 2.3. Проектування бази даних

### 2.3.1 Обґрунтування вибору Postgres

PostgreSQL є найкращим варіантом для нашого сайту з продажу квитків до кінотеатру, особливо з урахуванням того, що ми плануємо використовувати Next.js для розробки та Vercel для розгортання.

PostgreSQL відомий своєю надійністю та стабільністю. Це важливо для веб-сайту з продажу квитків, де потрібна постійна доступність та низька ймовірність втрати даних.

PostgreSQL відмінно працює з великими обсягами даних, що може бути важливим для нашого сайту, особливо коли розглядаються інформація про фільми, розклад сеансів та історія покупок.

PostgreSQL добре інтегрується з платформами хостингу, такими як Vercel. Це означає, що ми можемо легко налаштувати базу даних для нашого проекту на Vercel без проблем з сумісністю чи розгортанням.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 25   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

PostgreSQL має широкі можливості для розширення, що дозволяє нам зручно масштабувати наш сайт у майбутньому при зростанні обсягів даних та потоку користувачів.

PostgreSQL забезпечує високий рівень безпеки для наших даних, зокрема шифрування та механізми автентифікації. Це важливо для збереження конфіденційності та цілісності даних нашого сайту.

Отже, враховуючи всі ці фактори, використання PostgreSQL є оптимальним вибором для нашого проекту на Next.js та Vercel.

### 2.3.2 Використання ORM Prisma

Для зручності роботи з базою даних PostgreSQL ми вирішили використовувати ORM (Object-Relational Mapping) Prisma. Ось кілька причин, чому ми обрали саме цей інструмент:

Простота використання: Prisma надає зручний інтерфейс для взаємодії з базою даних через зрозумілі моделі та запити. Це дозволяє розробникам швидко розробляти та підтримувати код.

Статична типізація: Prisma підтримує статичну типізацію, що дозволяє виявляти помилки в коді на етапі компіляції. Це допомагає уникнути потенційних проблем з даними та забезпечує надійність програмного забезпечення.

Генерація SQL запитів: Prisma автоматично генерує SQL запити на основі виразів API, що зменшує ризик виникнення помилок та спрощує роботу з базою даних.

Міграції: Prisma надає вбудовані засоби для керування міграціями бази даних. Це дозволяє легко оновлювати схему бази даних та зберігати її у відповідності до вимог додатку.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 26   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Інтеграція з TypeScript: Prisma підтримує TypeScript, що дозволяє розробникам отримати переваги статичної типізації та автодоповнення під час роботи з моделями даних.

Загальною метою використання ORM Prisma є спрощення та полегшення взаємодії з базою даних PostgreSQL, що сприяє ефективнішій та надійній розробці веб-сайту для кінотеатру Cinemax.

### 2.3.3 Структура бази даних

Основні таблиці:

- Users: Інформацію про користувачів (ім'я, email, зображення).
- Movies: Дані про фільми (назва, опис, рік, тривалість, жанри, продюсери, рейтинг, кількість оцінок, зображення, трейлер).
- Showtimes: Інформація про сеанси (фільм, дата, час, доступні місця).
- Bookings: Інформація про бронювання (користувач, сеанс, місця, статус оплати).
- Reviews: Відгуки користувачів про фільми (користувач, фільм, оцінка, коментар).

Зв'язки:

- Користувач може мати кілька бронювань та відгуків.
- Відгук пов'язаний тільки з одним користувачем та одним фільмом.
- Фільм може мати кілька відгуків та сеансів.
- Сеанс пов'язаний тільки з одним фільмом та може бути пов'язаний тільки з одним бронюванням.
- Бронювання пов'язаний тільки з одним сеансом та тільки одним користувачем.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 27   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Така структура (див. Додаток 2) дозволяє зручно зберігати та маніпулювати даними, забезпечуючи гнучкість та масштабованість системи.

## 2.4. Проектування користувацького інтерфейсу

Проектування користувацького інтерфейсу є ключовим етапом у створенні веб-сайту для продажу квитків до кінотеатру Cinemax. Важливо створити інтерфейс, який буде не тільки привабливим та сучасним, але й зручним у використанні для користувачів. Основні компоненти інтерфейсу розроблені з урахуванням цих принципів.

### 2.4.1 Шапка сайту

Шапка сайту містить логотип, який робить бренд кінотеатру впізнаваним, навігаційне меню для швидкого доступу до різних розділів сайту та поле пошуку фільмів, що дозволяє користувачам швидко знаходити потрібні фільми.



Рисунок 2.1 — Дизайн шапки сайту

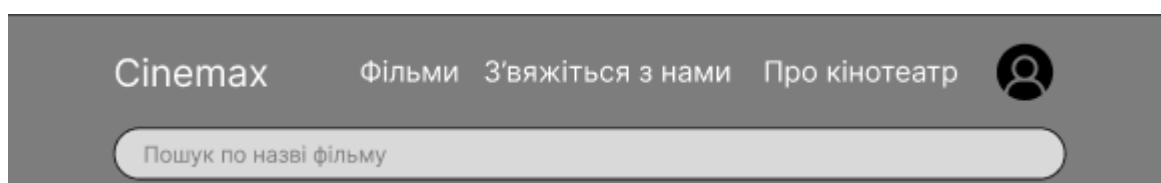


Рисунок 2.2 — Дизайн шапки сайту (версія для планшетів)

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 28   |

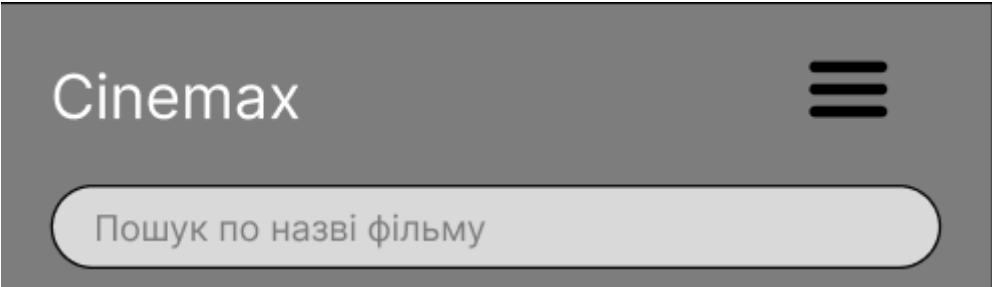


Рисунок 2.3 — Дизайн шапки сайту(мобільна версія)

2.4.2. Головна сторінка

Головна сторінка відображає список доступних фільмів з коротким описом, що дозволяє користувачам швидко ознайомитися з асортиментом кінотеатру. Тут також доступна можливість фільтрування фільмів за жанрами а також пагінація в кінці сторінки.

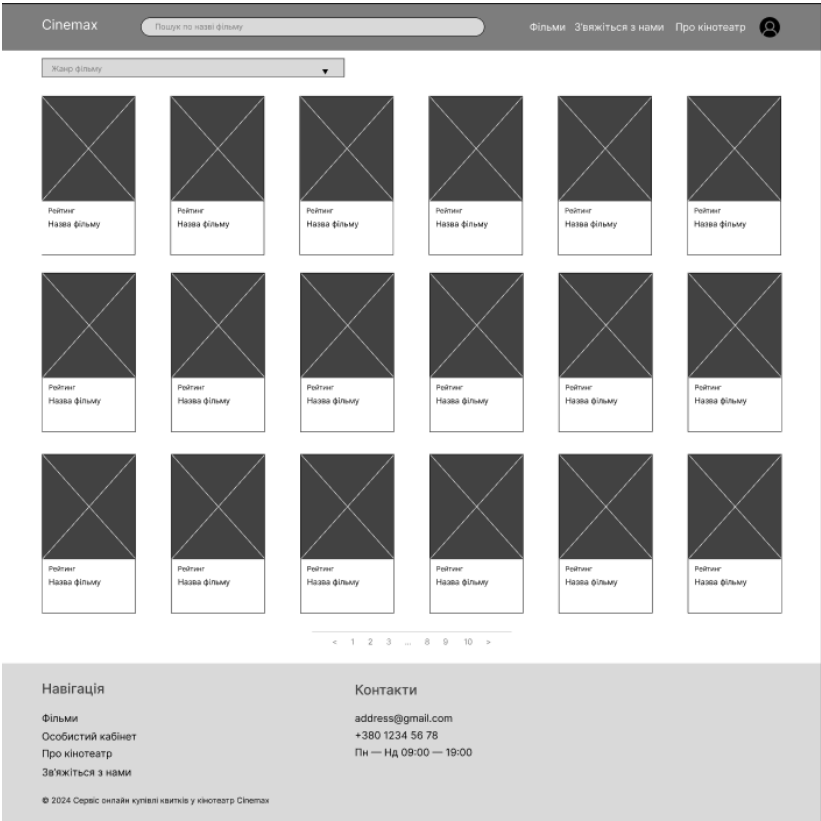


Рисунок 2.4  
— Дизайн

головної сторінки

2.4.3. Детальна сторінка фільму

Сторінка фільму містить детальну інформацію про обраний фільм, включаючи трейлер, розклад сеансів, що дозволяє користувачам отримати повну картину про фільм та зручно спланувати візит до кінотеатру.

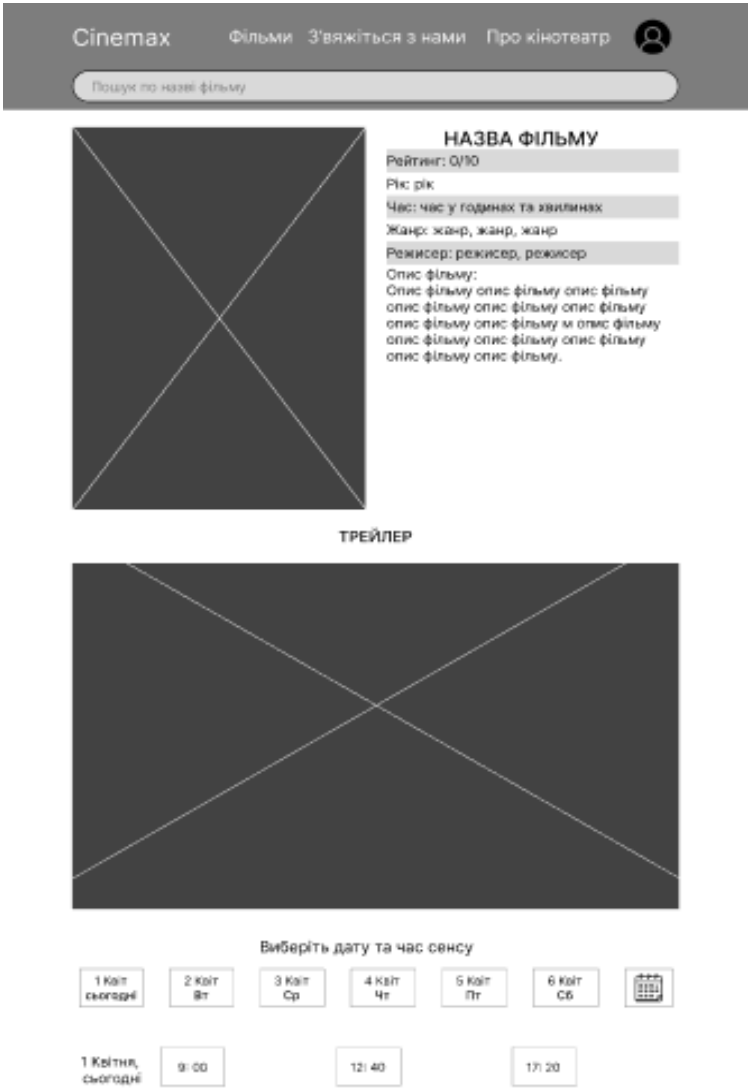


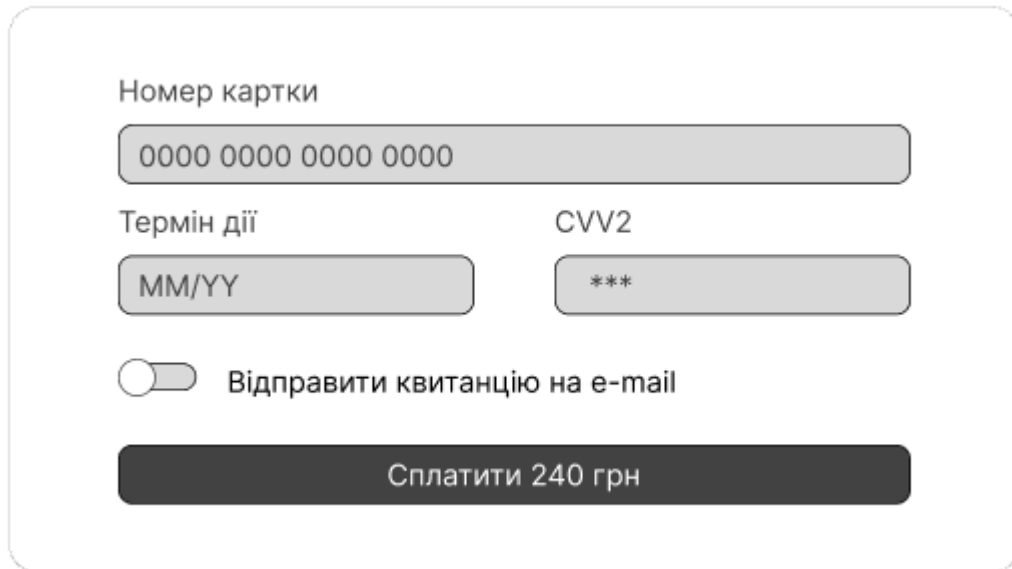
Рисунок 2.5 — Дизайн детальної сторінки фільма

А також в кінці сторінки розташована форма для надсилання відгуків про фільм та список відгуків від інших користувачів.



#### 2.4.4. Сторінка оплати

Сторінка оплати містить форму для введення платіжних даних та підтвердження бронювання, забезпечуючи зручний та безпечний процес оплати для користувачів.



The image shows a payment form with the following elements:

- Номер картки**: A text input field containing "0000 0000 0000 0000".
- Термін дії**: A text input field containing "MM/YY".
- CVV2**: A text input field containing "\*\*\*".
- Відправити квитанцію на e-mail**: A toggle switch that is currently turned off.
- Сплатити 240 грн**: A large, dark button with white text.

Рисунок 2.8 — Дизайн форми для оплати

#### 2.4.5. Сторінка про кінотеатр

Сторінка про кінотеатр містить детальну інформацію про сам кінотеатр, його історію, розташування та інші цікаві факти, що дозволяють користувачам краще ознайомитися з місцем, куди вони збираються.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 32   |

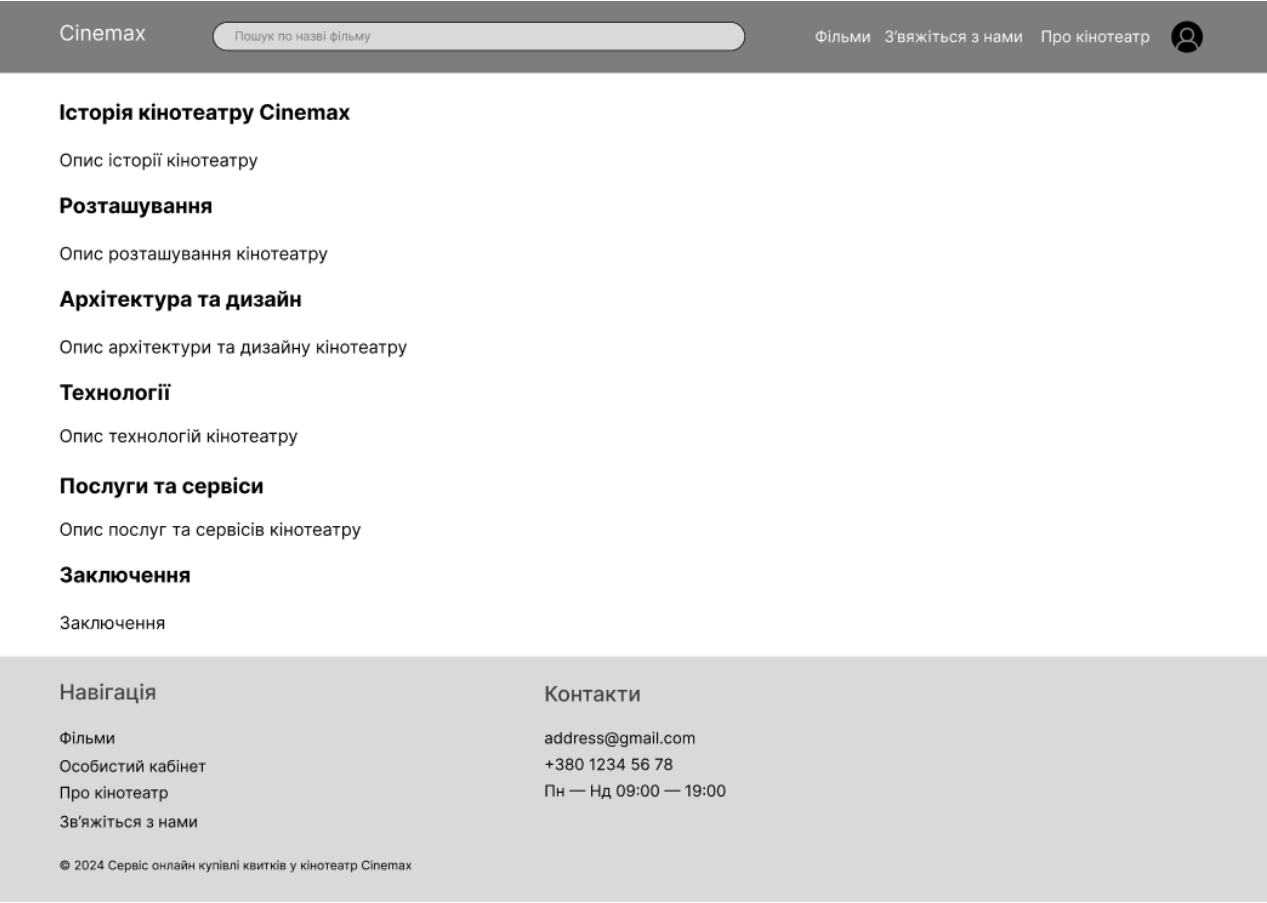


Рисунок 2.9 — Дизайн сторінки про кінотеатр

2.4.6. Сторінка зворотнього зв’язку

Сторінка зворотнього зв’язку надає контактні дані кінотеатру, такі як номер телефону, адреса та електронна пошта, а також форму для зворотнього зв’язку, що дозволяє користувачам легко зв’язатися з адміністрацією кінотеатру з будь-яких питань чи пропозицій.

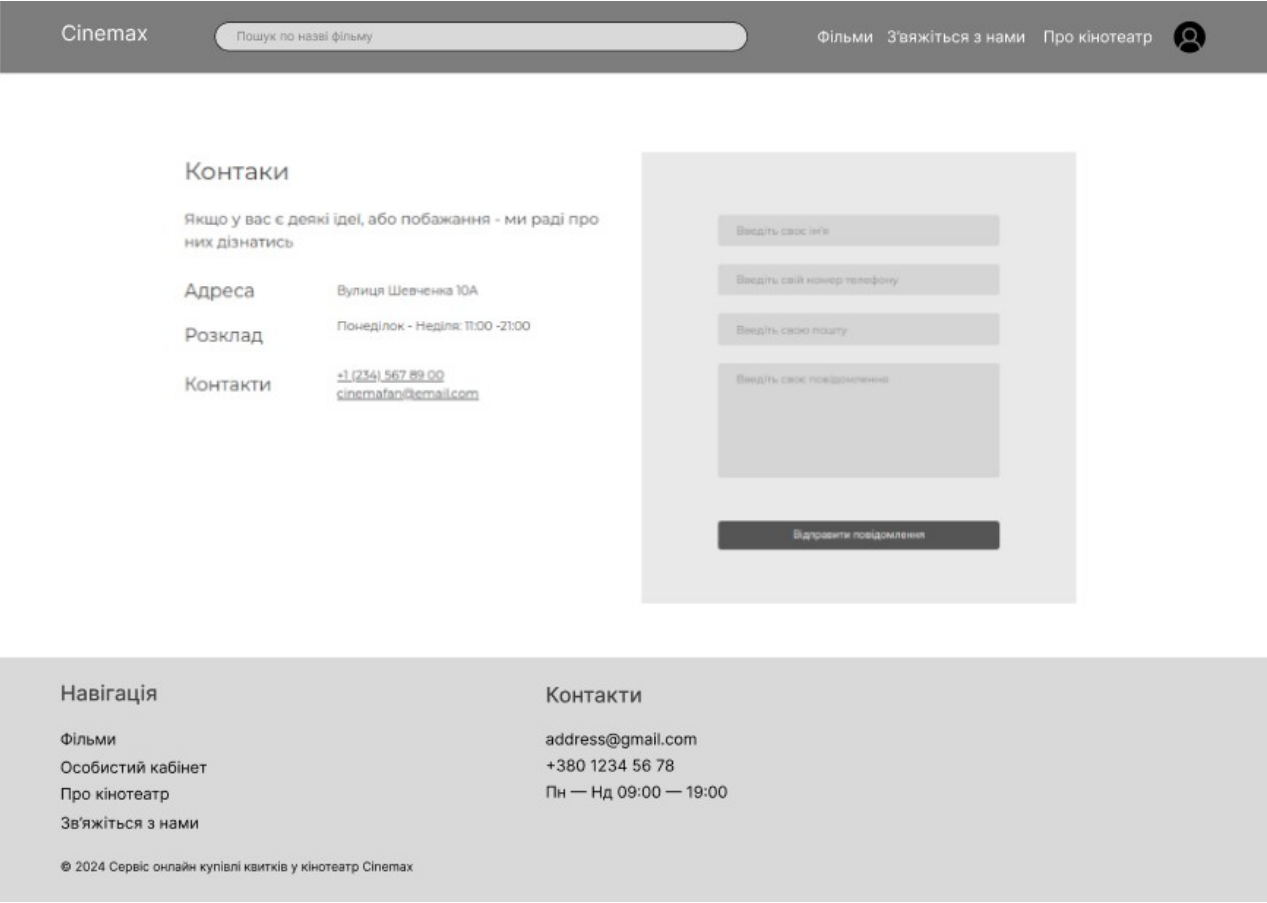


Рисунок 2.10 — Дизайн сторінки зворотнього зв’язку

2.4.7. Особистий кабінет користувача

Особистий кабінет користувача надає можливість перегляду історії бронювань та їх скасування, забезпечуючи користувачам контроль над їхніми бронюваннями та зручність у користуванні сервісом.

Cinemax

Пошук по назві фільму

ФільмиЗв'яжіться з намиПро кінотеатр

Мій профіль

Редагувати профіль



Електронна пошта: user@gmail.com

Ім'я: ім'я користувача

Реєстрація: 25.10.2022

Кількість коментарів: 13

Мої бронювання



Гарфілд у кіно

Дата і час сеансу: 25 травня 17:00

Ціна: 140 грн

Місця в залі: 33, 34

Статус оплати: оплачено

Скасувати бронювання



Королівство планети мавп

Дата і час сеансу: 27 квітня 20:40

Ціна: 70рн

Місця в залі: 21

Статус оплати: не оплачено

Оплатити

Скасувати бронювання

Навігація

Контакти

Фільми

Особистий кабінет

Про кінотеатр

Зв'яжіться з нами

address@gmail.com

+380 1234 56 78

Пн — Нд 09:00 — 19:00

© 2024 Сервіс онлайн купівлі квитків у кінотеатр Cinemax

Рисунок 2.11 — Дизайн сторінки особистого кабінету користувача

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 35   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## ВИСНОВОК ДО РОЗДІЛУ 2

У цьому розділі було розглянуто основні технології та алгоритми, що використовуються для розробки веб-сайту для продажу квитків до кінотеатру Cinemax. Було проаналізовано технології для фронтенду та бекенду, вибір бази даних та інструментів розробки, а також архітектуру системи. Окремо було висвітлено проектування бази даних та користувацького інтерфейсу, а також заходи щодо забезпечення безпеки даних. Ці компоненти дозволяють створити сучасний, продуктивний та зручний у використанні веб-сайт, який відповідає усім вимогам сучасного ринку та потребам користувачів.

Особлива увага була приділена інтеграції різних сервісів для підвищення функціональності сайту, таких як Auth0 для аутентифікації та Prisma для роботи з базою даних. Використання Next.js забезпечує потужні можливості серверного рендерингу та генерації статичних сторінок, що сприяє поліпшенню продуктивності та SEO-оптимізації. Крім того, впровадження сучасних методів розробки інтерфейсів за допомогою Tailwind CSS та Figma забезпечує високий рівень естетики та зручності для користувачів. Усі ці технології та інструменти сприяють створенню комплексного рішення для управління продажем квитків та покращення загального користувацького досвіду.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 36   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## РОЗДІЛ 3. ДЕТАЛІ РОЗРОБКИ СИСТЕМИ

### 3.1. Структура проекту

Проект має добре організовану структуру(див. Рис. 3.1), що полегшує його підтримку та розвиток. Ось детальний опис основних директорій і файлів, що складають проект:

1. /.next

Автоматично створювана директорія під час збірки Next.js, що містить скомпільовані файли. Втручання не потребує, використовується виключно фреймворком.

2. /node\_modules

Містить усі залежності та пакети, встановлені через npm(node package manager). Ця директорія також створюється автоматично під час встановлення залежностей.

3. /prisma

Використовується для роботи з Prisma ORM, що полегшує взаємодію з базою даних. Містить конфігураційні файли та схеми бази даних.

4. /public

Директорія для статичних файлів, таких як зображення, шрифти, іконки та інші ресурси, доступні напряму з браузера.

5. /src

Основна директорія з вихідним кодом додатку. Містить піддиректорії, кожна з яких відповідає за певний аспект функціональності додатку.

- /actions

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 37   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Містить файли, що визначають дії або функції для виконання специфічних задач, таких як обробка запитів або маніпуляція даними.
- /app  
Основна логіка додатку та конфігурація маршрутизації.
- /assets  
Ресурси, такі як зображення, стилі CSS, файли шрифтів тощо.
- /components  
Компоненти інтерфейсу, такі як кнопки, форми, модальні вікна тощо.
- /constants  
Файли з константами, що використовуються в додатку, такі кількість фільмів на сторінку тощо.
- /dto  
Data Transfer Objects (DTO) - об'єкти для передачі даних між різними частинами додатку.
- /interfaces  
Опис TypeScript інтерфейсів для типізації даних.
- /lib  
Бібліотеки та утиліти для використання в додатку.
- /scripts  
Скрипти для автоматизації різних задач, таких як налаштування середовища або запуск тестів.
- /types  
Додаткові типи TypeScript для додатку.
- global.d.ts  
Файл декларацій глобальних типів для TypeScript, що використовується для додавання типів у проєкті.

## 6. .env

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 38   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Файл налаштувань середовища, що містить змінні середовища, такі як ключі доступу до бази даних, API тощо.

7. `eslinttrc.json`

Файл конфігурації ESLint для налаштування правил лінтингу коду

8. `.gitignore`

Файл, що визначає, які файли та директорії ігнорує система контролю версій Git.

9. `next-env.d.ts`

TypeScript декларації для Next.js, які автоматично генеруються для забезпечення коректної типізації.

10. `next.config.mjs`

Конфігураційний файл для Next.js, що містить налаштування для фреймворку.

11. `package-lock.json`

Файл з точними версіями встановлених залежностей для забезпечення детермінованої збірки проєкту.

12. `package.json`

Файл конфігурації проєкту з метаданими, списком залежностей та скриптами для управління проєктом.

13. `postcss.config.mjs`

Конфігураційний файл для PostCSS, інструменту для трансформації CSS з використанням JavaScript плагінів.

14. `README.md`

Документація з загальною інформацією про проєкт, інструкціями щодо встановлення та запуску, а також іншими важливими деталями для розробників.

15. `tailwind.config.ts`

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 39   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

16.Конфігураційний файл для Tailwind CSS з налаштуваннями для цього CSS фреймворку.

17.tsconfig.json

Конфігураційний файл для TypeScript, що визначає параметри компіляції TypeScript до JavaScript.

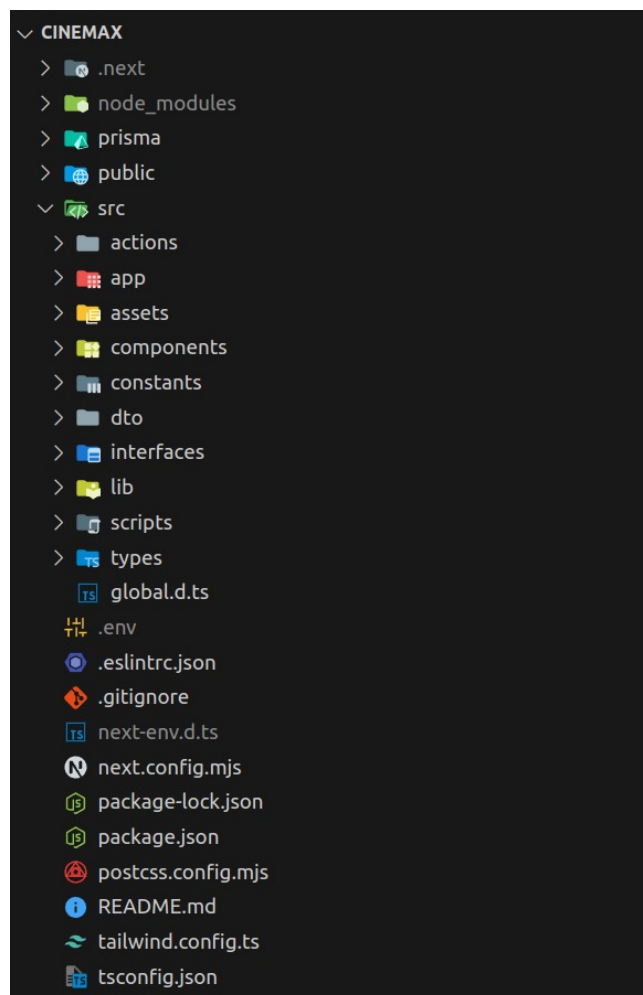


Рисунок 3.1 — Загальна структура проекту

Організація структури проекту сприяє логічному поділу відповідальностей між різними компонентами додатку. Це полегшує підтримку проекту, дозволяючи швидко знаходити необхідні файли та вносити зміни без ризику порушення функціональності інших частин системи.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 40   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## 3.2. Опис бази даних

Для кінотеатру Cinemax DB містить п'ять основних моделей: Movie, Showtime, User, Review, та Booking. Кожна з моделей описує різні аспекти системи, що дозволяє зберігати та управляти даними ефективно.

### 3.2.1. Таблиця Movie

Ця таблиця зберігає інформацію про фільми, включаючи їхню назву, опис, рік випуску, тривалість, жанри, режисерів, рейтинг, кількість голосів, URL зображення та трейлера. Кожен запис у цій таблиці представляє окремий фільм і містить всі його основні характеристики.

Таблиця 3.1 — Таблиця Movie

| Поле        | Тип         | Опис                     |
|-------------|-------------|--------------------------|
| id          | VARCHAR(36) | Унікальний ідентифікатор |
| title       | TEXT        | Назва фільму             |
| description | TEXT        | Опис фільму              |
| year        | INT         | Рік випуску фільму       |
| duration    | INT         | Тривалість (хвилини)     |
| genres      | TEXT[]      | Жанри фільму             |
| directors   | TEXT[]      | Режисери фільму          |
| rating      | FLOAT       | Рейтинг фільму           |
| votes       | INT         | Кількість голосів        |
| imageUrl    | TEXT        | URL зображення           |

### Кінець таблиці 3.1

| Поле       | Тип       | Опис                  |
|------------|-----------|-----------------------|
| trailerUrl | TEXT      | URL трейлера          |
| createdAt  | TIMESTAMP | Дата створення запису |
| updatedAt  | TIMESTAMP | Дата оновлення запису |

### 3.2.2. Таблиця Showtime

У цій таблиці зберігається інформація про сеанси показу фільмів. Кожен запис у цій таблиці пов'язаний з певним фільмом (за допомогою зовнішнього ключа movieId) та визначає дату, час початку сеансу та доступні місця для цього сеансу.

Таблиця 3.2 — Таблиця Showtime

| Поле           | Тип         | Опис                     |
|----------------|-------------|--------------------------|
| id             | VARCHAR(36) | Унікальний ідентифікатор |
| movieId        | VARCHAR(36) | Ідентифікатор фільму     |
| date           | DATETIME    | Дата показу              |
| time           | TEXT        | Час показу               |
| availableSeats | JSON        | Доступні місця           |
| createdAt      | TIMESTAMP   | Дата створення запису    |
| updatedAt      | TIMESTAMP   | Дата оновлення запису    |

### 3.2.3. Таблиця User

Ця таблиця містить інформацію про користувачів кінотеатру, таку як їхню електронну пошту, ім'я та URL зображення профілю. Кожен запис у цій таблиці відображає окремого користувача системи.

Таблиця 3.3 — Таблиця User

| Поле      | Тип         | Опис                     |
|-----------|-------------|--------------------------|
| id        | VARCHAR(36) | Унікальний ідентифікатор |
| email     | TEXT        | Електронна пошта         |
| name      | TEXT        | Ім'я користувача         |
| imageUrl  | TEXT        | URL зображення           |
| createdAt | TIMESTAMP   | Дата створення запису    |
| updatedAt | TIMESTAMP   | Дата оновлення запису    |

### 3.2.4. Таблиця Review

У цій таблиці зберігаються відгуки користувачів про фільми. Кожен запис у цій таблиці представляє конкретний відгук, що містить рейтинг, коментар та дату створення. Кожен відгук пов'язаний з користувачем та фільмом за допомогою зовнішніх ключів userId та movieId.

Таблиця 3.4 — Таблиця Review

| Поле | Тип         | Опис                     |
|------|-------------|--------------------------|
| id   | VARCHAR(36) | Унікальний ідентифікатор |

#### Кінець таблиці 3.4

| Поле      | Тип         | Опис                      |
|-----------|-------------|---------------------------|
| userId    | VARCHAR(36) | Ідентифікатор користувача |
| movieId   | VARCHAR(36) | Ідентифікатор фільму      |
| rating    | INT         | Рейтинг                   |
| comment   | TEXT        | Коментар                  |
| createdAt | TIMESTAMP   | Дата створення запису     |
| updatedAt | TIMESTAMP   | Дата оновлення запису     |

#### 3.2.5. Таблиця Booking

У цій таблиці зберігаються відгуки користувачів про фільми. Кожен запис у цій таблиці представляє конкретний відгук, що містить рейтинг, коментар та дату створення. Кожен відгук пов'язаний з користувачем та фільмом за допомогою зовнішніх ключів userId та movieId.

Таблиця 3.5 — Таблиця Booking

| Поле       | Тип         | Опис                      |
|------------|-------------|---------------------------|
| id         | VARCHAR(36) | Унікальний ідентифікатор  |
| userId     | VARCHAR(36) | Ідентифікатор користувача |
| showtimeId | VARCHAR(36) | Ідентифікатор сеансу      |
| seats      | TEXT[]      | Заброньовані місця        |
| isPayed    | BOOLEAN     | Статус оплати             |
| createdAt  | TIMESTAMP   | Дата створення запису     |
| updatedAt  | TIMESTAMP   | Дата оновлення запису     |

### 3.3. Розробка серверної частини

Цей підрозділ присвячений розробці функцій, які забезпечують взаємодію з базою даних у контексті створення серверної частини додатку. За допомогою цих функцій можна отримувати, зберігати, оновлювати та видаляти дані, такі як фільми, відгуки, користувачі та бронювання. Взаємодія з базою даних є невід'ємною частиною розробки серверної частини, оскільки це дозволяє ефективно управляти та обробляти дані, необхідні для функціонування додатку.

Короткий опис функцій для роботи з базою даних:

1. `getMovies(page?: number, limit?: number, filter?: FilterInterface): Promise<Movie[]>`

Ця функція використовується для отримання списку фільмів з бази даних з можливістю пагінації та фільтрації. Приймає параметри `page` (сторінка), `limit` (кількість записів на сторінці) та `filter` (фільтр за жанрами та/або назвою фільму).

2. `getMoviesLength(filter?: FilterInterface): Promise<number>`

Функція повертає загальну кількість фільмів у базі даних відповідно до вказаних фільтрів.

3. `getMovie(id: string): Promise<Movie | null>`

Ця функція призначена для отримання інформації про конкретний фільм за його унікальним ідентифікатором `id`.

4. `updateMovie(id: string, movie: Partial<MovieDto>): Promise<Movie | null>`

Функція використовується для оновлення існуючого запису про фільм у базі даних за його унікальним ідентифікатором `id`. Приймає об'єкт `Partial<MovieDto>`, який містить поля для оновлення.

5. `deleteMovie(id: string): Promise<Movie | null>`

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 45   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Ця функція використовується для видалення запису про фільм із бази даних за його унікальним ідентифікатором id.

6. `getAvailableGenres(): Promise<string[]>`

Функція повертає унікальні жанри всіх фільмів у базі даних.

7. `updateRating(id: string, rating: number): Promise<Movie | null>`

Ця функція використовується для оновлення рейтингу фільму в базі даних за його унікальним ідентифікатором id. Обчислює новий рейтинг та кількість голосів з урахуванням нового голосу.

8. `getShowtimes(movieId: string, date?: Date): Promise<Showtime[] | null>`

Ця функція використовується для отримання списку сеансів показу для певного фільму за його movieId. Опціонально можна передати конкретну дату, щоб отримати сеанси лише для цієї дати.

9. `getShowtime(id: string): Promise<Showtime | null>`

Ця функція призначена для отримання інформації про конкретний сеанс за його унікальним ідентифікатором id.

10. `saveShowtime(showtime: ShowtimeDto): Promise<Showtime | null>`

Функція використовується для збереження нового запису про сеанс у базі даних. Приймає об'єкт ShowtimeDto, який містить дані про новий сеанс.

11. `updateShowtime(id: string, showtime: Partial<ShowtimeDto>): Promise<Showtime | null>`

Ця функція використовується для оновлення існуючого запису про сеанс у базі даних за його унікальним ідентифікатором id. Приймає об'єкт Partial<ShowtimeDto>, який містить поля для оновлення.

12. `deleteShowtime(id: string): Promise<Showtime | null>`

Функція використовується для видалення запису про сеанс із бази даних за його унікальним ідентифікатором id.

13.getShowtimes(movieId: string, date?: Date): Promise<Showtime[] | null>

Ця функція використовується для отримання списку сеансів показу для певного фільму за його movieId. Опціонально можна передати конкретну дату, щоб отримати сеанси лише для цієї дати.

14.getShowtime(id: string): Promise<Showtime | null>

Ця функція призначена для отримання інформації про конкретний сеанс за його унікальним ідентифікатором id.

15.saveShowtime(showtime: ShowtimeDto): Promise<Showtime | null>

Функція використовується для збереження нового запису про сеанс у бази даних. Приймає об'єкт ShowtimeDto, який містить дані про новий сеанс.

16.updateShowtime(id: string, showtime: Partial<ShowtimeDto>) Promise<Showtime | null>

Ця функція використовується для оновлення існуючого запису про сеанс у бази даних за його унікальним ідентифікатором id. Приймає об'єкт Partial<ShowtimeDto>, який містить поля для оновлення.

17.deleteShowtime(id: string): Promise<Showtime | null>

Функція використовується для видалення запису про сеанс із бази даних за його унікальним ідентифікатором id.

18.getUsers(): Promise<User[] | null>

Ця функція використовується для отримання списку всіх користувачів з бази даних.

19.getUser(id: string): Promise<User | null>

Функція призначена для отримання інформації про користувача за його унікальним ідентифікатором id.

20.getUserByEmail(email: string): Promise<User | null>:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 47   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

Ця функція використовується для отримання інформації про користувача за його електронною поштою email.

21.checkIfUserWithEmailExists(email: string): Promise<boolean>

Функція перевіряє, чи існує користувач з вказаною електронною поштою email в базі даних.

22.saveUser(user: Auth0User): Promise<User | null>

Ця функція використовується для збереження нового користувача в базі даних. Приймає об'єкт Auth0User, який містить дані про нового користувача.

23.updateUser(id: string, userData: Partial<UserDto>): Promise<User | null>

Функція використовується для оновлення існуючого запису про користувача в базі даних за його унікальним ідентифікатором id. Приймає об'єкт Partial<UserDto>, який містить поля для оновлення.

24.deleteUser(id: string): Promise<User | null>

Ця функція використовується для видалення запису про користувача з бази даних за його унікальним ідентифікатором id.

25.getReviews(movieId?: string): Promise<Review[] | null>

Ця функція використовується для отримання списку відгуків про фільми. Опціонально можна передати movieId, щоб отримати відгуки лише для конкретного фільму. Результат впорядковується за датою створення у зворотньому порядку.

26.saveReview(review: ReviewDto): Promise<Review | null>

Ця функція використовується для збереження нового відгуку про фільм у базі даних. Приймає об'єкт ReviewDto, який містить дані про новий відгук.

27.updateReview(id: string, review: Partial<ReviewDto>): Promise<Review | null>

Функція використовується для оновлення існуючого запису про відгук у базі даних за його унікальним ідентифікатором id. Приймає об'єкт Partial<ReviewDto>, який містить поля для оновлення.

28.deleteReview(id: string): Promise<Review | null>

Ця функція використовується для видалення запису про відгук із бази даних за його унікальним ідентифікатором id.

29.getBookings(userId?: string): Promise<Booking[] | null>

Ця функція використовується для отримання списку бронювань користувача з бази даних. Опціонально можна передати userId, щоб отримати бронювання лише для конкретного користувача. Результат впорядковується за датою створення у зворотньому порядку.

30.saveBooking(booking: BookingDto): Promise<Booking | null>

Ця функція використовується для збереження нового бронювання в базі даних. Приймає об'єкт BookingDto, який містить дані про нове бронювання.

31.updateBooking(id: string, booking: Partial<BookingDto>): Promise<Booking | null>

Функція використовується для оновлення існуючого запису про бронювання в базі даних за його унікальним ідентифікатором id. Приймає об'єкт Partial<BookingDto>, який містить поля для оновлення.

32.deleteBooking(id: string): Promise<Booking | null>

Ця функція використовується для видалення запису про бронювання з бази даних за його унікальним ідентифікатором id.

### 3.4. Розробка клієнської частини

Розробка клієнської частини включає створення інтерфейсу користувача (UI), який забезпечує зручний та інтуїтивно зрозумілий доступ до функціональності веб-додатку. Це важливий етап, оскільки саме клієнтська частина визначає, як користувачі взаємодіють з вашим продуктом. Вона включає в себе як візуальні елементи, так і функціональність, які разом забезпечують оптимальний користувацький досвід. Розробка клієнської частини вимагає використання сучасних технологій та фреймворків, що дозволяють створювати швидкі, адаптивні та інтерактивні інтерфейси.

#### 3.4.1. Сторінка з фільмами

Компонент `MoviesPage` відповідає за відображення сторінки з фільмами, включаючи фільтрацію за жанрами, пошук за назвою, пагінацію та відображення сітки з фільмами. Він приймає параметри пошуку для динамічного рендерингу вмісту.

Основні завдання компонента:

1. Отримання параметрів пошуку:

`page`: номер поточної сторінки, якщо не вказано, використовується значення 1.

- `genres`: список жанрів, розділених комами, якщо не вказано, використовується порожній масив.
- `title`: назва фільму для пошуку, якщо не вказано, використовується порожній рядок.

2. Створення об'єкта фільтрів:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$0  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Об'єкт `searchFilters` формується на основі параметрів пошуку для фільтрації фільмів за жанрами і назвою.

### 3. Отримання даних:

Використовуються асинхронні функції, описані в попередньому підрозділі, для отримання даних з бази даних:

- `getMovies`: отримує список фільмів на основі поточної сторінки і фільтрів.
- `getMoviesLength`: отримує загальну кількість фільмів, що відповідають фільтрам.
- `getAvailableGenres`: отримує список доступних жанрів для фільтрації.

### 4. Розрахунок кількості сторінок:

- Кількість сторінок обчислюється на основі загальної кількості фільмів і кількості фільмів на сторінку (`MOVIES_PER_PAGE`).

### 5. Рендеринг компонентів:

- `Filter`: компонент для фільтрації фільмів за жанрами.
- `MoviesGrid`: компонент для відображення сітки з фільмами.
- `Pagination`: компонент для відображення пагінації.

Опис підкомпонентів:

#### 3.4.1.1. Компонент `Filter`

Компонент `Filter` відповідає за фільтрацію фільмів за жанрами. Він забезпечує вибір жанрів, відображає вибрані фільтри та дозволяє користувачам видаляти фільтри.

Основні завдання компонента:

#### 1. Ініціалізація стану:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$1  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Використовує `useState` для збереження вибраних фільтрів (`selectedFilter`).
- Отримання параметрів пошуку:
    - Використовує `useSearchParams` для отримання параметрів пошуку з URL.
    - Використовує `usePathname` для отримання поточного шляху.
    - Використовує `useRouter` для зміни поточного шляху в браузері.
  - Завантаження вибраних жанрів:
    - Використовує `useEffect` для завантаження вибраних жанрів з параметрів пошуку при завантаженні компонента або зміні параметрів пошуку.
  - Оновлення URL при зміні фільтрів:
    - Використовує `useEffect` для оновлення URL при зміні вибраних жанрів, встановлюючи нові параметри пошуку.
  - Обробка змін фільтрів:
    - Функція `handleChange` додає новий жанр до вибраних жанрів.
    - Функція `deleteFilter` видаляє вибраний жанр з фільтрів.

#### 3.4.1.2. Компонент **MoviesGrid**

Компонент **MoviesGrid** відповідає за відображення списку фільмів у вигляді сітки, де кожен фільм представлений як картка з інформацією про нього.

Основні завдання компонента:

- Відображення фільмів у вигляді сітки:
  - Використовує CSS класи для створення сітки, яка адаптується до різних розмірів екранів.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$2  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Кожен фільм відображається як елемент списку (<li>), що містить посилання на детальну сторінку фільму.
2. Навігація до сторінки фільму:
    - Використовує компонент Link з Next.js для створення посилання на сторінку з детальною інформацією про фільм.
  3. Відображення картки фільму:
    - В кожному елементі списку відображається компонент MovieCard, який містить інформацію про фільм, таку як зображення, рейтинг та назва

### 3.4.1.3. Компонент Pagation

Компонент Pagation відповідає за створення та відображення елементів пагінації на веб-сторінці, дозволяючи користувачам переходити між сторінками зі списком елементів, таких як фільми. Пагінація корисна для поділу великого обсягу даних на декілька сторінок для покращення користувацького досвіду.

Основні завдання компонента:

1. Визначення поточної сторінки:
  - Компонент використовує хук useSearchParams з Next.js для отримання поточних параметрів пошуку з URL.
  - Поточна сторінка визначається за параметром page, який перетворюється на число.
2. Створення URL для сторінок:
  - Функція createPageURL створює URL для кожної сторінки, з урахуванням параметрів пошуку.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$3  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Якщо номер сторінки виходить за межі допустимого діапазону, функція повертає undefined.

### 3. Створення списку сторінок:

- Масив pages містить номери сторінок, які відображаються у пагінації.

### 4. Відображення елементів пагінації:

- Кожна сторінка відображається як посилання (<a>), яке веде до відповідної сторінки.
- Додатково відображаються кнопки для переходу на попередню та наступну сторінку.
- Поточна сторінка виділяється за допомогою спеціального CSS класу styles.active.

## 3.4.2. Детальна сторінка фільма

Компонент MovieDetailsPage відображає сторінку з деталями про конкретний фільм, включаючи розклад сеансів, доступні місця, та відгуки користувачів. Він приймає параметри та пошукові параметри для динамічного рендерингу вмісту.

Ось основні завдання компонента:

#### 1. Отримання параметрів та пошукових параметрів:

- params: містить movieSlug, який ідентифікує фільм.
- searchParams: може містити date (обраний день) і showtimeId (ідентифікатор сеансу).

#### 2. Обробка дати та ідентифікатора сеансу:

- Якщо searchParams містить date, створюється об'єкт Date. Якщо ні, використовується поточна дата.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$4  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Якщо `searchParams` містить `showtimeId`, завантажується відповідний сеанс.

### 3. Отримання даних фільму:

- Функція `getMovie` використовується для отримання даних про фільм за `movieSlug`.
- Якщо фільм не знайдено, викликається функція `notFound` для відображення сторінки 404.

### 4. Отримання сеансів та відгуків:

- Функція `getShowtimes` використовується для отримання розкладу сеансів на обрану дату.
- Функція `getReviews` використовується для отримання відгуків про фільм.
- Для кожного відгуку завантажується відповідний користувач за допомогою `getUser`, щоб отримати інформацію про рецензентів.

### 5. Рендеринг компонентів:

- `MovieDetails`: відображає інформацію про фільм.
- `ShowtimePicker`: відображає доступні сеанси для вибраної дати.
- `ModalSeatsPicker`: відображає модальне вікно для вибору місць на конкретний сеанс.
- `Reviews`: відображає відгуки користувачів та дозволяє додавати нові відгуки через `createReview`.

Опис підкомпонентів:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$5  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

### 3.4.2.1. Компонент MovieDetails

Компонент MovieDetails відповідає за відображення детальної інформації про фільм. Він містить дані про фільм, такі як зображення, рейтинг, рік, тривалість, жанр, режисери, опис і трейлер.

Основні завдання компонента:

1. Відображення зображення фільму:
  - Використовує компонент Image з next/image для відображення зображення фільму з високою якістю та оптимізацією.
2. Відображення основної інформації про фільм:
  - Заголовок фільму.
  - Рейтинг у форматі зірок, кількість голосів.
  - Рік випуску.
  - Тривалість фільму у годинах та хвилинах.
  - Жанр(и) фільму.
  - Режисер(и).
  - Опис фільму.
3. Відтворення трейлера фільму:
  - Використовує підкомпонент Player для вбудовування відео трейлера з YouTube.

### 3.4.2.2. Компонент ShowtimePicker

Компонент ShowtimePicker відповідає за вибір дати та часу сеансу для перегляду фільму. Він надає користувачеві можливість вибирати дату зі списку доступних днів та показує доступні сеанси на вибрану дату.

Основні завдання компонента:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$6  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

1. Відображення доступних дат:

- Використовує функцію `getNextFormattedDays` для отримання списку дат на наступні 7 днів, включаючи форматовану дату та день тижня.
- Дати відображаються у вигляді списку, з можливістю вибору конкретної дати.

2. Відображення доступних сеансів:

- Після вибору дати показуються доступні сеанси на цю дату.
- Якщо на вибрану дату немає сеансів, відображається повідомлення про відсутність сеансів.

3. Інтерактивність:

- При виборі дати оновлюється стан компонента та URL-адреса з параметром `date`.
- При виборі часу сеансу, якщо користувач не авторизований, його перенаправляють на сторінку входу. Якщо авторизований, оновлюється URL-адреса з параметром `showtimeId`.

### 3.4.2.3. Компонент `ModalSeatsPicker`

Компонент `ModalSeatsPicker` відображає модальне вікно для вибору місць у кінозалі для конкретного сеансу. Цей компонент інтегрує форму вибору місць (`SeatsPickerForm`) і забезпечує навігацію до сторінки оплати після вибору місць.

Основні завдання компонента:

1. Відображення інформації про сеанс:

- Відображає заголовок фільму.
- Показує дату та час сеансу.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$7  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

2. Вибір місць:

- Використовує інтерактивну форму для вибору доступних місць у кінозалі.
- Відображає місця у вигляді чекбоксів, де заброньовані місця позначені відповідним стилем.

3. Обробка вибору місць:

- Додає або видаляє місця зі списку вибраних місць при взаємодії користувача.
- Показує загальну кількість вибраних місць та їх вартість.

4. Бронювання сеансу

- Після підтвердження вибору місць створює нове бронювання.

5. Навігація до сторінки оплати:

- При підтвердженні вибору місць оновлює URL і перенаправляє користувача на сторінку оплати.

#### 3.4.2.4. Компонент Reviews

Компонент Reviews відображає список відгуків про фільм разом з формою для додавання нового відгуку.

Основні завдання компонента:

1. Відображення списку відгуків:

- Показує кількість відгуків.
- Відображає список існуючих відгуків.

2. Форма для додавання відгуку:

- Виводить форму, яка дозволяє користувачам залишити відгук.
- Перевіряє, чи користувач авторизований для відправлення відгуку.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | \$8  |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

- Надає можливість користувачам оцінити фільм за допомогою повзунка.
- Дозволяє користувачам залишити коментар до свого відгуку.
- Показує попередження для неавторизованих користувачів.

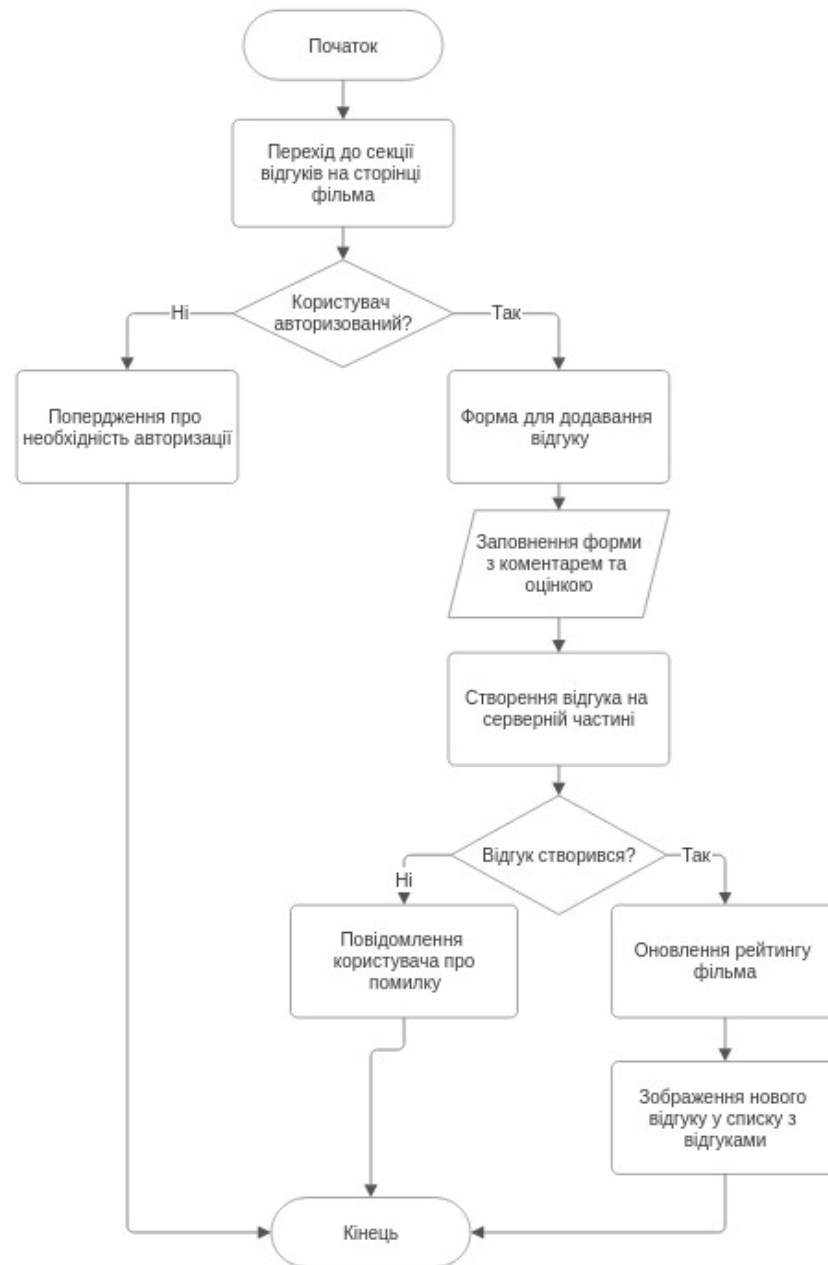


Рисунок 3.2 — Алгоритм надсилання відгуків про фільм

### 3.4.3. Сторінка оплати

Сторінка оплати (PaymentPage) призначена для відображення інформації про бронювання і надання можливості користувачеві завершити оплату. Вона отримує дані про бронювання, сеанс та фільм, і дозволяє користувачеві провести оплату через форму.

Основні завдання компонента:

1. Інформація про бронювання:

- Витягує дані про бронювання за допомогою унікального paymentSlug.
- Якщо бронювання не знайдено, перенаправляє користувача на сторінку "404 Not Found".

2. Інформація про сеанс та фільм:

- Отримує дані про сеанс на основі showtimeId з бронювання.
- Отримує дані про фільм на основі movieId з сеансу.
- Відображає заголовок фільму і дату сеансу.

3. Форма оплати:

- Використовує компонент PaymentForm для відображення форми оплати.
- Передає до PaymentForm ціну бронювання та функцію для оновлення статусу оплати.

#### 3.4.3.1. Компонент PaymentForm

Компонент PaymentForm надає інтерфейс для введення платіжної інформації та здійснення оплати. Він обробляє введені дані, здійснює запит на

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 60   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

сервер для обробки платежу, і при успішній оплаті перенаправляє користувача на сторінку з фільмами.

Основні завдання компонента:

1. Ввід платіжних даних:

- Форма містить поля для введення номера картки, терміну дії та CVV2 коду.
- Поля мають обмеження на довжину введених даних для забезпечення коректності.

2. Обробка форми:

- Використовує хук `useFormState` для зберігання стану форми та виконання дії при її відправленні.
- Використовує хук `useFormStatus` для відстеження статусу відправки форми.
- Перевіряє правильність введених даних.
- Якщо дані правильні, змінює статус оплати бронювання на сплачено.

3. Перенаправлення після оплати:

- Після успішної оплати, користувач перенаправляється на сторінку з фільмами.

### 3.4.4. Сторінка особистого кабінета користувача

Сторінка особистого кабінету користувача надає користувачам доступ до їхніх особистих даних, інформації про бронювання та відгуки. Вона дозволяє переглядати деталі профілю, історію бронювань і здійснювати різні дії, такі як редагування профілю, оплата бронювань та їх скасування.

Основні завдання компонента:

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 61   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

1. Інформація про користувача:

- Відображає зображення профілю користувача.
- Показує електронну пошту, ім'я, дату реєстрації та кількість відгуків користувача.
- Містить кнопку для редагування профілю.

2. Інформація про бронювання:

- Показує кількість здійснених бронювань.
- Для кожного бронювання відображає деталі, такі як назва фільму, дата та час сеансу, ціна, місця в залі та статус оплати.
- Містить посилання для оплати бронювань, якщо вони ще не оплачені.
- Містить кнопку для скасування бронювання.

### 3.5. SEO-оптимізація

SEO-оптимізація (Search Engine Optimization) є критичним аспектом веб-розробки, спрямованим на покращення видимості сайту в пошукових системах. Вона включає набір стратегій і технік, які допомагають забезпечити високі позиції веб-додатку в результатах пошуку, що сприяє збільшенню органічного трафіку. Ефективна SEO-оптимізація охоплює як технічні аспекти, так і контентну складову, з метою створення зручного, релевантного та легко доступного ресурсу для користувачів і пошукових систем.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 62   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

### 3.5.1. Структуровані дані

Структуровані дані допомагають пошуковим системам краще розуміти зміст сторінок вашого сайту. Наприклад, для сторінки з фільмами можна використовувати [schema.org](https://schema.org) для опису фільмів та їх деталей.

```
import Head from 'next/head';

const MoviePage = ({ movie }: { movie: Movie }) => {
  const jsonLd = {
    "@context": "https://schema.org",
    "@type": "Movie",
    "name": movie.title,
    "image": movie.imageUrl,
    "description": movie.description,
    "director": movie.directors.join(', '),
    "datePublished": movie.createdAt,
    "genre": movie.genres.join(', ')
  }

  return (
    <>
      <Head>
        <script
          type="application/ld+json"
          dangerouslySetInnerHTML={{ __html: JSON.stringify(jsonLd) }}
        />
      </Head>
      <main>
        | Вміст сторінки
      </main>
    </>
  );
}
```

You, 6 days ago • add movie details

Рисунок 3.3 — Приклад використання структурованих типів [schema.org](https://schema.org)

### 3.5.2. Заголовки та метадані

Заголовки та мета-описи повинні бути унікальними для кожної сторінки. Це допомагає покращити видимість у пошукових системах і підвищує CTR (Click-Through Rate).

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 63   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

```
export const metadata: Metadata = {
  title: 'Фільми | Cinemax',
  description: 'Перегляд доступних фільмів у кінотеатрі Cinemax',
}
```

Рисунок 3.4 — Приклад статичних метаданих для сторінки з фільмами

```
export async function generateMetadata({ params }: MovieDetailsPageProps): Promise<Metadata> {
  const movie: Movie | null = await getMovie(params.movieSlug);

  return {
    title: `${movie?.title.toUpperCase()} | Cinemax`,
    description: `Перегляньте інформацію про фільм ${movie?.title}, включаючи опис, дату виходу, режисера та`
  }
}
```

Рисунок 3.5 — Приклад динамічних метаданих для детальної сторінки фільма

### 3.5.3. Внутрішні посилання

Внутрішні посилання покращують навігацію і допомагають пошуковим системам зрозуміти структуру вашого сайту.

Next.js оптимізує SEO для посилань за допомогою компонента Link, шляхом генерації чистих і оптимізованих URL-адрес, які враховують ієрархію сторінок та ключові слова. Крім того, Next.js додає відповідні HTML атрибути до посилань, такі як `rel="preload"`, `as="script"`, `as="style"`, щоб забезпечити попереднє завантаження ресурсів. Це допомагає пошуковим системам ефективно індексувати контент сторінок, а також поліпшує загальний рейтинг SEO вашого веб-сайту. Крім того, Next.js використовує рендерінг на стороні сервера (SSR) або статичне попереднє рендерінг (SSG), що також може позитивно вплинути на SEO, роблячи ваш веб-сайт більш доступним та зрозумілим для пошукових систем.

```

<Link href={` /movies/${movie.id}`}>
  <MovieCard movie={movie} key={movie.id} />
</Link>

```

Рисунок 3.6 — Приклад внутрішнього посилання

### 3.5.4. Оптимізація зображень

SEO-оптимізація зображень в Next.js відіграє важливу роль у покращенні видимості вашого веб-сайту у пошукових системах. Компонент Image в Next.js не лише забезпечує ліниве завантаження зображень, що допомагає зменшити час завантаження сторінок, але і автоматично додає значення атрибута alt, що відповідає назві зображення, для поліпшення індексації пошуковими системами. Крім того, ви можете використовувати атрибут sizes, щоб оптимізувати розміри зображень для різних пристроїв та екранів.

Правильне використання атрибутів alt та sizes, разом з лінивим завантаженням зображень, допомагає покращити швидкість завантаження сторінок, забезпечити кращий досвід користувача та підвищити індексацію вашого контенту пошуковими системами.

```

<Image
  src={movie.imageUrl}
  alt={movie.title}
  fill
  sizes="(max-width: 768px) 50vw, (max-width: 1200px) 33vw, 16vw"
/>

```

Рисунок 3.7 — Приклад оптимізації зображення

### 3.5.5. Адаптивний дизайн

Адаптивний дизайн є важливою складовою веб-розробки, оскільки він забезпечує оптимальний вигляд та функціональність сайту на будь-яких пристроях, від маленьких мобільних телефонів до великих настільних моніторів. Tailwind CSS, як модульний CSS-фреймворк, забезпечує потужні засоби для створення адаптивних інтерфейсів. Він дозволяє використовувати класи, які адаптуються в залежності від розміру екрану, наприклад, md:, lg:, xl: і так далі, щоб забезпечити консистентний та привабливий вигляд сторінки на різних пристроях.

Наприклад, застосовуючи клас lg:grid-cols-3, ми можемо розділити компонент MoviesGrid, який відображатиметься у форматі "grid" лише на екранах великого розміру та більше (див. рис. 3.7.). Використання Tailwind CSS спрощує розробку адаптивного дизайну, дозволяючи легко контролювати вигляд та поведінку елементів на різних пристроях.

```
<ul className={` ${styles.grid} grid grid-cols-2 lg:grid-cols-3 xl:grid-cols-4 2xl:grid-cols-6 gap-2 md:gap-4`} >
  {movies.map(movie => (
    <li key={movie.id}>
      <Link href={`/movies/${movie.id}`} >
        <MovieCard movie={movie} key={movie.id} />
      </Link>
    </li>
  ))}
</ul>
```

Рисунок 3.8 — Приклад адаптивного дизайну Tailwind CSS

Також використання @media в CSS дозволяє створювати адаптивний дизайн, що відповідає характеристикам пристрою.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 66   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

```

@media (min-width: 768px) {
  .player {
    height: 25em;
  }
}

@media (min-width: 1024px) {
  .player {
    height: 30em;
  }
}

```

Рисунок 3.9 — Приклад адаптивного дизайну CSS

На рис. 3.8 видно, що висота плеєра на детальній сторінці фільму має різну висоту в залежності від екрану. Це дозволяє забезпечити оптимальний вигляд веб-сторінки на різних пристроях.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 67   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## ВИСНОВОК ДО РОЗДІЛУ 3

Розділ включає детальний опис структури проєкту, опис бази даних та розробку серверної та клієнтської частин додатку.

Структура проєкту демонструє організованість та логічний поділ відповідальностей між різними компонентами системи. Це сприяє зручній підтримці та розвитку проєкту, полегшуючи пошук необхідних файлів та внесення змін без ризику порушення функціональності інших частин системи.

Опис бази даних включає п'ять основних моделей: Movie, Showtime, User, Review та Booking, що дозволяє зберігати та управляти даними ефективно. Кожна з таблиць має чітко визначені поля та відображає різні аспекти системи.

Розробка серверної частини включає опис функцій, які забезпечують взаємодію з базою даних. Ці функції дозволяють отримувати, зберігати, оновлювати та видаляти дані, необхідні для функціонування додатку.

Розробка клієнтської частини передбачає створення інтерфейсу користувача, який забезпечує зручний та інтуїтивно зрозумілий доступ до функціональності веб-додатку. Це включає в себе розробку як візуальних елементів, так і функціональності, забезпечуючи оптимальний користувацький досвід.

У цілому, даний розділ надає комплексний огляд процесу розробки проєкту, від структури проєкту до розробки бази даних та функціоналу серверної та клієнтської частин додатку.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 68   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## РОЗДІЛ 4. ПРЕДСТАВЛЕННЯ РОЗРОБЛЕНОЇ СИСТЕМИ

У цьому розділі буде представлено розгортання системи та бази даних на платформі Vercel, а також продемонстровано роботу розробленої системи. Ми детально розглянемо процес налаштування та запуску проекту на Vercel, включаючи необхідні технічні кроки для забезпечення безперебійної роботи сайту та бази даних. Після цього буде проведена демонстрація основних функціональних можливостей системи, що дозволить оцінити її ефективність та зручність використання, включаючи реєстрацію користувачів, перегляд фільмів та бронювання квитків.

### 4.1. Розгортання системи

Розгортання системи та бази даних на платформі Vercel є важливим етапом для забезпечення доступності та функціональності системи. Спершу необхідно підготувати проект, перевіривши код і налаштувавши конфігураційні файли.

Для розгортання сайту на Vercel потрібно зареєструватися на платформі, інтегрувати Vercel з репозиторієм проекту, налаштувати змінні середовища (див рис. 4.1) та запустити процес розгортання. Vercel автоматично збере і розгорне проект, надаючи URL-адресу для доступу до сайту (див. рис. 4.2).

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 69   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

```
1 POSTGRES_URL=  
2 POSTGRES_PRISMA_URL=  
3 POSTGRES_URL_NO_SSL=  
4 POSTGRES_URL_NON_POOLING=  
5 POSTGRES_USER=  
6 POSTGRES_HOST=  
7 POSTGRES_PASSWORD=  
8 POSTGRES_DATABASE=  
9 AUTH0_SECRET=  
10 AUTH0_BASE_URL=  
11 AUTH0_ISSUER_BASE_URL=  
12 AUTH0_CLIENT_ID=  
13 AUTH0_CLIENT_SECRET=
```

Рисунок 4.1 — Змінні середовища для Postgres та Auth0

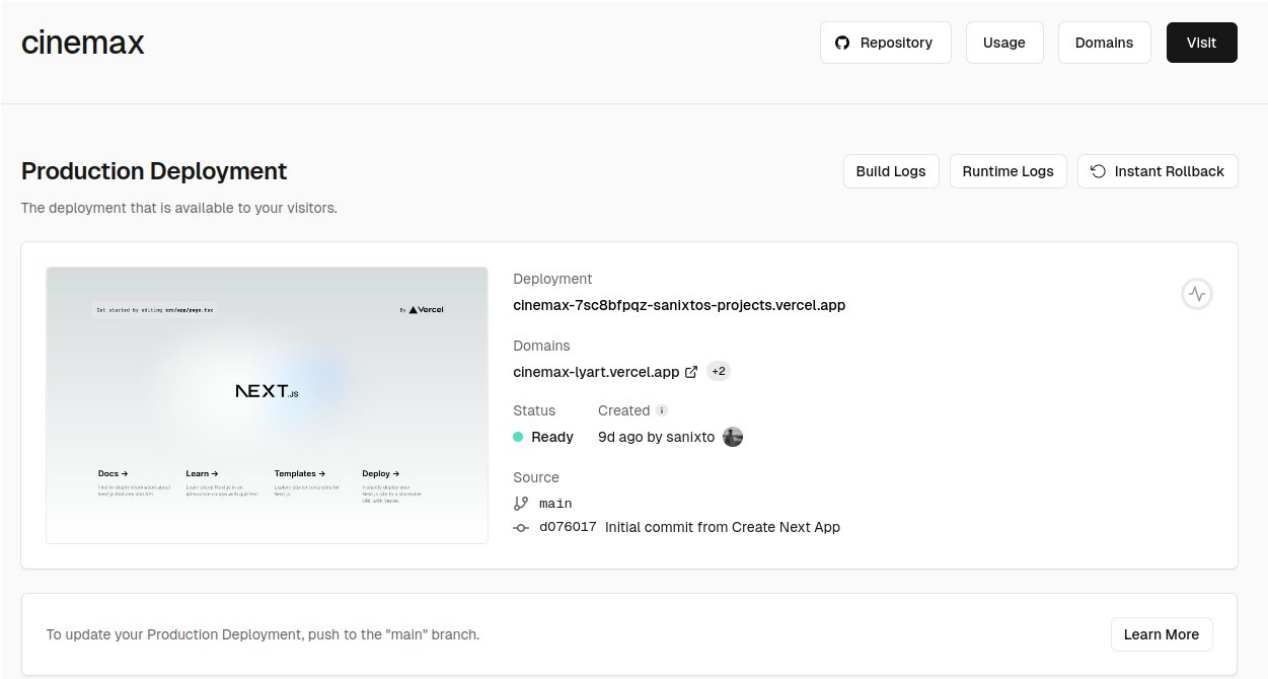


Рисунок 4.2 — Розгорнутий проєкт на Vercel

База даних розгорнута на Vercel Postgres, що забезпечує зручну інтеграцію та ефективне управління даними. Для цього використовується файл `schema.prisma` (див. рис. 4.3) і команди Prisma. Спочатку слід налаштувати схему бази даних у файлі `schema.prisma`. Потім, за допомогою команд `prx`

prisma db push і npx prisma generate, створюється та оновлюється структура бази даних на Vercel Postgres.

```
model Movie {
  id          String  @id @default(uuid())
  title       String
  description  String
  year        Int
  duration    Int
  genres       String[]
  directors   String[]
  rating      Float
  votes       Int
  imageUrl    String  @map("image_url")
  trailerUrl   String  @map("trailer_url")
  showtimes   Showtime[]
  reviews     Review[]
  createdAt   DateTime @default(now()) @map("created_at")
  updatedAt   DateTime @updatedAt @map("updated_at")
}
```

Рисунок 4.3 — Таблиця Movie, описана в schema.prisma

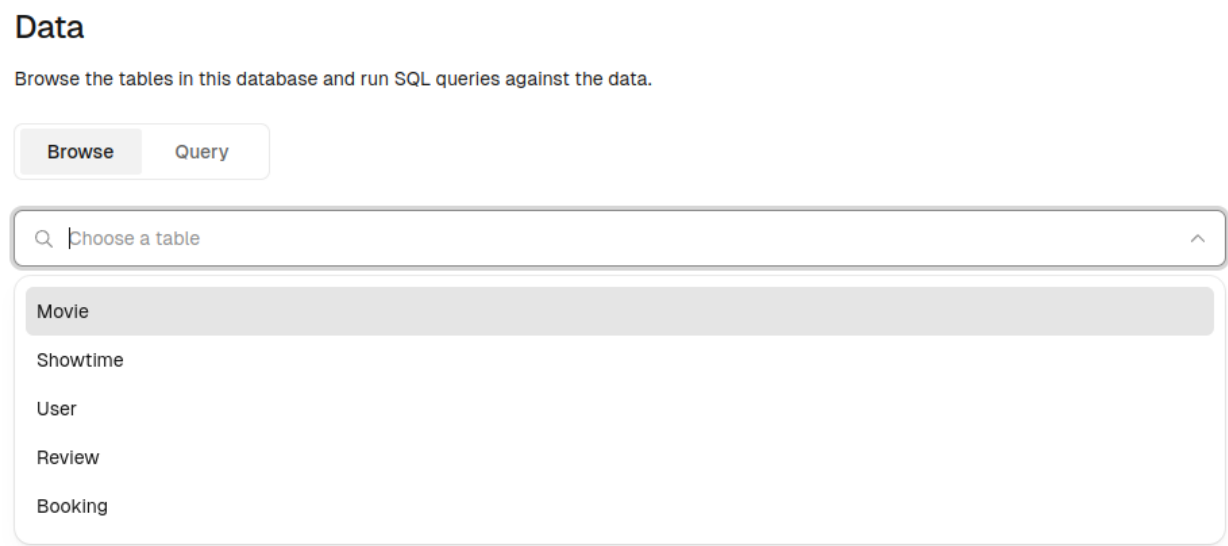


Рисунок 4.4 — Розгорнута база даних на Vercel

Data

Browse the tables in this database and run SQL queries against the data.

Browse

Query

| id                                   | title          | description                                                                                                                                                                                                                                                                                                                                                                                                    | year | duration |
|--------------------------------------|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|----------|
| 9d142844-6554-4967-bd07-09c4fd9b7812 | ґарфілд у кіно | <div> <div> Улюбленець мільйонів – рудий кіт ґарфілд повертається на великі екрани! Він ненавидить понеділки, обожнює лазанью і кожен день проводить біля телевізора. Проте цього разу на нього чекає несподівана пригода за межами дому. Після випадкової зустрічі зі своїм давно загубленим батьком – замураним вуличним котом – ґарфілд і його друг Оді змушені залишити безтурботне життя та </div> </div> | 2024 | 101      |

Рисунок 4.5 — Розгорнута таблиця Movie на Vercel

4.2.Демонстрація системи

Демонстрація роботи системи охоплює огляд основних функцій та можливостей вебсайту, а також взаємодію користувачів із системою. Після успішного розгортання сайту та бази даних, користувачі можуть отримати доступ до системи за наданою URL-адресою.

Перше, що бачить користувач, це головна сторінка сайту, яка надає зручний інтерфейс для навігації. На головній сторінці розташовано меню, що дозволяє переходити до різних розділів сайту, таких як каталог фільмів, сторінка входу/реєстрації, особистий кабінет користувача тощо.



Рисунок 4.6 — Меню сайту

Каталог фільмів демонструє можливості системи щодо пошуку (див. рис. 4.7), фільтрації (див. рис. 4.8) контенту(див. рис. 4.9). Користувач може переглядати список доступних фільмів, використовуючи різні критерії для пошуку.

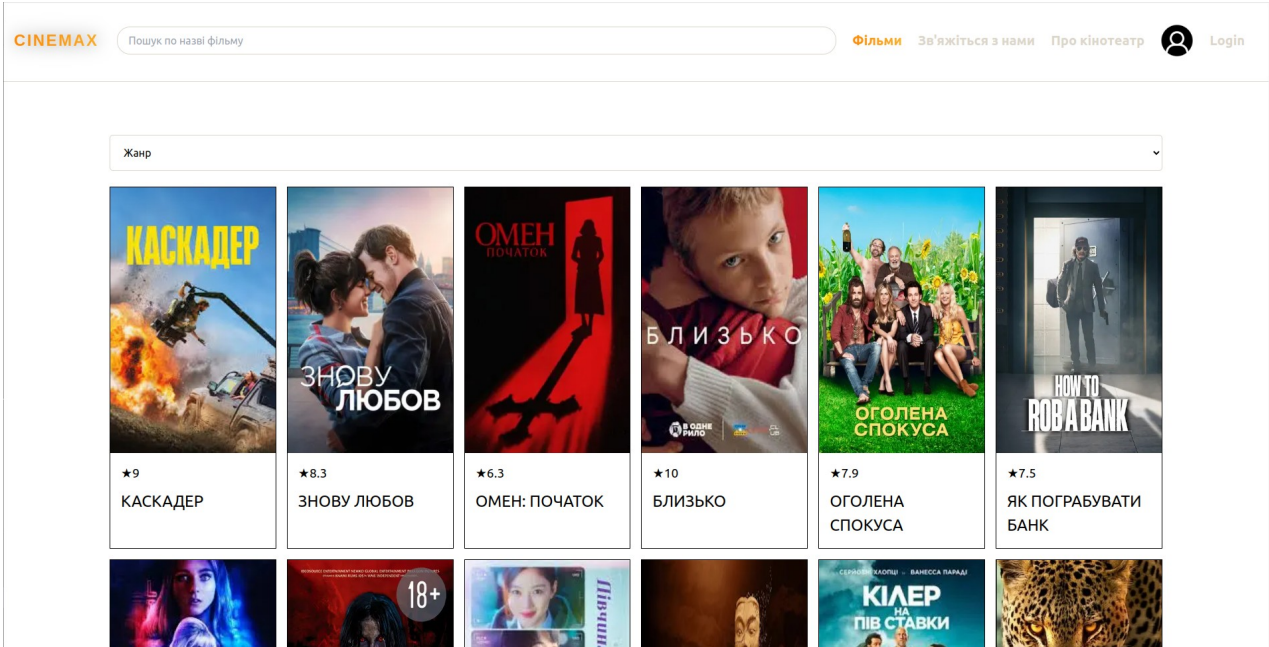


Рисунок 4.7 — Сторінка з фільмами

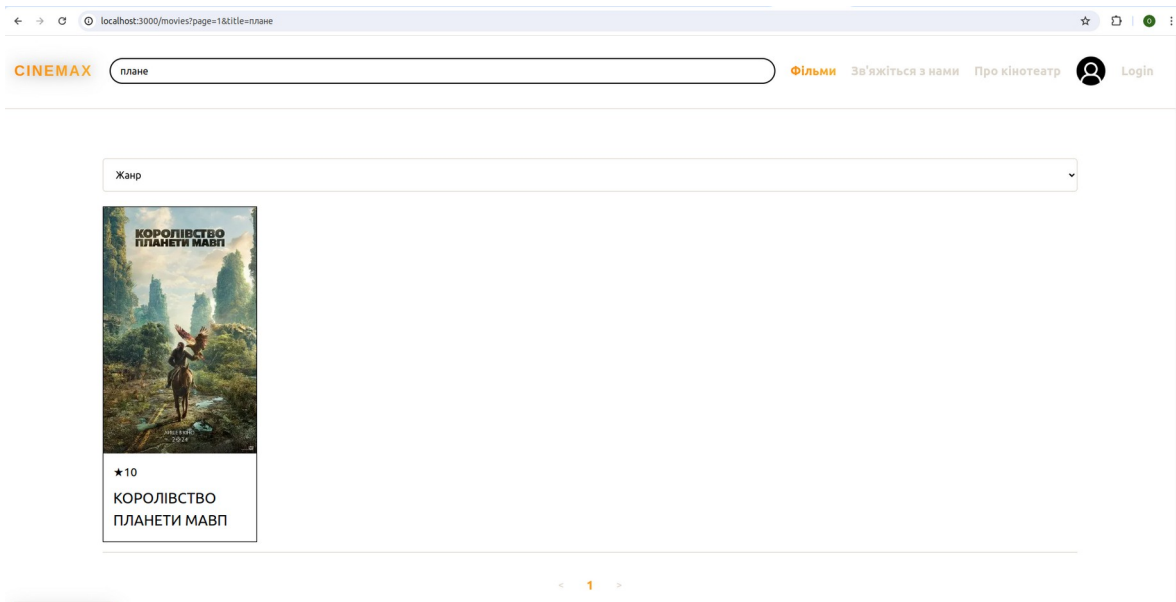


Рисунок 4.8 — Пошук фільмів за назвою

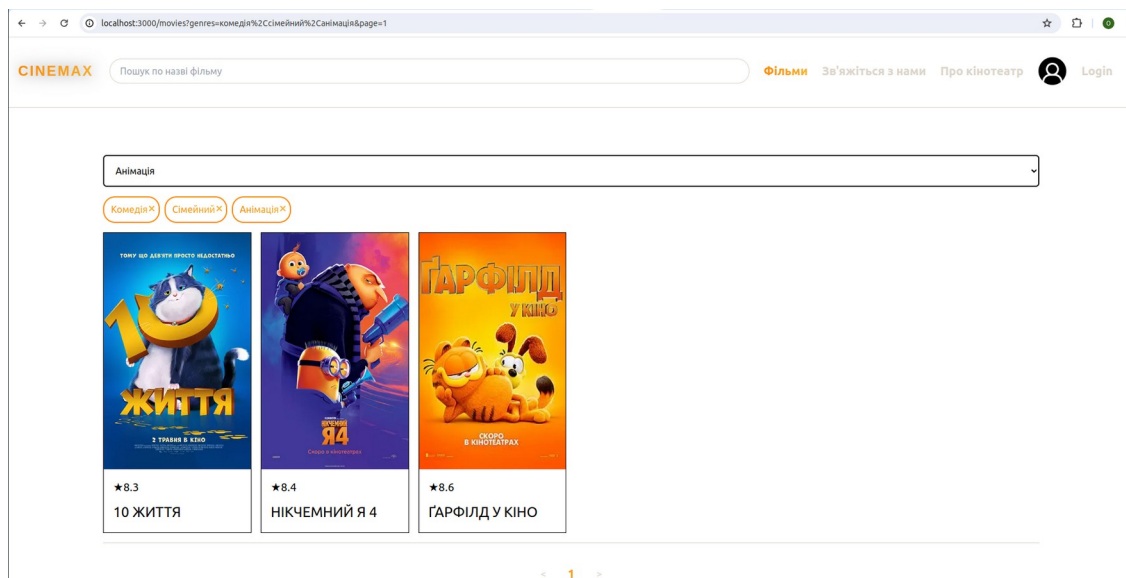


Рисунок 4.9 — Фільтрація фільмів за жанрами

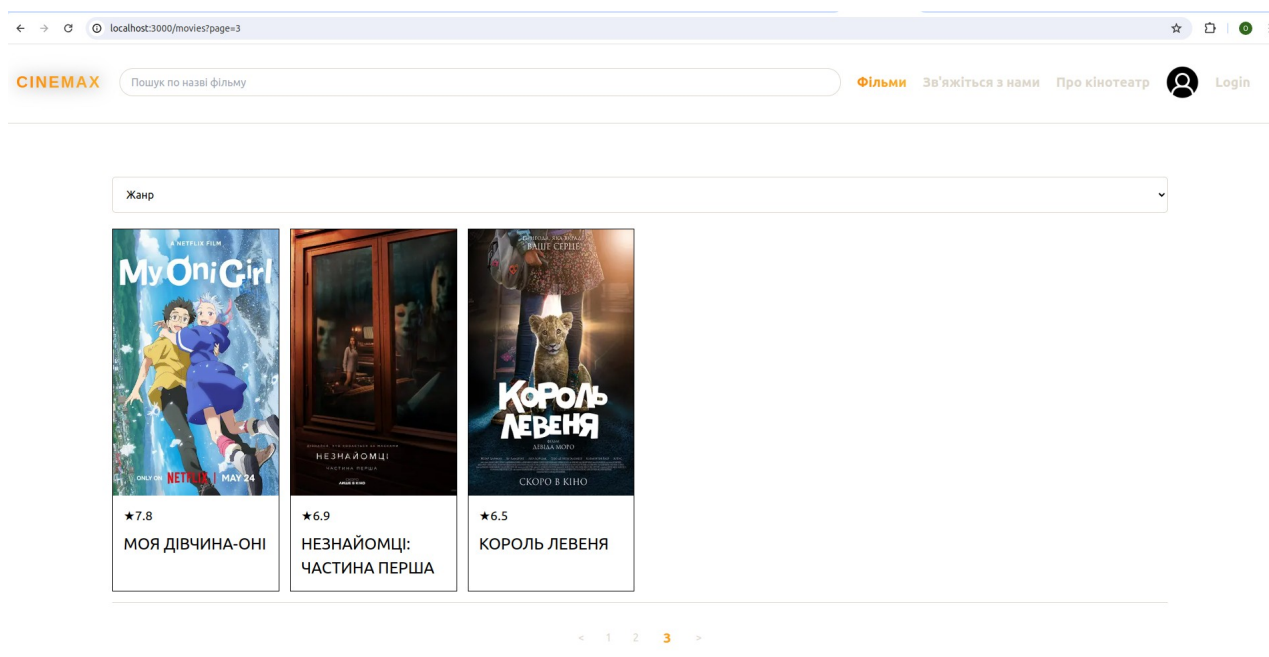


Рисунок 4.10 — Пагінація

Натискаючи на обраний фільм, користувач потрапляє на сторінку з детальною інформацією про нього, де можна знайти опис, трейлер та відгуки.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 74   |

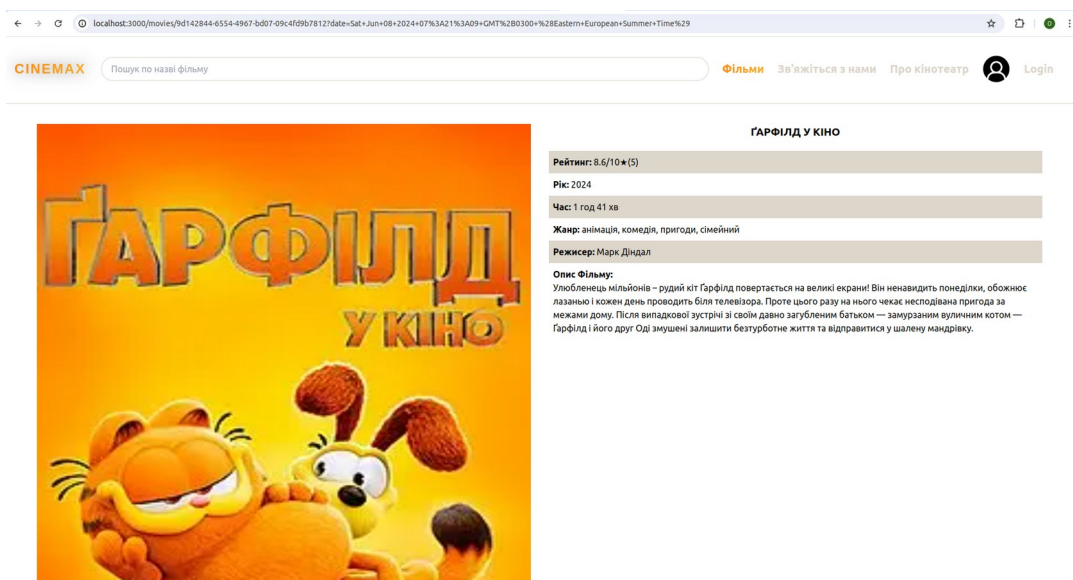


Рисунок 4.11 — Детальна сторінка фільма

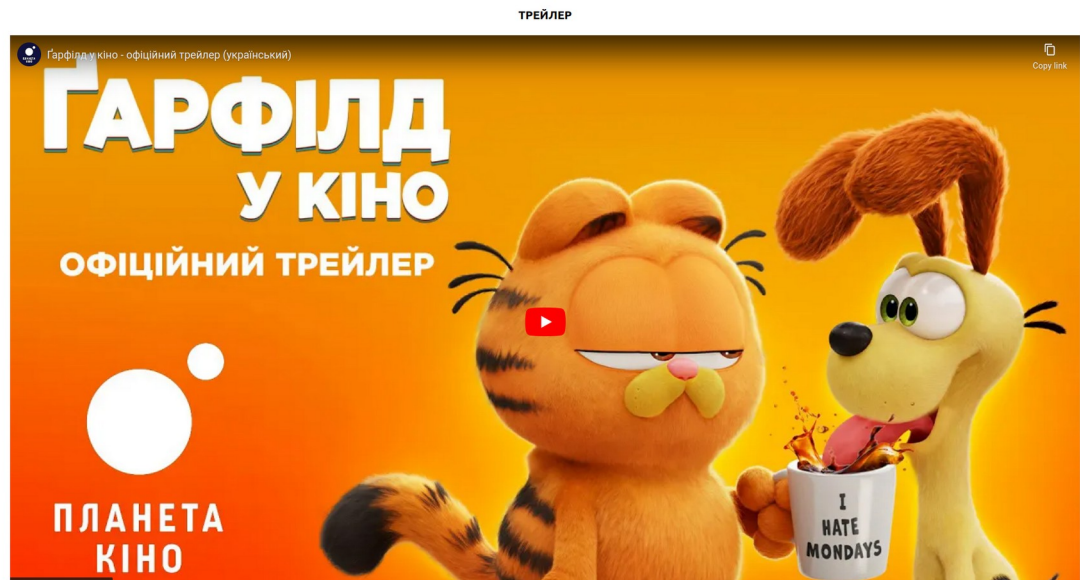


Рисунок 4.12 — Трейлер фільму

Однією з ключових функцій системи є можливість купівлі квитків на сеанси. Для демонстрації цього процесу користувачеві слід обрати фільм, а потім перейти до розділу вибору дати та часу сеансу (див. рис. 4.13) на детальній сторінці фільма (див рис. 4.11). На цій сторінці можна вибрати зручний сеанс, дату та час перегляду.

**Виберіть дату та час сенсу**

8 червня  
субота

9 червня  
неділя

10 червня  
понеділок

11 червня  
вівторок

12 червня  
середа

13 червня  
четвер

8 червня,  
субота

9:00

13:00

17:00

Рисунок 4.13 — Розділ вибору дати та часу сеансу

**Виберіть дату та час сенсу**

8 червня  
субота

9 червня  
неділя

10 червня  
понеділок

11 червня  
вівторок

12 червня  
середа

13 червня  
четвер

10 червня,  
понеділок

12:00

16:00

20:00

Рисунок 4.14 — Демонстрація зміни доступних сеансів після вибору дати

Після вибору сеансу, якщо користувач не авторизований, то його перенаправляє на сторінку входу/реєстрації (див. рис. 4.15).

**Ласкаво просимо!**

Зареєструйтеся на сайті сіпетах, щоб продовжити.

Зареєструватися

Уже маєте обліковий запис? [Увійти](#)

ЛБО

Продовжити з Google

Продовжити з Apple

Рисунок 4.15 — Сторінка входу/реєстрації

Якщо користувач авторизований то після вибору сеансу відкривається модальне вікно інтерактивного вибору місця в залі (див. рис. 4.16).

|     |      |          |        |      |                    |            |
|-----|------|----------|--------|------|--------------------|------------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк.<br>76 |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |            |

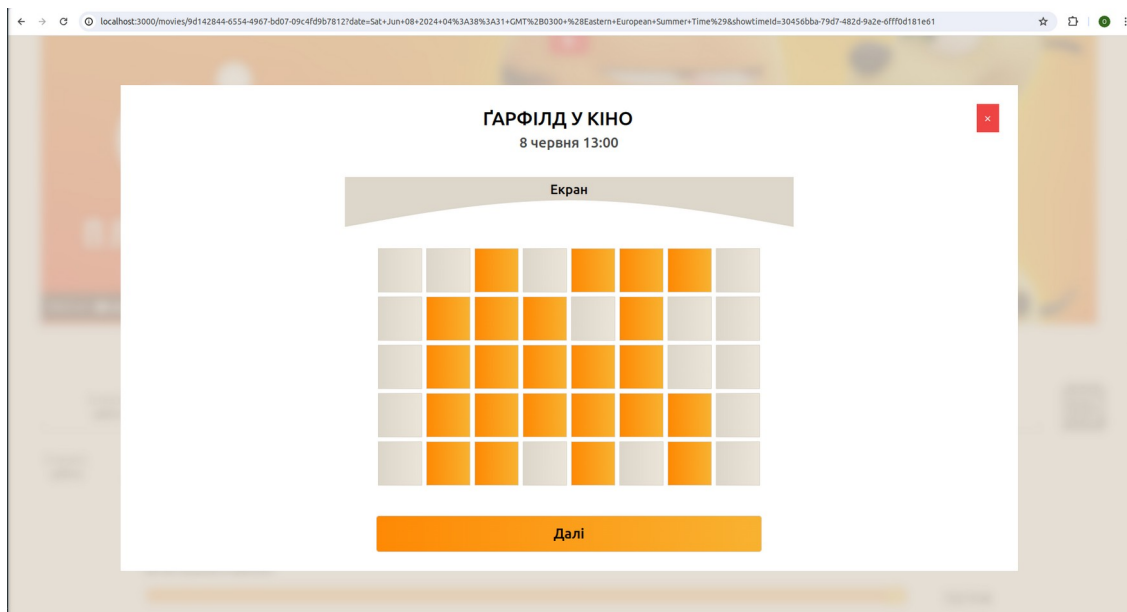


Рисунок 4.16 — Модальне вікно інтерактивного вибору місця в залі

Після вибору місць користувач може побачити їхні номери та загальну суму (див рис. 4.17).

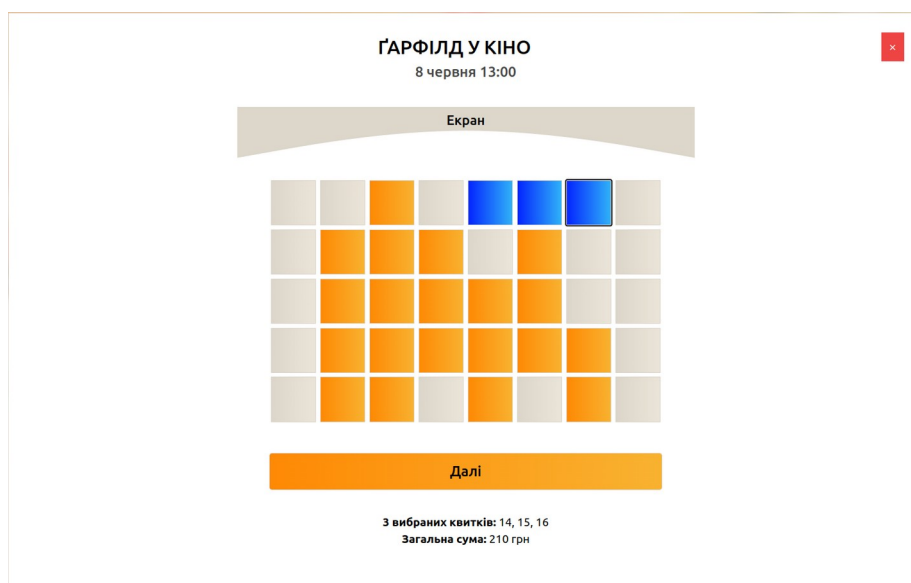


Рисунок 4.17 — Демонстрація вибору місць

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 77   |

Після підтвердження вибору користувача перенаправляє на сторінку оплати, де він може оплатити бронювання або натиснути ‘Оплатити пізніше’ (див. рис. 4.18).

ГАРФІЛД У КІНО

8 червня 2024 р. 03:00

Номер картки

0000 0000 0000 0000

Термін дії

24/07

CVV2

123

Оплатити 210 грн

Оплатити пізніше

Рисунок 4.18 — Сторінка оплати

Сторінка оплати буде тільки симулювати процес оплати, проводячи валідацію введених даних та зміну статусу оплати бронювання в результаті успішної оплати.

Після натискання ‘Оплатити пізніше’ користувач переходить на сторінку ‘Особистий кабінет користувача’, де знаходиться інформація про користувача та його бронювання (див. рис. 4.19).

Мій профіль



Електронна пошта: user1@gmail.com

Ім'я: user1

Реєстрація: 8 червня 2024 р. 04:36

Кількість відгуків 0

Редагувати профіль

Мої бронювання(1)



ГАРФІЛД У КІНО

Дата і час сеансу: 8 червня 2024 р. 03:00

Місця в залі: 14, 15, 16

Ціна: 210

Статус оплати: Не оплачено

Оплатити

Скасувати бронювання

Рисунок 4.19 — Особистий кабінет користувача

В кабінеті можемо побачити, що у нас з'явилося нове бронювання зі статусом 'Не оплачено'. Кнопка 'Оплатити' поверне нас на сторінку оплати (див. рис. 4.18). На це раз вже натиснемо оплатити 210 грн та побачимо, що статус оплати змінився (див. рис. 4.20).

Мої бронювання(1)



ГАРФІЛД У КІНО

Дата і час сеансу: 8 червня 2024 р. 03:00

Місця в залі: 14, 15, 16

Ціна: 210

Статус оплати: Оплачено

Скасувати бронювання

Рисунок 4.20 — Демонстрація зміни статусу оплати

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 79   |

Діаграму алгоритму купівлі квитка дивіться в Додатку 3.

Повернемося на сторінку фільму. Після розділу вибору сеансу розташований розділ з відгуками та формою для створення нового відгуку (див. рис. 4.21).

**Відгуки(5)**

Як ви оцінюєте фільм?

10/10★

Напишіть відгук

Надіслати відгук



**Влад**  
6 червня 2024 р. 13:34

8/10★

Прикольний фільм



10/10★

Рисунок 4.21 — Розділ з відгуками

Якщо користувач неавторизований, то він побачить попередження (див. рис. 4.22).

**Відгуки(5)**

Щоб залишити відгук, будь ласка, авторизуйтеся.



**Влад**  
6 червня 2024 р. 13:34

8/10★

Прикольний фільм



**Богдан Помазанка**

10/10★

Рисунок 4.22 — Розділ з відгуками для неавторизованого користувача

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 80   |

Після заповнення форми та надсилання, у списку з’явиться новий щойно створений відгук (див. рис. 4.23).

Відгуки(6)

Як ви оцінюєте фільм?

10/10★

Напишіть відгук

Надіслати відгук

user1

8 червня 2024 р. 07:25

Це мій новий відгук!

10/10★

8/10★

Рисунок 4.22 — Демонстрація створення нового відгуку

Також можемо побачити оновлення рейтингу фільму (див. рис. 4.22 та рис. 4.11).

ГАРФІЛД У КІНО

Рейтинг: 8.8/10★(6)

Рік: 2024

Час: 1 год 41 хв

Жанр: анімація, комедія, пригоди, сімейний

Режисер: Марк Діндал

Опис Фільму:

Улюбленець мільйонів – рудий кіт Гарфілд повертається на великі екрани! Він ненавидить понеділки, обожнює лазання і кожен день проводить біля телевізора. Проте цього разу на нього чекає несподівана пригода за межами дому. Після випадкової зустрічі зі своїм давно загубленим батьком — замурзаним вуличним котом — Гарфілд і його друг Оді змушені залишити безтурботне життя та відправитися у шалену мандрівку.

Рисунок 4.22 — Демонстрація оновлення рейтингу фільма

## ВИСНОВОК ДО РОЗДІЛУ 4

Розділ 4 детально висвітлив процес розгортання та демонстрацію роботи розробленої системи. Відповідно до вимог, система була успішно розгорнута на платформі Vercel, з використанням Vercel Postgres для бази даних. Основні функції системи, такі як реєстрація користувачів, перегляд каталогу фільмів, детальна інформація про фільми та процес купівлі квитків, були продемонстровані та оцінені на предмет їх зручності та ефективності.

Показано, що розроблений веб-сайт забезпечує інтуїтивно зрозумілий інтерфейс, що дозволяє користувачам легко взаємодіяти з системою. Процес купівлі квитків організований максимально зручно, з інтерактивним вибором місць та швидким оформленням замовлення. Таким чином, система повністю відповідає сучасним вимогам до онлайн-сервісів для продажу квитків, забезпечуючи високий рівень зручності та задоволеності користувачів.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 82   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## ВИСНОВКИ

В результаті проведеного дослідження та розробки, був створений ефективний і зручний у використанні веб-сайт для продажу квитків до кінотеатру Cinemax. Кожен розділ роботи детально висвітлив різні аспекти проекту, від аналізу існуючих рішень та вимог, до розробки архітектури, вибору технологій і демонстрації готової системи.

У першому розділі було проведено аналіз існуючих рішень, що дозволило визначити ключові вимоги для успішного функціонування веб-сайту: зручний інтерфейс, широкий функціонал для користувачів та адміністраторів та висока швидкість роботи. Впровадження цих вимог забезпечило створення конкурентоспроможного продукту, здатного задовольнити потреби сучасних користувачів і підвищити ефективність роботи кінотеатру.

У другому розділі було розглянуто основні технології та алгоритми, використані для розробки веб-сайту. Було проаналізовано інструменти для фронтенду та бекенду, вибір бази даних та інструментів розробки, а також архітектуру системи. Інтеграція сервісів Auth0 для аутентифікації та Prisma для роботи з базою даних, разом із використанням Next.js, Tailwind CSS та Figma, забезпечили високу продуктивність і зручність використання веб-сайту.

У третьому розділі був описаний детальний процес розробки проекту, включаючи структуру проекту, опис бази даних, розробку серверної і клієнтської частин додатку та SEO-оптимізацію. Проект продемонстрував організованість і логічний поділ відповідальностей між різними компонентами

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 83   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

системи, що сприяє зручній підтримці та розвитку проекту. Опис бази даних та функціональності серверної частини забезпечив ефективне управління даними, а розробка клієнтської частини гарантувала інтуїтивно зрозумілий користувацький інтерфейс.

У четвертому розділі було детально описано процес розгортання системи на платформі Vercel з використанням Vercel Postgres для бази даних. Демонстрація основних функцій системи, таких як реєстрація користувачів, перегляд каталогу фільмів, детальна інформація про фільми та процес купівлі квитків, показала, що веб-сайт забезпечує високу зручність та ефективність для користувачів. Інтерактивний вибір місць та швидке оформлення замовлення підтвердили відповідність системи сучасним вимогам до онлайн-сервісів для продажу квитків.

Загалом, виконана робота дозволила створити сучасний, продуктивний та зручний у використанні веб-сайт для продажу квитків до кінотеатру Cinemax, який відповідає всім вимогам сучасного ринку та потребам користувачів.

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 84   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Fandago [Електронний ресурс] — Режим доступу до ресурсу: <https://www.fandango.com/> дата звернення (04.01.2024)
2. Movietickets.com [Електронний ресурс] - Режим доступу до ресурсу: <https://www.movietickets.com/> дата звернення (04.01.2024)
3. BookMyShow [Електронний ресурс] - Режим доступу до ресурсу: <https://in.bookmyshow.com/> дата звернення (04.01.2024)
4. The Complete Guide to 17 Key Principles of Design [Електронний ресурс] / [Sowmiya V](#) - Режим доступу до ресурсу: <https://sowmiyeh.medium.com/the-complete-guide-to-17-key-principles-of-design-and-how-they-can-help-you-create-a-compelling-7d25b913772e> дата звернення (15.01.2024)
5. Документація Next.js [Електронний ресурс] - Режим доступу до ресурсу: <https://nextjs.org/> дата звернення (06.03.2024)
6. How to Build a Fullstack App with Next.js, Prisma, and Vercel Postgres [Електронний ресурс] - Режим доступу до ресурсу: <https://vercel.com/guides/nextjs-prisma-postgres> дата звернення (01.02.2024)
7. JavaScript | MDN [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> дата звернення (05.03.2024)
8. Документація Tailwind CSS [Електронний ресурс] - Режим доступу до ресурсу: <https://tailwindcss.com/> дата звернення (15.03.2024)

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 85   |

9. Organization of Schemas [Електронний ресурс] - Режим доступу до ресурсу: <https://schema.org/docs/schemas.html> дата звернення (12.04.2024)
10. First Contentful Paint (FCP) [Електронний ресурс] - Режим доступу до ресурсу: <https://web.dev/articles/fcp> дата звернення (14.04.2024)
11. Why Prisma ORM? [Електронний ресурс] - Режим доступу до ресурсу: <https://www.prisma.io/docs/orm/overview/introduction/why-prisma> дата звернення (20.02.2024)
12. Pure CSS Custom Checkbox Style [Електронний ресурс] - Режим доступу до ресурсу: <https://moderncss.dev/pure-css-custom-checkbox-style/> дата звернення (28.03.2024)
13. Документація React [Електронний ресурс] - Режим доступу до ресурсу: <https://react.dev/> дата звернення (04.03.2024)
14. How to Deploy a Next.js App Secured By Auth0 to Vercel [Електронний ресурс] / Dan Arias - Режим доступу до ресурсу: <https://developer.auth0.com/resources/guides/web-app/nextjs/vercel-deployment> дата звернення (25.01.2024)
15. Best practices for React iframes [Електронний ресурс] / Chiamaka Umeh - Режим доступу до ресурсу: <https://blog.logrocket.com/best-practices-react-iframes/> дата звернення (24.03.2024)
16. How to Compare Arrays in PostgreSQL [Електронний ресурс] - Режим доступу до ресурсу: <https://popsql.com/learn-sql/postgresql/how-to-compare-arrays-in-postgresql> дата звернення (30.03.2024)
17. Introducing storage on Vercel [Електронний ресурс] - Режим доступу до ресурсу: <https://vercel.com/blog/vercel-storage#vercel-postgres-complex-data-made-easy> дата звернення (02.04.2024)

18. Article vs Section? - freeCodeCamp [Електронний ресурс] - Режим доступу до ресурсу: <https://forum.freecodecamp.org/t/article-vs-section/493173> дата звернення (20.03.2024)
19. What is difference between use-client & use-server in next Js 14? [Електронний ресурс] - Режим доступу до ресурсу: <https://www.linearloop.io/blog/nextjs-use-client-vs-use-Server> дата звернення 17.01.2024)
20. Документація Typescript [Електронний ресурс] - Режим доступу до ресурсу: <https://www.typescriptlang.org/> дата звернення (10.03.2024)
21. Next JS vs React : Which Framework to choose for Front end in 2024? [Електронний ресурс] /Jaydeep Patadiya - Режим доступу до ресурсу: <https://radixweb.com/blog/nextjs-vs-react> дата звернення (01.01.2024)
22. Довідник Client-server model [Електронний ресурс] - Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Client%E2%80%93server\\_model](https://en.wikipedia.org/wiki/Client%E2%80%93server_model) дата звернення (12.01.2024)
23. What is UUID V4? [Електронний ресурс] - Режим доступу до ресурсу: <https://gcore.com/dev-tools/gen-uuid-v4> дата звернення (04.03.2024)
24. Object.assign() [Електронний ресурс] - Режим доступу до ресурсу: [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/assign](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/assign) дата звернення (05.03.2024)
25. Date/Time Types [Електронний ресурс] - Режим доступу до ресурсу: <https://www.postgresql.org/docs/current/datatype-datetime.html> дата звернення (26.04.2024)
26. Vercel Postgres [Електронний ресурс] - Режим доступу до ресурсу: <https://vercel.com/docs/storage/vercel-postgres> дата звернення (30.04.2024)

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
| Зм. | Арк. | № докум. | Підпис | Дата |                    | 87   |

27.Learn Next.js [Електронний ресурс] - Режим доступу до ресурсу:  
<https://nextjs.org/learn/dashboard-app> дата звернення (04.03.2024)

28.Five Ways to Improve Your Site's Ranking (SEO) [Електронний ресурс]  
— Режим доступу до ресурсу: [https://www.mtu.edu/umc/services-  
/websites/seo/](https://www.mtu.edu/umc/services-websites/seo/) дата звернення (24.04.2024)

|     |      |          |        |      |                    |      |
|-----|------|----------|--------|------|--------------------|------|
|     |      |          |        |      | ІАЛЦ.467200.003 ПЗ | Арк. |
|     |      |          |        |      |                    | 88   |
| Зм. | Арк. | № докум. | Підпис | Дата |                    |      |

## **ДОДАТОК 1**

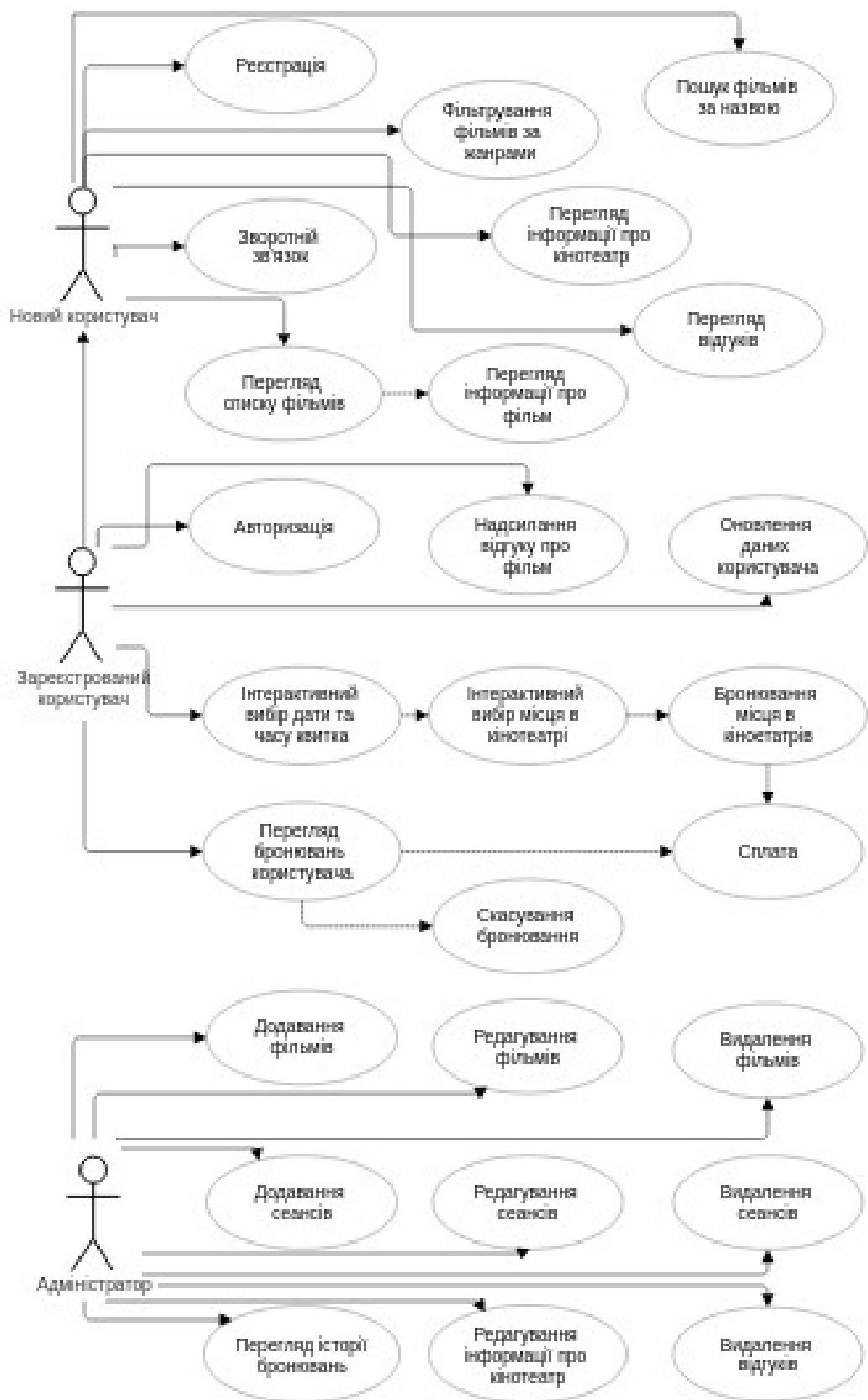
Сайт для продажу квитків до кінотеатру

**Діаграма різновиду використання (структурна схема)**

ІАЛЦ.467200.004 Д1

Аркушів 1

**Київ 2024 рж**



|           |                   |          |        |      |                                                                                                        |                                        |       |         |
|-----------|-------------------|----------|--------|------|--------------------------------------------------------------------------------------------------------|----------------------------------------|-------|---------|
|           |                   |          |        |      | ІАЛЦ.467200.004 Д1                                                                                     |                                        |       |         |
| Зм.       | Арк.              | № докум. | Підпис | Дата | <b>Сайт для продажу квитків до кінотеатру</b><br>Діаграма різновиду використання<br>(структурна схема) | Літ.                                   | Аркуш | Аркушів |
| Розробив  | Гакавий О. О.     |          |        |      |                                                                                                        |                                        | 1     | 1       |
| Перевірив | Череватенко О. В. |          |        |      |                                                                                                        | КПІ ім. Ігоря Сікорського, ФІОТ, ІП-04 |       |         |
| Реценз.   | Коган А. В.       |          |        |      |                                                                                                        |                                        |       |         |
| Н. Контр. | Волокита А. М.    |          |        |      |                                                                                                        |                                        |       |         |
| Затвердив |                   |          |        |      |                                                                                                        |                                        |       |         |

## **ДОДАТОК 2**

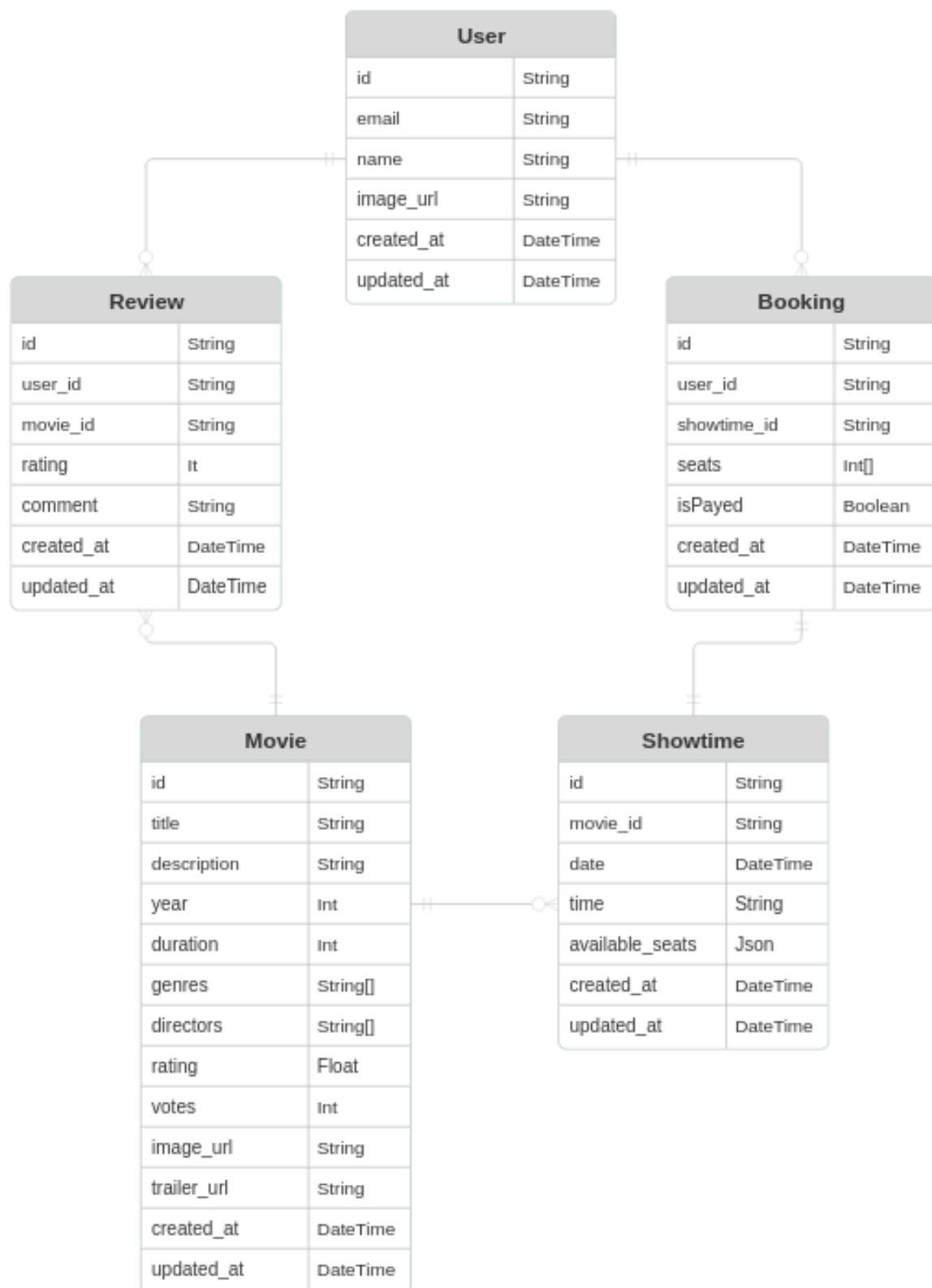
Сайт для продажу квитків до кінотеатру

**Діаграма даних (Функціональна схема)**

ІАЛЦ.467200.005 Д2

Аркушів 1

Київ 2024 р



|           |                  |          |        |      |                                                                                   |  |  |  |                                           |       |         |   |  |
|-----------|------------------|----------|--------|------|-----------------------------------------------------------------------------------|--|--|--|-------------------------------------------|-------|---------|---|--|
|           |                  |          |        |      | ІАЛЦ.467200.005 Д2                                                                |  |  |  |                                           |       |         |   |  |
|           |                  |          |        |      |                                                                                   |  |  |  |                                           |       |         |   |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до кінотеатру<br>Діаграма даних<br>(Функціональна схема) |  |  |  | Літ.                                      | Аркуш | Аркушів |   |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                                   |  |  |  |                                           |       | 1       | 1 |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                                   |  |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |   |  |
| Реценз.   | Коган А. В.      |          |        |      |                                                                                   |  |  |  |                                           |       |         |   |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                                   |  |  |  |                                           |       |         |   |  |
| Затвердив |                  |          |        |      |                                                                                   |  |  |  |                                           |       |         |   |  |

## **ДОДАТОК 3**

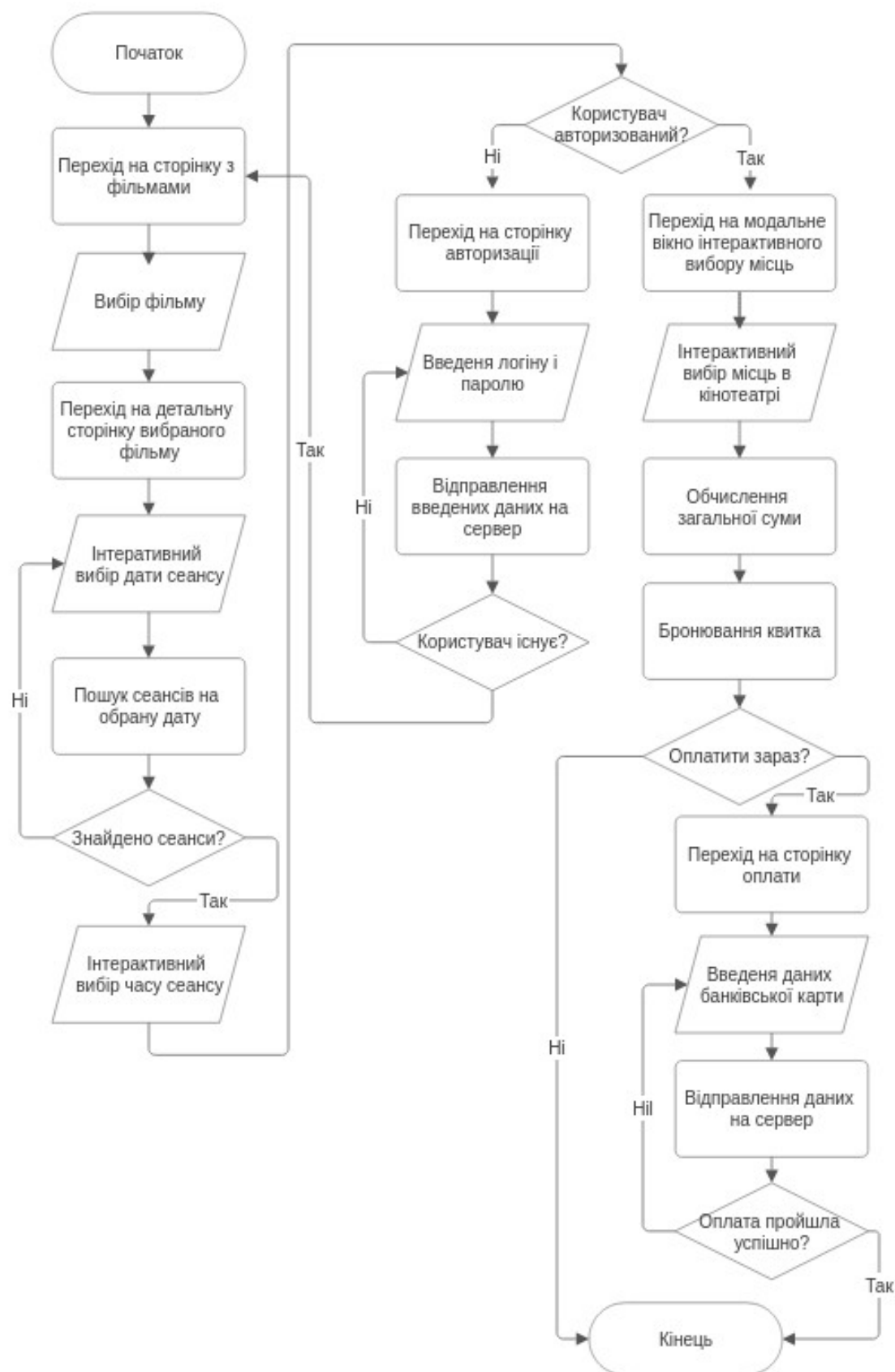
Сайт для продажу квитків до кінотеатру

**Алгоритм дій програмного забезпечення  
(Принципова схема)**

ІАЛЦ.467200.006 ДЗ

Аркушів 1

Київ 2024 р



|           |                   |          |        |      |                                                                                                       |                                        |       |         |  |  |  |
|-----------|-------------------|----------|--------|------|-------------------------------------------------------------------------------------------------------|----------------------------------------|-------|---------|--|--|--|
|           |                   |          |        |      | ІАЛЦ.467200.006 ДЗ                                                                                    |                                        |       |         |  |  |  |
| Зм.       | Арк.              | № докум. | Підпис | Дата | Сайт для продажу квитків до кінотеатру<br>Алгоритм дій програмного забезпечення<br>(Принципова схема) | Літ.                                   | Аркуш | Аркушів |  |  |  |
| Розробив  | Гакавий О. О.     |          |        |      |                                                                                                       |                                        |       |         |  |  |  |
| Перевірив | Череватенко О. В. |          |        |      |                                                                                                       |                                        |       |         |  |  |  |
| Реценз.   | Коган А. В.       |          |        |      |                                                                                                       | КПІ ім. Ігоря Сікорського, ФІОТ, ІП-04 |       |         |  |  |  |
| Н. Контр. | Волокита А. М.    |          |        |      |                                                                                                       |                                        |       |         |  |  |  |
| Затвердив |                   |          |        |      |                                                                                                       |                                        |       |         |  |  |  |
|           |                   |          |        |      |                                                                                                       | 1                                      | 1     |         |  |  |  |

## **ДОДАТОК 4**

Сайт для продажу квитків до кінотеатру

Текст програмного коду

ІАЛЦ.467200.007 Д4

Аркушів 39

**Київ 2024**

```
// app/layout.tsx:

import type { Metadata } from 'next';
import { UserProvider } from '@auth0/nextjs-auth0/client';

import './globals.css';
import Header from '@components/header';
import Footer from '@components/footer';

export const metadata: Metadata = {
  title: `Cinemax`,
  description: `Сервіс онлайн купівлі квитків у кінотеатр Cinemax`,
};

export default function RootLayout({
  children,
}: Readonly<{
  children: React.ReactNode;
}>) {
  return (
    <html lang="en">
      <UserProvider>
        <body>
          <Header />
          {children}
          <Footer />
        </body>
      </UserProvider>
    </html>
  );
}

// components/header/index.jsx

import Link from 'next/link';

import styles from './index.module.css';
import HamburgerMenu from './hamburger-menu';
import Navigation from './navigation';
import SearchField from './search-field';
import AuthButton from './auth-button';

export default function Header() {
  return (
    <header className={styles.header}>
      <div className="flex justify-between items-center mb-1">
        <div>
          <Link className={styles.logo} href="/movies">
            Cinemax
          </Link>
        </div>
        <div className="hidden lg:flex flex-1 p-1">
          <SearchField placeholder="Пошук по назві фільму" />
        </div>
        <div className="md:hidden">
          <HamburgerMenu />
        </div>
      </div>
    </header>
  );
}
```

|           |                  |          |        |      |                                                                     |       |         |  |
|-----------|------------------|----------|--------|------|---------------------------------------------------------------------|-------|---------|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                  |       |         |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текст програмного коду |       |         |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                     |       |         |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                     |       |         |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                     |       |         |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                     |       |         |  |
| Затвердив |                  |          |        |      | КПП ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                           |       |         |  |
|           |                  |          |        |      | Літ.                                                                | Аркуш | Аркушів |  |
|           |                  |          |        |      |                                                                     |       | 39      |  |

```

        <div><AuthButton /></div>
      </div>
      <div className="hidden md:block">
        <Navigation />
      </div>
    </div>
    <div className="lg:hidden pt-1">
      <SearchField placeholder="Пошук по назві фільму" />
    </div>
  </header>
);
}

// components/header/searchField.jsx

'use client';

import { useEffect, useState } from 'react';
import { usePathname, useSearchParams, useRouter } from 'next/navigation';

import styles from './search-field.module.css';

interface SearchFieldProps {
  placeholder: string,
}

export default function SearchField({ placeholder }: SearchFieldProps) {
  const router = useRouter();
  const pathname: string = usePathname();
  const searchParams = useSearchParams();
  const [inputTitle, setInputTitle] = useState<string>('');

  useEffect(() => {
    const searchedTitle = searchParams.get('title');
    if (searchedTitle) setInputTitle(searchedTitle);
  }, [searchParams]);

  useEffect(() => {
    const params = new URLSearchParams(searchParams.toString())

    if (inputTitle) params.set('title', inputTitle);
    else params.delete('title');

    router.push(`${pathname}?${params.toString()}`);
  }, [inputTitle, pathname, router, searchParams]);

  const handleChange = (e: React.ChangeEvent<HTMLInputElement>) => {
    setInputTitle(e.target.value);
  }

  const handleFocus = () => {
    if (!pathname.startsWith('/movies')) {
      router.replace('/movies');
    }
  }
}

```

|           |                  |          |        |      |                                                                    |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду | Літ.                                      | Аркуш | Аркушів |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |                                           |       |         |

```

return (
  <input
    className={styles.search}
    type="text"
    placeholder={placeholder}
    onChange={handleChange}
    onFocus={handleFocus}
    value={inputTitle}
  />
);
}

// components/header/hamburgerMenu.tsx

'use client';

import Image from 'next/image';
import { useState } from 'react';

import styles from './hamburger-menu.module.css';
import hamburgerMenuImg from '../../assets/hamburger-menu.png';
import NavLink from './nav-link';

export default function HamburgerMenu() {
  const [isOpen, setIsOpen] = useState<boolean>(false);

  const toggleMenu = () => {
    setIsOpen(!isOpen);
  };

  return (
    <>
      <button onClick={toggleMenu}>
        <Image src={hamburgerMenuImg} alt="Hamburger menu icon" width={50}
height={50} />
      </button>
      {isOpen && (
        <nav className={styles.menu} onClick={toggleMenu}>
          <ul>
            <li>
              <NavLink href="/movies">Фільми</NavLink>
            </li>
            <li>
              <NavLink href="/feedback">Зв&#39;яжіться з нами</NavLink>
            </li>
            <li>
              <NavLink href="/about">Про кінотеатр</NavLink>
            </li>
            <li>
              <NavLink href="/profile">Особистий кабінет</NavLink>
            </li>
          </ul>
        </nav>
      )}
    </>
  );
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

}

// components/header/navigation.tsx

import styles from './navigation.module.css';
import NavLink from './nav-link';
import AuthButton from './auth-button';
import ProfileButton from './profile-button';

export default function Navigation() {
  return (
    <nav className={styles.nav}>
      <ul>
        <li>
          <NavLink href='/movies'>Фільми</NavLink>
        </li>
        <li>
          <NavLink href='/feedback'>Зв'язіться з нами</NavLink>
        </li>
        <li>
          <NavLink href='/about'>Про кінотеатр</NavLink>
        </li>
        <li>
          <ProfileButton />
        </li>
        <li>
          <AuthButton />
        </li>
      </ul>
    </nav>
  );
}

// components/header/navi-link.tsx

'use client';

import Link from 'next/link';
import { usePathname } from 'next/navigation';
import { ReactNode } from 'react';

import styles from './nav-link.module.css';

interface NavLinkProps {
  href: string,
  children: ReactNode,
}

export default function NavLink({href, children}: NavLinkProps) {
  const path: string = usePathname();

  return (
    <Link
      href={href}
      className={
        path.startsWith(href)

```

|           |                  |          |        |      |                                                                    |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш | Аркушів |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |         |

```
        ? `${styles.link} ${styles.active}`
        : styles.link
    }
  >
    {children}
  </Link>
);
}

// componentns/header/auth-button.tsx

'use client';

import { useUser } from '@auth0/nextjs-auth0/client';

import styles from './nav-link.module.css';
import { LOGIN_PATH, LOGOUT_PATH } from '@/constants';

export default function AuthButton() {
  const { user, error, isLoading } = useUser();
  if (isLoading || error) return

  const path = user ? LOGOUT_PATH : LOGIN_PATH;
  const text = user ? 'Logout' : 'Login';

  return <a className={styles.link} href={path}>{text}</a>;
}

// componentns/footer/index.tsx

import Link from 'next/link';
import styles from './index.module.css';

export default function Footer() {
  return (
    <footer className={`${styles.footer} grid grid-cols-1 lg:grid-cols-2`} >
      <section>
        <h4>Навігація</h4>
        <ul className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-1">
          <li><Link href="/movies">Фільми</Link></li>
          <li><Link href="/feedback">Зв'язатися з нами</Link></li>
          <li><Link href="/about">Про кінотеатр</Link></li>
          <li><Link href="/profile">Особистий кабінет</Link></li>
        </ul>
      </section>
      <section>
        <h4>Контакти</h4>
        <ul className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-1">
          <li><Link href="mailto:address@gmail.com">address@gmail.com</Link></li>
          <li><Link href="tel:+38012345678">+380 1234 56 78</Link></li>
          <li>Пн – Нд 09:00 – 19:00</li>
        </ul>
      </section>
      <section>
        <p>©2024 Сервіс онлайн купівлі квитків у кінотеатр Cinemax</p>
      </section>
    </footer>
  );
}
```

|           |                  |          |        |      |                                                                 |  |                                           |       |
|-----------|------------------|----------|--------|------|-----------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                              |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                 |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                 |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                 |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                 |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                 |  |                                           |       |

```

    </section>
  </footer>
);
}

// app/movies.page.tsx

import Filter from '@components/filter';
import MoviesGrid from '@components/movies-grid/movies-grid';
import Pagination from '@components/pagination';
import { MOVIES_PER_PAGE } from '@constants';
import { FilterInterface } from '@interfaces/filter.interface';
import { getAvailableGenres, getMovies, getMoviesLength } from '@lib/movies';
import { Movie } from '@prisma/client';
import { Metadata } from 'next';

export const metadata: Metadata = {
  title: 'Фільми | Cinemax',
  description: 'Перегляд доступних фільмів у кінотеатрі Cinemax',
}

interface MoviesPageProps {
  searchParams: {
    page?: string,
    genres?: string,
    title?: string,
  }
}

export default async function MoviesPage({ searchParams }: MoviesPageProps) {
  const currentPage: number = Number(searchParams?.page) || 1;
  const searchFilters: FilterInterface = {
    genres: searchParams?.genres?.split(',') || [],
    title: searchParams?.title || '',
  }

  const movies: Movie[] = await getMovies(currentPage, MOVIES_PER_PAGE,
searchFilters);
  const moviesLength: number = await getMoviesLength(searchFilters);
  const totalPages: number = Math.ceil(moviesLength / MOVIES_PER_PAGE);
  const availableGenres: string[] = await getAvailableGenres();

  return (
    <main className="p-4 md:p-20 lg:px-40">
      <Filter filter={{genres: availableGenres}} />
      <MoviesGrid movies={movies} />
      <Pagination totalPages={totalPages} />
    </main>
  );
}

// componentns/filter/index.tsx

'use client';

import { useEffect, useState } from 'react';

```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |  |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|----|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |    |  |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |    |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                    |  |  |                                           |       |         | 39 |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       |         |    |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |    |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |    |  |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |  |

```

import { usePathname, useSearchParams, useRouter } from 'next/navigation';

import styles from './index.module.css';
import FilterItem from './filter-item';
import { FilterInterface, FilterKey } from '@/interfaces/filter.interface';

interface FilterProps {
  filter: FilterInterface,
}

export default function Filter({ filter }: FilterProps) {
  const router = useRouter();
  const pathname: string = usePathname();
  const searchParams = useSearchParams();
  const [selectedFilter, setSelectedFilter] =
    useState<FilterInterface>({ genres: []});

  useEffect(() => {
    const searchedGenres = searchParams.get('genres')?.split(',');
    setSelectedFilter({
      genres: searchedGenres,
    });
  }, [searchParams]);

  const { genres } = selectedFilter;

  useEffect(() => {
    const params = new URLSearchParams(searchParams.toString())
    const searchedGenres = searchParams.get('genres')?.split(',');

    if (genres && genres.length > 0) {
      params.set('genres', genres.join(','));
      if (genres.length !== searchedGenres?.length) {
        params.set('page', '1');
      }
    } else {
      params.delete('genres');
    }
    router.push(`${pathname}?${params.toString()}`);
  }, [genres, searchParams, router, pathname]);

  const handleChange = (e: React.ChangeEvent<HTMLSelectElement>) => {
    const targetValue = e.target.value;
    if (genres && genres.includes(targetValue)) return;
    const newGenres = genres ? [...genres, targetValue] : [targetValue];
    setSelectedFilter(prev => ({...prev, genres: newGenres}));
  }

  const deleteFilter = (name: FilterKey, value: string) => {
    const newFilter = {...selectedFilter};
    if (name === 'genres' && Array.isArray(newFilter.genres)) {
      newFilter.genres = newFilter.genres.filter((item: string) => item !==
value);
    }
    setSelectedFilter(newFilter);
  }

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

    }

    return (
      <article>
        <section>
          <select className={styles.select} name="filter-genre" id="filter-genre"
onChange={handleChange} defaultValue={'genre'}>
            <option disabled value={'genre'}>Жанр</option>
            {filter.genres?.sort().map(genre => (
              <option key={genre} value={genre}>{genre}</option>
            ))}
          </select>
        </section>
        <section>
          <ul className={styles.filters}>
            {selectedFilter.genres?.map(genre => (
              <li key={genre}>
                <FilterItem filter={{name: 'genres', value: genre}}
deleteFilter={deleteFilter} />
              </li>
            ))}
          </ul>
        </section>
      </article>
    );
  }
}

```

```
// components/filter/filter-item.tsx
```

```
import styles from './filter-item.module.css';
```

```
import { FilterKey } from '@/interfaces/filter.interface';
```

```
interface FilterItemProps {
  filter: {
    name: FilterKey,
    value: string,
  }
  deleteFilter: (name: FilterKey, filter: string) => void,
}

```

```
export default function FilterItem({ filter, deleteFilter }: FilterItemProps) {
  return (
    <button className={styles.filter} onClick={() => deleteFilter(filter.name,
filter.value)}>
      {filter.value}
      <span>x</span>
    </button>
  );
}

```

```
// components/movies-grid/index.tsx
```

```
import Link from 'next/link';
```

```
import styles from './movies-grid.module.css';
```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

import MovieCard from './movie-card';
import { Movie } from '@prisma/client';

interface MoviesGridProps {
  movies: Movie[],
}

export default function MoviesGrid({ movies }: MoviesGridProps) {
  return (
    <ul className={` ${styles.grid} grid grid-cols-2 lg:grid-cols-3 xl:grid-cols-4 2xl:grid-cols-6 gap-2 md:gap-4`} >
      {movies.map(movie => (
        <li key={movie.id}>
          <Link href={` /movies/${movie.id}`} >
            <MovieCard movie={movie} key={movie.id} />
          </Link>
        </li>
      ))}
    </ul>
  );
}

// componentns/movies-grid/movies-card.tsx

import Image from 'next/image';

import styles from './movie-card.module.css';
import { formatRating } from '@/lib/formatData';
import { Movie } from '@prisma/client';

interface MovieCardProps {
  movie: Movie,
}

export default function MovieCard({ movie }: MovieCardProps) {
  return (
    <article className={styles.card}>
      <header>
        <div className={styles["img-container"]} >
          <Image
            src={movie.imageUrl}
            alt={movie.title}
            fill
            sizes="(max-width: 768px) 50vw, (max-width: 1200px) 33vw, 16vw"
          />
        </div>
      </header>
      <div className={styles.info}>
        <span>★{formatRating(movie.rating)}</span>
        <h3>{movie.title}</h3>
      </div>
    </article>
  );
}

// componentns/pagination/index.tsx

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

'use client';

import { usePathname, useSearchParams } from 'next/navigation';

import styles from './index.module.css';

interface PaginationProps {
  totalPages: number,
}

export default function Pagination({ totalPages }: PaginationProps) {
  const pathname: string = usePathname();
  const searchParams = useSearchParams();
  const currentPage: number = Number(searchParams.get('page')) || 1;

  const createPageURL = (pageNumber: number) => {
    if (pageNumber < 1 || pageNumber > totalPages) return;
    const params = new URLSearchParams(searchParams.toString());
    params.set('page', pageNumber.toString());
    return `${pathname}?${params.toString()}`;
  };

  const pages = Array.from({ length: totalPages }, (_, i) => i + 1);

  return (
    <article className={styles.container}>
      <a href={createPageURL(currentPage - 1)} >{'<'}/></a>
      { pages.map(page => (
        <a className={currentPage === page ? styles.active : undefined}
key={page} href={createPageURL(page)}>{page}</a>
      ))}
      <a href={createPageURL(currentPage + 1)}>{'>'}/></a>
    </article>
  );
}

// app/movies/[moviesSlug]

import { notFound } from 'next/navigation';

import { createReview } from '@actions/reviews';
import ModalSeatsPicker from '@components/modalSeatsPicker';
import MovieDetails from '@components/movie-details';
import ShowtimePicker from '@components/showtime-picker';
import Reviews from '@components/reviews';
import { getMovie } from '@lib/movies';
import { getReviews } from '@lib/reviews';
import { getShowtime, getShowtimes } from '@lib/showtimes';
import { getUser } from '@lib/users';
import { Movie, Review, Showtime, User } from '@prisma/client';
import { createBooking } from '@actions/bookings';
import Head from 'next/head';
import { Metadata } from 'next';

```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|----|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |    |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |  |                                           |       |         |    |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |    |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       |         | 39 |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |    |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |    |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |    |

```

export async function generateMetadata({ params }: MovieDetailsPageProps):
Promise<Metadata> {
  const movie: Movie | null = await getMovie(params.movieSlug);

  return {
    title: `${movie?.title.toUpperCase()} | Cinemax`,
    description: `Перегляньте інформацію про фільм ${movie?.title}, включаючи
опис, дату виходу, режисера та жанр.`,
  }
}

interface MovieDetailsPageProps {
  params: {
    movieSlug: string,
  },
  searchParams: {
    date?: string,
    showtimeId?: string,
  }
}

export default async function MovieDetailsPage({ params, searchParams }:
MovieDetailsPageProps) {
  const selectedDateStr: string | undefined = searchParams?.date;
  const date = selectedDateStr ? new Date(selectedDateStr) : new Date();
  const showtimeId: string | undefined = searchParams?.showtimeId;

  const movie: Movie | null = await getMovie(params.movieSlug);

  if (!movie) notFound();

  let showtime: Showtime | null = null;
  if (showtimeId) showtime = await getShowtime(showtimeId);

  const showtimes: Showtime[] = await getShowtimes(movie.id, date) ?? [];
  const reviews: Review[] = await getReviews({ movieId: movie.id }) ?? [];
  const usersWithUndefined: (User | undefined)[] = await
Promise.all(reviews.map(async (review) => {
  const reviewer: User | null = await getUser(review.userId);
  if (reviewer) return reviewer;
})));
const users: User[] = usersWithUndefined.filter((user): user is User => user !
== undefined);

const jsonLd = {
  "@context": "https://schema.org",
  "@type": "Movie",
  "name": movie.title,
  "image": movie.imageUrl,
  "description": movie.description,
  "director": movie.directors.join(', '),
  "datePublished": movie.createdAt,
  "genre": movie.genres.join(', ')
}

return (

```

|           |                  |          |        |      |                                                                    |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду | Літ.                                      | Аркуш | Аркушів |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |                                           |       |         |

```

<>
  <Head>
  <script
    type="application/ld+json"
    dangerouslySetInnerHTML={{ __html: JSON.stringify(jsonLd) }}
  />
</Head>
<main className="p-5 lg:px-10">
  <MovieDetails movie={movie} />
  <ShowtimePicker showtimes={showtimes} />
  <ModalSeatsPicker showtime={showtime} movie={movie}
action={createBooking} />
  <Reviews reviews={reviews} reviewers={users}
action={createReview.bind(undefined, movie.id)} />
</main>
</>
);
}

// components/movie-details/index.tsx

import Image from 'next/image';

import styles from './index.module.css';
import Player from './player';
import { formatRating } from '@lib/formatData';
import { Movie } from '@prisma/client';

interface MovieDetailsProps {
  movie: Movie,
}

export default function MovieDetails({ movie }: MovieDetailsProps) {
  const hours = Math.floor(movie.duration / 60);
  const minutes = movie.duration % 60;

  return (
    <article className={styles.article}>
      <div className='grid grid-cols-1 md:grid-cols-2'>
        <section>
          <div className={styles["img-container"]}>
            <Image
              src={movie.imageUrl}
              alt={movie.id}
              sizes="(max-width: 768px) 100vw, 50vw"
              priority
              fill />
          </div>
        </section>
        <section className={styles.details}>
          <h2>{movie.title}</h2>
          <ul>
            <li><b>Рейтинг:</b>
{formatRating(movie.rating)}/10*({movie.votes})</li>
            <li><b>Рік:</b> {movie.year}</li>
          </ul>
        </section>
      </div>
    </article>
  );
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

        <li><b>Час:</b> {hours && `${hours} год`} {minutes && `${minutes}
хв `}</li>
        <li><b>Жанр:</b> {movie.genres.join(', ')}</li>
        <li><b>Режисер:</b> {movie.directors.join(', ')}</li>
        <li>
            <b>Опис Фільму:</b>
            <p>{movie.description}</p>
        </li>
    </ul>
</section>
</div>
<section>
    <h4>Трейлер</h4>
    <Player url={movie.trailerUrl} />
</section>
</article>
);
}

// componentns/movie-details/player.tsx

import styles from './player.module.css';

export default function url({url} : {url: string}) {
    return (
        <div className={styles.player}>
            <iframe
                width="100%"
                height="100%"
                src={url}
                title="YouTube video player"
                allow="accelerometer; autoplay; clipboard-write; encrypted-media;
gyroscope; picture-in-picture"
                referrerPolicy="strict-origin-when-cross-origin"
                allowFullScreen
            ></iframe>
        </div>
    );
}

// componentns/showtime-picker/index.tsx

'use client';

import Image from 'next/image';
import { usePathname, useRouter, useSearchParams } from 'next/navigation';
import { useEffect, useState } from 'react';
import { useUser } from '@auth0/nextjs-auth0/client';

import { getNextFormattedDays } from '@lib/formatDate';
import styles from './index.module.css';
import calendarImg from '@assets/calendar.png';
import DateData from '@interfaces/dateData.interface';
import { LOGIN_PATH } from '@constants';
import { Showtime } from '@prisma/client';

```

|           |                  |          |        |      |                                                                            |  |  |  |
|-----------|------------------|----------|--------|------|----------------------------------------------------------------------------|--|--|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                         |  |  |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | <b>Сайт для продажу квитків до<br/>кінотеатру</b><br>Текс програмного коду |  |  |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                            |  |  |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                            |  |  |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                            |  |  |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                            |  |  |  |
| Затвердив |                  |          |        |      | КПП ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                                  |  |  |  |

```

interface ShowtimePickerProps {
  showtimes: Showtime[],
}

export default function ShowtimePicker({ showtimes }: ShowtimePickerProps) {
  const [selectedDate, setSelectedDate] = useState<DateData | null>(null);
  const router = useRouter();
  const pathname: string = usePathname();
  const searchParams = useSearchParams();
  const { user, error, isLoading } = useUser();

  const datesData = getNextFormattedDays(new Date(), 7);

  useEffect(() => {
    setSelectedDate(datesData[0]);
  }, []);

  useEffect(() => {
    const currentSearchParams = new URLSearchParams(searchParams.toString());
    if (selectedDate) {
      currentSearchParams.set('date', selectedDate.date.toString());
    } else {
      currentSearchParams.delete('date');
    }
    const newUrl = `${pathname}?${currentSearchParams.toString}`;
    router.replace(`${newUrl}`, { scroll: false });
  }, [selectedDate, pathname, searchParams, router]);

  const handleDateClick = (date: DateData) => {
    setSelectedDate(date);
  }

  const handleTimeClick = (showtime: Showtime) => {
    if (!isLoading && !error && !user) router.push(LOGIN_PATH);
    const currentSearchParams = new URLSearchParams(searchParams.toString());
    currentSearchParams.set('showtimeId', showtime.id);
    const newUrl = `${pathname}?${currentSearchParams.toString}`;
    router.replace(`${newUrl}`, { scroll: false });
  }

  return (
    <article className={styles.article}>
      <h3>Виберіть дату та час сенсу</h3>
      <section>
        <ul className={styles.list}>
          {datesData.map(dateData => (
            <li
              key={dateData.dateAndMonth}
              onClick={() => handleDateClick(dateData)}
              className={searchParams.get('date') === dateData.date.toString() ?
styles.active : ''}
            >
              <span>{dateData.dateAndMonth}</span>
              <span><b>{dateData.dayOfWeek}</b></span>
            </li>
          ))}
        </ul>
      </section>
    </article>
  )
}

```

|           |                  |          |        |      |                                                                     |       |         |  |
|-----------|------------------|----------|--------|------|---------------------------------------------------------------------|-------|---------|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                  |       |         |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текст програмного коду |       |         |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                     |       |         |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                     |       |         |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                     |       |         |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                     |       |         |  |
| Затвердив |                  |          |        |      | КПП ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                           |       |         |  |
|           |                  |          |        |      | Літ.                                                                | Аркуш | Аркушів |  |
|           |                  |          |        |      |                                                                     |       | 39      |  |

```

    </ul>
    <div className={styles.calendar}>
      <Image src={calendarImg} alt="Calendar icon" sizes="(max-width: 768px)
20vw, 8vw" fill/>
    </div>
  </section>
  <section>
    <h4>{selectedDate?.dateAndMonth}, <b>{selectedDate?.dayOfWeek}</b></h4>
    <ul className={styles.list}>
      {showtimes.length > 0 ? (
        showtimes.map(showtime => (
          <li
            key={showtime.id}
            onClick={() => handleTimeClick(showtime)}
          >
            <span>{showtime.time}</span>
          </li>
        ))
      ) : <p>Немає сеансів на вибраний день.</p>
    }
  </ul>
</section>
</article>
)
}

// componentns/modal-seats-picker/index.tsx

'use client';

import { useRouter, useSearchParams, usePathname } from 'next/navigation';
import Link from 'next/link';

import styles from './index.module.css';
import { ChangeEvent, useEffect, useState } from 'react';
import { formatDateAndTime } from '@lib/formatDate';
import { TICKET_PRICE } from '@constants';
import SeatsPickerForm from './seats-picker-form';
import { Movie, Showtime } from '@prisma/client';

interface ModalSeatsPickerProps {
  showtime: Showtime | null,
  movie: Movie | null,
  action: (showtimeId: string, seats: string[], prevState: { bookingId:
string }, formData: FormData) => Promise<{ bookingId: string }>,
}

export default function ModalSeatsPicker({ showtime, movie, action }:
ModalSeatsPickerProps) {
  const router = useRouter();
  const pathname: string = usePathname();
  const searchParams = useSearchParams();

  const [selectedSeats, setSelectedSeats] = useState<string[]>([]);

  useEffect(() => {

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

    setSelectedSeats([]);
  }, [showtime]);

  if (!showtime || !movie || !action) return null;

  let date = showtime && formatDateAndTime(showtime!.date).split('
').slice(0,2).join(' ');

  const handleChange = (e: ChangeEvent<HTMLInputElement>) => {
    const { checked, value } = e.target;
    if (checked) {
      setSelectedSeats(prev => [...prev, value]);
    } else {
      setSelectedSeats(selectedSeats?.filter(seat => seat !== value));
    }
  }

  return (
    <dialog className={styles.modal}>
      <div className={styles.content}>
        <Link href={pathname} className="float-right">
          <button type="button" className="bg-red-500 text-white p-3">x</button>
        </Link>
        <h3 className="text-center">{movie?.title.toUpperCase()}</h3>
        <time dateTime={showtime.date.toString()}>{date} {showtime.time}</time>
        <div className="flex flex-col items-center mt-10">
          <div className={styles.svg}>
            <svg viewBox="0 0 1440 320">
              <path
                fill="#ddd6cb"
                d="M0,160L120,138.7C240,117,480,75,720,74.7C960,75,1200,117,1320,
138.7L1440,160L1440,0L1320,0C1200,0,960,0,720,0C480,0,240,0,120,0L0,0Z"
              ></path>
            </svg>
          </div>
          <div className={styles.screen}>Екран</div>
          <SeatsPickerForm
            availableSeats={JSON.parse(showtime.availableSeats as string)}
            handleChange={handleChange}
            action={action.bind(undefined, showtime.id, selectedSeats)}
            disabled={selectedSeats.length <= 0}
          />
          {selectedSeats.length > 0 && (
            <div className={styles.summary}>
              <b>{selectedSeats.length} вибраних квитків:</b>
{selectedSeats.join(', ')} <br />
              <b>Загальна сума:</b> {TICKET_PRICE * selectedSeats.length} грн
            </div>
          )}
        </div>
      </div>
    </dialog>
  );

```

|           |                  |          |        |      |                                                                     |       |         |  |
|-----------|------------------|----------|--------|------|---------------------------------------------------------------------|-------|---------|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                  |       |         |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текст програмного коду |       |         |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                     |       |         |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                     |       |         |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                     |       |         |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                     |       |         |  |
| Затвердив |                  |          |        |      | КПП ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                           |       |         |  |
|           |                  |          |        |      | Літ.                                                                | Аркуш | Аркушів |  |
|           |                  |          |        |      |                                                                     |       | 39      |  |

```

};

// components/modal-seats-picker/seats-picker-form.tsx

'use client';
import { useFormState, useFormStatus } from 'react-dom';
import styles from './seats-picker-form.module.css';
import { useRouter } from 'next/navigation';
import { useEffect } from 'react';

interface SeatsPickerForm {
  availableSeats: boolean[][],
  handleChange: (e: React.ChangeEvent<HTMLInputElement>) => void,
  action: (prevState: { bookingId: string }, formData: FormData) =>
  Promise<{ bookingId: string }>,
  disabled: boolean,
}

export default function SeatsPickerForm({ availableSeats, handleChange, action,
disabled }: SeatsPickerForm) {
  const router = useRouter();
  const [state, formAction] = useFormState(action, { bookingId: '' });
  const { pending } = useFormStatus();

  useEffect(() => {
    if (state.bookingId) {
      router.push(`/payment/${state.bookingId}`);
    }
  }, [state.bookingId, router]);

  return (
    <form className={styles.hall} action={formAction}>
      {availableSeats?.map((row, rowIndex) => {
        return (
          <div key={`row-${rowIndex}`} className={styles.row}>
            {row.map((seat, seatIndex) => (
              <input
                key={`seat-${rowIndex}-${seatIndex}`}
                className={seat ? styles.seat : `${styles.seat} ${styles.booked}`}
                type="checkbox"
                value={`-${rowIndex+1}-${seatIndex}`}
                onChange={handleChange}
                title={seat ? `Місце ${rowIndex+1}-${seatIndex}` :
'Заброньовано'}
                disabled={!seat}
              />
            ))}
          </div>
        );
      })}
      <div className="flex justify-center">
        <button type="submit" disabled={disabled}>
          {pending ? 'Завантаження...' : 'Далі'}
        </button>
      </div>
    </form>
  );
}

```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                    |  |  |                                           |       |         |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |

```

    </form>
  );
}

// components/reviews/index.tsx

'use client';

import { useUser } from '@auth0/nextjs-auth0/client';

import styles from './index.module.css';
import ReviewsList from './reviews-list';
import ReviewForm from './review-form';
import { Review, User } from '@prisma/client';

interface ReviewsProps {
  reviews: Review[] | null,
  reviewers: User[] | null,
  action: (prevState: { errorMessages: string[] }, formData: FormData) =>
  Promise<{ errorMessages: string[] }>,
}

export default function Reviews({ reviews, reviewers, action }: ReviewsProps) {
  const { user } = useUser();
  return (
    <article className={` ${styles.article} px-2 md:px-5 lg:md-10 xl:px-20 2xl:px-40`} >
      <h3>Відгуки({reviews?.length || 0})</h3>
      {user ? (
        <ReviewForm action={action} />
      ) : (
        <div className="warning">
          <h4>Щоб залишити відгук, будь ласка, авторизуйтеся.</h4>
        </div>
      )}
      <ReviewsList reviews={reviews} reviewers={reviewers} />
    </article>
  );
}

// components/reviews/review-form.tsx

'use client';

import styles from './review-form.module.css';
import { useState } from 'react';
import { useFormState, useFormStatus } from 'react-dom';

interface ReviewFormProps {
  action: (prevState: { errorMessages: string[] }, formData: FormData) =>
  Promise<{ errorMessages: string[] }>,
}

export default function ReviewForm({ action }: ReviewFormProps) {
  const [state, formAction] = useFormState(action, {errorMessages: [] });
  const { pending } = useFormStatus();

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

const [rating, setRating] = useState<number>(10);
const [comment, setComment] = useState<string>('');

const handleChangeRating = (e: React.ChangeEvent<HTMLInputElement>) => {
  setRating(Number(e.currentTarget.value));
}

const handleChangeComment = (e: React.ChangeEvent<HTMLTextAreaElement>) => {
  setComment(e.currentTarget.value);
}

const handleSubmit = (formData: FormData) => {
  formAction(formData);
  setRating(10);
  setComment('');
}

return (
  <form className={styles.form} action={handleSubmit}>
    <h4>Як ви оцінюєте фільм?</h4>
    <div className="flex justify-space mb-3">
      <input
        className={styles.slider}
        type="range"
        name="rating"
        id="rating"
        min={0}
        max={10}
        value={rating}
        onChange={handleChangeRating}
        style={{ '--value': `${(rating) * 10}%` }}
      />
      <span>{rating}/10★</span>
    </div>
    <h4>Напишіть відгук</h4>
    <textarea
      className="mb-3"
      name="comment"
      id="comment"
      rows={10}
      value={comment}
      onChange={handleChangeComment}
    ></textarea>
    <div className="flex justify-end">
      <button type="submit">
        {pending ? 'Надсилання...' : 'Надіслати відгук'}
      </button>
    </div>
    {state.errorMessages.length > 0 && (
      <ul className={styles.errors}>
        {state.errorMessages.map(errorMsg => (
          <li key={errorMsg}>{errorMsg}</li>
        ))}
      </ul>
    )}
  </form>
)

```

|           |                  |          |        |      |                                                                     |       |         |  |
|-----------|------------------|----------|--------|------|---------------------------------------------------------------------|-------|---------|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                  |       |         |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текст програмного коду |       |         |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                     |       |         |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                     |       |         |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                     |       |         |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                     |       |         |  |
| Затвердив |                  |          |        |      | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                           |       |         |  |
|           |                  |          |        |      | Літ.                                                                | Аркуш | Аркушів |  |
|           |                  |          |        |      |                                                                     |       | 39      |  |

```

    );
}

// components/reviews/review-list.tsx

import Image from 'next/image';
import styles from './reviews-list.module.css';
import { formatDateAndTime } from '@lib/formatDate';
import { Review, User } from '@prisma/client';

interface ReviewsListProps {
  reviews: Review[] | null,
  reviewers: User[] | null,
}

export default function ReviewsList({ reviews, reviewers }: ReviewsListProps) {
  return (
    <ul>
      {reviews && reviews.map(review => {
        const reviewer: User = reviewers?.find(reviewer => reviewer.id ===
review.userId)!;
        return(
          <li className={` ${styles.item}` } key={review.id} >
            <section className={styles.info}>
              <span className={styles.rating}>{review.rating}/10</span>
              <div className={styles.img}>
                <Image src={reviewer.imageUrl} alt={reviewer.name} sizes="(max-
width: 768px) 15vw, 5vw" fill />
              </div>
              <h4>{reviewer.name}</h4>
              <time dateTime={review.createdAt.toString()}>
                {formatDateAndTime(review.createdAt)}
              </time>
            </section>
            {review?.comment &&
              <section className={styles.comment}>
                {review.comment}
              </section>
            }
          </li>
        )
      })}
    </ul>
  )
}

// app/payment/[paymentSlug]/page.tsx

import styles from './page.module.css';
import { updatePaymentStatus } from '@actions/bookings';
import PaymentForm from '@components/payment-form';
import { getBooking } from '@lib/booking';
import { formatDateAndTime } from '@lib/formatDate';
import { getMovie } from '@lib/movies';
import { getShowtime } from '@lib/showtimes';
import { Booking, Movie, Showtime } from '@prisma/client';
import { notFound } from 'next/navigation';

```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                    |  |  |                                           |       |         |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |

```

interface PaymentPageProps {
  params: {
    paymentSlug: string,
  },
}

export default async function PaymenPage({ params }: PaymentPageProps) {
  const { paymentSlug } = params;

  const booking: Booking | null = await getBooking(paymentSlug);

  if (!booking) notFound();

  const showtime: Showtime | null = await getShowtime(booking.showtimeId);
  if (!showtime) throw new Error('Something went wrong');
  const movie: Movie | null = await getMovie(showtime.movieId);

  return (
    <main className="p-5 lg:px-10 py-20 xl:py-40">
      <article className={` ${styles.article} w-full md:w-3/4 lg:w-2/3 xl:w-1/2
p-10 lg:p-40`} >
        <section>
          <h2>{movie?.title.toUpperCase()}</h2>
          <time dateTime={showtime.date.toString()} >
            {formatDateAndTime(showtime.date)}
          </time>
        </section>
        <PaymentForm
          price={booking.price}
          action={updatePaymentStatus.bind(undefined, booking)}
        />
      </article>
    </main>
  );
}

// componentns/payment-form/index.tsx

'use client';

import Link from 'next/link';
import styles from './index.module.css';
import { useRouter } from 'next/navigation';
import { useEffect } from 'react';
import { useFormState, useFormStatus } from 'react-dom';

interface PaymentFormProps {
  price: number,
  action: (prevState: { isPayed: boolean }, formData: FormData) =>
Promise<{ isPayed: boolean }>,
}

export default function PaymentForm({ price, action }: PaymentFormProps) {
  const router = useRouter();
  const [state, formAction] = useFormState(action, { isPayed: false });

```

|           |                  |          |        |      |                                                                    |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш | Аркушів |
| Перевірів | Череватенко О.В. |          |        |      |                                                                    |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |         |

```

const { pending } = useFormStatus();

useEffect(() => {
  if (state.isPayed) {
    router.replace('/profile');
  }
}, [state.isPayed, router]);

return (
  <form className={styles.form} action={formAction}>
    <div>
      <label htmlFor="card-number">Номер картки</label>
      <input
        type="number"
        name="card-number"
        id="card-number"
        minLength={16}
        maxLength={16}
        placeholder='0000 0000 0000 0000'
      />
    </div>
    <div className="flex flex-row">
      <div>
        <label htmlFor="validity">Термін дії</label>
        <input
          type="number"
          name="validity"
          id="validity"
          minLength={4}
          maxLength={4}
          placeholder='24/07'
        />
      </div>
      <div>
        <label htmlFor="cvv2">CVV2</label>
        <input
          type="number"
          name="cvv2"
          id="cvv2"
          maxLength={3}
          minLength={3}
          placeholder='123'
        />
      </div>
    </div>
    <div>
      <button type="submit">
        {pending ? 'Зачекайте...' : `Оплатити ${price}`} грн
      </button>
    </div>
    <div className="flex">
      <Link href={'/profile'}>Оплатити пізніше</Link>
    </div>
  </form>
)
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```
// app/profile/page.tsx

import { getSession, Session } from '@auth0/nextjs-auth0';

import styles from './page.module.css';
import Auth0User from '@/interfaces/auth0User.interface';
import { getUserByEmail } from '@/lib/users';
import { User } from '@prisma/client';
import { deleteBooking, getBookings } from '@/lib/booking';
import { getShowtime } from '@/lib/showtimes';
import { getMovie } from '@/lib/movies';
import { getReviews } from '@/lib/reviews';
import UserInfo from '@/components/profile/user-info';
import UserBooking from '@/components/profile/user-booking';

export default async function Profile() {
  const session: Session | null | undefined = await getSession();
  let user: User | null = null;
  if (session && session.user) {
    const auth0User = session.user as Auth0User;
    try {
      user = await getUserByEmail(auth0User.email);
    } catch (e) {
      console.log(e);
    }
  }

  if (!user) throw new Error('401 Unauthorized');

  const bookings = await getBookings(user.id);
  const reviews = await getReviews({ userId: user.id });

  return (
    <main className="p-10 lg:px-10 xl:px-60">
      <article className={styles.article}>
        <button>Редагувати профіль</button>
        <h2>Мій профіль</h2>
        <UserInfo user={user} amountOfReviews={reviews?.length ?? 0} />
      </article>
      <article className={styles.article}>
        <h2>Мої бронювання({bookings?.length})</h2>
        {bookings?.map(async booking => {
          const showtime = await getShowtime(booking.showtimeId);
          const movie = await getMovie(showtime!.movieId);
          return <UserBooking
            key={booking.id}
            booking={booking}
            movie={movie!}
            showtime={showtime!}
          />
        })}
      </article>
    </main>
  );
}
```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```
// components/profile/user-info.tsx

import Image from 'next/image';

import styles from './user-info.module.css';
import { formatDateAndTime } from '@lib/formatDate';
import { User } from '@prisma/client';

interface UserInfoProps {
  user: User,
  amountOfReviews: number,
}

export default function UserInfo({ user, amountOfReviews }: UserInfoProps) {
  return (
    <div className={styles.container}>
      <Image src={user.imageUrl} alt="User's picture" width={200} height={200}
priority />
      <ul>
        <li><b>Електронна пошта: </b>{user.email}</li>
        <li><b>Ім'я: </b>{user.name}</li>
        <li><b>Реєстрація: </b>{formatDateAndTime(user.createdAt)}</li>
        <li><b>Кількість відгуків </b>{amountOfReviews}</li>
      </ul>
    </div>
  );
}

// components/profile/user-booking.tsx

import Image from 'next/image';
import Link from 'next/link';

import styles from './user-bookings.module.css';
import { Booking, Movie, Showtime } from '@prisma/client';
import { formatDateAndTime } from '@lib/formatDate';

interface UserBookingsProps {
  booking: Booking,
  movie: Movie,
  showtime: Showtime,
}

export default function UserBooking({ booking, movie, showtime }:
UserBookingsProps) {
  return (
    <section className={styles.section}>
      <div>
        <div className={styles.img}>
          <Image src={movie!.imageUrl} alt={movie!.title} priority fill />
        </div>
        <div className={styles.info}>
          <h3>{movie?.title}</h3>
          <ul>
```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                    |  |  |                                           |       |         |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |

```

    <li>
      <b>Дата і час сеансу: </b>
      <time dateTime={showtime!.date.toString()}>
        {formatDateAndTime(showtime!.date)}
      </time>
    </li>
    <li>
      <b>Ціна: </b>{booking.price}
    </li>
    <li>
      <b>Місця в залі: </b>{booking.seats.join(', ')}
    </li>
    <li>
      <b>Статус оплати: </b>{booking.isPaid ? 'Оплачено' : 'Не
оплачено'}
    </li>
  </ul>
</div>
</div>
<div className={styles.buttons}>
  {!booking.isPaid && (
    <Link href={`\payment/${booking.id}`}>
      Оплатити
    </Link>
  )}
  <div><button>Скасувати бронювання</button></div>
</div>
</section>
)
}

```

```
// actions/reviews.tsx
```

```
'use server';
```

```
import { revalidatePath } from 'next/cache';
import { getSession, Session } from '@auth0/nextjs-auth0';
```

```
import { saveReview } from '@lib/reviews';
import Auth0User from '@interfaces/auth0User.interface';
import { getUserByEmail } from '@lib/users';
import { updateRating } from '@lib/movies';
import { User } from '@prisma/client';
```

```
interface createReviewState {
  errorMessages: string[],
}
```

```
export async function createReview(movieId: string, prevState:
createReviewState, formData: FormData): Promise<createReviewState> {
  const session: Session | null | undefined = await getSession();
  let user: User | null = null;
  if (session && session.user) {
    const auth0User = session.user as Auth0User;
    try {
      user = await getUserByEmail(auth0User.email);
    }
  }
}
```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

    } catch(e) {
      console.log(e);
    }
  }
  const rating: number = Number(formData.get('rating')?.toString());
  const comment: string | null = formData.get('comment')?.toString() ?? null;

  const errorMessages: string[] = [];

  if (!rating) {
    errorMessages.push('Rating is required.')
  }

  if (rating < 0 || rating > 10) {
    errorMessages.push('Rating must be greater than 0 and less than 10.')
  }

  if (errorMessages.length > 0) return { errorMessages };
  if (!user) throw new Error('401 Unauthorized')
  try {
    const review = await saveReview({
      userId: user?.id,
      movieId,
      rating,
      comment,
    });
    if (review) {
      await updateRating(movieId, rating);
    }
  } catch (e) {
    console.log(e);
  }

  revalidatePath(`/movies/${movieId}`, 'page');
  return { errorMessages };
}

// actions/bookings.tsx

'use server';

import { revalidatePath } from 'next/cache';
import { getSession, Session } from '@auth0/nextjs-auth0';

import Auth0User from '@/interfaces/auth0User.interface';
import { getUserByEmail } from '@/lib/users';
import { Booking, User } from '@prisma/client';
import { saveBooking, updateBooking } from '@/lib/booking';
import { TICKET_PRICE } from '@/constants';

interface createBookingState {
  bookingId: string,
}

```

|           |                  |          |        |      |                                                                    |  |                                           |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш   |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | Аркушів |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |                                           | 39      |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |         |

```

export async function createBooking(showtimeId: string, seats: string[],
prevState: createBookingState, formData: FormData): Promise<createBookingState>
{
  const session: Session | null | undefined = await getSession();
  let user: User | null = null;
  if (session && session.user) {
    const auth0User = session.user as Auth0User;
    try {
      user = await getUserByEmail(auth0User.email);
    } catch(e) {
      console.log(e);
    }
  }

  if (!user) throw new Error('401 Unauthorized');
  let booking: Booking | null = null;
  try {
    booking = await saveBooking({
      userId: user?.id,
      showtimeId,
      seats,
      price: seats.length * TICKET_PRICE,
      isPayed: false,
    });
  } catch (e) {
    console.log(e);
  }
  if (!booking) return prevState;
  revalidatePath(`/}`, 'layout');
  return { bookingId: booking.id };;
}

interface updatePaymentStateStatus {
  isPayed: boolean,
}

export async function updatePaymentStatus(
  booking: Booking,
  prevState: updatePaymentStateStatus,
  formData: FormData
): Promise<updatePaymentStateStatus> {
  const updatedBooking: Booking | null = await updateBooking(booking.id,
{ isPayed: !booking.isPayed});
  if (!updatedBooking) return prevState;
  revalidatePath('/profile', 'page');
  return { isPayed: updatedBooking?.isPayed };
}

// lib/movies.ts

import prisma from '../lib/prisma';
import { Movie } from '@prisma/client';
import { FilterInterface } from '@interfaces/filter.interface';
import MovieDto from '@dto/movie.dto';

```

|           |                  |          |        |      |                                                                            |       |         |  |
|-----------|------------------|----------|--------|------|----------------------------------------------------------------------------|-------|---------|--|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                         |       |         |  |
| Зм.       | Арк.             | № докум. | Підпис | Дата | <b>Сайт для продажу квитків до<br/>кінотеатру</b><br>Текс програмного коду |       |         |  |
| Розробив  | Гакавий О. О.    |          |        |      |                                                                            |       |         |  |
| Перевірив | Череватенко О.В. |          |        |      |                                                                            |       |         |  |
| Реценз.   | Коган А.В.       |          |        |      |                                                                            |       |         |  |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                            |       |         |  |
| Затвердив |                  |          |        |      | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04                                  |       |         |  |
|           |                  |          |        |      | Літ.                                                                       | Аркуш | Аркушів |  |
|           |                  |          |        |      |                                                                            |       | 39      |  |

```

export async function getMovies(page?: number, limit?: number, filter?:
FilterInterface): Promise<Movie[]> {
  let offset: number = 0;

  if (page && limit) {
    offset = (page - 1) * limit;
  }

  try {
    return await prisma.movie.findMany({
      where: {
        genres: {
          hasEvery: filter?.genres,
        },
        title: {
          contains: filter?.title?.toLowerCase(),
        }
      },
      take: limit,
      skip: offset,
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch movies data.');
```

```

  }
}

export async function getMoviesLength(filter?: FilterInterface): Promise<number>
{
  try {
    return await prisma.movie.count({
      where: {
        genres: {
          hasEvery: filter?.genres,
        },
        title: {
          contains: filter?.title?.toLowerCase(),
        }
      },
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch movies data length.');
```

```

  }
}

export async function getMovie(id: string): Promise<Movie | null> {
  try {
    return await prisma.movie.findUnique({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to fetch movie data with id${id}`);
  }
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

export async function updateMovie(id: string, movie: Partial<MovieDto>):
Promise<Movie> {
  try {
    return await prisma.movie.update({
      where: { id },
      data: movie
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to update movie data with id ${id}.`);
  }
}

export async function deleteMovie(id: string): Promise<Movie> {
  try {
    return await prisma.movie.delete({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to delete movie data with id ${id}.`);
  }
}

export async function getAvailableGenres(): Promise<string[]> {
  try {
    const data = await prisma.movie.findMany({
      distinct: ['genres'],
      select: { genres: true },
    });
    const splitedGenres = data.map(item => item.genres).flat();
    const uniqueGenres = Array.from(new Set(splitedGenres));
    return uniqueGenres;
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch movies genres.');
```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

    },
  });
} catch (error) {
  console.error('Database Error:', error);
  throw new Error(`Failed to update movie rating with id${id}`);
}
}

// lib/showtimes.ts

import { Showtime } from '@prisma/client';
import prisma from '../lib/prisma';
import ShowtimeDto from '@dto/showtime.dto';

export async function getShowtimes(movieId: string, date?: Date):
Promise<Showtime[]> {
  let formattedDate: Date | undefined;
  if (date) formattedDate = new Date(date.toISOString().split('T')[0]);

  try {
    const data = await prisma.showtime.findMany({
      where: {
        movieId,
        date: formattedDate,
      }
    });
    return data;
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch showtimes data.');
```

```

  }
}

export async function getShowtime(id: string): Promise<Showtime | null> {
  try {
    return await prisma.showtime.findUnique({ where: { id: id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch showtimes data.');
```

```

  }
}

export async function saveShowtime(showtime: ShowtimeDto): Promise<Showtime |
null> {
  try {
    return await prisma.showtime.create({ data: showtime });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to save showtime data.');
```

```

  }
}

export async function updateShowtime(id: string, showtime:
Partial<ShowtimeDto>): Promise<Showtime> {
  try {
    return await prisma.showtime.update({
```

|           |                  |          |        |      |                                                                    |  |                                           |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш   |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | Аркушів |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |                                           | 39      |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |         |

```

        where: { id },
        data: showtime
    });
} catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to update showtime data with id ${id}.`);
}
}

export async function deleteShowtime(id: string): Promise<Showtime> {
    try {
        return prisma.showtime.delete({ where: { id } });
    } catch (error) {
        console.error('Database Error:', error);
        throw new Error(`Failed to delete showtime data with id ${id}.`);
    }
}

// lib/users.ts

import prisma from '../lib/prisma';
import { User } from '@prisma/client';
import Auth0User from '@interfaces/auth0User.interface';
import UserDto from '@dto/user.dto';

export async function getUsers(): Promise<User[]> {
    try {
        return await prisma.user.findMany();
    } catch (error) {
        console.error('Database Error:', error);
        throw new Error('Failed to fetch users data.');
```

|           |                  |          |        |      |                                                                    |  |                                           |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш   |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | Аркушів |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |                                           | 39      |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |         |

```

    if (data) return true;
    return false;
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to fetch user data with email ${email}.`);
  }
}

export async function saveUser(user: Auth0User): Promise<User> {
  const name = user.name === user.email ? user.nickname : user.name;
  try {
    return await prisma.user.create({
      data: {
        name: name,
        email: user.email,
        imageUrl: user.picture,
      }
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to save user data in database.');
```

```

  }
}

export async function updateUser(id: string, userData: Partial<UserDto>):
Promise<User> {
  try {
    return await prisma.user.update({
      where: { id },
      data: userData,
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to update user data with id ${id}.`);
  }
}

export async function deleteUser(id: string): Promise<User> {
  try {
    return await prisma.user.delete({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to delete user data with id ${id}.`);
  }
}

// lib/reviews.ts
import { Review } from '@prisma/client';
import prisma from '../lib/prisma';
import ReviewDto from '@dto/review.dto';

interface getReviewsParams {
  movieId?: string,
  userId?: string,

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

}

export async function getReviews({ movieId, userId }: getReviewsParams):
Promise<Review[]> {
  try {
    return await prisma.review.findMany({
      where: {
        movieId: movieId,
        userId: userId,
      },
      orderBy: {
        createdAt: 'desc'
      }
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch reviews data.');
```

```

  }
}

export async function saveReview(review: ReviewDto): Promise<Review> {
  try {
    return await prisma.review.create({ data: review });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to save review data.');
```

```

  }
}

export async function updateReview(id: string, review: Partial<ReviewDto>):
Promise<Review> {
  try {
    return await prisma.review.update({
      where: { id },
      data: review
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to update review data with id ${id}.`);
  }
}

export async function deleteReview(id: string): Promise<Review> {
  try {
    return await prisma.review.delete({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to delete review data with id ${id}.`);
  }
}

// lib/bookings.ts

import { Booking } from '@prisma/client';
import prisma from '../lib/prisma';
import BookingDto from '@dto/booking.dto';
```

|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |  |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |  |                                           |       |         |

```

export async function getBookings(userId?: string): Promise<Booking[]> {
  try {
    return await prisma.booking.findMany({
      where: {
        userId,
      },
      orderBy: {
        createdAt: 'desc'
      }
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch bookings data.');
```

```

  }
}

export async function getBooking(id: string): Promise<Booking | null> {
  try {
    return await prisma.booking.findUnique({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to fetch bookings data.');
```

```

  }
}

export async function saveBooking(booking: BookingDto): Promise<Booking> {
  try {
    return await prisma.booking.create({ data: booking });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error('Failed to save booking data.');
```

```

  }
}

export async function updateBooking(id: string, booking: Partial<BookingDto>):
Promise<Booking> {
  try {
    return await prisma.booking.update({
      where: { id },
      data: booking
    });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to update booking data with id ${id}.`);
  }
}

export async function deleteBooking(id: string): Promise<Booking> {
  try {
    return await prisma.booking.delete({ where: { id } });
  } catch (error) {
    console.error('Database Error:', error);
    throw new Error(`Failed to delete booking data with id ${id}.`);
  }
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

// lib/formatDate.ts

import DateData from "@/interfaces/dateData.interface";

export function getNextFormattedDays(startDate: Date, amount: number):
DateData[] {
    const datesData: DateData[] = [];

    for (let i = 0; i < amount; i++) {
        const curDate: Date = new Date(startDate);
        curDate.setDate(startDate.getDate() + i);

        const dateAndMonth: string = curDate.toLocaleDateString('uk-UA', { day:
'numeric', month: 'long' });
        let dayOfWeek: string = curDate.toLocaleDateString('uk', { weekday:
'long' });

        datesData.push({ date: curDate, dateAndMonth, dayOfWeek });
    }
    return datesData;
}

export function formatDateAndTime(date: Date) {
    const optionsDate: Intl.DateTimeFormatOptions = { day: 'numeric', month:
'long', year: 'numeric' };
    const optionsTime: Intl.DateTimeFormatOptions = { hour: '2-digit', minute: '2-
digit' };

    const formattedDate: string = date.toLocaleDateString('uk-UA', optionsDate);
    const formattedTime: string = date.toLocaleTimeString('uk-UA', optionsTime);

    return `${formattedDate} ${formattedTime}`;
}

// lib/formatData.ts

export const formatRating = (rating: number) => {
    if (Number.isInteger(rating)) {
        return rating;
    } else {
        return rating.toFixed(1);
    }
}

// dto/movie.dto.ts

export default interface MovieDto {
    title: string,
    description: string,
    year: number,
    duration: number,
    genres: string[],
    directors: string[],
    rating: number,
    votes: number,

```

|           |      |                  |        |      |                                                                    |  |  |                                           |       |         |
|-----------|------|------------------|--------|------|--------------------------------------------------------------------|--|--|-------------------------------------------|-------|---------|
|           |      |                  |        |      | ІАЛЦ.467200.007 Д4                                                 |  |  |                                           |       |         |
|           |      |                  |        |      |                                                                    |  |  |                                           |       |         |
| Зм.       | Арк. | № докум.         | Підпис | Дата |                                                                    |  |  |                                           |       |         |
| Розробив  |      | Гакавий О. О.    |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  |  | Літ.                                      | Аркуш | Аркушів |
| Перевірив |      | Череватенко О.В. |        |      |                                                                    |  |  |                                           |       | 39      |
| Реценз.   |      | Коган А.В.       |        |      |                                                                    |  |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. |      | Волокита А. М.   |        |      |                                                                    |  |  |                                           |       |         |
| Затвердив |      |                  |        |      |                                                                    |  |  |                                           |       |         |

```

    imageUrl: string,
    trailerUrl: string,
  }

// dto/showtime.dto.ts

export default interface ShowtimeDto {
  movieId: string,
  date: Date,
  time: string,
  availableSeats: boolean[][],
}

// dto/user.dto.ts

export default interface UserDto {
  name: string,
  email: string,
  imageUrl: string,
}

// dto/review.dto.ts

export default interface ReviewDto {
  userId: string,
  movieId: string,
  rating: number,
  comment: string | null
}

// dto/booking.dto.ts

export default interface BookingDto {
  userId: string,
  showtimeId: string,
  seats: string[],
  price: number,
  isPaid: boolean,
}

// constants/index.ts

export const MOVIES_PER_PAGE = 24;

export const LOGIN_PATH = '/api/auth/login';
export const LOGOUT_PATH = '/api/auth/logout';

export const TICKET_PRICE = 70
// prisma/schema.prisma

generator client {
  provider = "prisma-client-js"
}

datasource db {
  provider = "postgresql"

```

|           |                  |          |        |      |                                                                    |  |                                           |       |         |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|---------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |         |
|           |                  |          |        |      |                                                                    |  |                                           |       |         |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |         |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш | Аркушів |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           |       | 39      |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |         |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |         |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |         |

```

// Uses connection pooling
url = env("POSTGRES_PRISMA_URL")
// Uses direct connection, ⚠ make sure to keep this to
`POSTGRES_URL_NON_POOLING`
// or you'll have dangling databases from migrations
directUrl = env("POSTGRES_URL_NON_POOLING")
}

model Movie {
  id          String    @id @default(uuid())
  title       String
  description  String
  year        Int
  duration    Int
  genres      String[]
  directors   String[]
  rating      Float
  votes       Int
  imageUrl    String    @map("image_url")
  trailerUrl  String    @map("trailer_url")
  showtimes   Showtime[]
  reviews     Review[]
  createdAt   DateTime @default(now()) @map("created_at")
  updatedAt   DateTime @updatedAt @map("updated_at")
}

model Showtime {
  id          String    @id @default(uuid())
  movie       Movie     @relation(fields: [movieId], references: [id])
  movieId     String    @map("movie_id")
  date        DateTime
  time        String
  availableSeats Json    @map("available_seats")
  booking     Booking?
  createdAt   DateTime @default(now()) @map("created_at")
  updatedAt   DateTime @updatedAt @map("updated_at")
}

model User {
  id          String    @id @default(uuid())
  email       String    @unique
  name        String
  imageUrl    String    @map("image_url")
  reviews     Review[]
  bookings    Booking[]
  createdAt   DateTime @default(now()) @map("created_at")
  updatedAt   DateTime @updatedAt @map("updated_at")
}

model Review {
  id          String    @id @default(uuid())
  user        User      @relation(fields: [userId], references: [id])
  userId      String    @map("user_id")
  movie       Movie     @relation(fields: [movieId], references: [id])
  movieId     String    @map("movie_id")
  rating      Int

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |

```

comment      String?
createdAt    DateTime @default(now()) @map("created_at")
updatedAt    DateTime @updatedAt @map("updated_at")

@@index([userId], name: "idx_review_user_id")
@@index([movieId], name: "idx_review_movie_id")
}

model Booking {
  id          String    @id @default(uuid())
  user        User      @relation(fields: [userId], references: [id])
  userId      String    @map("user_id")
  showtime    Showtime  @relation(fields: [showtimeId], references: [id])
  showtimeId  String    @unique @map("showtime_id")
  seats       String[]
  price       Int
  isPayed     Boolean
  createdAt   DateTime @default(now()) @map("created_at")
  updatedAt   DateTime @updatedAt @map("updated_at")

  @@index([userId], name: "idx_booking_user_id")
}

```

|           |                  |          |        |      |                                                                    |  |                                           |       |
|-----------|------------------|----------|--------|------|--------------------------------------------------------------------|--|-------------------------------------------|-------|
|           |                  |          |        |      | ІАЛЦ.467200.007 Д4                                                 |  |                                           |       |
| Зм.       | Арк.             | № докум. | Підпис | Дата |                                                                    |  |                                           |       |
| Розробив  | Гакавий О. О.    |          |        |      | Сайт для продажу квитків до<br>кінотеатру<br>Текс програмного коду |  | Літ.                                      | Аркуш |
| Перевірив | Череватенко О.В. |          |        |      |                                                                    |  |                                           | 39    |
| Реценз.   | Коган А.В.       |          |        |      |                                                                    |  | КПІ ім. Ігоря Сікорського,<br>ФІОТ, ІП-04 |       |
| Н. Контр. | Волокита А. М.   |          |        |      |                                                                    |  |                                           |       |
| Затвердив |                  |          |        |      |                                                                    |  |                                           |       |