

Міністерство освіти і науки України
Національний університет «Запорізька Політехніка»
Кафедра програмних засобів

ЗВІТ

з самостійної роботи

з дисципліни «Веб технології та Веб дизайн»

Прийняли:
Старший викладач: С. Д. Леощенко

2023

Реферат	3
Вступ.....	4
Аналіз предметної області	5
Опис теми.....	5
Постановка завдання	5
Межі системи.....	5
Функціональні вимоги	6
Висновки за розділом.....	6
Матеріали і методи	6
Опис засобів розробки	6
Вибір мови програмування.....	7
Структурна схема розробки.....	8
Висновки за розділом.....	8
Основні рішення щодо реалізації компонентів системи	9
Проектування дизайну застосунку.....	9
Вайрфрейми програми:.....	9
Мокапи програми:.....	12
Прототип програми:.....	14
Опис прийнятих рішень	15
Дизайн навігаційної панелі	15
Дизайн сторінки каталогу	15
Кнопка "Повернутися до початку"	16
Розгортання серверу.....	16
Висновки за розділом.....	17
Експлуатація, тестування та експериментальне дослідження програми.....	17
Призначення й умови застосування програми.....	17
Методики та результати тестування	18
Чек-лист тестування	18
Тестування за сценаріями.....	19
Сценарій 1	19
Сценарій 2	20
Сценарій 3	20
Сценарій 4	20
Сценарій 5	20
Висновки	21
Успіхи.....	21
Виклики.....	21
Загальні висновки.....	22
Цінність проекту.....	22

Реферат

Ключові слова – Перегляд фільмів, каталог фільмів, фільми, серіали, самостійна робота, веб технології, веб дизайн.

Об'єкт дослідження – Метою цього дослідження є розробка веб-каталогу фільмів під назвою "Flickster", який спрощує процес перегляду та вибору фільмів і серіалів зі світової кіноіндустрії.

Предмет дослідження – Предметом дослідження є застосування веб-технологій та принципів дизайну для створення інтуїтивно зрозумілого та зручного для користувача каталогу фільмів.

Мета роботи – Мета дослідження: Метою цієї роботи є створення платформи для любителів кіно для легкого пошуку та вибору фільмів чи серіалів з великої бази даних. Проєкт спрямований на покращення користувацького досвіду та сприяння глибшому залученню людей до кіноіндустрії.

Матеріали, методи та технічні засоби – Матеріали, методи та техніки: У дослідженні було використано різноманітні технології та методи, зокрема JavaScript, React, Next.JS, та TailwindCS. Ці інструменти були використані для створення динамічного, адаптивного та естетично привабливого веб-сайту.

Результати – Очікуваним результатом цього дослідження стала розробка повнофункціонального веб-сайту-каталогу фільмів Flickster, який слугуватиме всеосяжною платформою для кінолюбителів.

Висновки – По завершенню виконання даної роботи було отримано практичні навички з проєктування, розробки та тестування програмного забезпечення, а також досвід роботи з популярними фреймворками та

бібліотеками. Також було вивчено, як розгорнати програмний продукт на віддаленому сервері, покращуючи доступність проекту.

Галузь використання – Розроблений веб-сайт Flickster слугуватиме цінним ресурсом для спільноти кіноманів, надаючи їм спрощену платформу для пошуку та вибору улюблених фільмів та серіалів.

Вступ

У цифрову епоху каталоги фільмів перетворилися з простих статичних баз даних на динамічні інтерактивні платформи. Цей проект, "Flickster", є втіленням цієї еволюції. Актуальність цієї теми зумовлена зростаючим попитом на цифрові платформи, що задовольняють потреби кінолюбителів.

Вибір саме цієї теми зумовлений потребою у спрощеній платформі, яка полегшує користувачам пошук і вибір фільмів чи серіалів. Зважаючи на величезну кількість контенту, доступного у світовій кіноіндустрії, така зручна платформа, як Flickster, може значно покращити користувацький досвід.

Крім того, розробка Flickster відповідає сучасній тенденції використання веб-технологій та принципів дизайну для створення інтуїтивно зрозумілих та цікавих користувацьких інтерфейсів. Використовуючи популярні фреймворки та бібліотеки, такі як JavaScript, React, Next.JS, та TailwindCSS, цей проект не тільки продемонструє застосування цих технологій, але й підкреслить їхній потенціал у створенні ефективних та естетично привабливих веб-застосунків.

Отже, цей проект - це не лише розробка каталогу фільмів, але й використання веб-технологій та принципів дизайну для створення зручної, ефективної та цікавої платформи для любителів кіно.

Аналіз предметної області

Опис теми

Темою цього проекту є розробка веб-каталогу фільмів "Flickster". Метою є створення інтуїтивно зрозумілої та зручної платформи, на якій кінолюбителі зможуть легко шукати та обирати фільми чи серіали з великої бази даних.

Постановка завдання

Завданням цього проекту є розробка веб-сайту, який надає спрощену платформу для перегляду та вибору фільмів і серіалів. Веб-сайт буде відображати каталог фільмів, кожен з яких матиме детальну інформацію, а також пропонуватиме різні варіанти сортування та фільтрації для покращення користувацького досвіду.

Межі системи

Система складатиметься щонайменше з п'яти головних сторінок. На головній сторінці буде відображатися каталог фільмів з різними варіантами сортування та фільтрації. Кожен фільм у каталозі буде посиланням на окрему сторінку з більш детальною інформацією про фільм. Система також включатиме функцію пошуку, яка дозволить користувачам знаходити конкретні фільми або серіали. Веб-сайт буде створено з використанням JavaScript, React, Next.JS, та TailwindCSS, і він не використовуватиме базу даних. На сторінці каталогу буде показано список фільмів з мініатюрами та короткими описами. На сторінці

інформації про фільм відображатиметься детальна інформація про конкретний фільм, включно з повним описом, списком акторів та оцінками користувачів. Сторінка пошуку дозволить користувачам шукати конкретні фільми чи серіали за назвою, жанром чи іншими критеріями.

Функціональні вимоги

Функціональні вимоги до системи включають відображення каталогу фільмів, надання детальної інформації про кожен фільм, різні варіанти сортування та фільтрації, а також можливість пошуку конкретних фільмів чи серіалів. Веб-сайт також повинен бути адаптивним та естетично привабливим.

Висновки за розділом

Цей розділ забезпечив чітке розуміння обсягу та вимог проєкту. Було визначено межі системи та окреслено функціональні вимоги. Проєкт передбачає розробку веб-каталогу фільмів, який відповідатиме потребам кінолюбителів і використовуватиме популярні веб-технології та принципи дизайну.

Матеріали і методи

Опис засобів розробки

Інструменти розробки, обрані для цього проєкту, включають JavaScript, React, Next.JS, та TailwindCSS. Ці інструменти широко використовуються у веб-розробці та пропонують широкий спектр можливостей та функцій. JavaScript - це основна мова, яка

використовується для веб-розробки. React - це бібліотека JavaScript для створення користувацьких інтерфейсів, Next.JS - фреймворк для серверного рендерингу React-застосунків, а TailwindCSS - CSS-фреймворк для швидкого створення кастомних дизайнів.

Вибір мови програмування

При виборі мови програмування для веб-розробки слід враховувати кілька факторів. Одним з ключових критеріїв є наявність доповнень та комплексів для спрощення розробки. JavaScript, з його розгалуженою екосистемою бібліотек та фреймворків, пропонує безліч інструментів та ресурсів для спрощення розробки. Він також має потужну підтримку серед серверів і добре масштабується, що робить його гарним вибором для веб-розробки.

Для реалізації веб-сайту Flickster ми порівняли JavaScript з Python та Ruby. JavaScript був обраний замість Python та Ruby через його потужну підтримку серед серверів, масштабованість та крос-платформенну сумісність. JavaScript також має велику екосистему бібліотек і фреймворків, включаючи React, що має вирішальне значення для побудови користувацького інтерфейсу веб-сайту Flickster.

Критерій	JavaScript	Python	Ruby
Популярність	Висока	Помірна	Помірна
Варіанти застосування	Frontend, Backend, Mobile	Data Science, машинне навчання, веб-розробка	Веб-розробка, скрипting
Синтаксис	C-подібний	На основі відступів	C-подібний
Типізація	Динамічна	Динамічна	Динамічна

Швидкодія	Гарна	Чудова	Гарна
Масштабованість	Чудова	Гарна	Гарна
Підтримка спільноти	Висока	Висока	Висока
Простота вивчення	Помірна	Висока	Висока

Структурна схема розробки

Розробка веб-сайту Flickster відбудуватиметься за шаблоном NVC (N-рівнева архітектура). Цей патерн передбачає поділ застосунку на три основні шари: рівень презентації, рівень бізнес-логіки та рівень доступу до даних. Рівень презентації відповідає за користувальський інтерфейс та взаємодію з користувачем. Рівень бізнес-логіки буде обробляти дані, введені користувачем, і взаємодіяти з даними системи. Рівень доступу до даних відповідає за всі взаємодії з системою зберігання даних.

Висновки за розділом

У цьому розділі було надано детальний огляд інструментів та методів, використаних при розробці веб-сайту Flickster. Обрана мова програмування, JavaScript, пропонує широкий спектр можливостей і функцій, що робить її підходящим вибором для веб-розробки. Паттерн NVC забезпечує чітку структуру для процесу розробки, гарантуючи, що застосунок добре організований і простий в обслуговуванні. Порівняння мов програмування підтвердило, що JavaScript є найкращим вибором для цього проєкту.

Основні рішення щодо реалізації компонентів системи

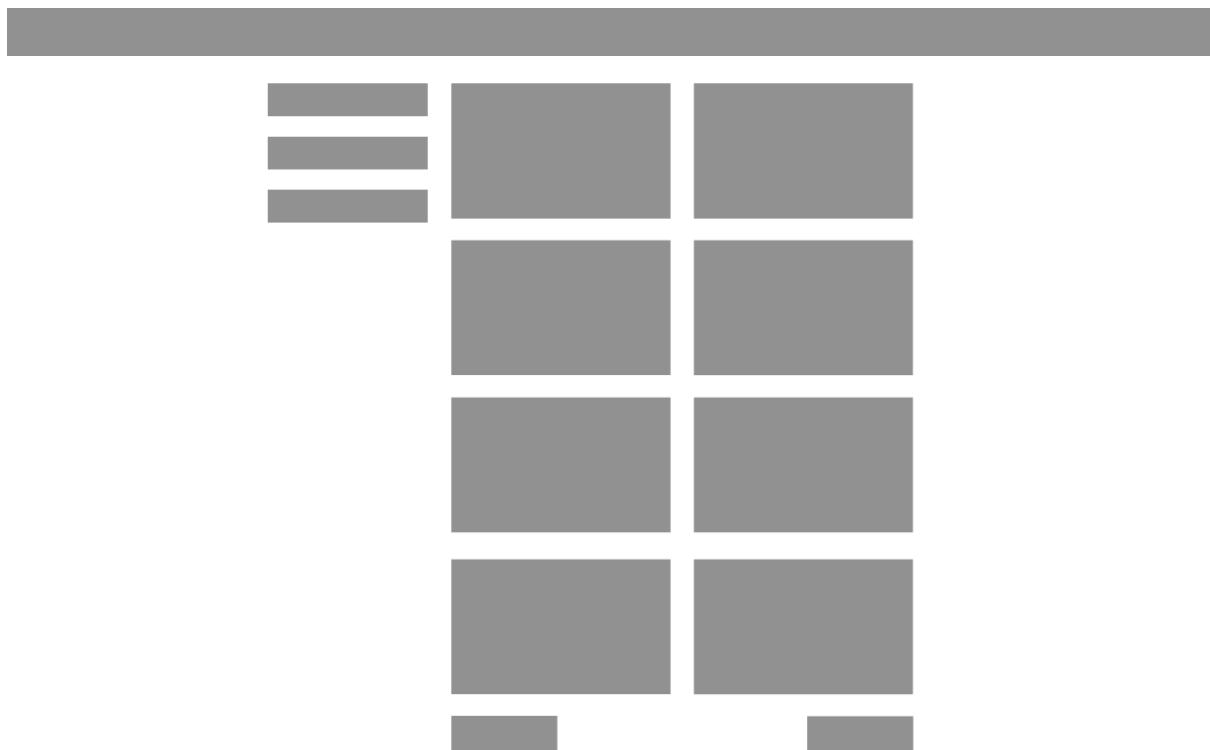
Проєктування дизайну застосунку

Дизайн веб-сайту Flickster буде базуватися на принципах простоти, зручності та естетики. Веб-сайт буде адаптивним, що забезпечить його гарний вигляд на всіх пристроях, від настільних комп'ютерів до мобільних телефонів.

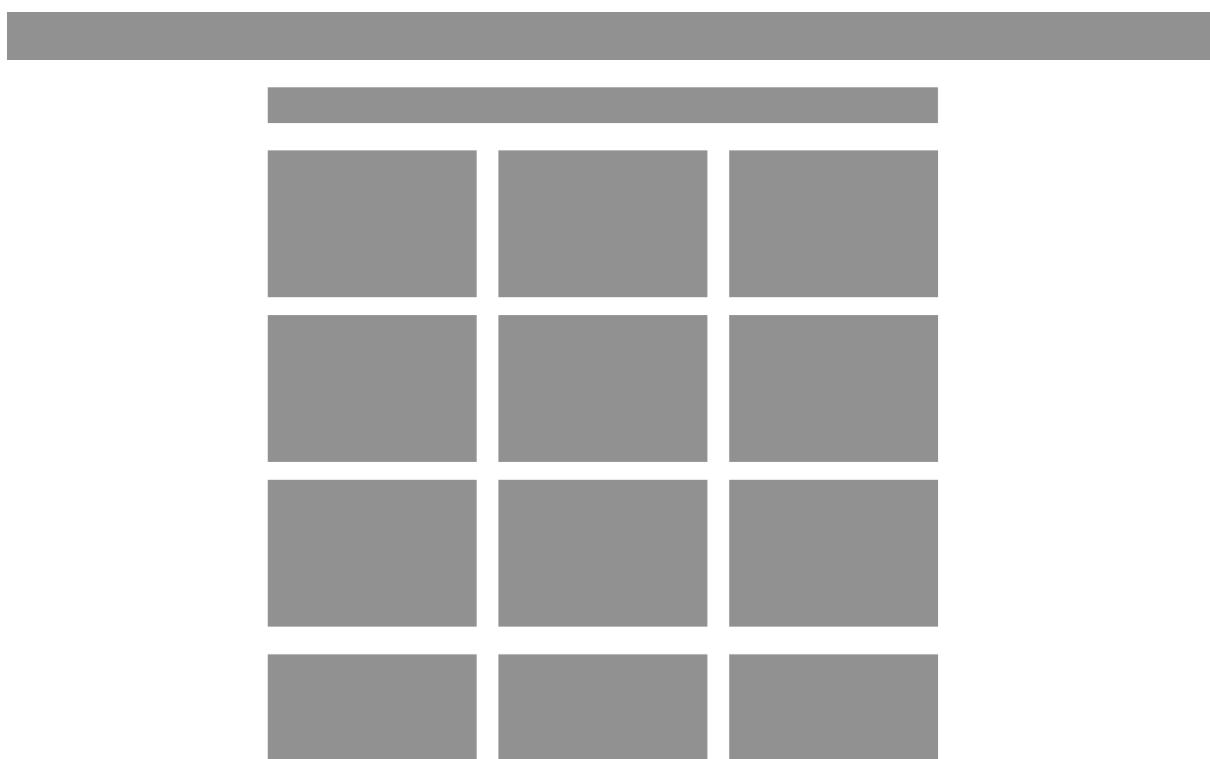
Вайрфрейми програми:



Вайрфрейм головної сторінки (Рис. 1.1)



Вайрфрейм сторінки каталогу (Рис 1.2)



Вайрфрейм сторінки жанрів (Рис. 1.3)



Вайрфрейм інформаційної сторінки (Рис. 1.4)

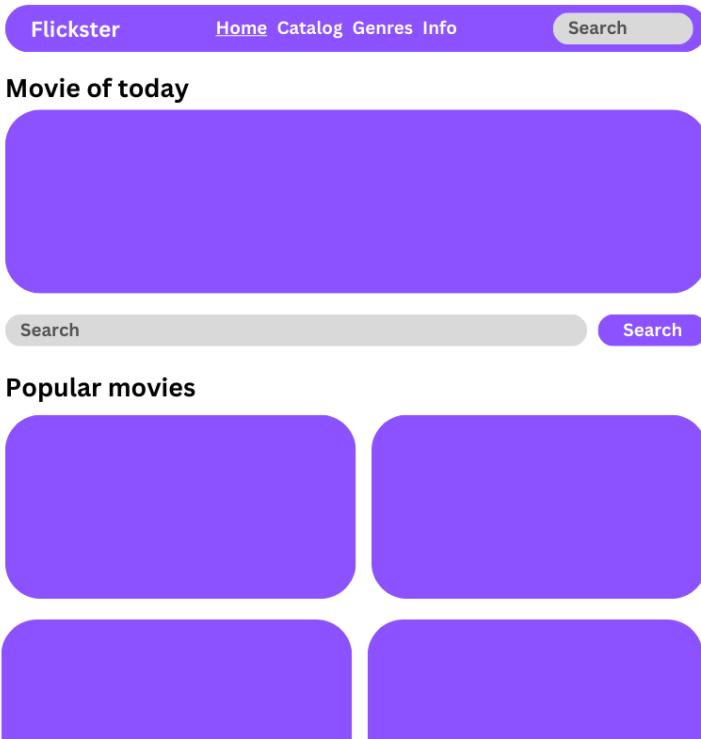


Вайрфрейм сторінки фільму (Рис. 1.5)

Мокапи програми:

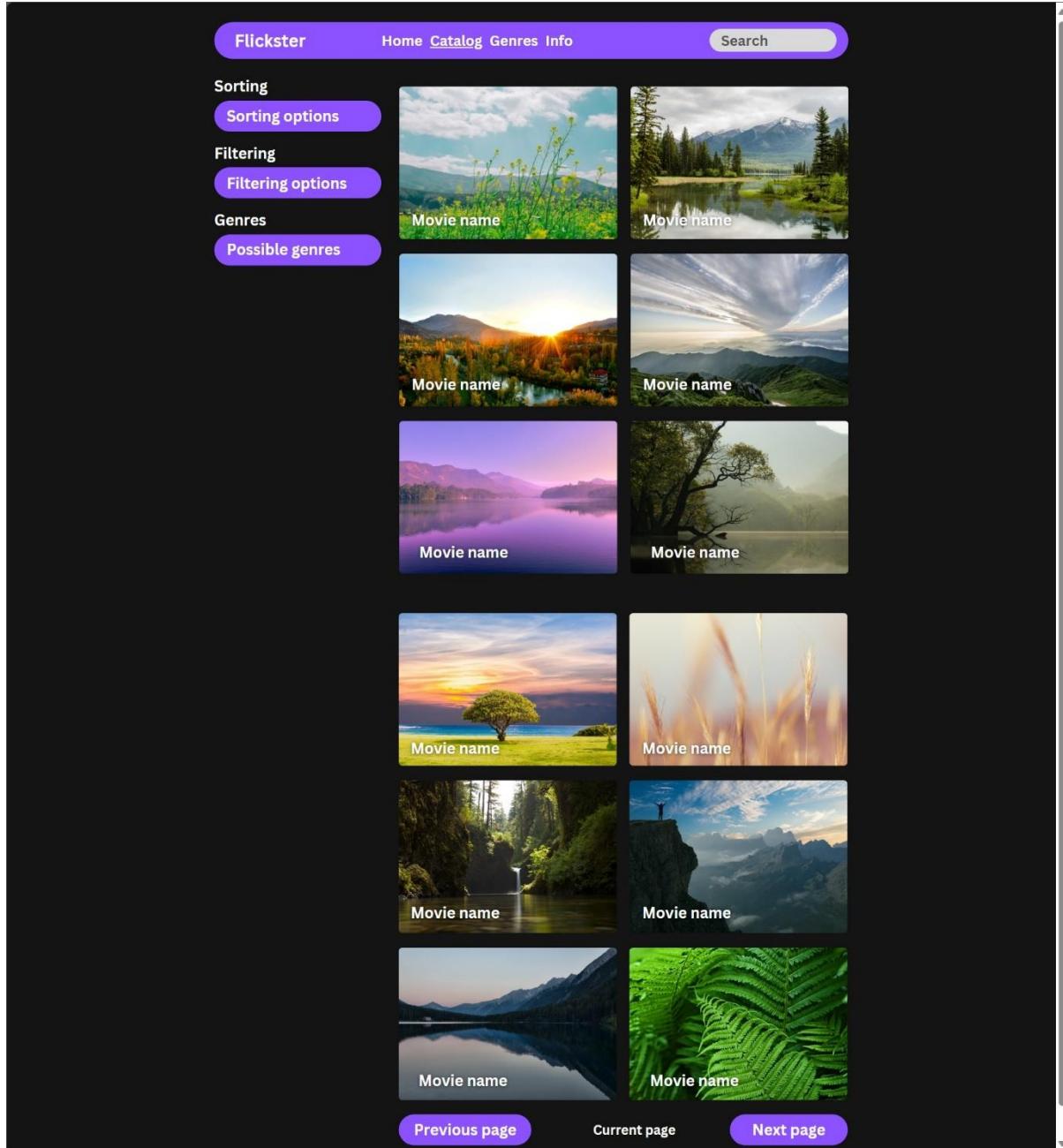


Мокап сторінки каталогу (Рис 2.1)



Мокап головної сторінки (Рис. 2.2)

Прототип програми:



Прототип сторінки каталогу (Рис. 3.1)

Опис прийнятих рішень

Рішення щодо дизайну та реалізації застосунку Flickster були в першу чергу спрямовані на покращення користувальського досвіду та зручності використання на різних розмірах екранів.

Дизайн навігаційної панелі

Панель навігації була розроблена таким чином, щоб адаптуватися до різних розмірів екрану. На менших екранах навігаційна панель розташована в нижній частині екрана, надаючи чотири кнопки розділів для навігації. Таке дизайнське рішення було прийнято для того, щоб панель навігації залишалася функціональною і легкодоступною на невеликих пристроях.

На більших екранах панель навігації розташована у верхній частині екрана і містить логотип проекту як посилання на домашню сторінку та пошуковий рядок для пошуку фільмів. Таке дизайнське рішення було прийнято для того, щоб забезпечити візуально привабливіший і зручніший інтерфейс на великих екранах, зберігаючи при цьому функціональність.

Панель навігації також є липкою, тобто зберігає фіксоване положення на екрані. Таке рішення було прийнято для покращення юзабіліті та зручності користування, гарантуючи, що панель навігації завжди доступна, незалежно від позиції прокрутки сторінки користувачем.

Дизайн сторінки каталогу

На сторінці каталогу поруч із сіткою фільмів розміщено секцію фільтрації та сортування. На великих екранах ця секція є липкою по всій

висоті розділу, тобто вона слідує за користувачем вниз під час прокрутки. На менших екранах вона залишається вгорі.

Таке дизайнерське рішення було прийнято для того, щоб покращити користувацький досвід і полегшити сортування та фільтрацію каталогу фільмів. Зберігаючи опції сортування та фільтрації видимими під час прокрутки, ми прагнемо забезпечити більш плавний та ефективний перегляд фільмів.

Кнопка "Повернутися до початку"

Внизу результатів пошуку на сторінці пошуку фільмів є кнопка "Повернутися до початку". Ця кнопка призначена для покращення користувацького досвіду, надаючи користувачам швидкий і простий спосіб повернутися на початок сторінки, особливо при роботі з великою кількістю результатів пошуку.

Кнопка має вигляд шеврона, спрямованого вгору, який є загальновизнаним символом руху вгору або навігації. Таке дизайнерське рішення було прийнято для того, щоб кнопка була легко впізнаваною та інтуїтивно зрозумілою у використанні.

На завершення, дизайнерські рішення, прийняті для застосунку Flickster, керувалися принципами зручності використання, користувацького досвіду та оперативності реагування. Враховуючи потреби та вподобання різних груп користувачів, ми змогли створити веб-застосунок, який є не лише функціональним, але й візуально привабливим та зручним для навігації.

Розгортання серверу

Сервер розробки можна розгорнути за допомогою інструментів Next.JS. Для цього потрібно налаштувати локальне середовище розробки, створити новий застосунок Next.JS і запустити його на локальному сервері, використавши для цього консольну команду npm run dev. Після того, як застосунок повністю розроблений і протестований, його можна розгорнути на віддаленому хостингу, наприклад, Vercel, використовуючи Vercel CLI або зручний користувачкий інтерфейс на відповідному сайті, де сам проект можна завантажити, наприклад, через GitHub репозитарій.

[Посилання на розгорнуту версію застосунку через Vercel](#)

[Посилання на GitHub репозитарій](#)

Висновки за розділом

У цьому розділі було надано детальний огляд процесу розробки та реалізації веб-сайту Flickster. Прийняті дизайнерські рішення, такі як вибір макету та функціональності, керувалися принципами простоти, зручності та естетики. Процес розгортання, як на локальному сервері, так і на віддаленому хостингу, був детально описаний.

Експлуатація, тестування та експериментальне дослідження програми

Призначення й умови застосування програми

Застосунок Flickster створений для того, щоб забезпечити комплексний перегляд фільмів. Застосунок призначений для всіх, хто зацікавлений у дослідженні різних фільмів, фільтруванні та сортуванні їх

на основі різних критеріїв, а також у тому, щоб дізнатися більше про свої улюблені фільми.

Застосунок розроблений таким чином, щоб бути зручним і доступним для користувача, з адаптивним дизайном, який підлаштовується під різні розміри екранів. Він призначений для використання на різних пристроях, включаючи настільні комп'ютери, ноутбуки, планшети та мобільні телефони.

Методики та результати тестування

Чек-лист тестування

Для програми Flickster чек-лист тестування включає такі пункти, як:

- Навігація: Перевірити всі посилання та кнопки, щоб переконатися, що вони працюють коректно і ведуть на потрібні сторінки.
- Адаптивність: Перевірити застосунок на різних розмірах екрану, щоб переконатися, що він реагує і відображається коректно.
- Функціональність: Перевірити всі функції застосунку, включаючи перегляд фільмів, фільтрацію, сортування та пошук.
- Продуктивність: Перевірити продуктивність застосунку, щоб переконатися, що він швидко завантажується і працює без збоїв.

Компонент/Функція	Функціональність	Швидкодія	Зручність
Навігація	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Адаптивність	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Функціональність	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Продуктивність	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Тестування за сценаріями

Для застосунку Flickster тестування за сценаріями включає наступні етапи:

- Сценарій 1: Користувач переходить на головну сторінку і переглядає популярні та актуальні фільми.
- Сценарій 2: Користувач переходить на сторінку фільмів і фільтрує фільми за жанрами.
- Сценарій 3: Користувач переходить на сторінку жанрів і вибирає жанр для перегляду фільмів.
- Сценарій 4: Користувач переходить на сторінку "Про нас" і читає надану інформацію.
- Сценарій 5: Користувач використовує рядок пошуку, щоб знайти певний фільм.

У кожному сценарії ми тестуємо функціональність застосунку, переконуючись, що він працює так, як очікувалося, а будь-які проблеми чи помилки виявляються та вирішуються.

Сценарій 1

Результати тестування - Успіх

Нотатки - головна сторінка завантажується успішно, перегляд фільмів та переход за посиланнями працює коректно.

Сценарій 2

Результати тестування - Успіх

Нотатки - сторінка каталогу відкривається успішно та відображається коректно, фільтрація фільмів за жанрами працює коректно, сортування та фільтрація працюють коректно.

Сценарій 3

Результати тестування - Успіх

Нотатки - сторінка жанрів відкривається успішно, вибір жанру для перегляду фільмів працює коректно.

Сценарій 4

Результати тестування - Успіх

Нотатки - інформаційна сторінка відкривається успішно та відображається коректно.

Сценарій 5

Результати тестування - Успіх

Нотатки - сторінка пошуку відкривається успішно та відображається коректно.

Висновки

Розробка програми Flickster була корисним і повчальним досвідом. Проект дозволив нам застосувати та поглибити наше розуміння веб-технологій та принципів дизайну, а також дав нам практичний досвід у розробці, тестуванні та розгортанні програмних продуктів.

Успіхи

Застосунок Flickster було успішно розроблено та розгорнуто. Він забезпечує комплексний досвід перегляду фільмів з адаптивним дизайном, який пристосовується до різних розмірів екрану. Застосунок включає такі функції, як перегляд фільмів, фільтрація, сортування та пошук, які були ретельно протестовані і працюють належним чином.

При розробці застосунку ми керувалися принципами простоти, зручності та естетики. Прийняті дизайнерські рішення, такі як макет і функціональність застосунку, призвели до створення зручного та візуально привабливого веб-застосунку.

Застосунок був успішно розгорнутий на Vercel, популярній платформі для розміщення веб-застосунків. Це зробило застосунок доступним для широкої аудиторії та покращило його доступність.

Виклики

Під час розробки застосунку Flickster ми зіткнулися з кількома проблемами. Однією з головних проблем була робота з фільмами, які не мали зображень. Спочатку ми розглядали можливість використання компонента Image у Next.js для відображення зображення-заповнювача

для таких фільмів. Однак ми вирішили відфільтрувати отримані дані і виключити фільми, які не мають зображень. Такий підхід спростив процес обробки даних і покращив продуктивність застосунку.

Ще однією проблемою, з якою ми зіткнулися, був доступ до даних фільму, оскільки вони все ще завантажувалися. Ми вирішили цю проблему, перевіривши змінну `isLoading`. Цей підхід гарантував, що доступ до даних буде отримано лише тоді, коли стан буде повністю завантажено, запобігаючи будь-яким потенційним помилкам або проблемам.

Загальні висновки

Розробка застосунку Flickster була цінним практичним досвідом. Ми отримали практичні навички з проєктування, розробки та тестування програмного забезпечення, а також досвід роботи з популярними фреймворками та бібліотеками. Ми також навчилися розгорнати програмний продукт на віддаленому сервері, покращуючи доступність проєкту.

Процес розробки також посилив важливість користувачького досвіду та юзабіліті у розробці програмного забезпечення. Ми зрозуміли важливість розробки застосунків, які є простими у використанні та навігації, а також забезпечують безперебійний та ефективний перегляд веб-сторінок.

Цінність проєкту

Застосунок Flickster слугує цінним ресурсом для спільноти кіноманів, надаючи їм спрощену платформу для пошуку та вибору улюблених фільмів і телешоу. Він демонструє використання популярних

веб-технологій та принципів дизайну для створення ефективного та естетично привабливого веб-застосунку.

На завершення, розробка застосунку Flickster була успішним і збагачуючим досвідом. Ми не лише створили функціональний і зручний веб-застосунок, але й отримали цінний досвід і знання у сфері веб-розробки та дизайну.