



Приватний заклад вищої освіти  
**Одеський технологічний університет «ШАГ»**  
Кафедра інформаційних технологій та фундаментальної підготовки

Випускна кваліфікаційна робота бакалавра

**Інформаційна система управління ліцензованим відео контентом (на  
прикладі Netflix)**

на здобуття бакалаврського рівня вищої освіти  
зі спеціальності «122 Комп'ютерні науки»

Виконавці проєкту

| № | П.І.Б.                   | Група    |
|---|--------------------------|----------|
| 1 | Коваленко Я. В.          | КН-Д-211 |
| 2 | Головань І. В.           | КН-П-212 |
| 3 | Кійко Б. І.              | КН-П-211 |
| 4 | Цапкалов-Старченко Г. Г. | КН-Д-211 |
| 5 | Подрушняк П. В.          | КН-П-    |

Автор звіту

\_\_\_\_\_ Головань Ігор Вікторович

|                   |                                                           |
|-------------------|-----------------------------------------------------------|
| Науковий керівник | Доцент кафедри ІТ та ФП, к.т.н., доцент<br>Данилець Є. В. |
|-------------------|-----------------------------------------------------------|

Одеса – 2025

## АНОТАЦІЯ

Головною метою даного проекту було створення зручного та інтуїтивно зрозумілого веб-застосунку та мобільного додатка, який надасть можливість безкоштовного перегляду найвідоміших фільмів та серіалів, користуючись зручним застосунком, або сайтом. У процесі розробки ми вивчали схожі продукти, такі як Netflix, IMDB, Disney Plus, з метою зрозуміти потреби користувача щодо функціонала, найпопулярніших звичок та інтуїтивного розуміння.

Наша платформа має дуже зручний та зрозумілий інтерфейс, починаючи з реєстрації або авторизації, можливості одразу обирати фільми/серіали з рекомендованих платформою, або користуватись зручним пошуком, де користувач також отримує багато додаткової інформації щодо фільму, а також зручне редагування профілю та надходження повідомлень щодо улюблених жанрів.

При розробці ми брали акцент на користувачах та їхньому досвіду від знаходження на нашій платформі, особливу увагу проводили на наступні пункти:

- Створення зрозумілого інтерфейсу під кожного користувача.
- Додавання найвідоміших жанрів на головний екран.
- Можливість почати перегляд навіть без реєстрації.
- Можливість дізнатися більше інформації щодо фільму/серіалу перед початком перегляду.
- Створення мобільного застосунку, замість змушення дивитись фільми з мобільного браузера.

## ЗМІСТ

|                                           |    |
|-------------------------------------------|----|
| Анотація                                  | 1  |
| Вступ                                     | 3  |
| Технічне завдання до проекту              | 4  |
| Вимоги до функціоналу                     | 5  |
| Преамбула                                 | 6  |
| РОЗДІЛ 1. Загальна характеристика системи | 7  |
| РОЗДІЛ 2. Реалізація клієнтської частини  | 9  |
| РОЗДІЛ 3. Результати випробувань          | 12 |
| ВИСНОВКИ                                  | 15 |
| ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ               | 16 |

## ТЕХНІЧНЕ ЗАВДАННЯ ДО ПРОЄКТУ

### **Загальний опис.**

Веб Застосунок та мобільний додаток призначений для стрімінгу фільмів та серіалів, який дозволяє користувачам переглядати контент. Головна функція та ідея веб застосунку полягає в доступності широкого вибору контенту та легкого інтуїтивного використання для користувача який споживає цю послугу.

### **Призначення:**

Веб Застосунок та додаток орієнтується на користувачів які хочуть швидко знаходити фільми та серіали без багатьох зайвих кроків. ПЗ створюється за прикладом замовлення транспортних послуг, у той же час має складатись за принципом універсальності, мати змогу змінити змінити цільове призначення.

### *Головна сторінка*

З метою забезпечення привабливого першого враження з Веб Застосунком та додатком потрібно реалізувати головну сторінку з банером відображення рекомендованого фільму. Реалізації кнопок сторінок які доступні у верхній частині сайту а саме Головна, Фільми, Серіали, Мій список, У тренді та кнопок Логіну та Реєстрації. А також представлення доступного сучасного контенту з категорії Фільми і Серіали.

### *Категорії*

Для представлення структурованого контенту на сторінках Фільми й Серіали потрібно реалізувати систему розділів фільмів, кожен з якого повинні бути відображені у вигляді горизонтального списку у вигляді окремих карток що містять у собі рейтинг, назву та жанр.

### *Особистий кабінет*

У персональному розділі Веб Застосунку повинно бути відображення основної інформації про користувача а саме ім'я, емейл, налаштунок підписки, служба підтримки.

### **Технології**

У рамках реалізації Веб Застосунку та додатку було застосовано сучасні й зручні технології які дозволяють створити потрібний функціонал та гарний інтерфейс користувача.

- HTML - використаний для створення сторінок та розмітки.
- Figma - використаний для створення UI/UX частини.
- CSS - використаний для стилізації застосунку та елементів у ньому.
- JavaScript - використаний для реалізації логіки інтерфейсу та зміни контенту без перезавантаження сторінок.
- Git - використаний для контролю змінень.

За допомогою цього підходу сталося можливим реалізувати Веб Застосунок який не залежить від фреймворків або складних збірок що дозволяє Веб Застосунку існувати без постійного перепису коду через оновлення фреймворку та зрозумілістю для будь-якого розробника.

Завдяки цьому Веб Застосунок може працювати як у сучасних браузерях та пристроях так й у застарілих пристроях та браузерях.

Вихідний код повинен оформлятися у відповідності до загальноприйнятих стандартів, на зразок стандартів GNU <http://www.gnu.org/prep/standards/>, у тому числі з додержанням вимог до коментування та документування коду.

## ВИМОГИ ДО ФУНКЦІОНАЛУ

- Реєстрація/авторизація
  - Можливість створення облікового запису
  - Pop-up вікна входу та реєстрації
  - Перевірка правильності введених даних
  - Можливість виходу із аккаунту в будь який момент
- Головна сторінка
  - Навігаційне меню з пунктами як - Головна, Фільми, Серіали, Обране, Про нас, Увійти, Зареєструватися
  - Тематичні розділи
- Каталог фільмів із пошуком
  - Постер фільмів із відображенням назви, рейтингу та жанру
  - Динамічний пошук
  - Плеєр
- Слайдер логотипів компаній
  - Ручне керування стрілками
- Персональний кабінет
  - Відображення інформації про аккаунт(ім'я, емейл, підписка)
- Відповідність москир
  - Колористика та відповідання інтерфейсу із UI/UX
- Підтримка широкого спектру браузерів
- Сумісність із різними операційними системами
- Форма зворотнього зв'язку
  - Через форму
- Сторінка Про Нас
- Сторінка Політика конфіденційності
- Сторінка Умови використання
- Інтерфейс та навігація
  - Візуально зрозумілий дизайн
  - Ефекти взаємодії
  - Динамічний курсор
  - Навігація між сторінками
  - Відображення поточної активної сторінки

## ПРЕАМБУЛА

Основним референсом для розробки платформи став стримінговий сервіс Netflix - один із найвідоміших сайтів для перегляду фільмів і серіалів. Його було обрано завдяки зрозумілій структурі, широкій функціональності та вдалим UX-рішенням. Однак під час аналізу були виявлені типові недоліки: складна навігація, перевантаженість контентом, відсутність чіткої персоналізації та необхідність обов'язкової підписки. Це дало поштовх до створення нашого проєкту - стримінгового сервісу, який не потребує підписки й забезпечує швидкий, зручний доступ до відеоконтенту без зайвих бар'єрів.

Платформа орієнтована на користувачів, які цінують простоту, адаптивність та сучасний дизайн. Вона пропонує інтерфейс, у якому акцент зроблено на головне - сам контент, а не навколишні елементи. Для глибшого розуміння потреб аудиторії були проведені опитування, сформовані карти емпатії, CJM та побудовані сценарії взаємодії. Ми також дослідили поведінку користувачів на інших платформах, щоб адаптувати структуру під звичні шаблони, при цьому усуваючи їхні недоліки.

Було повністю розроблено унікальний візуальний стиль: від назви та логотипу до кольорової палітри та типографіки, яка підкреслює сучасність та технологічність бренду. Створені макети інтерфейсу охоплюють ключові екрани платформи - головну сторінку, сторінку фільму, профіль користувача, фільтрацію та пошук - і повністю адаптовані під смартфони, планшети та десктопи.

Завдяки комплексному підходу - від досліджень до створення повноцінної дизайн-системи - проєкт демонструє, як можна переосмислити користувацький досвід і зробити цифровий продукт по-справжньому орієнтованим на людину.

## РОЗДІЛ 1

### ЗАГАЛЬНА ХАРАКТЕРИСТИКА СИСТЕМИ

У відповідності до технічного завдання (ТЗ) система була спроектована за клієнт-серверною розподіленою архітектурою. У складі системи було умовно відокремлено наступні архітектурні частини:

- «Mockup»,
- «Backend»,
- «Frontend».

Частина «Mockup» являє собою проекцію інтерфейсу користувача. Ця частина включає у себе створення макетів у Figma, які демонструють вигляд, компонування, логіку та структуру. Макети виконані з урахуванням UI/UX дизайну та колористики. Частина розміщена за адресою: <https://www.figma.com/design/ik4H8ADhr5mtEhUZ7i4iQ4/NETFLIX?node-id=0-1&p=f&t=UxndbvdeTrnIR48O-0>

Частина «Frontend» являє собою реалізацію клієнтської частини Веб Застосунку з якою взаємодіє користувач. Вона забезпечує візуальне представлення інтерфейсу з динамічною поведінкою усіх елементів на сторінці та логіки роботи у браузері. Інтерфейс цієї частини був реалізований згідно макетів створених в Figma з урахуванням колористики та швидкого реагування на дії користувача разом із структурованою навігацією між розділами веб застосунку. Частина розміщена за адресою: <https://github.com/Flor1d4/NEXTVIEW>

Частина «Backend»

## РОЗДІЛ 2

### РЕАЛІЗАЦІЯ КЛІЄНТСЬКОЇ ЧАСТИНИ

Клієнтська частина Веб Застосунка реалізована з використанням наступних технологій:

Засоби розроблення:

- Мова програмування HTML для розробки сторінок
- CSS для можливості оформлення за прикладом мокапу
- JavaScript для реалізації логіки сайту та керуванням подій на ньому

У межах даного проекту використання Фреймворків є технічно невиправданим бо одними з основних потреб було висока продуктивність та довгострокова підтримка.

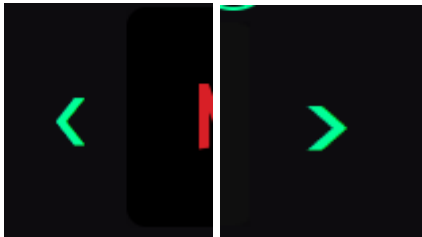
Обґрунтування вибору наведено таблицю з порівняннями:

| Критерії                                            | HTML+JS+CSS      | React/Angular                                     |
|-----------------------------------------------------|------------------|---------------------------------------------------|
| Простота налаштування                               | Висока           | Потребує CLI та інструменти збірки                |
| Прозорість                                          | Логіка зрозуміла | Частина логіки прихована                          |
| Гнучкість                                           | Повний контроль  | На основі шаблонів                                |
| Час запуску                                         | Миттєвий         | Потребує збірки та розгортання                    |
| Підтримка старих браузерів                          | Так              | Потребує Поліфіли                                 |
| Залежності                                          | Відсутні         | React, Framework, Webpack                         |
| Швидкість завантаження                              | Висока           | Критично для слабких пристроїв                    |
| Сумісність і стабільність                           | Без ризиків      | Можливість виходу плагінів з ладу після оновлення |
| Можливість підключення до фреймворків у майбутньому | Можливо та легко | Потребує час та підтримки                         |

У складі клієнтської частини реалізовано наступні функції:

- Динамічний пошук фільмів
- Робота з улюбленими фільмами
- Зміна контенту сторінки із залежністю від ролі користувача а саме авторизований або гість
- Модальні вікна
- Слайдер гортання площадок
- Картки фільмів
- Відображення поточної сторінки
- Перегляд фільмів
- Динамічний курсор
- Шапка навігації
- Сторінки Веб Застосунку

На початку функцій які зустрічають користувача Веб Застосунка це Карусель яка дозволяє промотувати логотипи компаній. Її функціональність забезпечена за допомогою функцій `slideLeft()` та `slideRight()`. Також присутнє відстеження поточного положення логотипів на слайдері.



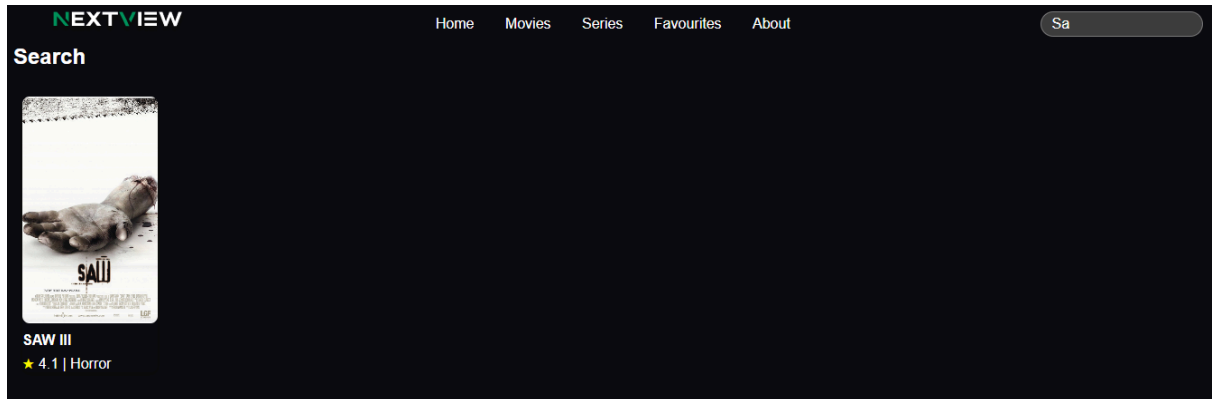
```
<script>
let currentSlide = 0;
const track = document.getElementById("carouselTrack");
const itemWidth = 220; // ширина блоку + gap

function slideLeft() {
  if (currentSlide > 0) {
    currentSlide--;
    track.style.transform = `translateX(-${currentSlide * itemWidth}px)`;
  }
}

function slideRight() {
  const items = document.querySelectorAll(".carousel-item");
  if (currentSlide < items.length - 3) {
    currentSlide++;
    track.style.transform = `translateX(-${currentSlide * itemWidth}px)`;
  }
}
</script>
```

При пошуку фільму коли користувач починає писати текст вмикається подія `input` яка миттєво починає фільтрувати фільми у залежності того який

текст користувач ввів. Завдяки чому проходить швидкий зворотній зв'язок та отримання результату



```
const input = document.getElementById("searchInput");
input.addEventListener("input", () => {
  const query = input.value.toLowerCase().trim();
  const filtered = movies.filter(movie =>
    movie.title.toLowerCase().includes(query)
  );
  renderResults(filtered);
});
```

Картка відображення фільма результату пошука реалізована у наступній функції як `renderResults()` витягуючи елемент масиву для створення картки з інформацією про фільм. Та якщо текст у пошуковій строці змінюється чи видаляється, функція автоматично змінює відображення пошуку для користувача

```
function renderResults(filtered) {
  const container = document.getElementById("results-container");
  container.innerHTML = "";

  filtered.forEach(movie => {
    const card = document.createElement("div");
    card.className = "card";
    card.innerHTML = `
      
      <p>${movie.title}</p>
      <div class="movie-rating">
        <span class="star">★</span>
        <span class="score">${movie.rating}</span>
        <span class="meta">| ${movie.genre}</span>
      </div>`;
    container.appendChild(card);
  });
}
```

З повним кодом проєкту можна ознайомитись у репозиторії <https://github.com/Flor1d4/NEXTVIEW>

### РОЗДІЛ 3 РЕЗУЛЬТАТИ ВИПРОБУВАНЬ

Результатом розробки клієнтської частини є Веб Застосунок який реалізує сучасний інтерфейс із динамічними компонентами. Основна мета це забезпечення зручної навігації та взаємодії із застосунком

При запуску застосунку користувачу відображається шапка з пунктами меню Home, Movies, Series, Favourites, About. А також у лівій частині Логотип та у правій частині кнопки Sign Up та Login.

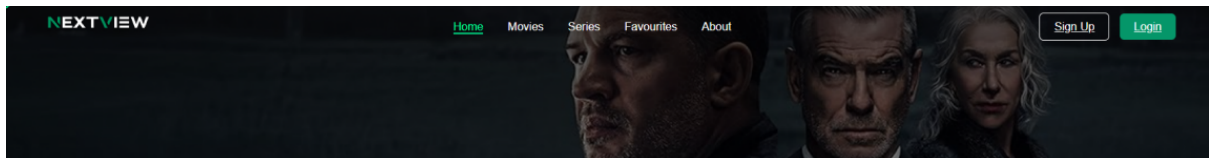


Рисунок 1 - Головне меню з навігацією

Однією з найважливіших функцій які випробувались - це реалізація вікон входу та реєстрації. Замість окремих сторінок було обрано рішення реалізувати поп-ап вікна, які з'являються поверх поточної сторінки. Реєстрація та вхід виконуються за допомогою стандартних полів (емейл, пароль), а при неправильному введенні даних відображається відповідне повідомлення. Це знижує навантаження на навігацію та зберігає контекст користувача.

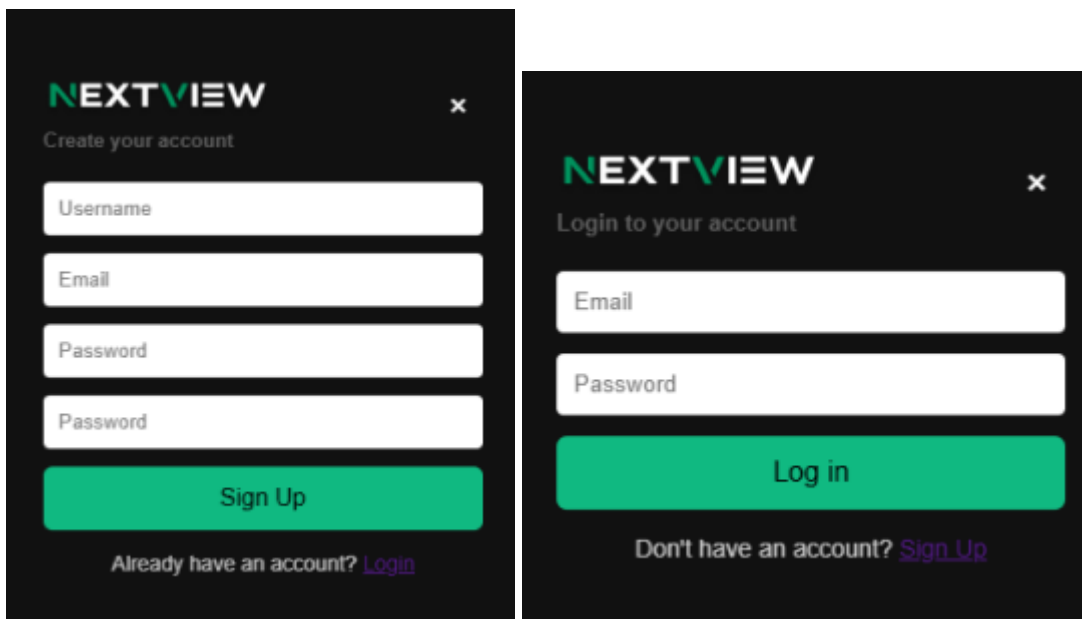


Рисунок 2 - Поп-ап вікна входу та реєстрації

У якості ще одного елементу який підлягав перевірці це горизонтальний слайдер логотипів стрімінгових платформ. Вона має анімацію та дозволяє перегортати логотипи вліво або вправо. Таке рішення було реалізовано без сторонніх бібліотек за допомогою на JavaScript, завдяки чому дозволило досягти кращої продуктивності та контролю.

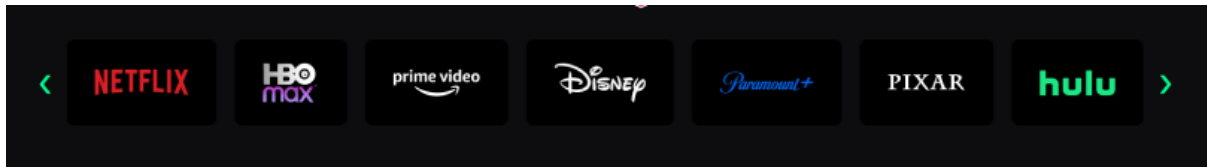


Рисунок 3 - Карусель логотипів платформ

Після успішного входу, користувач переходить до списку фільмів реалізованого у вигляді карток з постерами, назвами, рейтингами та жанрами. При натисканні на картку відкривається детальна інформація про фільм.

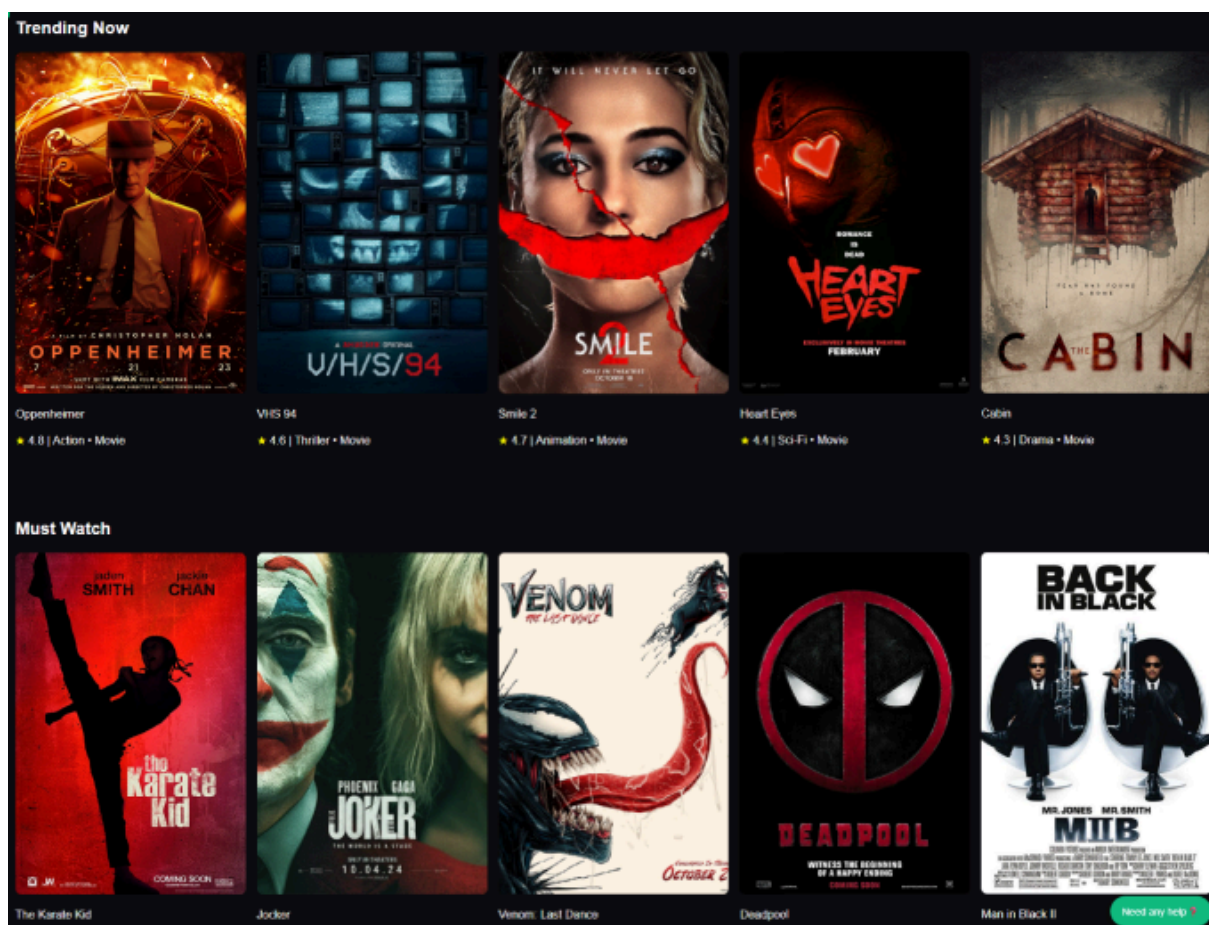


Рисунок 4 - Показ карток фільмів на сторінці

В результаті випробувань було підтверджено працездатність усіх ключових елементів інтерфейсу, а саме - від навігації до авторизації та взаємодії з контентом. Інтерфейс адекватно адаптується під різні розміри екранів (від мобільних до десктопних), а динамічні елементи працюють стабільно на більшості популярних браузерів.

У підсумку, всі функції клієнтської частини, що були регламентовані технічним завданням було реалізовано та випробувано. Поточна версія додатку, що реалізує клієнтську частину, доступна у репозиторії <https://github.com/Flor1d4/NEXTVIEW>

## ВИСНОВКИ

У результаті розробки було створено Веб Застосунок орієнтований на користувача для полегшення взаємодії та перегляду контенту. Основною метою було розробити зручний та зрозумілий для клієнта інтерфейс що працює без навантаження фреймворками. Для реалізації інтерфейсу було обрано стандартні веб технології JavaScript, HTML і CSS, оскільки вони допомагають нам уникнути непотрібних залежностей. Це також вирішує проблеми сумісності між браузерами та платформами спрощуючи його підтримку.

Веб Застосунок спроектовано, реалізовано та випробувано з урахуванням користувачів які взаємодіють з Веб Застосунком. Система складена за розподіленою трьох частин які розміщені нижче за адресами:

- «Mockup»  
<https://www.figma.com/design/ik4H8ADhr5mtEhUZ7i4iQ4/NETFLIX?node-id=0-1&p=f&t=UxndbvdeTrnIR48O-0> ,
- «Backend»,
- «Frontend» <https://github.com/Flor1d4/NEXTVIEWW>

Увесь функціонал працює завдяки JavaScript. Код якого був поділений на окремі файли для зручності змін його підтримки та використання (popup.js), (login.js), (signup.js), (search.js).

У результаті тестів було підтверджено працездатність застосунку та відповіданню технічного завдання із створенням зручного та швидкого клієнт-інтерфейсу.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Стратегія розвитку інформаційного суспільства в Україні. Розпорядження Кабінету Міністрів України від 15 травня 2013 р. № 386-р. // Урядовий портал. URL: <https://www.kmu.gov.ua/npas/246420577>
2. Стратегія кібербезпеки України. Указ Президента України від 14 травня 2021 року // Урядовий кур'єр від 28.08.2021 — № 165. URL: <https://zakon.rada.gov.ua/laws/show/447/2021/print>
3. Mozilla Developer Network (MDN). Офіційна документація з HTML, CSS, JavaScript. URL: <https://developer.mozilla.org/>
4. W3Schools. Вебдовідник з фронтенд-технологій. URL: <https://www.w3schools.com/>
5. Шевчук О.В. Основи веб-дизайну та технологій створення сайтів: навчальний посібник. — Київ: Видавничий центр КНУ, 2020. — 215 с.
6. Булгакова О.С. Інформатика: візуальне програмування. Навчальний посібник. — Херсон: Олді-Плюс, 2020. — 312 с. ISBN: 978-966-289-039-6
7. Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник. — Київ: КНЕУ, 2017. — 110 с.
8. Величко О.М., Гордієнко Т.Б. Основи системного аналізу і прийняття оптимальних рішень. — Херсон, 2021. — 672 с. ISBN: 978-966-289-553-7
9. Олександр Смичніков. Параноя або сучасні реалії: як зберегти конфіденційність у мережі. — ФЬЮЧЕР МЕДІА. URL: <https://mind.ua/openmind/20183329-paranoya-abo-suchasni-realiyi-yak-zberegti-konfidencijnist-u-merezhi>
10. Bootstrap 5. Офіційна документація. URL: <https://getbootstrap.com/docs/>
11. Чайковська М.П. Концептуально-методологічні засади управління маркетинговими ІТ-проектами. — Херсон, 2021. — 370 с. ISBN: 978-966-289-537-7
12. Google Developers: Web Fundamentals. Документація з безпечної та адаптивної розробки. URL: <https://developers.google.com/web/>
13. Чемакіна О.В. Дизайн системи візуальної інформації. Навчальний посібник / Чемакіна О.В., Рубцов А.Л., Свірко В.О. - Херсон. - 2019. - 200 с. ISBN: 978-966-289-216-1
14. ECMAScript 2021 Language Specification. URL: <https://tc39.es/ecma262/>