

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА
ФРАНКА**

Факультет прикладної математики та інформатики
(повне найменування назва факультету)

Кафедра інформаційних систем
(повна назва кафедри)

ДИПЛОМНА РОБОТА

**ВЕБЗАСТОСУНОК ІЗ ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ RUBY ТА
RUBY ON RAILS**

Виконала: студентка групи ПМІ-42
спеціальності 122 – комп'ютерні науки
(шифр і назва спеціальності)

Омелюх І. О.

(підпис)

(прізвище та ініціали)

Керівник _____ **Стельмащук В. В.**
(підпис) (прізвище та ініціали)

Рецензент _____
(підпис) (прізвище та ініціали)

2023

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ, ОГЛЯД ВІДОМИХ РІШЕНЬ ТА ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ.....	6
1.1 Постановка задачі.....	6
1.2 Огляд відомих рішень.....	7
IMDB.....	7
UAkino.....	8
1.3 Методики реалізації та опис використаних технологій розробки програм.....	8
1.3.1 Ruby.....	9
1.3.2 Ruby on Rails.....	9
1.3.3 Stimulus.JS.....	11
1.3.4 PostgreSQL.....	11
1.3.5 Elasticsearch.....	12
1.3.6 Tailwind CSS.....	12
1.3.7 Архітектурний шаблон MVC.....	13
1.3.8 Дизайн шаблони.....	13
1.3.9 Ruby gems.....	14
1.4.1 Devise.....	14
1.4.2 Rails-i18n.....	14
1.4.2 Wicked_pdf.....	14
1.3.10 Visual Studio Code.....	15
1.3.11 GitHub.....	15
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ВІДПОВІДНО ДО ВИМОГ.....	16

2.1 Загальний опис та характеристика продукту.....	16
2.3 База даних проекту.....	17
2.4 Підключення TMDb API.....	18
РОЗДІЛ 3. СЦЕНАРІЙ РОБОТИ ВЕБ-АПЛІКАЦІЇ.....	20
ВИСНОВКИ.....	26
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	27

ВСТУП

За останні роки відбулися серйозні події, які перевернули життя багатьох людей у цьому світі, змусивши нас адаптуватися до нових способів роботи, навчання, спілкування і загалом до нового життя. Пандемія COVID-19 і локдауни спричинили закриття всіх місць розваг, зокрема кінозалів та театрів, тому перегляд фільмів став популярною розвагою, джерелом хорошого настрою та, навіть, розвитку. Протягом цих років суспільство при звичалося до, так званого “онлайн життя”. У багатьох людей робота та навчання повністю або частково проводиться дистанційно, що значно економить час на добирання в офіс чи додому. Завдяки цьому вони мають можливість проводити більше часу за переглядом телеекранів, тому актуальність фільмів та серіалів залишиться і після закінчення пандемії.

Згодом відбулося повномасштабне вторгнення Російської Федерації на нашу землю, яке перетворило життя на “до” та “після”. Українці щохвилини перевіряли новини та перебували у постійному стресі. Для нас перегляд фільмів вечорами став можливістю тимчасово втекти від реальності та відволіктися, щоб наступного дня з новими силами підтримувати нашу країну та донатити ЗСУ.

Часто знайти підходящий фільм непросто. Якись фільми нам рекомендують друзі, про якись фільми ми чуємо з інших джерел на подібні соцмереж чи реклами, але у потрібний момент не можемо згадати назву потрібного фільму. З’являється потреба записувати або десь зберігати їх. Багато людей використовують нотатки та інші записники, але, погодьтеся, це не найкращий і, однозначно, не найзручніший варіант.

З огляду на вищеописані причини, метою цієї роботи є створення веб-аплікації, яка буде зручною для зберігання та пошуку фільмів, відображення опису, жанрів, акторів, рейтинг та дату випуску фільмів,

також можна буде переглянути подібні фільми. Сайт матиме приємний інтерфейс, який буде інтуїтивно зрозумілий і комфортний для користувача. У ньому також забезпечуватиметься можливість реєстрації, пошуку та слідкування за користувачами та відображення їх улюблених фільмів.

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ, ОГЛЯД ВІДОМИХ РІШЕНЬ ТА ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

1.1 Постановка задачі

Завданням дипломної роботи є проектування веб-аплікації, призначеної для зручного відображення та надання інформації про фільми: назва, короткий опис, дата виходу, рейтинг, акторський склад та плакат. На сайті повинні бути списки популярних фільмів та з високим рейтингом, до кожного фільму можна переглянути список йому подібних. Також потрібно реалізувати авторизацію користувача, який, в свою чергу, буде мати змогу слідкувати за іншими користувачами. Для авторизованого користувача відкривається можливість додавання фільмів в улюблені та надсилання цього списку на пошту у форматі PDF, також він може переглянути улюблені фільми інших користувачів. Окрім цього, обов'язковою є локалізація сайту: у ньому повинні бути додані українська та англійська мови. Доступ до програмного забезпечення повинен здійснюватися із будь-якого сучасного персонального комп'ютера або телефону з доступом в інтернет.

Специфікація ставить наступні вимоги перед програмним продуктом:

- а) отримання інформації із відкритої бази даних фільмів: TMDB
- б) зміна мови
- в) відображення списку фільмів на вибраній мові
- г) можливість переглянути детальну інформацію про фільми
- г) відображення популярних, високо оцінених та подібних фільмів
- д) пошук фільмів за назвою
- е) створення/редагування/видалення особистого акаунту користувача
- є) додавання фільмів в улюблені (видалення з улюблених)
- ж) надсилання списку улюблених фільмів на пошту
- з) пошук користувачів

- и) слідування за іншими користувачами
- і) блокування певних функцій для неавторизованого користувача

1.2 Огляд відомих рішень

IMDb

Домашня сторінка: <https://www.imdb.com/>

IMDb – повна онлайн-база даних фільмів, вебсайт, який надає інформацію про мільйони фільмів і телевізійних програм, а також про їх акторський склад і знімальну групу. Назва є аббревіатурою від Internet Movie Database.

IMDb пропонує кілька послуг як частину більшої вебпрограми: містить інформацію та статистику про фільми, телешоу та відеоігри, а також про акторів, режисерів та інших професіоналів кіноіндустрії. Ця інформація може включати списки акторів і членів знімальної групи, дати виходу фільму та інформацію про касові збори, короткий опис сюжету, трейлери, біографії акторів і режисерів та інші дрібниці.

Користувачі також можуть надсилати відгуки про фільми та телешоу за шкалою від одного до десяти, які потім використовуються для створення середньозваженого значення всіх відгуків користувачів, які відобразатимуться на сторінці фільму чи телешоу. Крім того, ці рейтинги використовуються для складання списків IMDb із найвищим і найнижчим рейтингом фільмів і телешоу.

Недоліки:

- а) немає можливості для користувачів взаємодіяти: слідувати та переглядати список улюблених фільмів інших користувачів
- б) немає можливості отримання списку улюблених фільмів на пошту немає української мови

UAkino

Домашня сторінка: <https://uakino.club/>

UAkino – український вебсайт, з можливістю онлайн перегляду величезної кількості фільмів, серіалів, аніме, мультфільмів, а також завантаження їх за допомогою торрента. Усе можна дивитися у високій якості та українській озвучці.

Недоліки:

- а) немає можливості для користувачів взаємодіяти: слідкувати та переглядати список улюблених фільмів інших користувачів
- б) немає можливості отримання списку улюблених фільмів на пошту
- в) немає англійської мови
- г) немає ліцензії на фільми (піратство)

1.3 Методики реалізації та опис використаних технологій розробки програм

Для написання цього вебзастосунку найкраще підійшов фреймворк Ruby on Rails – це фулстек фреймворк, написаний мовою програмування Ruby, і є доступним для багатьох операційних систем, таких як Mac OS X, Windows і Linux . Фреймворк базується на шаблоні модель - представлення - контролер (MVC) і постачається з багатим набором інструментів, що охоплює основні функції, які стосуються проблем інтерфейсу і серверу. Для створення користувацьких інтерфейсів я використовую CSS фреймворк – Tailwind CSS та додаю інтерактивність до сторінок за допомогою Stimulus JS. Дані в цьому проекті зберігаються в СКБД PostgreSQL і Elastic, останній з яких я використовую для пошуку користувачів.

У цьому застосунку використовується архітектурний шаблон MVC для розбиття компонентів програми на три рівні, розподілення відповідальності та встановлення зв'язків між ними. Також реалізовані

дизайн шаблони: Form Object – для розподілення логіки та дотримання принципів програмування; Singleton – для реалізації налаштувань сайту. Як середовище програмування використано Visual Studio Code.

1.3.1 Ruby

Ruby – це інтерпретована, повністю об'єктно-орієнтована мова програмування з чіткою динамічною типізацією. Мова вирізняється високою ефективністю розробки програм і увібрала в себе найкращі риси Perl, Java, Python, Smalltalk, Eiffel, Ada і Lisp. Ruby поєднує в собі Perl-подібний синтаксис з об'єктно-орієнтованим підходом мови програмування Smalltalk. Також деякі риси запозичено із мов програмування Python, Lisp, Dylan та CLU. Ruby використовується в різних сферах: від системного адміністрування до розробки складних динамічних вебзастосунків. Основним напрямком є, звичайно, веброботка. Вже згаданий вище Ruby on Rails — дуже крутий і ефективний інструмент, який використовувався для створення багатьох відомих сервісів. Зокрема, Twitter, Airbnb, Crunchbase, Dribbble, Bloomberg, Netflix, SoundCloud, Upwork, Hulu, Basecamp, Kickstarter та багато інших.

Мова активно застосовується в системному адмініструванні, а також має інструменти для мобільних додатків (Ruboto, RubyMotion та ін.). Код, написаний на Ruby, може бути зрозумілим навіть людині, яка не розуміється на програмуванні.

1.3.2 Ruby on Rails

Ruby on Rails (RoR) — це фреймворк розробки вебзастосунків, написаний мовою програмування Ruby. Його розроблено, щоб полегшити програмування вебзастосунків, роблячи припущення про те, що потрібно кожному розробнику для початку. Він дозволяє вам писати менше коду, досягаючи при цьому більшого, ніж багато інших мов і фреймворків.

Rails побудовано на наборі попередньо визначених шаблонів, бібліотек і фреймворків, які дозволяють новачкам і професіоналам швидко впроваджувати різні функції, такі як надсилання пошти, читання даних із бази даних SQL та інші. Наприклад, Rails реалізує шаблон об'єктно-реляційного відображення (ORM) під назвою Active Record, який дає розробникам можливість взаємодіяти з базою даних за допомогою об'єктів Ruby.

На RoR були створені такі проекти як Redmine, Twitter, Shopify, Basecamp, GitHub, Kickstarter, Airbnb та інші.

Переваги Ruby on Rails:

- а) швидкість розробки – найважливішою перевагою Ruby on Rails є висока швидкість розробки, яка досягається завдяки простому синтаксису, а також великому вибору готових рішень для різних типів завдань
- б) спочатку тести, потім код – часто в процесі розробки великих проектів виникає питання про тестування і не рідкість, коли немає додаткових коштів на окрему команду тестувальників. У Rails є вирішення цієї проблеми. Якщо порівнювати RoR з іншими фреймворками в контексті тестування програми, то ви знайдете масу готових рішень для будь-якого виду тестів, чи то інтеграційні, чи юніт. Всі ці бібліотеки працюють із коробки. В ідеалі в проекті на Ruby on Rails код не пишеться доти, доки під нього не написані тести. RoR ідеологія передбачає початкове використання методів BDD (Behavior Driven Development) або TDD (Test Driven Development).
- в) величезна кількість готових рішень – тут є велика колекція бібліотек (RubyGems), де можна знайти практично все, що потрібно для реалізації певної ідеї

г) високий рівень захисту даних – зараз нерідко в мережі публікуються статті про злам різних ресурсів. Розробники фреймворку Ruby on Rails дуже серйозно поставилися до проблеми захисту даних. У RoR спочатку є шифрування паролів, даних кредитних карток та інших особистих даних користувача, також виключені SQL ін'єкції та XSS атаки. Усі вхідні параметри екрануються за замовчуванням.

1.3.3 Stimulus.JS

Stimulus JS – це бібліотека JavaScript, призначена для спрощення процесу додавання інтерактивності до вебсторінок. Він спеціально розроблений для роботи з серверними фреймворками, такими як Ruby on Rails та дозволяє легко додавати динамічну поведінку до вебсторінок, не вимагаючи складного коду JavaScript.

Однією з головних переваг використання Stimulus JS у додатку Rails є те, що він дозволяє підтримувати код JavaScript упорядкованим і придатним для обслуговування. Замість того, щоб мати багато спеціального коду, розкиданого по програмі, можна використовувати контролери Stimulus для керування поведінкою окремих елементів на сторінці. Це може полегшити налагодження коду та внесення змін у майбутньому.

1.3.4 PostgreSQL

PostgreSQL (Postgres) є потужною системою керування реляційною базою даних із відкритим кодом. Він широко використовується у веброботці, включаючи програми написані на Ruby on Rails. Однією з головних переваг використання Postgres у додатку Rails є його масштабованість і надійність. Postgres розроблений для обробки великих обсягів даних і може використовуватися для створення

високопродуктивних вебзастосунків, які вимагають складних запитів даних і транзакцій.

1.3.5 Elasticsearch

Elasticsearch — це пошукова система з відкритим вихідним кодом, яка широко використовується завдяки потужним пошуковим можливостям. Він призначений для використання з великими наборами даних і часто використовується у веброботці, включаючи програми Ruby on Rails.

Однією з головних переваг використання Elasticsearch у додатку Rails є його здатність надавати швидкі та точні результати пошуку. Він призначений для роботи з великими обсягами даних і може бути легко збільшений або зменшений залежно від потреб програми.

1.3.6 Tailwind CSS

Tailwind CSS — це CSS-фреймворк, який надає набір попередньо визначених класів CSS, які можна використовувати для стилізації елементів HTML. Він розроблений, щоб допомогти розробникам швидко створювати користувацькі інтерфейси без необхідності писати власний CSS.

Використання Tailwind CSS у програмі Rails може допомогти покращити узгодженість, зручність обслуговування та швидкість реагування інтерфейсів користувача. Набір попередньо визначених класів CSS допомагає швидко створювати користувацькі інтерфейси користувача, які матимуть чудовий вигляд, а також полегшуватимуть підтримку та оновлення коду з часом.

1.3.7 Архітектурний шаблон MVC

Rails має каталог додатків під назвою `app/` з трьома підкаталогами: `models`, `views` і `controllers`. Це архітектурний шаблон модель-подання-контролер (MVC), який забезпечує відокремлення

бізнес-логіки від вхідної інформації та логіки представлення, пов'язаної з графічним інтерфейсом користувача (GUI).

Під час взаємодії з додатком браузер надсилає запит, який отримує вебсервер і передає механізму маршрутизації Rails. Маршрутизатор отримує запит і перенаправляє до відповідного методу класу контролера на основі шаблону URL-адреси маршрутизації. Тоді контролер бере на себе роботу. У деяких випадках контролер негайно візуалізує перегляд для браузера. Для динамічних сайтів контролер взаємодіє з моделлю. Після виклику моделі контролер відтворює остаточний вигляд (HTML, CSS і зображення) і повертає повну вебсторінку в браузер користувача.

1.3.8 Дизайн шаблони

Form Object – це шаблон проектування, який використовується для інкапсуляції пов'язаної з формою логіки та поведінки за межами рівня моделі та контролера. Він забезпечує спосіб обробки складних форм або форм, які потребують додаткової обробки перед взаємодією з базою даних. Традиційно програми Rails обробляють подання форм безпосередньо в контролері та покладаються на моделі ActiveRecord для визначення схеми бази даних і перевірок. Однак, оскільки форми стають складнішими, це може призвести до роздутих контролерів, які важко обслуговувати.

Об'єкти форми забезпечують спосіб вилучення специфічної для форми логіки в окремий клас, зменшуючи складність контролера та зосереджуючи рівень моделі на завданнях, пов'язаних із збереженням.

Singleton — це шаблон проектування, який обмежує створення екземпляра класу одним об'єктом. Це гарантує наявність лише одного екземпляра класу в усій програмі. Він часто використовується для керування глобальними або спільними ресурсами, які повинні мати єдину точку доступу. Шаблон Singleton можна використовувати для

представлення таких об'єктів, як параметри конфігурації, підключення до бази даних або стан усієї програми. Він допомагає підтримувати узгодженість і дозволяє уникнути дублювання або конфліктів, які можуть виникнути через кілька екземплярів одного класу.

1.3.9 Ruby gems

1.4.1 Devise

Devise — це популярне рішення аутентифікації для програм Ruby on Rails. Це бібліотека, яку можна легко інтегрувати в програму Rails для виконання поширених завдань автентифікації, таких як реєстрація користувача, вхід, скидання пароля та керування сесією. **1.4.2 Rails-i18n**

Rails-i18n, також відома як «Інтернаціоналізація Rails», — це бібліотека Ruby, яка забезпечує підтримку інтернаціоналізації (i18n) і локалізації для програм Ruby on Rails. Це важливий компонент для створення додатків, які можна легко перекладати кількома мовами, дозволяючи розробникам адаптувати свої додатки до різних регіонів і орієнтуватися на глобальну аудиторію.

1.4.2 Wicked_pdf

Wicked PDF — це бібліотека Ruby, яка забезпечує зручний спосіб генерувати PDF-документи з шаблонів HTML у програмах Ruby on Rails. Вона дозволяє динамічно створювати PDF-файли на основі даних і макета вашої програми, використовуючи можливості HTML і CSS для налаштування.

1.3.10 Visual Studio Code

Visual Studio Code (VS Code) — широко використовуваний редактор вихідного коду, розроблений Microsoft. Це легкий редактор, який зручно та просто налаштовувати, призначений для розробників, які працюють на різних мовах програмування та платформах.

1.3.11 GitHub

GitHub — це онлайн-платформа розробки програмного забезпечення, яка використовується для зберігання, відстеження та співпраці над програмними проектами. Це дозволяє розробникам завантажувати власні файли коду та співпрацювати з іншими розробниками над проектами з відкритим кодом. GitHub також служить сайтом соціальної мережі, на якому розробники можуть відкрито спілкуватися, співпрацювати та презентувати свою роботу.

GitHub дозволяє розробникам програмного забезпечення та інженерам безкоштовно створювати віддалені загальнодоступні сховища в хмарі. Після того, як ви налаштували репозиторій на GitHub, ви можете скопіювати його на свій пристрій, локально додавати та змінювати файли, а потім «відправляти» свої зміни назад у сховище, де ваші зміни відображаються для всіх.

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ВІДПОВІДНО ДО ВИМОГ

Під час розробки даного додатку було використано вищезгадані технології: мова програмування Ruby, середовище програмування Visual Studio Code, фреймворк Ruby on Rails, вебсервіс контролю версій GitHub. Як базу даних для аплікації було обрано PostgreSQL. Усі вони були описані у попередніх розділах. Для реалізації функціоналу було створено відповідний користувацький інтерфейс.

2.1 Загальний опис та характеристика продукту

Дана веб-аплікація була створена для вирішення наступного ряду вимог:

- а) відображення популярних, високо оцінених та подібних фільмів
- б) зміна мови
- в) відображення списку фільмів на вибраній мові
- г) можливість переглянути детальну інформацію про фільми
- г) створення/редагування/видалення особистого акаунту користувача
- д) пошук користувачів та фільмів за назвою
- е) додавання фільмів в улюблені (видалення з улюблених)
- є) надсилання списку улюблених фільмів на пошту
- ж) слідкування за іншими користувачами
- з) блокування певних функцій для неавторизованого користувача

Система має один тип класів користувачів – користувач. Користувач може переглянути сторінку з переліком фільмів, додаткові інформацію про кожен фільм, список улюблених фільмів, шукати конкретний фільм, змінити мову та побачити всю інформацію вибраною мовою. Може переглядати інших користувачів, у яких також відображаються їх улюблені

фільми. Також користувач може переходити на сторінки з створенням та редагуванням особистого акаунту.

2.3 База даних проекту

База даних створюється локально і одразу підключається до програми.

Схема бази даних:

Users table

Column	Type	Description
email	string	user's email
encrypted_password	string	secured user's password
nickname	string	user's nickname
created_at	datetime	automatically created timestamps
updated_at	datetime	automatically created timestamps

Relationships table

Column	Type	Description
follower_id	integer	who is following me
followee_id	integer	who am i following
created_at	datetime	automatically created timestamps
updated_at	datetime	automatically created timestamps

Movies table

Column	Type	Description
--------	------	-------------

tmdb_id	integer	movie_id from tmdb
created_at	datetime	automatically created timestamps
updated_at	datetime	automatically created timestamps

Favorites table

Column	Type	Description
movie_id	bigint	id of liked movie from local db
user_id	bigint	user's id
created_at	datetime	automatically created timestamps
updated_at	datetime	automatically created timestamps

2.4 Підключення TMDB API

TMDB ([The Movie Database](#)) — це популярна база даних для фільмів і телешоу. Весь вміст було додано та оновлено користувачами, починаючи з 2008 року. TMDB зосереджено на зборі даних з усього світу та охоплює широкий спектр даних різними мовами.

За допомогою API TMDB можна програмно отримувати та використовувати дані, надані TMDB. Служба TMDB API призначена для тих, хто зацікавлений у використанні даних фільму, телешоу або актора та, якщо потрібно, зображень у своїй програмі. Щоб використовувати TMDB у програмі, потрібно налаштувати API за допомогою платформи [TMDB for developers](#)

Додавання ключа API

Щоб додати ключ API, потрібно створити новий файл `tmdb.rb` за таким шляхом: `config/initializers/`. Та додати ось цей запис:

`Tmdb::Api.key("YOUR-KEY-HERE"),`
який містить ключ API, створений на попередньому кроці.

РОЗДІЛ 3. СЦЕНАРІЙ РОБОТИ ВЕБ-АПЛІКАЦІЇ

Увійшовши в програму, користувача зустріне перелік популярних фільмів.

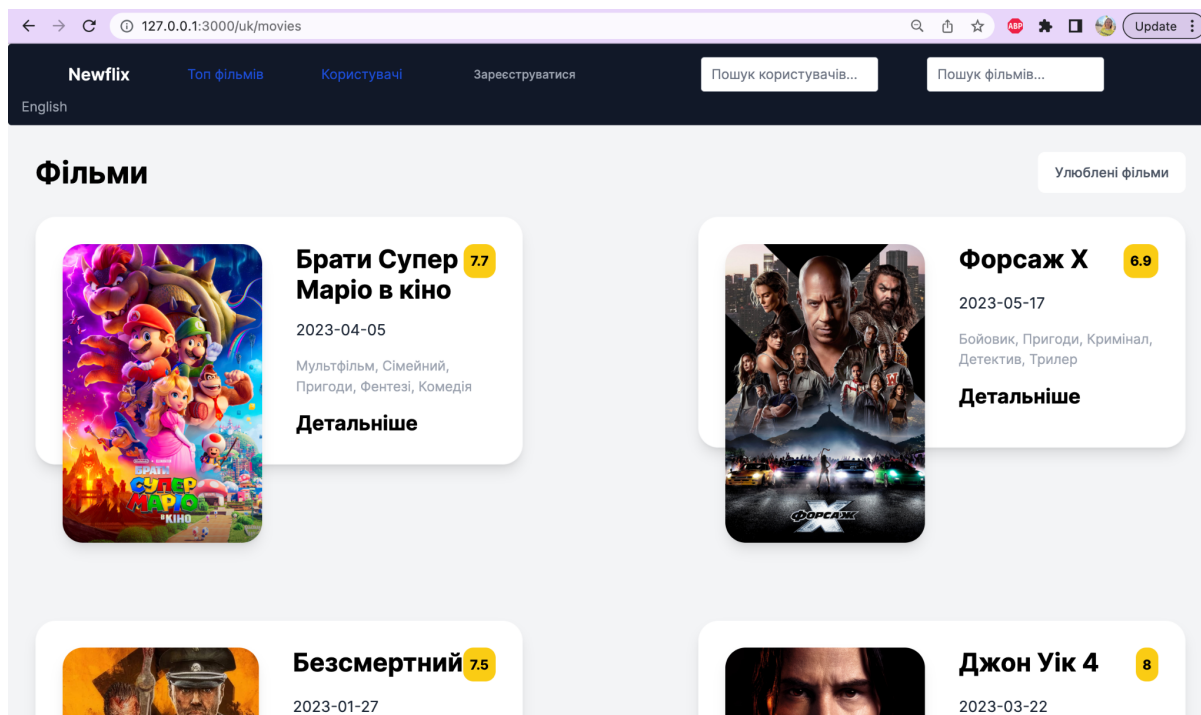


Рисунок 3.1 – перелік популярних фільмів

На рисунку у правому верхньому куті знаходиться кнопка, натиснувши на яку можна побачити перелік улюблених фільмів зареєстрованого користувача, в іншому випадку відображається таке повідомлення, зображене на рис. 3.2.

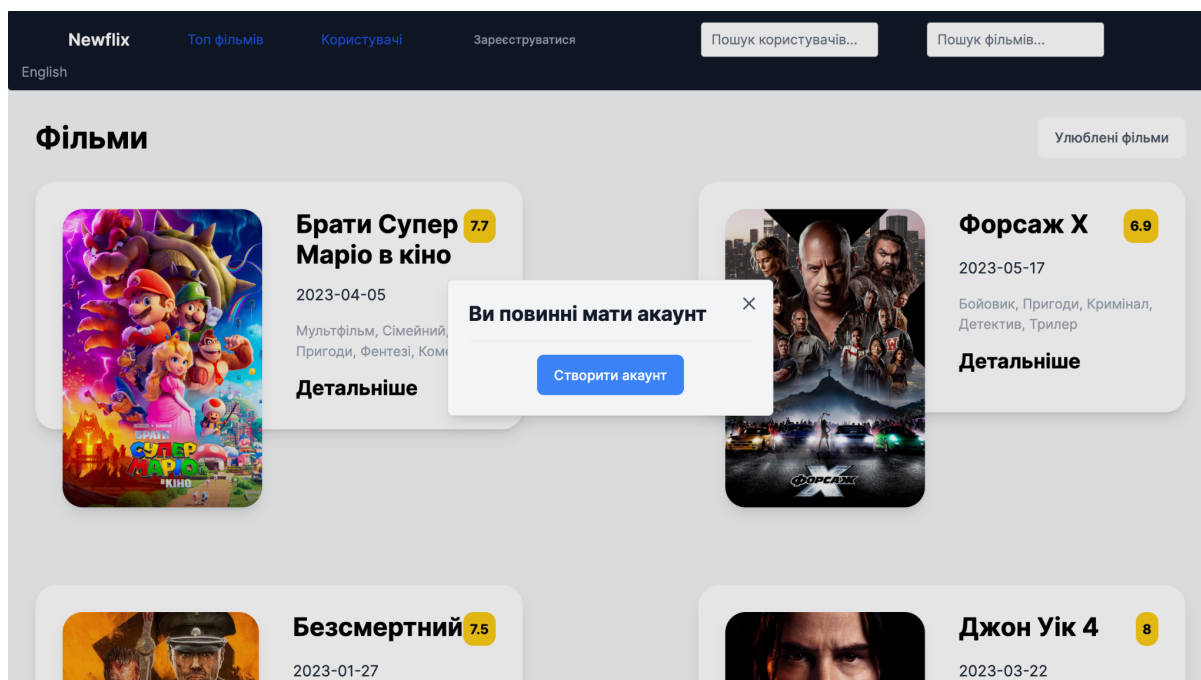


Рисунок 3.2 – пропозиція зареєструватися

Зареєстрований користувач може переглядати фільми, додані в улюблені, рис. 3.3.

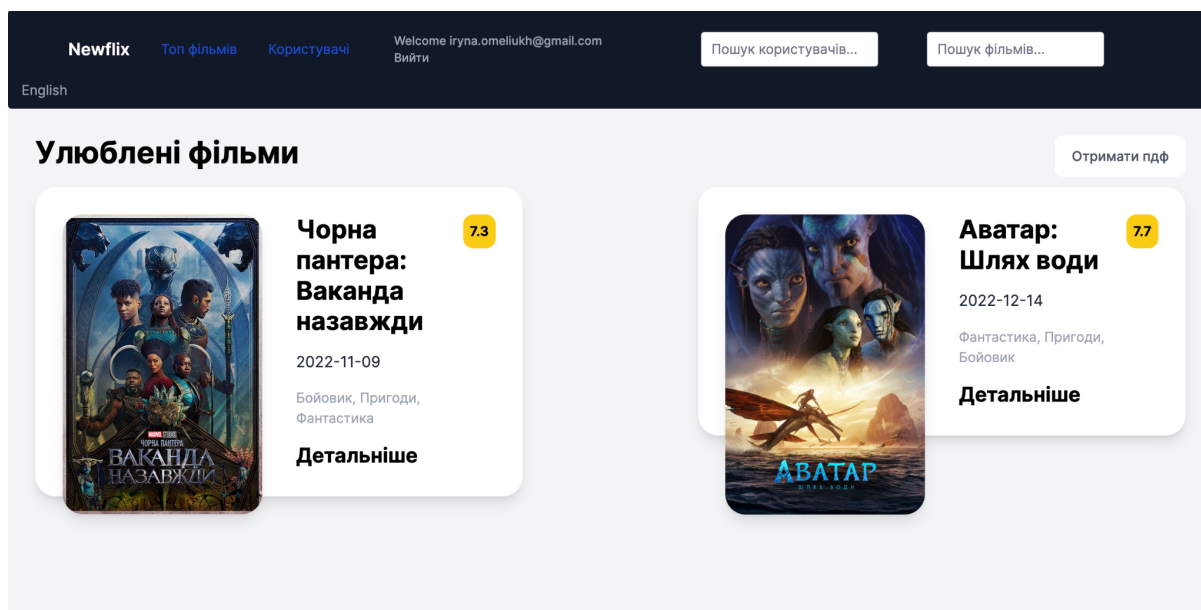


Рисунок 3.3 – улюблені фільми

Список улюблених фільмів можна надіслати на пошту у форматі PDF, рис. 3.4 та 3.5.

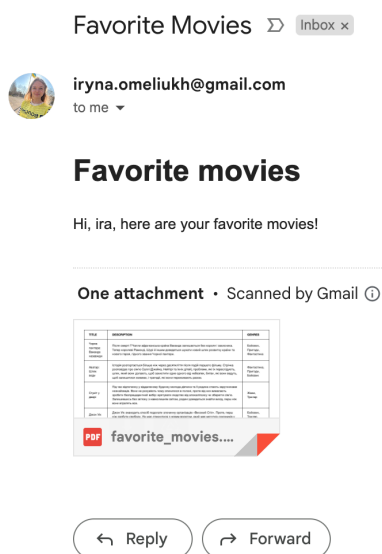


Рисунок 3.4 – лист з файлом

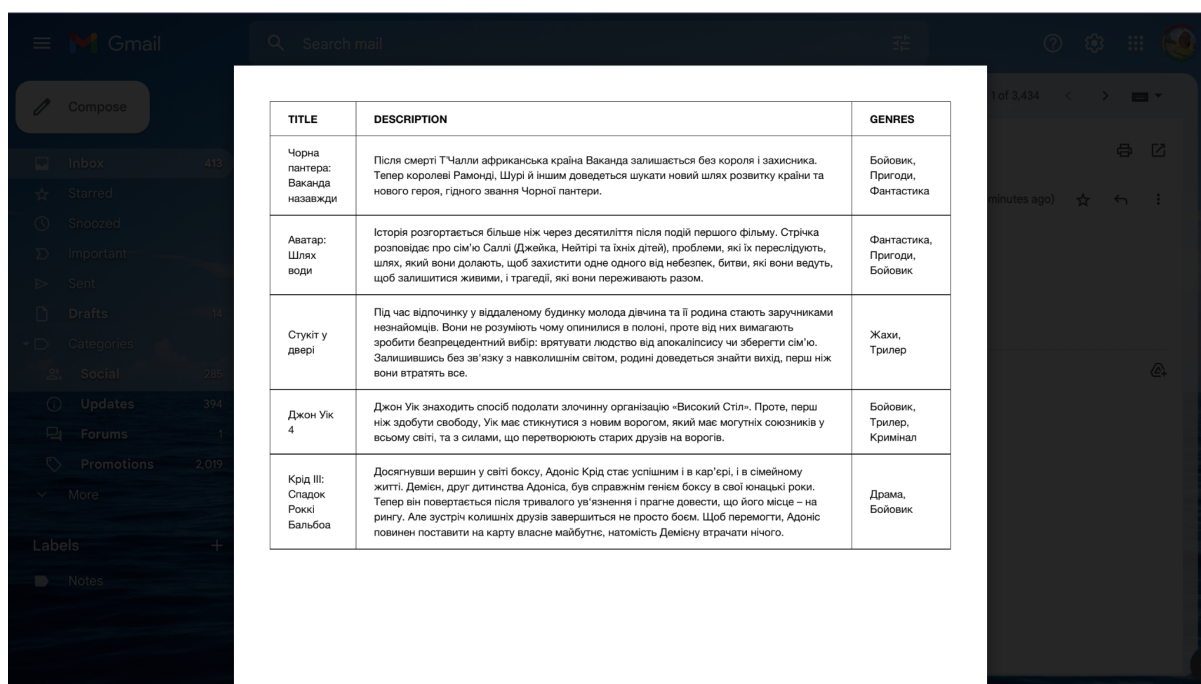


Рисунок 3.5 – вміст файлу

На сайті можна шукати фільми, ввівши ключове слово з назви фільму, що є зображене на рис. 3.6.

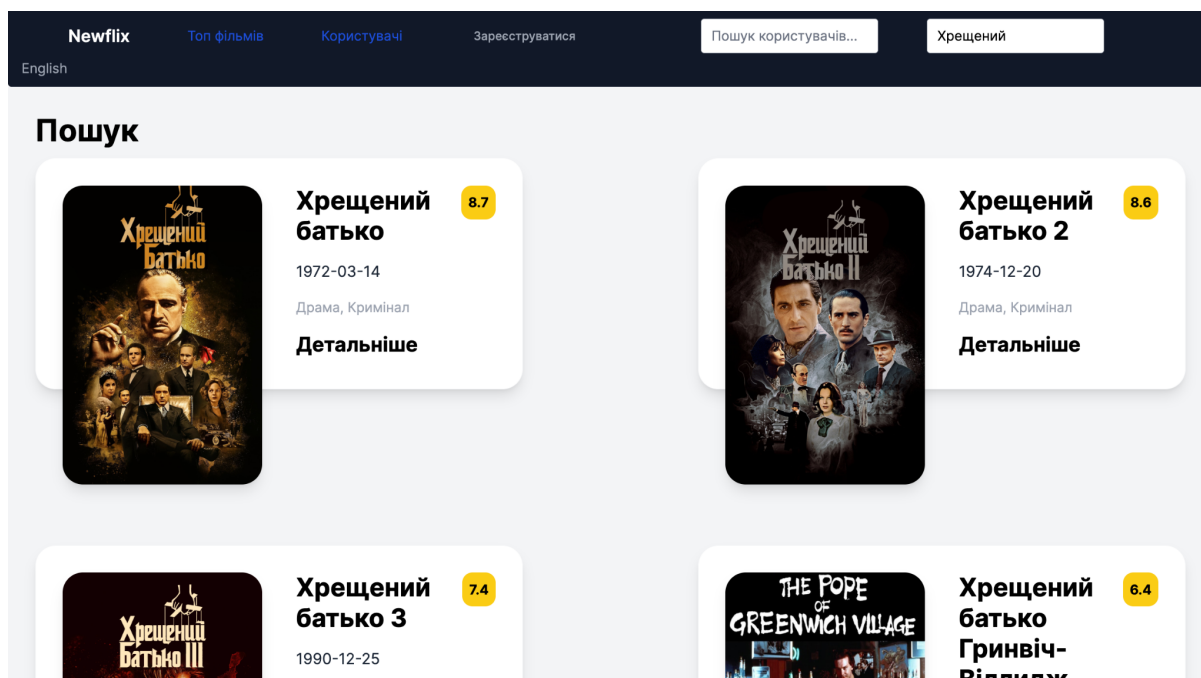


Рисунок 3.6 – пошук фільмів

Щоб переглянути більш детальну інформацію щодо певного фільму потрібно натиснути кнопку “Детальніше”. Результат цього бачимо на рис. 3.7.

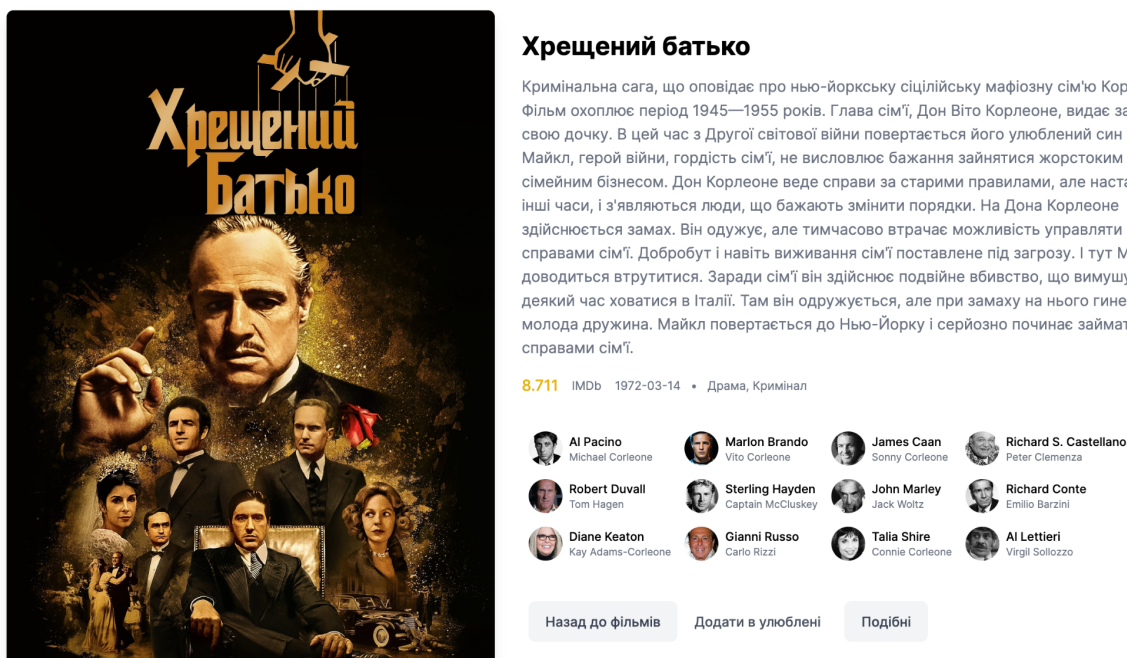


Рисунок 3.7 – детальна інформація про фільм

Тут зареєстрований користувач може додати фільм в улюблені та переглянути підбірку подібних фільмів, натиснувши кнопку “Подібні”, рис. 3.8.

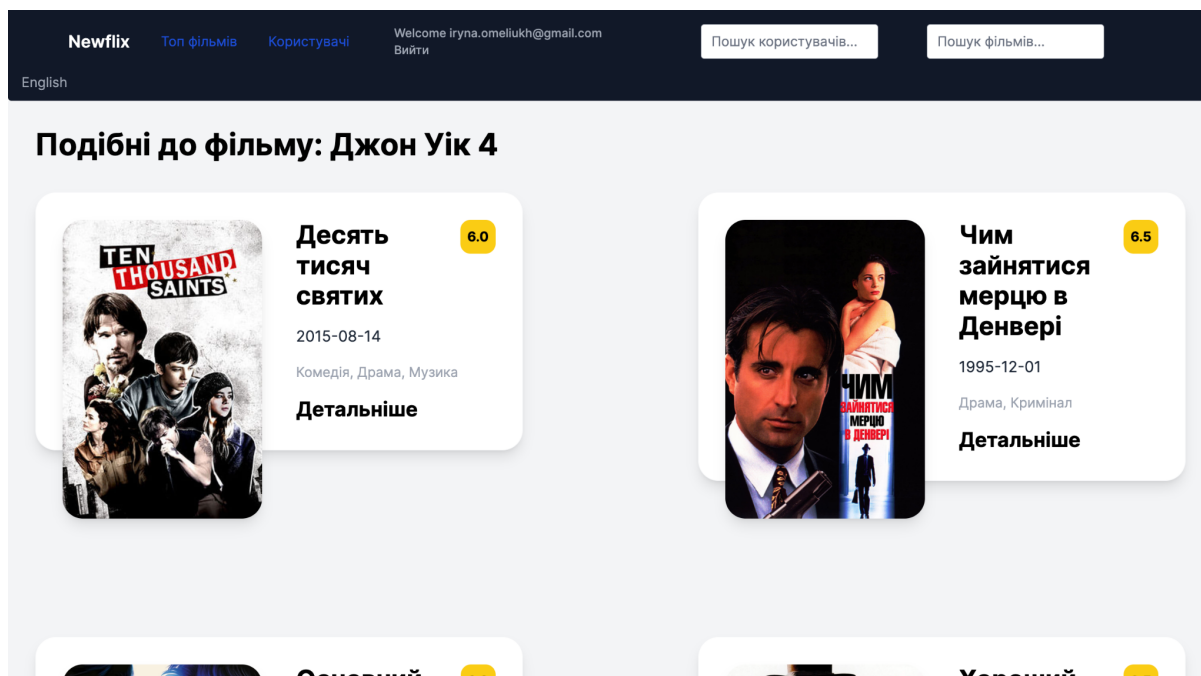


Рисунок 3.8 – подібні фільми

На рис. 3.9 зображений профіль існуючого користувача, який може переглянути незареєстрований користувач.

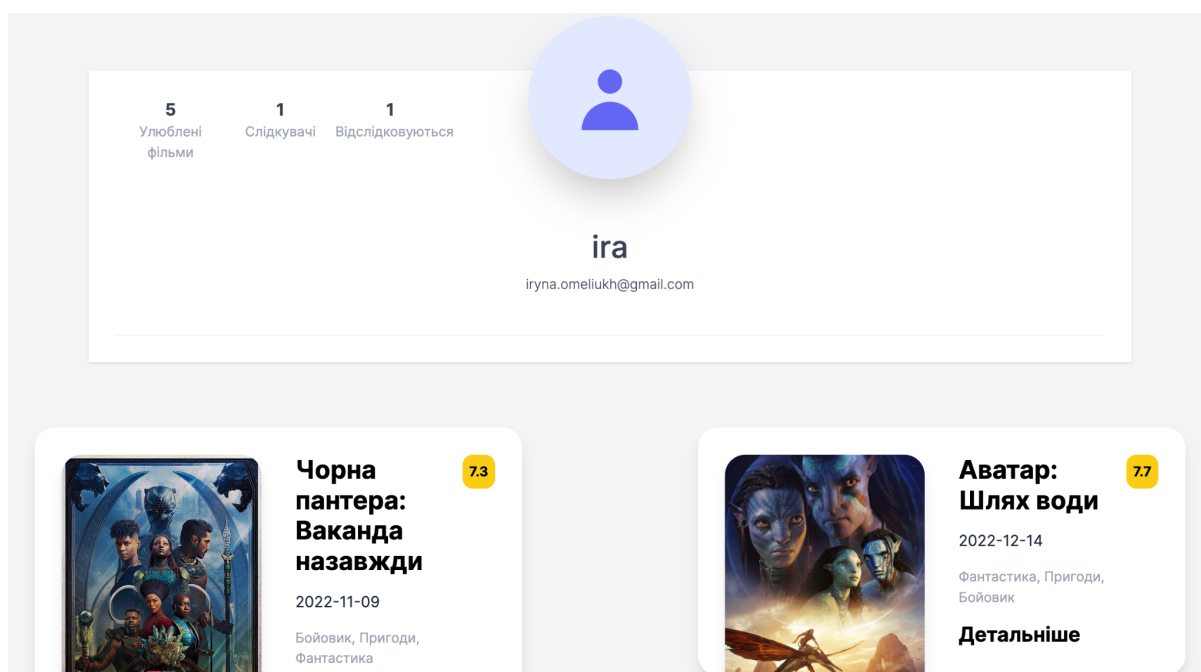


Рисунок 3.9 – профіль існуючого користувача

Користувач може змінювати мову, а також, після авторизації, редагувати свій профіль (рис. 3.10), переглядати та відслідковувати інших користувачів (рис. 3.11).

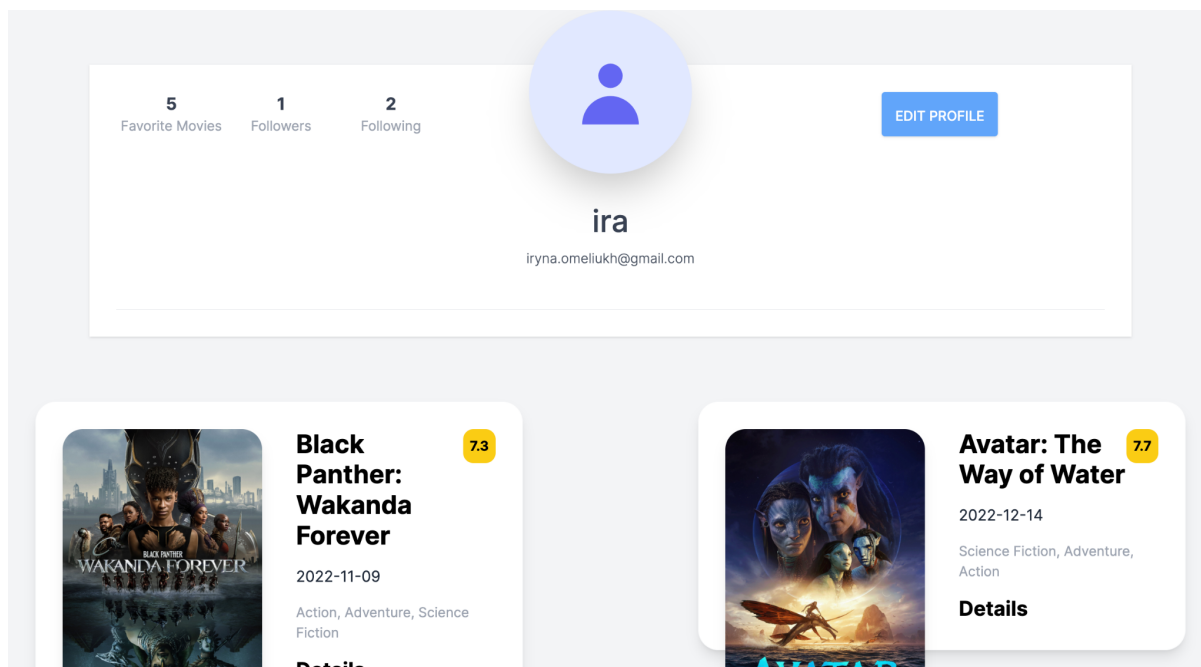


Рисунок 3.10 – власний профіль зареєстрованого користувача

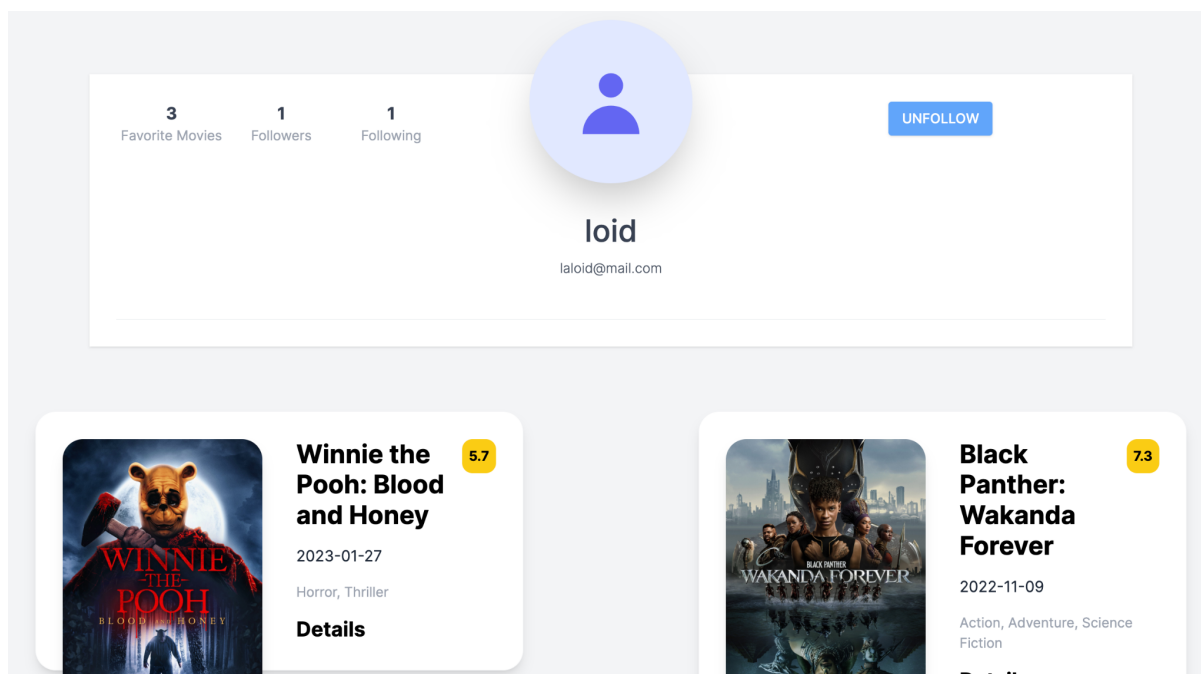


Рисунок 3.11 – профіль іншого користувача

ВИСНОВКИ

В цій дипломній роботі було розроблено вебзастосунок для вибору фільмів за допомогою мови програмування Ruby та з використанням фреймворку Ruby on Rails.

Основна мета роботи полягала у створенні зручного та функціонального сервісу для вибору фільмів, додавання їх до списку улюблених, а також відстеження активності користувачів. Це було досягнуто шляхом розробки моделей, контролерів та представлень, використання бази даних для зберігання інформації про фільми та користувачів, а також реалізації функцій для зручного взаємодії з додатком.

Під час написання даної програми було використано чимало новітніх технологій (Ruby, Ruby on Rails, PostgreSQL, Stimulus.JS та інші). Також я розмістила вихідний код на платформі GitHub, який можна переглянути за посиланням <https://github.com/IrynaOmeliukh/Newflix>

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Hartl M. Ruby on Rails Tutorial / Hartl M. – Addison-Wesley Professional, 2021. – 816 p.
2. Thomas D. Agile Web Development with Rails 7 / Thomas D. – Pragmatic Bookshelf, 2022. – 474 p.
3. David A. Black The Well-Grounded Rubyist / David A. Black – Manning Publications, 2014. – 584 p.
4. David Bryant Copeland Sustainable Web Development with Ruby on Rails / David Bryant Copeland – Amazon Digital Services LLC - KDP Print US, 2020. – 472 p.