

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра прикладних інформаційних систем

122 «Комп'ютерні науки»

(шифр і назва спеціальності)

«Прикладне програмування»

(назва освітньої програми)

Кваліфікаційна робота бакалавра

на тему: «Веб-сервіс із підтримки діяльності кінотеатру»

Виконала _____
(Підпис)

Брік Катерина Вадимівна
(прізвище, ім'я, по батькові)

Керівник Бойко Юлія Петрівна
(прізвище, ім'я, по батькові)

(Резолюція «До захисту»)

Попередній захист:

(Висновок: “До захисту в екзаменаційній комісії”)

Завідувач кафедри _____ Плескач В.Л.
(Дата) (Підпис) (Прізвище, ініціали)

Київ – 2021

Київський національний університет імені Тараса Шевченка
Факультет інформаційних технологій
Кафедра прикладних інформаційних систем

Назва теми: «Веб-сервіс із підтримки діяльності кінотеатру»

Освітня програма: Прикладне програмування
Спеціальність: Комп'ютерні науки

ПІБ

Підпис

Брік Катерина Вадимівна

Назва роботи українською та англійською мовами

Веб-сервіс із підтримки діяльності кінотеатру

Web service to support the activities of the cinema

Мета бакалаврської роботи, завдання

Мета бакалаврської роботи: підвищення ефективності процесу перегляду кінофільмів та замовлення білетів на кіносеанс

План роботи:

1. Проаналізувати сучасні підходи до розроблення і впровадження веб сервісів
2. Аналіз архітектурних рішень і вибір програмних засобів для реалізації веб-сервісу
3. Програмна реалізація веб-сервісу підтримки діяльності кінотеатру

Бойко Юлія Петрівна, доцент, кандидат технічних наук

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

№ п/п	Назва етапів бакалаврської роботи	Термін виконання етапів бакалаврської роботи	Відмітка про виконання
1.	Вибір теми та наукового керівника бакалаврської роботи	26.10.2020	
2.	Видача завдання бакалаврської роботи	23.11.2020	
3.	Настановча групова співбесіда з бакалаврської роботи	01.12.2020	
4.	Затвердження плану бакалаврської роботи	18.02.2021	
5.	Підбір та вивчення літературних та інших джерел з теми дослідження	25.02.2021	
6.	Підготовка і подання науковому керівнику першого варіанту I розділу роботи	05.03.2021	
7.	Підготовка і подання науковому керівнику першого варіанту II розділу роботи	09.04.2021	
8.	Підготовка і подання науковому керівнику першого варіанту III розділу роботи	07.05.2021	
9.	Подання роботи у першому варіанті	11.05.2021	
10.	Оформлення пояснювальної записки бакалаврської роботи	12.05.2021	
11.	Подання бакалаврської роботи на попередній захист	24.05.2021	
12.	Врахування зауважень керівника і подання роботи в остаточному варіанті (з відповідним висновком про допуск) на кафедру	28.05.2021	
13.	Затвердження роботи в цілому (підготовка письмового відгуку керівника, письмова рецензія на бакалаврську роботу)	11.06.2021	
14.	Захист бакалаврської роботи	16.06.2021	

Здобувач вищої освіти _____ (підпис)

Керівник _____ (підпис)

ЗМІСТ

АНОТАЦІЯ	5
ВСТУП	6
РОЗДІЛ 1: АНАЛІЗ СУЧАСНИХ ПІДХОДІВ ДО РОЗРОБЛЕННЯ І ВПРОВАДЖЕННЯ ВЕБ-СЕРВІСІВ НА ТЕМАТИКУ КІНОМИСТЕЦТВА	7
1.1 Опис предметної області веб-системи кіномистецтва та перегляду фільмів	7
1.2 Огляд відомих реалізацій веб-системи кіномистецтва та перегляду фільмів	9
1.3 Формулювання вимог на розробку ТЗ	15
Висновки	16
РОЗДІЛ 2: АНАЛІЗ АРХІТЕКТУРНИХ РІШЕНЬ І ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ ВЕБ-СЕРВІСУ	17
2.1 Аналіз методологій для побудови архітектурного рішення	17
2.2 Реалізація структури програмного додатку	20
2.3 Обґрунтування середовища і мови програмування	24
Висновки	27
РОЗДІЛ 3: ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СЕРВІСУ ПІДТРИМКИ ДІЯЛЬНОСТІ КІНОТЕАТРУ	27
3.1 Алгоритм роботи системи веб-сервісу підтримки діяльності кінотеатру	27
3.2 Обґрунтування середовища і мови програмування	30
3.3 Програмування веб-сервісу підтримки діяльності кінотеатру	30
ВИСНОВКИ	40
СПИСОК ЛІТЕРАТУРИ	41
Додаток А. Основний лістинг	43

АНОТАЦІЯ

Дана дипломна робота присвячена створенню адаптивних динамічних веб-сервісів з використанням сучасних інструментів розробки та проектування баз даних.

У даній роботі досліджено сучасні веб-системи кіномистецтва та перегляду фільмів.

Розглянуто найкращі реалізації подібних систем, виділено їх особливості та на основі даного аналізу сформовано технічне завдання.

Проаналізовано ряд методологій проектування, побудовано діаграми та структурні схеми. Побудовано UML-діаграми, розроблено структурну схему веб-системи кіномистецтва та перегляду фільмів.

На основі цього побудовано алгоритм функціонування веб-системи кіномистецтва та перегляду фільмів.

Здійснено програмну реалізацію веб-системи кіномистецтва та перегляду фільмів.

Загальний обсяг роботи: 60 сторінок, 34 рисунків, 4 таблиці, 12 посилань.

Ключові слова: веб-застосунок, клієнт-сервіс, кінотеатр.

ABSTRACT

This thesis is devoted to the creation of adaptive dynamic web services using modern tools for database development and design.

In this paper, modern web systems of cinematography and movie viewing are studied.

The best realizations of similar systems are considered, their features are allocated and on the basis of the given analysis the technical task is formed.

A number of design methodologies are analyzed, diagrams and structural schemes are constructed. UML-diagrams were built, the structural scheme of the web system of cinematography and movie viewing was developed.

Based on this, the algorithm of functioning of the web system of cinematography and movie viewing is built.

The software implementation of the web system of cinematography and watching movies has been implemented.

Total volume of work: 60 pages, 34 figures, 4 tables, 12 links.

Keywords: web application, customer service, cinema.

ВСТУП

Актуальність дослідження. Предметна область кіномистецтва та перегляду фільмів не втратила свою актуальність і донині. Люди люблять фільми як мистецтво з декількох причин, і часто це пояснюється їх лаконічністю, поєднанням різних видів творчої діяльності, застосуванням новітніх технологій та ефектів. На сьогодні фільми мають чи не найширшу аудиторію порівняно з іншими видами мистецтва. Водночас, дуже часто постає питання про те чи варто йти в кіно або ж витратити час на перегляд фільму вдома – таким чином, знайти справді цікавий фільм серед багатьох і на будь-який смак допомагають рецензії. Тому формування відкритої веб-системи фільмів з можливістю їх оцінювання та написання рецензій на них є досить цікавою та корисною задачею.

Метою дипломної роботи є підвищення ефективності процесу перегляду кінофільмів та замовлення білетів на кіносеанс.

Завдання дослідження:

- аналіз сучасних підходів до розробки та впровадження веб-сервісів;
- аналіз архітектурних рішень і вибір програмних засобів для реалізації веб-сервісу;
- програмна реалізація веб-сервісу підтримки діяльності кінотеатру.

Об'єктом дослідження у цій роботі є Веб-сервіс з підтримки діяльності кінотеатру, а **предметом** – сучасні технології розробки та проектування веб-ресурсів.

РОЗДІЛ 1: АНАЛІЗ СУЧАСНИХ ПІДХОДІВ ДО РОЗРОБЛЕННЯ І ВПРОВАДЖЕННЯ ВЕБ-СЕРВІСІВ НА ТЕМАТИКУ КІНОМИСТЕЦТВА

Веб-додаток — додаток, де користувач виступає в ролі клієнта, а в ролі сервера – віртуальна серверна машина. Оглядач Інтернету реалізується як тонкі клієнти. В його функціонал входить відображення веб-сторінок, і є частиною власне операційної системи, проте його супровід виконує постачальник ОС. Логічна пам'ять пристрою зберігається на сервері, а сам оглядач лише представляє інформацію, що завантажена з сервера і передає певні дані користувача. Перевагою даного підходу є значна незалежність клієнтів від тої чи іншої ОС, таким чином, веб-ресурс має міжплатформенну і кросплатформенну природу.

Веб-додаток відповідає на запити клієнтів, виконує певні операції і відправляє їх клієнту через мережу з використанням протокола HTTP.

Веб-додаток може бути клієнтом інших служб, наприклад, бази даних або іншого веб-додатку, розташованого на іншому сервері.

В останні роки значного розповсюдження набула технологія Ajax. Сторінки ресурсу не потрібно перевантажувати повністю, а лише завантажують з сервера ЗМІНИ веб-сторінки, що робить їх більш інтерактивними і продуктивними.

1.1 Опис предметної області веб-системи кіномистецтва та перегляду фільмів

Кінематограф є галуззю водночас мистецтва та культури з економічною складовою, що об'єднує всі види професійної діяльності, яка відповідає за виробництво, поширення, збереження та демонстрацію фільмів, а також навчально-науковою роботою (див. кінознавство).

Великий обсяг займає кіноіндустрія – частина промисловості, що займається виробництвом фільмів, сучасних технологій та спецефектів. Багато

країн містять кінематографію як важливу галузь економіки, і варто зазначити що осередки виробництва – це кіностудії. Готові фільми представлені у кінотеатрах, по телебаченню, на сучасних носіях інформації.

Словом кінематографія іноді позначають також:

-Кіномистецтво — вид образотворчого мистецтва, що винайдений у 19 столітті і став найпопулярнішим у 20 столітті.

-Кінознімання — комплекс пристроїв та методів, що необхідні для знімання рухомих об'єктів на кіноплівку, і для наступного відображення отриманих знімків, шляхом проектування їх на екран.

Виробництво кінофільмів може відбуватись за межами систем голівудських кіностудій, тому носить назву незалежне (або інді) кіновиробництво.

Незалежний фільм («фільм інді») — це фільм створений за власні кошти автора кінофільму, без будь-якого спонсорства. Багато причин, в тому числі бізнес, розвиток мистецтва та технологій викликали високий та потужний ріст виробництва інді-фільмів.

Проте стало набагато більш важко вмовити професійних фахівців на певні експерименти у творчості, адже інновації такого роду можуть нести значні ризики.

У аспекті бізнесу - витрати на виробництво високобюджетних фільмів призводило до виникнення певних консервативних тенденцій у виборі команди кінопроектів і акторського складу. Проблема тільки поглибилась переходом до співфінансування. Більше двох третин фільмів, які випустила у прокат студія «Ворнер Броз.» у 2000-му році, мали фінансування з різноманітних джерел, а у 1987-му році таких було лише десять відсотків. Режисер-початківець майже не має можливості отримати кошти та зняти фільм у відомій студії, за умови якщо він не має солідного та визнаного досвіду у сфері кіно чи у телебаченні. Також надзвичайно рідко виходять на екрани фільми з маловідомими акторами у головних ролях [1].

До того, як з'явилися альтернативні ресурси кіномистецтва, вартість техніки була теж значним критерієм для можливості збагатити фільм спецефектами чи анімацією.

На сьогодні, спеціальні технології і програмне забезпечення для пост-виробництва можуть бути встановлені на будь-який домашній девайс, що схожий на комп'ютер. Подібні технології як Ді-Ві-Ді (DVD), програмне забезпечення як «Adobe Premiere Pro», «Final Cut Pro», «Final Cut Express», «iMovie» роблять кіновиробництво відносно недорогим для середньостатистичного зацікавленого користувача [2]

Поява у світі технологій цифрового відео (DV) означало, що кіновиробництво стало доступнішим та демократичнішим. Виробник має можливість, за бажання, зняти фільм, здійснити монтаж, записати звук та обрати потрібну музику, і створити остаточний варіант на домашньому комп'ютері. Варто зазначити, що хоча засоби виробництва стали доступнішими, фінансування, розповсюдження та маркетинг вкрай важко здійснюється за межами традиційно прийнятої системи. Переважна більшість незалежних кіновиробників вимушені з'являтися на тематичних конференціях та кінофестивалях, аби їхні проекти та фільми були помічені і їх можна було продати для розповсюдження [3].

Основне кіно, здається, вступає в дивну фазу своєї творчості. Наші мультиплекси тепер наповнені продовженнями і ремейками, і, здається, у кінорежисерів швидко закінчуються ідеї. Фільми, що отримали нагороди 2016 року, мали дещо різноманітний характер, а деякі надзвичайно спонукали до роздумів і мали дуже потужні послання. І все ж деякі інші чудові фільми не привернули уваги, яку вони заслуговували, хоча вони демонстрували частину історії, яка надзвичайно актуальна сьогодні.

Одним з багатьох таких прикладів був фільм "Приховані фігури", справжня історія, створена на початку 1960-х років про трьох блискучих афроамериканських жінок-математиків, які відіграли важливу роль у найперші

часи НАСА. Їх роль була вирішальною для успіху космічної гонки в той час, коли США та Росія були посеред холодної війни, і якби результат був вирішений будь-яким іншим способом, ніж він, наслідки могли потенційно означати кінець НАСА. Це був улюблений улюблений фільм минулого року, і він містив усі елементи, щоб змусити завоювати більше нагород та успіху. У фільмі було чудове повідомлення, чудові вистави, які були і веселими, і також мали серйозні переваги. Це була фантастична справжня історія, яка вражала на багатьох рівнях і мала справжній фактор вау. І все ж фільм чомусь пройшов на нагородах.

Величезний обсяг захоплюючих справжніх історій означав, що останні роки в кіно було більше представлених у кіно.

На великому екрані були представлені Стів Джобс, Алан Тьюрінг, Стівен Хокінг та багато інших. Звичайно, сценарій, кінематографія, художній відділ, звуковий дизайн та режисура відіграють важливу роль у успіху будь-якого фільму, але, по суті, справжня, необроблена історія в кінцевому рахунку є ключем до будь-якого кіно успіху. Якщо це нудна історія, вона, швидше за все, зробить нудний фільм або навіть не вийде на екрани. Можливо, спосіб виробництва фільмів та фінансова модель, що використовується зараз, означає, що виробники грають у безпеці, майже забезпечуючи комерційний успіх. Звідси продовження та подібні жанри фільмів, що створюються, з кількома помітними фільмами, які виділяються з маси. Було зроблено два фільми про Стіва Джобса, які за своєю суттю були дуже різними, тож саме так інтерпретується та створюється історія.

1.2 Огляд відомих реалізацій веб-системи кіномистецтва та перегляду фільмів

На сьогоднішній день здобули значної популярності сайти з кінофільмів – такі веб-додатки можуть включати в себе можливості підписки на них,

перегляду фільмів, публікації рецензій, формування тематичних статей та багато інших функцій.

Оглянемо найвідоміші сайти з кіномистецтва та окреслимо їх основні переваги й недоліки (табл.1.1).

Таблиця 1.1 – Порівняльний аналіз веб-систем з кіномистецтва та перегляду фільмів

Назва	Опис	Переваги	Недоліки
Kinopoisk.ru	Найвідоміший кіно-сайт серед країн колишнього СНГ. Доступний перегляд трейлерів до нових фільмів, читання та написання рецензій, наявні функції оцінювання фільмів. На сайті наявний власний топ-250 за версією користувачів	Найсвіжіші новини, може витримувати значну кількість користувачів одночасно, зручний інтерфейс.	Блокування сайту у декотрих країнах, що унеможлиблює роботу з ним (якщо не використовувати обхід блокування), занадто багато інформації, що не є дуже добре для підвищення юзабіліті.

	pecyocy.		
--	----------	--	--

Продовження таблиці 1.1

Назва	Опис	Переваги	Недоліки
http://snimifilm.com/	Даний сайт відомий своїми аналізами кіноіндустрії, анонсами різноманітних тематичних подій та кінофестивалів. Також наявні поради і статті про виробництво фільмів та інтерв'ю досвідчених режисерів.	Надзвичайно інформативний сайт, наявність анти-спам контролю, наявність зв'язку з соціальними мережами.	Погане колірне рішення та загальне оформлення сайту, занадто багато реклами, застарілі новини.
www.kinote.info	Анімаційний інтерактивний сайт про арт-кіно з новинами, інтерв'ю, оглядами та анонсами фестивалів.	Наявність календарю подій, багато інформації, що є рідкісною.	Дуже застарілий дизайн, погана швидкість завантаження сторінки.

Продовження таблиці 1.1

Назва	Опис	Переваги	Недоліки
kinomania.ru	Сайт, що має щоденну підтримку та розвиток. Тут є багато новин, фільмів, рецензій, трейлерів, тощо. На форумі є можливість спілкування з іншими кінолюбителями.	Вдала класифікація та гід сайту, непогане оформлення.	Досить посередній серед аналогів.
www.filmz.ru	Сайт фокусується на анонсах блокбастерів і артбастерів, драм і комедій, фільмів-катастроф та серіалів. На сайті наявні блоги, графіки виходу новинок, відгуки та кіноклуб.	Багато динамічних та інтерактивних елементів, швидке завантаження сторінки.	Інформаційне перенасичення.

Для детальнішого розуміння розглянемо інтерфейси аналогів заданих систем (рис. 1.1 – 1.5). На рисунку 1.1 зображено зовнішній вигляд головної сторінки веб-системи кіномистецтва Kinopoisk.ru. На рисунку 1.2 можна побачити одну із додаткових сторінок сайту під назвою «Головні героїні».

Слід зазначити, що інтерфейс від сайту є досить деталізованим, що займає багато часу на розбір його функціоналу і, відповідно, у користувача складається враження про відносно великий ступінь складності веб-ресурсу.

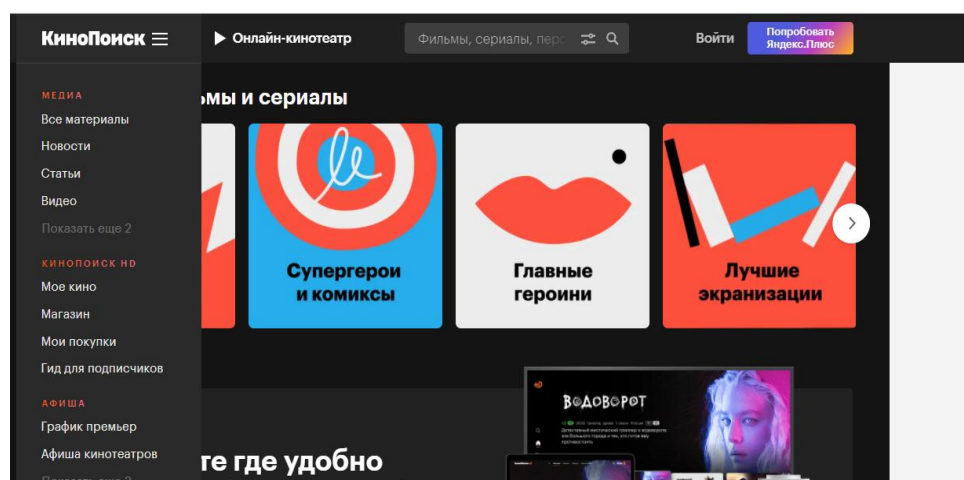


Рисунок 1.1 – Головна сторінка сайту Kinopoisk.ru

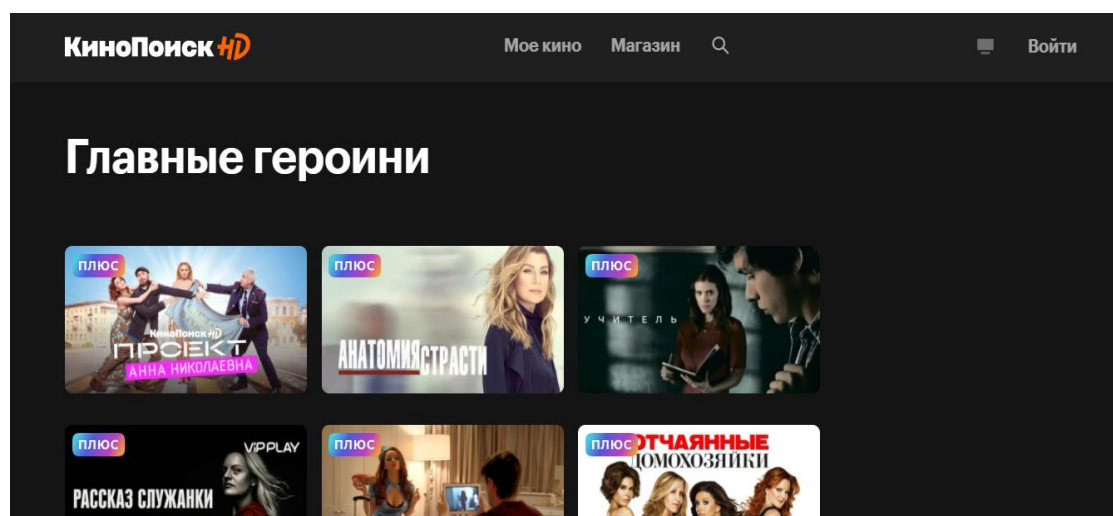


Рисунок 1.2 – Додаткова сторінка сайту під назвою «Головні героїні»

Сторінка авторизації в системі виглядає наступним чином, як на рисунку 1.3.

Також наявна і стандартна процедура входу через соціальні мережі, реєстрація і вхід за QR-кодом.

Авторизація на даному сайті є зручною і наочною, дозволяє використати

різноманітні опції для входу у мережу.

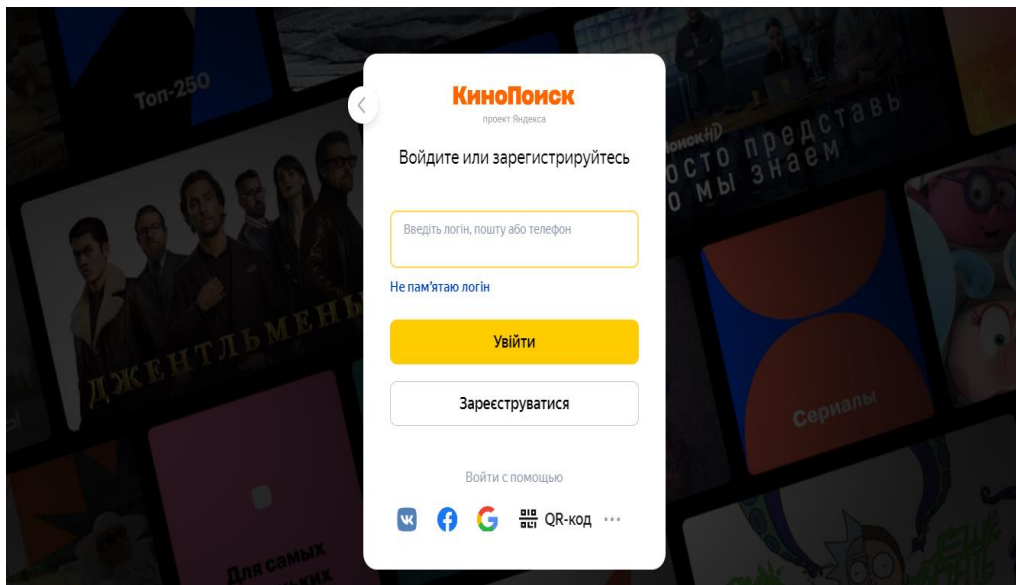


Рисунок 1.3 – Сторінка авторизації в Kinopoisk.ru

Наступними будуть скріншоти з сайту-аналогу веб-системи кіномистецтва та перегляду кінофільмів <http://snimifilm.com/>. На рисунках 1.4 та 1.5 показано зовнішній вигляд основної сторінки сайту, та одного з багатьох його розділів.

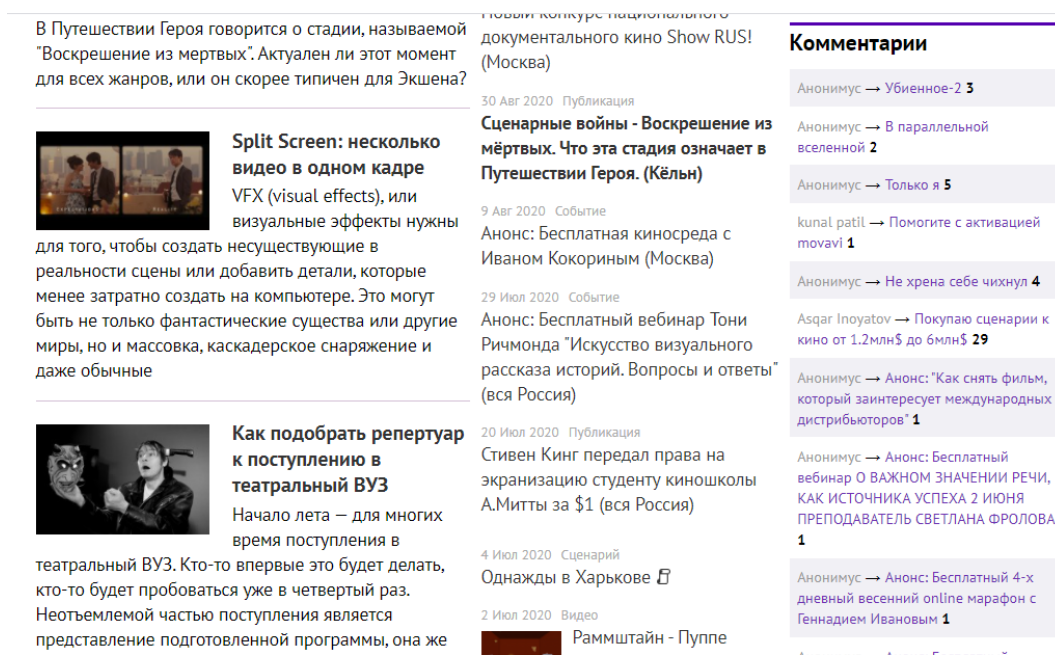


Рисунок 1.4 – Основна сторінка сайту <http://snimifilm.com>

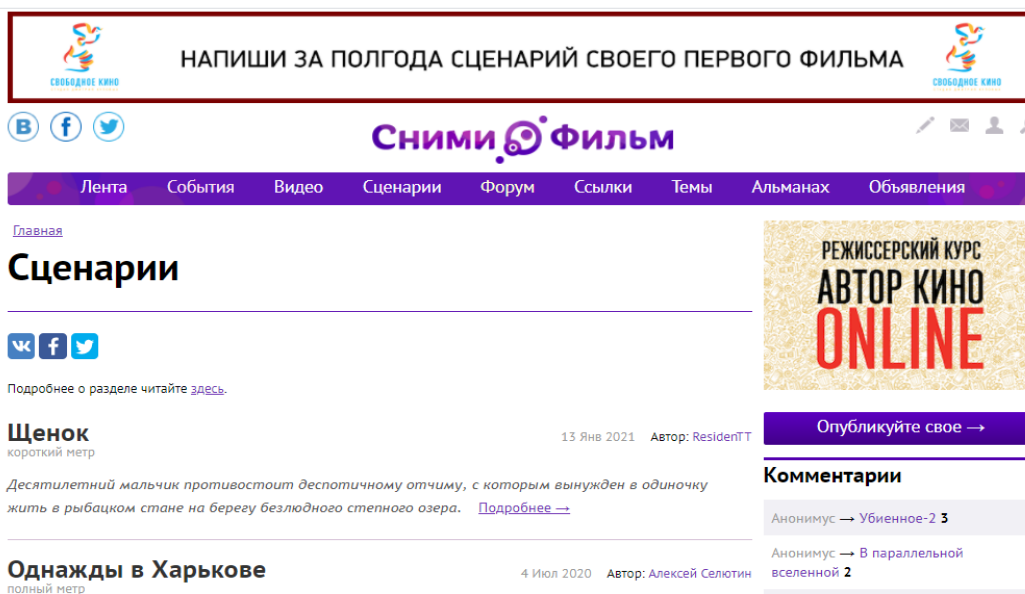


Рисунок 1.5 – Одна з додаткових сторінок сайту

Як бачимо, у другого аналога більш консервативне оформлення порівняно з першим, за рахунок чого він значно програє у функціональності.

1.3 Формулювання вимог на розробку ТЗ

Веб-система повинна відповідати таким вимогам:

1. Інтеграція з браузерами Google Chrome, Firefox, Microsoft Edge.
2. Інтеграція з ОС Windows, Linux.
3. Система повинна витримувати навантаження до 1000 осіб одночасно.
4. Інтерфейс повинен бути дружнім і зрозумілим.
5. Система виведення помилок наявна.
6. Система тестування гнучка та не містить багів.
7. Сервер захищено від втрати даних.
8. Можливість реєстрації.
9. Можливість вільного доступу до даних.

Висновки

У даному розділі розглянуто предметну область кіномистецтва та перегляду фільмів, що є популярними на сьогодні. Виявлено типові переваги та недоліки даних систем, розглянуто їх можливості, особливості, переваги, недоліки, на основі чого здійснено порівняльний аналіз. Сформовано ряд вимог на ТЗ, які дозволять уникнути неприємностей при розробці алгоритмів та впровадженні їх у програмне забезпечення.

РОЗДІЛ 2: АНАЛІЗ АРХІТЕКТУРНИХ РІШЕНЬ І ВИБІР ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ ВЕБ-СЕРВІСУ

В даному розділі проводиться опис роботи по розробці веб-сервісу для підтримки діяльності кінотеатру. Для якісної розробки будуть проведені наступні роботи:

- аналіз вихідних даних;
- розробка стратегії створення веб-застосунку;
- реалізація веб-застосунку;
- опис технологій та додатків для веб-застосунку;
- опис функціоналу;
- тестування веб-застосунку.

2.1 Аналіз методологій для побудови архітектурного рішення

Проектування веб-системи потребує оновлених підходів щодо алгоритмів. Тому перевагу в основному буде надано не стільки класичним алгоритмам, як UML-діаграмам, які показують розвиток системи в динаміці [21-26].

UML (англ. Unified Modeling Language) — уніфікована мова моделювання, що реалізується у парадигмі об'єктно-орієнтованого програмування(ООП). Також є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення, написання коду. UML є мовою широкого профілю, це відкритий стандарт, що використовує візуально-графічні позначення для створення абстрактного виду системи, яка називається UML-моделлю. UML був створений для визначення, проектування, візуалізації й документування в основному програмних систем. UML не є мовою програмування, але в засобах реалізації UML-моделей як інтерпретованого коду можливе застосування кодогенерації [4].

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (англ. association relationship);

- включення (англ. include relationship);
- розширення (англ. extend relationship);
- узагальнення (англ. generalization relationship) [5].

Варто зазначити, що загальні властивості варіантів використання можуть бути представлені у виді трьох різних способів, а точніше — за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації — це одне з фундаментальних понять у мові UML і у загальному, в певному об'ємі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (англ. include) у мові UML — це різновид відношення залежності між стандартним варіантом використання і його спеціальним випадком. До цього, відношенням залежності (англ. dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) призводить до зміни іншого елемента (залежного).

Відношення розширення (англ. extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого використовується базовим не завжди, а тільки при виконанні спеціальних додаткових установ [6].

Першою важливою діаграмою, яка допоможе визначити значну частину потрібних функцій, вбудованих в систему, є діаграма прецедентів. Діаграма прецедентів представлена графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи, яку можемо уявити як прямокутник,, асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами [7].

Сутність представленої діаграми полягає в такому: проєктована система передається у вигляді безлічі сутностей чи акторів, які взаємодіють певним чином із системою за допомогою ситуативних варіантів використання. Варіант використання (англ. use case) існує для описання послуг, які актору надаються системою. Якщо перефразувати, кожен випадок використання визначає певний набір дій, який реалізує система при взаємодії з актором. Те, яким чином буде

реалізована взаємодія акторів із системою при вищесказаному можливо не зазначати..

Наступним видом UML-діаграми, який часто використовують при проектуванні саме веб-орієнтованих додатків, є діаграма діяльності. Такі діаграми мають дуже багато схожих властивостей із звичайними алгоритмами, проте водночас вони є більш динамічними та підлаштованими до сучасних систем.

Діаграма діяльності (англ. activity diagram) — в UML, графічний вид графу діяльностей. Граф діяльностей це різновид графу станів скінченного автомату, вершинами якого є деякі обумовлені дії, а переходи відбуваються по завершенню певних дій [8].

Дія (англ. action) є однією з фундаментальних одиниць означення поведінки в специфікації. Дія аналізує отриману множину вхідних сигналів, та конвертує їх на множину вихідних сигналів та даних. Одна, або обидві із цих множин водночас, можуть бути порожніми. Реалізація дії відповідає виконанню певної дії. Схоже до цього, реалізація діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може реалізуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії потрібні аналізувати отримані дані, перетворювати їх та проводити над ними тестування, певні дії можуть вимагати деякої послідовності. Специфікація діяльності (на вищих рівнях сумісності) дозволяє реалізацію декількох (логічних) потоків, та існування певних механізмів синхронізації для забезпечення правильності виконання дій у відповідно правильному порядку[8].

Діаграми активностей створюються з певної обмеженої кількості фігур, що з'єднані стрілочками.

Найважливіші типи фігур:

- скруглені прямокутники мають означення дії;
- ромби мають означення рішення;

- риси мають означення початку (розподілу) чи кінця (об'єднання) паралельних активностей;
- чорний зафарбований круг має означення старту (початкового стану) процесу;
- чорний зафарбований круг в колі має означення кінця (кінцевого стану).

2.2 Реалізація структури програмного додатку

Перш за все, необхідно звернути увагу на загальну структурну схему системи, яку буде відтворено на рисунку 2.1. На ній видно усі основні блоки, на які поділений сайт.

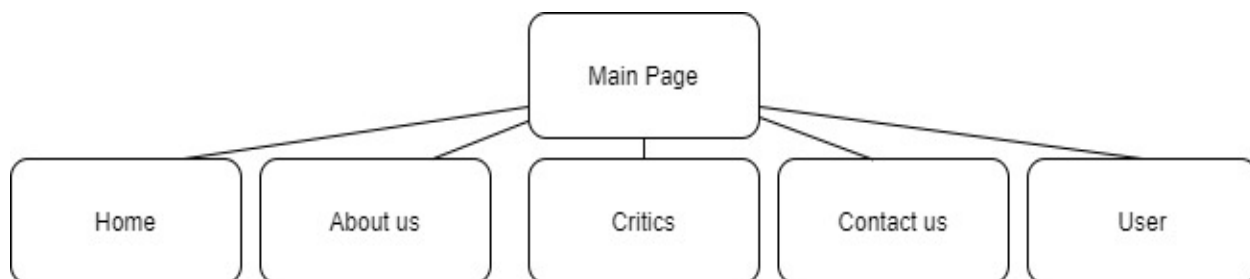


Рисунок 2.1 – Структурна схема веб-системи

На рисунку 2.2 зображена діаграма прецедентів для веб-застосунку кіномистецтва. Акторами є юзер, рецензент та адмін, тобто ролі всього три, хоча в подальшому планується їх розширення. Як заведено завжди, адмін має найбільше повноважень з усіх [18], [19], [20].

Аналіз предметної області і проектування є першими етапами в життєвому циклі створення програмного рішення. Одним з результатів цього етапу є діаграма варіантів використання (Use Case), що описує основні групи користувачів системи і варіанти її використання.

Предметна область – це частина реального світу, дані та особливості якої будуть відображені в розробляється програмному рішенні. Наприклад, в якості предметної області можна вибрати бухгалтерію будь-якого підприємства, відділ кадрів, банк, магазин і т. д. Предметна область нескінченна і містить як важливі

поняття і дані, так і малозначні або взагалі нічого не значущі дані. Так, якщо в якості предметної області вибрати облік товарів на складі, то поняття «накладна» і «рахунок-фактура» є важливими, а то, що співробітниця, приймаюча накладні, має двох дітей - це для обліку товарів неважливо. Однак з точки зору відділу кадрів дані про наявність дітей є важливими. Таким чином, значимість даних залежить від вибору предметної області [9].

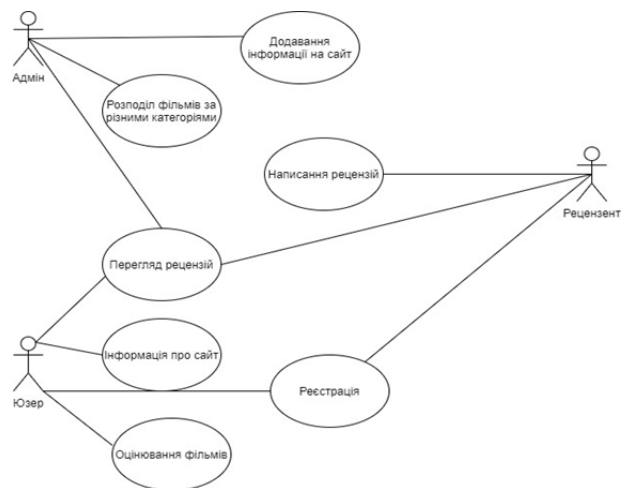
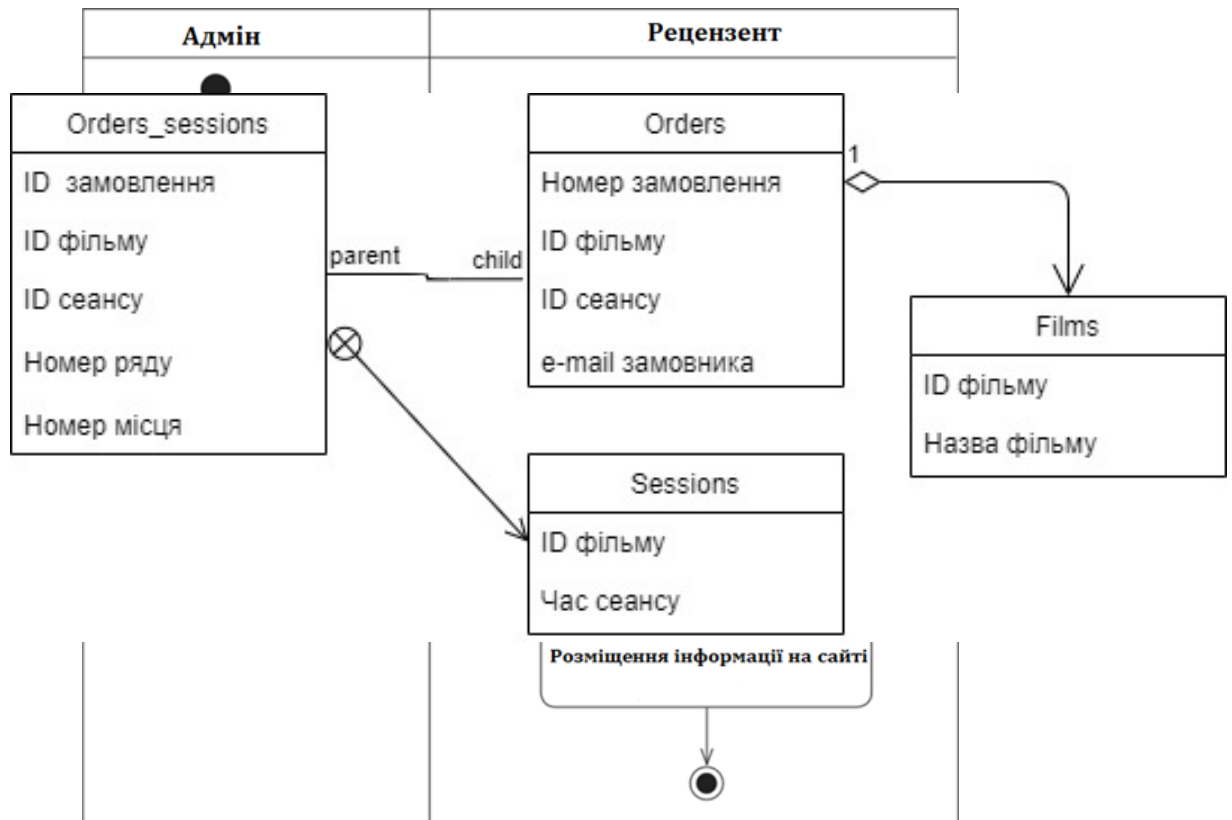


Рисунок 2.2 – Діаграма прецедентів веб-системи з кіномистецтва

Діаграма діяльності показує послідовність дій у випадку адміна та рецензента по відношенню до веб-системи кіномистецтва. Діаграма має багато

спільного зі звичайними алгоритмами, проте є більш динамічним, зручним і сучасним інструментом. Її зображено на рисунку 2.3.

Рисунок 2.3 – Діаграма діяльності веб-системи з кіномистецтва



На рисунку 2.4 і 2.5 зображені діаграма класів та діаграма компонентів.

Рисунок 2.4 – Діаграма класів

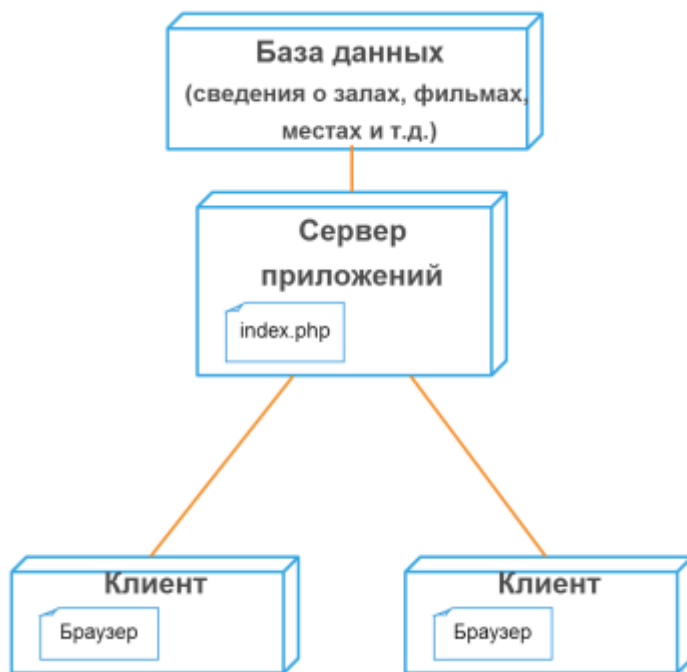


Рисунок 2.5 – Діаграма компонентів

Діаграма компонентів має досить загальний вигляд, оскільки орієнтована на веб-сервіси та їх універсальну клієнт-серверну технологію.

Взаємодія з базою даних реалізується в системі наступним чином. При вході на сторінку кінофільму з бази завантажуються доступні сеанси. Сеанси генеруються у вигляді кнопок з посиланнями на замовлення білетів.

При натисненні на сеанс генерується імітація залу з вибором доступних місць з бази на даний сеанс. Після замовлення в базі відмічається номер замовлення та обрані місця. При наступному замовленні на цей сеанс замовлені раніше місця вже недоступні [13],[14].

2.3 Обґрунтування середовища і мови програмування

HTML (англ. HyperText Markup Language — мова розмітки гіпертексту) є мовою тегів, завдяки якій реалізують розмітку веб-сторінок у глобальній мережі Інтернет. Реалізується це наступним чином: спочатку запускається процес отримання веб-браузерами HTML-документів з локальної пам'яті і подальша

трансформація певних файлів у мультимедіа. HTML описує структуру веб-сторінки семантичним способом і спочатку включені сигнали для зовнішнього вигляду потрібного документа.

Елементи HTML є значущими структурними блоками сторінок HTML. За допомогою конструкцій HTML, зображення та інші об'єкти, для прикладу інтерактивні і динамічні елементи, можна побудувати в окрему сторінку. HTML має ефективні на практиці засоби і методи потрібної структуризації інформації, позначаючи структурну семантику тексту, наприклад заголовки, абзаци, списки, посилання, цитати та інші елементи. Елементи HTML визначені тегами, написаними з використанням кутових дужок [10].

HTML може впроваджувати та інтегрувати сценарії та скрипти, для прикладу JavaScript, що має вплив на поведінку та вміст веб-сторінок. Включення CSS задає певний вигляд і компоновання вмісту. World Wide Web Consortium (W3C), який супроводжує стандарти HTML та CSS, заохочує реалізацію на CSS в порівнянні з явним презентаційним HTML з 1997 року. [11]

HTML реалізує засоби для:

- створення структурованого документа методом визначення структурного складу тексту: заголовки, абзаци, списки, таблиці, цитати та інше;
- отримання інформації із мережі інтернет, використовуючи гіперпосилання;
- створення інтерактивних форм;
- включення зображень, відео, аудіо доріжок, та інших медіа об'єктів до тексту.

За результатами опитування на сайті <https://dou.ua/>, станом на 2019 рік має місце наступна діаграма розподілу мов програмування за популярністю (рисунок 2.6):

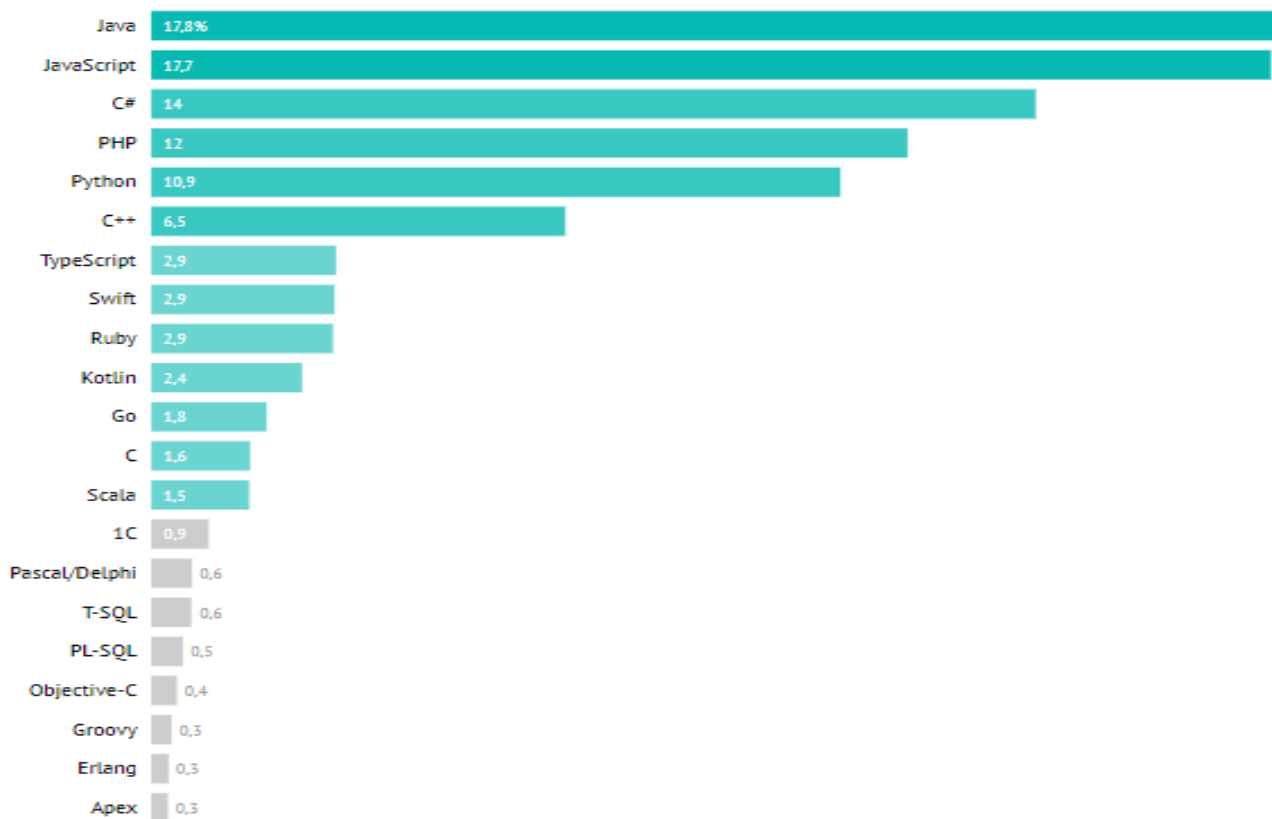


Рисунок 2.6 – Діаграма розподілу мов програмування станом на 2019 рік

JavaScript посідає одну із провідних позицій, а у своїй ніші програмування веб-додатків і взагалі випереджає всі інші мови. Тому було вирішено обрати саме дану мову для розробки веб-системи [12].

JavaScript® (часто просто JS) – інтерпретована або JIT-компільована, об'єктно-орієнтована мова з функціями першого класу. Найбільш широке застосування знаходить як мова сценаріїв веб-сторінок, але також використовується і в інших програмних продуктах, наприклад, node.js або Apache CouchDB. JavaScript – це прототипно-орієнтована, мультипарадигмальна мова з динамічною типізацією, що підтримує об'єктно-орієнтований, імперативний і декларативний (наприклад, функціональне програмування) стилі програмування.

Розглянемо ряд редакторів для написання коду Javascript та оберемо з них кращий [15], [16], [17].

Таблиця 2.1 – Порівняльний аналіз редакторів для написання коду Javascript

Назва	Переваги	Недоліки
Komodo Edit	Мультимовність, спливаючі підказки, автозавершення коду, відстежування змін.	Включає лише базові функції роботи з кодом.
Bluefish	Перевірка орфографії, автозавершення коду, автозбереження.	Не підходить для дизайну сайтів.
Vim	Нескінченна історія відміни, велика кількість плагінів, потужний пошук і заміна, інтеграція із сторонніми інструментами.	Не підходить для редагування коду.
Visual Studio Code	Містить вбудований інтелектуальний модуль, відладка коду безпосередньо з редактора, запуск додатка для відладки і приєднання до запусчених додатків	Не помічено.

Отже, було вирішено обрати для розробки редактор Visual Studio Code – легкий, але потужний редактор початкового коду. У первинній конфігурації використовується для редагування коду на JavaScript, TypeScript і Node.JS, а за допомогою розширень підтримує C++, C#, Python і PHP.

Висновки

У даному розділі розглянуто архітектурні рішення для веб-сервісу за допомогою методології проектування UML, розроблено відповідні діаграми. Обґрунтовано середовище та мову програмування для розробки продукту.

РОЗДІЛ 3: ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СЕРВІСУ ПІДТРИМКИ ДІЯЛЬНОСТІ КІНОТЕАТРУ

3.1 Алгоритм роботи системи веб-сервісу підтримки діяльності кінотеатру

Для проектування якісного програмного додатку, перш за все, слід правильно розробити і подати алгоритм його роботи. Веб-додатки тут не є винятком.

Алгоритм (латинізов. Algorithmi за араб. ім'ям перського математика аль-Хорезмі) — набір вказівників, що будуть описувати покроковий порядок дій для досягнення потрібного результату за умови, що кількість дій скінченна; система правил виконання дискретного процесу, для якої характерне прагнення до досягнення мети за певну кількість часу. Для візуалізації алгоритмів є відомий засіб і це блок-схеми.

Для комп'ютерних програм алгоритм є списком деталізованих інструкцій, що реалізують процес обчислення і від початку до кінця, відбувається певна послідовність логічних змін. При тому, переходи зі стану до стану можуть бути недетермінованими і випадковими.

Поняття алгоритму має математичні корні. Обчислювальні процеси алгоритмічного характеру (наприклад, прості дії над цілими числами, решето Ератосфена) відомі вже багато тисяч років і знайшли своє практичне значення. Термін алгоритму знайшов свою трактування ж аж на початку ХХ століття.

В методології алгоритм можна пояснити як повністю аксіоматичний термін. З методології виходить абсолютно нове визначення алгоритму як оптимального методу з наближенням до прогнозованого абсолюту. Зробивши все в послідовності алгоритму за певних обмежень і граничних умов задачі, маємо цілком ідеальне рішення нагальних проблем науково-практичного характеру певної галузі. На сьогодні алгоритм будь-якої діяльності у

формалізованому вигляді є основою навчання на прикладах за подібністю. На основі подібності алгоритмів у різних галузях була сформована концепція (теорія) експертних систем.

На рисунку 3.1 зображено алгоритм функціонування веб-сервісу підтримки діяльності кінотеатру.



Рисунок 3.1 – Алгоритм функціонування веб-сервісу підтримки діяльності кінотеатру

3.2 Обґрунтування середовища і мови програмування

Розглянемо ряд редакторів для написання коду Javascript та оберемо з них кращий.

Назва	Переваги	Недоліки
Komodo Edit	Мультимовність, спливаючі підказки, автозавершення коду, відстежування змін.	Включає лише базові функції роботи з кодом.
Bluefish	Перевірка орфографії, автозавершення коду, автозбереження.	Не підходить для дизайну сайтів.
Vim	Нескінченна історія відміни, велика кількість плагінів, потужний пошук, заміна, інтеграція зі сторонніми ресурсами.	Не підходить для редагування коду.
Notepad	Простий для розуміння, займає мало місця, відладка коду власне з редактора, запуск додатка для зневадження і приєднання до запусчених додатків.	Не помічено.

Таблиця 3.1 – Порівняльний аналіз редакторів для написання коду Javascript

Отже, було вирішено обрати для розробки редактор Notepad – легкий, але потужний редактор для створення початкового коду. У первинній конфігурації дуже поширений для редагування та корекції коду на JavaScript, TypeScript і Node.JS, декотрі його розширення підтримують C++, C#, Python і PHP.

3.3 Програмування веб-сервісу підтримки діяльності кінотеатру

На рисунках 3.2 – 3.9 зображено процес розробки у середовищі Notepad веб-сервісу системи підтримки діяльності кінотеатру.

Проектування програмного продукту здійснювалось наступним чином: спершу було розроблено логічний каркас і структуру веб-сервісу, а уже потім пов'язано графічний інтерфейс користувача із базою даних.

На початковому етапі важливо було створити в каталозі чітко структурований набір сторінок сайту. Усі сторінки необхідно було чітко продумати, перш за все грамотно оформити їх зовнішній вигляд, зробити їх коректними відповідно до сучасних стандартів, а саме: додати метатеги, визначити кодування сторінки, пов'язати файл зі сторінками стилів (css, bootstrap), провести структурування сторінки і виявити розділи, відповідальні за інформаційне наповнення.

```

1 <!doctype html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <!-- Bootstrap CSS -->
9 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCW98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERd"
10 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
11 <link rel="stylesheet" href="Styleheaven.css" />
12
13 <title>MoviesHeaven - Toy Story</title>
14 </head>
15 <body>
16 <!-- Navbar -->
17 <div class="container">
18 <div id="accueil">
19 <div class="sticky-top">
20 <nav class="navbar navbar-expand-lg navbar-light bg-light">
21 <a class="navbar-brand disabled" href="index.html"> Movies Heaven</a>
22 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle nav">
23 <span class="navbar-toggler-icon"></span>
24 </button>
25 <div class="collapse navbar-collapse" id="navbarNav">
26 <ul class="navbar-nav">
27 <li class="nav-item active">
28 <a class="nav-link" href="index.html">HOME</a>
29 </li>
30 <li class="nav-item">
31 <a class="nav-link" href="#about">ABOUT US</a>
32 </li>
33 <li class="nav-item">
34 <a class="nav-link" href="critics.html">CRITICS</a>
35 </li>
36 <li class="nav-item">
37 <a class="nav-link" href="contactus.html">CONTACT US</a>
38 </li>
39 </ul>

```

Рисунок 3.2 – Процес розробки головної сторінки сайту

Рисунок 3.3 ілюструє поглиблену розробку сайту, включно зі створенням слайдеру та його окремих блоків. Слайдер є одним з найважливіших і найпопулярніших елементів на різноманітних веб-сервісах з декількох причин. По-перше, він візуально привабливий для користувача, а отже, стимулює інтерес цільової аудиторії та мотивує вивчати матеріали сайту. По-друге, він дозволяє заощадити місце, адже в одному блоці слайдера можна демонструвати відразу декілька різних анонсів, які інакше довелося б розміщувати окремо, що призвело б до втрати місця на сторінці.

Функціонал слайдерів для сайтів містить різні опції: найпростіші слайдери можуть працювати лише у послідовному режимі, складніші слайдери дають можливість «перегорнути» сторінку вручну, обрати чи перейти на матеріал або розгорнути повнорозмірне зображення тощо. Але загальний алгоритм роботи будь-якого слайдера приблизно однаковий.

У функціях, відповідальних за роботу слайдера, часто вказують швидкість перегортання слайдів, задають умови показу попереднього/наступного слайда або слайда за номером, задають візуальне оформлення слайдів, встановлюють додаткові ефекти перегортання. При цьому за «виведення» слайдів у потрібному місці сторінки відповідальний HTML-код, за зовнішній вигляд і дизайн слайдера — CSS, а безпосередньо за роботу слайдера — сценарій JavaScript, особливо, функції бібліотеки jQuery.

```

61 </div>
62 <div class="carousel-item">
63 <a href="TOY-STORY.html"> </a>
64 </div>
65 <div class="carousel-item">
66 <a href="ICE-AGE.html"></a>
67 </div>
68 <div class="carousel-item">
69 <a href="INTERSTELAR.html"></a>
70 </div>
71 </div>
72 <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">
73 <span class="carousel-control-prev-icon" aria-hidden="true"></span>
74 <span class="sr-only">Previous</span>
75 </a>
76 <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">
77 <span class="carousel-control-next-icon" aria-hidden="true"></span>
78 <span class="sr-only">Next</span>
79 </a>
80 </div>
81 <div id="about">
82 <h1 class="principal">ABOUT US</h1>
83 <p class="description">Film, also called movie, motion picture or moving picture, is a visual art used to simulate experiences that communicate ideas, stories, percept
84 The moving images of a film are created by photographing actual scenes with a motion-picture camera, by photographing drawings or miniature models using traditional animation techniqu
85 Traditionally, films were recorded onto celluloid film stock through a photochemical process and then shown through a movie projector onto a large screen. Contemporary films are often
86 Films are cultural artifacts created by specific cultures. They reflect those cultures, and, in turn, affect them. Film is considered to be an important art form, a source of popular
87 The individual images that make up a film are called frames. In the projection of traditional celluloid films, a rotating shutter causes intervals of darkness as each frame, in turn,
88 The name "film" originates from the fact that photographic film (also called film stock) has historically been the medium for recording and displaying motion pictures. Many other term
89 </div>
90 </div>
91 <!-- Pied de page-->
92 <footer id="baadepage">
93 <ul>

```

Рисунок 3.3 – Процес розробки головної сторінки сайту (продовження)

Рисунки 3.4-3.5 ілюструють процес розробки фронтенду сторінки кінокритики. Так, як і при розробці попередньої сторінки, застосовується тег `div`, що слугує головним інструментом для створення блокової верстки веб-сервісу.

Перевагами блокової верстки є

- кешування блоків браузером, задля швидкого завантаження сторінок;
- необмежені можливості для створення дизайну різної складності;
- чітке відділення контенту від робочої частини;
- простота реалізації програмного продукту.

Недоліком блокової верстки відносно табличної верстки є складність у розробці та реалізації для простих сайтів, також на старих браузерах може відмовляти кросбраузерність, але при цьому блокова верстка відмінно індексується пошуковиками та багато програм правильно читають контент сайту, що допомагає користувачам швидко завантажувати сторінки, а також за допомогою поєднання блоків і CSS можливо створювати авторські професійні дизайни.

Популярним також є використання тегу `a href` для навігації на сайті при переходах на додаткові сторінки – де містяться рецензії користувачів на кінороботи. Таких сторінок може бути нескінченна кількість – вони будуть відібрані модератором та додані впродовж цілого життя веб-сервісу.

Сторінка критики структурно є нескладною, проте блокова верстка потребує ретельної та докладної розробки для укомплектування професійного та сучасного дизайну.

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <!-- Bootstrap CSS -->
9 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCW9/SFmGE8FJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkPFJwJ8ERd
10 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
11 <link rel="stylesheet" href="Styleheaven.css" />
12
13 <title>MoviesHeaven - Critics</title>
14 </head>
15 <body>
16 <!-- Navbar -->
17 <div class="container">
18 <div class="sticky-top">
19 <nav class="navbar navbar-expand-lg navbar-light bg-light">
20 <a class="navbar-brand disabled" href="index.html"> Movies Heaven</a>
21 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label
22 <span class="navbar-toggler-icon"></span>
23 </button>
24 <div class="collapse navbar-collapse" id="navbarNav">
25 <ul class="navbar-nav">
26 <li class="nav-item active">
27 <a class="nav-link" href="index.html">HOME</a>
28 </li>
29 <li class="nav-item">
30 <a class="nav-link" href="index.html#about">ABOUT US</a>
31 </li>
32 <li class="nav-item">
33 <a class="nav-link" href="critics.html">CRITICS</a>
34 </li>
35 <li class="nav-item">
36 <a class="nav-link" href="contactus.html">CONTACT US</a>
37 </li>
38 </ul>
39 </div>
</div>
</div>
</body>
</html>
```

Рисунок 3.4 – Розробка сторінки сайту Критика

```
85 <div class="row">
86 <p class="col-6 auteur">by Audrey | 25/08/2018 </p>
87 <span class="col-6 etoilles"><i class="glyphicon glyphicon-star color"></i><i class="glyphicon glyphicon-star color"></i><i class="gl
88 </span>
89 </div>
90 <h1 class="card-title">Toy Story</h1>
91
92 <p class="card-text">Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>
93 <a href="TOY-STORY.html" class="seecritic">See the critic <i class="glyphicon glyphicon-menu-right"></i></a>
94
95 </div>
96 </div>
97 </div>
98
99 <div class="col-sm-12 col-md-6 col-lg-3">
100 <div class="card">
101 <div class="card-header">
102 <a href="ICE-AGE.html">
104 <div class="card-body">
105 <div class="row">
106 <p class="col-6 auteur">by Carolina | 25/08/2018 </p>
107 <span class="col-6 etoilles"><i class="glyphicon glyphicon-star color"></i><i class="glyphicon glyphicon-star color"></i><i class
108 </span>
109 </div>
110 <h1 class="card-title">Ice Age</h1>
111
112 <p class="card-text">Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p>
113 <a href="ICE-AGE.html" class="seecritic">See the critic <i class="glyphicon glyphicon-menu-right"></i></a>
114
115 </div>
116 </div>
117 </div>
118
119 <div class="col-sm-12 col-md-6 col-lg-3">
120 <div class="card">
121 <div class="card-header">
122 <a href="INTERSTELIAR.html"></a>
123 </div>
```

Рисунок 3.5 – Розробка сторінки сайту Критика (продовження)

На рисунках 3.6-3.8 фрагментарно наведені приклади процесу створення сторінок Сконтактувати, Створити акаунт та Скинути пароль.

Під час розробки даних сторінок використовується багато схожих принципів веб-програмування. Задіяні попередні методики, але тут додається поняття форм.

Форма HTML є документом, що створений з використанням елементів мови розмітки HTML. Основна функція – це збирання даних від користувачів. Після заповнення користувачем форми і запуску процесу її обробки, інформація з неї потрапляє в програму, обробляючись на сервері. Додаток під назвою Common Gateway Interface (CGI) обробляє її. Саме так користувачі взаємодіють з Web через Internet.

Окрім того, додається поняття активних елементів HTML, найчастіше це поля для введення даних, кнопки, випадаючі списки та діалогові вікна.

Під час розробки форм важливо потурбуватись як і про бекенд, так і про фронтенд веб-сервісу: інформація від користувачів повинна надсилатись на сервер і коректно зчитуватись, поля можуть допускати чи не допускати певні типи даних, водночас, сторінка з формою повинна мати дружній та неускладнений для користувача вигляд, щоб у нього було бажання взаємодіяти з веб-ресурсом і авторизуватись на сервері.

Елемент <INPUT> є важливим для визначення області у центрі форми, де збираються дані. Елемент є активним полем для взаємодії користувача з системою (зазвичай один рядок тексту). Необхідна наявність атрибуту NAME для визначення найменування змінної поля.

```

45 </nav>
46 </div>
47 </div>
48 <!-- space entre nav et ... -->
49
50 <section id="contact">
51 <div id="c.us">
52 <h1>Contact Us</h1>
53 </div>
54 <div id="contenu" >
55 <form>
56
57 <div class="form-row">
58 <div class="form-group col-md-6 offset-md-3"><br/>
59 <label for="inputEmail4">Email</label>
60 <input type="email" class="form-control" id="inputEmail4" placeholder="Email"><br/>
61 </div>
62 <div class="form-group col-md-6 offset-md-3"><br/>
63 <label for="inputName">Name</label>
64 <input type="text" class="form-control" id="inputName" placeholder="Name">
65 </div>
66 </div>
67 <div class="form-row">
68 <div class="form-group col-md-6 offset-md-3">
69 <label for="inputCity">City</label>
70 <input type="text" class="form-control" id="inputCity">
71 <br/>
72 </div>
73 </div>
74 <div class="form-row" id="prout">
75 <div class="form-group col-md-6 offset-md-3">
76 <label for="inputMessage">Your message</label>
77 <input type="text" class="form-control" id="inputCity">
78 </div>
79 </div>
80 <div class="row" id="sub">
81 <button class="btn btn-primary" type="submit">Send</button>
82 </div>
83 </form>
</div>

```

Hyper Text Markup Language file length: 5575 lines: 101 Ln: 10 Col: 105 Sel: 0 | 0 Unix (LF) UTF-8 INS

Рисунок 3.6 – Розробка сторінки сайту Зконтактувати

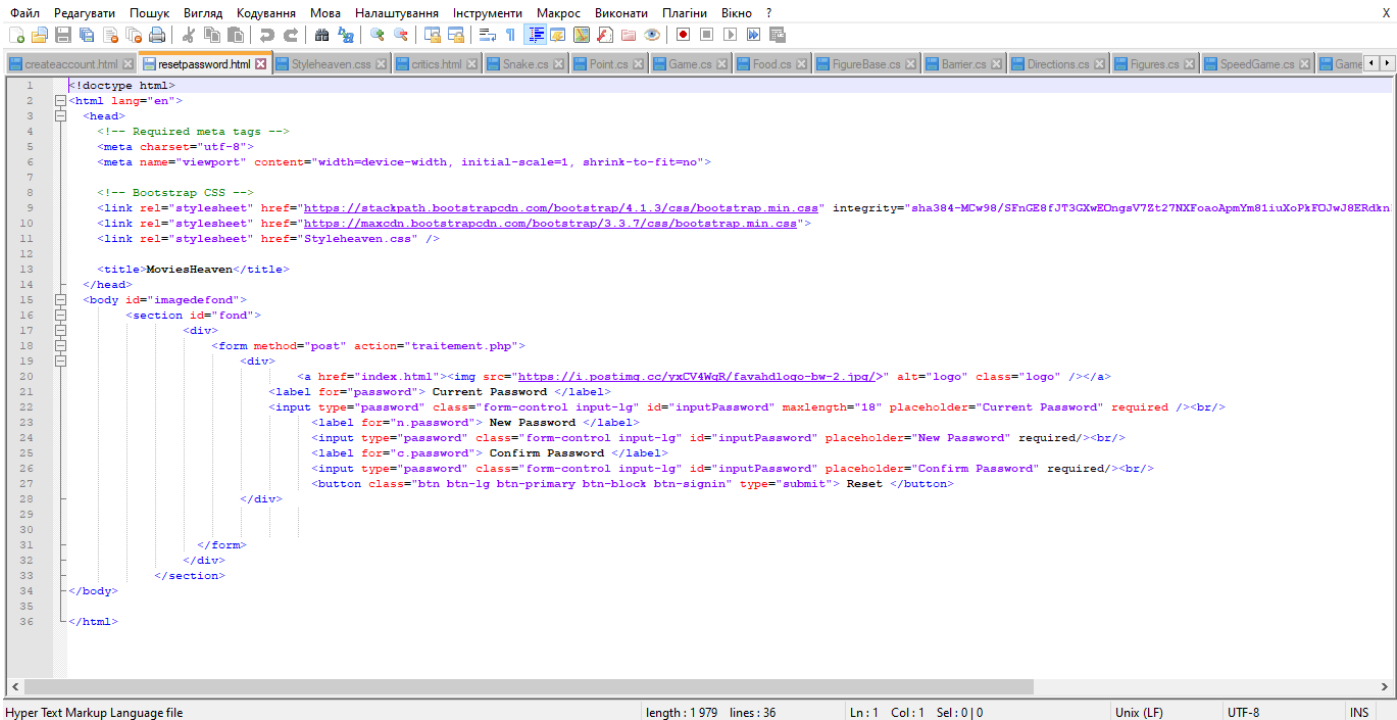
```

29 <label class="radio-inline">
30 <input type="radio" name="sex" id="inlineCheckbox2" value="female" />
31 Female
32 </label><br/>
33 <br/><label for="">
34 Birth Date</label>
35 <div class="row">
36 <div class="col-xs-4 col-md-4">
37 <select class="form-control">
38 <option value="Month">Month</option>
39 </select>
40 </div>
41 <div class="col-xs-4 col-md-4">
42 <select class="form-control">
43 <option value="Day">Day</option>
44 </select>
45 </div>
46 <div class="col-xs-4 col-md-4">
47 <select class="form-control">
48 <option value="Year">Year</option>
49 </select>
50 </div>
51 </div><br/>
52 <br><label for="Email"> Email Address </label>
53 <input type="email" name="Email" class="form-control input-lg" id="inputEmail" maxlength="18" placeholder="Email address" required /><br/>
54 <label for="username"> Username </label>
55 <input type="text" name="Username" class="form-control input-lg" id="input1.name" maxlength="18" placeholder="ID" required /><br/>
56 <label for="password"> Password</label>
57 <input type="password" class="form-control input-lg" id="inputPassword" placeholder="Password" required/><br/>
58 <label for="c_password"> Confirm Password</label>
59 <input type="password" class="form-control input-lg" id="inputPassword" placeholder="Password" required/><br/>
60 <div id="remember" class="checkbox">
61 <input type="checkbox" name="terms"> I agree with the <a href="#"> Terms and Conditions</a>.</div>
62 </div>
63 <button class="btn btn-lg btn-primary btn-block btn-signin" type="submit"> Sign up </button>
64 </div>
65 </form>
66 </div>
67 </section>

```

Hyper Text Markup Language file length: 4323 lines: 73 Ln: 15 Col: 26 Sel: 0 | 0 Unix (LF) UTF-8 INS

Рисунок 3.7 – Розробка сторінки сайту Створити акаунт



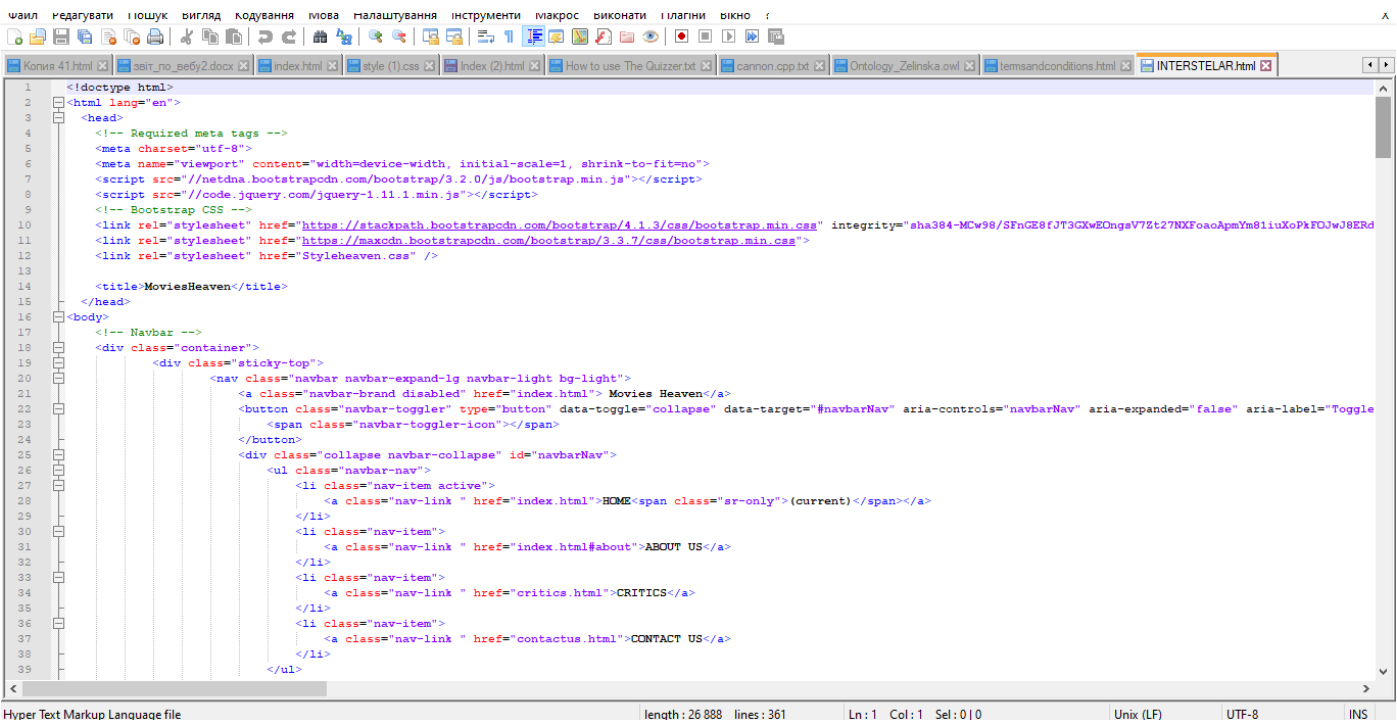
```

1 <!doctype html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7
8 <!-- Bootstrap CSS -->
9 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCW98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaAoApm81iuXoPkFOJwJ8ERdIn"
10 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
11 <link rel="stylesheet" href="Styleheaven.css" />
12
13 <title>MoviesHeaven</title>
14 </head>
15 <body id="imagedefond">
16 <section id="fond">
17 <div>
18 <form method="post" action="traitement.php">
19 <div>
20 <a href="index.html"></a>
21 <label for="password"> Current Password </label>
22 <input type="password" class="form-control input-lg" id="inputPassword" maxlength="18" placeholder="Current Password" required /><br/>
23 <label for="n.password"> New Pasword </label>
24 <input type="password" class="form-control input-lg" id="inputPassword" placeholder="New Password" required/><br/>
25 <label for="c.password"> Confirm Password </label>
26 <input type="password" class="form-control input-lg" id="inputPassword" placeholder="Confirm Password" required/><br/>
27 <button class="btn btn-lg btn-primary btn-block btn-signin" type="submit"> Reset </button>
28 </div>
29 </form>
30 </div>
31 </section>
32 </body>
33 </html>

```

Hyper Text Markup Language file length: 1979 lines: 36 Ln: 1 Col: 1 Sel: 0|0 Unix (LF) UTF-8 INS

Рисунок 3.8 – Розробка сторінки сайту Скинути пароль



```

1 <!doctype html>
2 <html lang="en">
3 <head>
4 <!-- Required meta tags -->
5 <meta charset="utf-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
7 <script src="https://netdna.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js"></script>
8 <script src="https://code.jquery.com/jquery-1.11.1.min.js"></script>
9 <!-- Bootstrap CSS -->
10 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCW98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaAoApm81iuXoPkFOJwJ8ERdIn"
11 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
12 <link rel="stylesheet" href="Styleheaven.css" />
13
14 <title>MoviesHeaven</title>
15 </head>
16 <body>
17 <!-- Navbar -->
18 <div class="container">
19 <div class="sticky-top">
20 <nav class="navbar navbar-expand-lg navbar-light bg-light">
21 <a class="navbar-brand disabled" href="index.html"> Movies Heaven</a>
22 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle"
23 <span class="navbar-toggler-icon"></span>
24 </button>
25 <div class="collapse navbar-collapse" id="navbarNav">
26 <ul class="navbar-nav">
27 <li class="nav-item active">
28 <a class="nav-link" href="index.html">HOME<span class="sr-only"> (current)</span></a>
29 </li>
30 <li class="nav-item">
31 <a class="nav-link" href="index.html#about">ABOUT US</a>
32 </li>
33 <li class="nav-item">
34 <a class="nav-link" href="critics.html">CRITICS</a>
35 </li>
36 <li class="nav-item">
37 <a class="nav-link" href="contactus.html">CONTACT US</a>
38 </li>
39 </ul>

```

Hyper Text Markup Language file length: 26 888 lines: 361 Ln: 1 Col: 1 Sel: 0|0 Unix (LF) UTF-8 INS

Рисунок 3.9 – Розробка сторінки критики на прикладі фільму

Інтерстеллар

На рисунках 3.10-3.18 показані основні розроблені на даній мові вікна сайту і бази даних.

Спочатку при запуску html-файлу на екрані з'явиться головна сторінка сайту (рис.3.10).

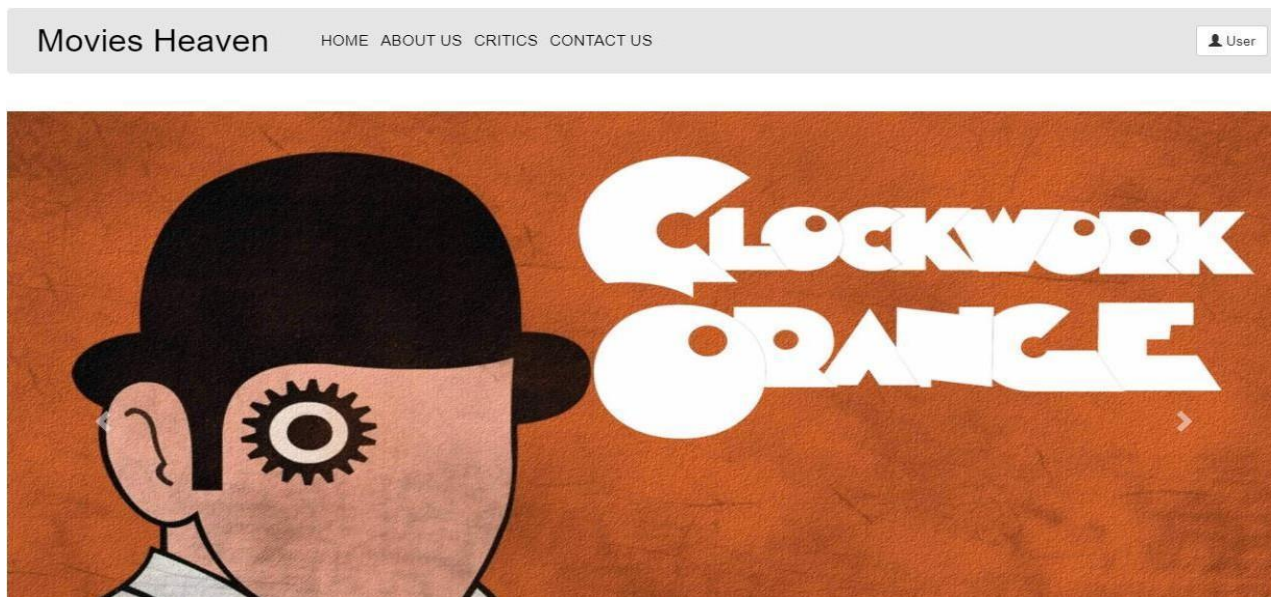


Рисунок 3.10 – Головна сторінка сайту кіномистецтва

При натисненні на кнопку User в правому верхньому куті буде здійснено перехід на сторінку реєстрації чи входу (рис. 3.11, 3.12).

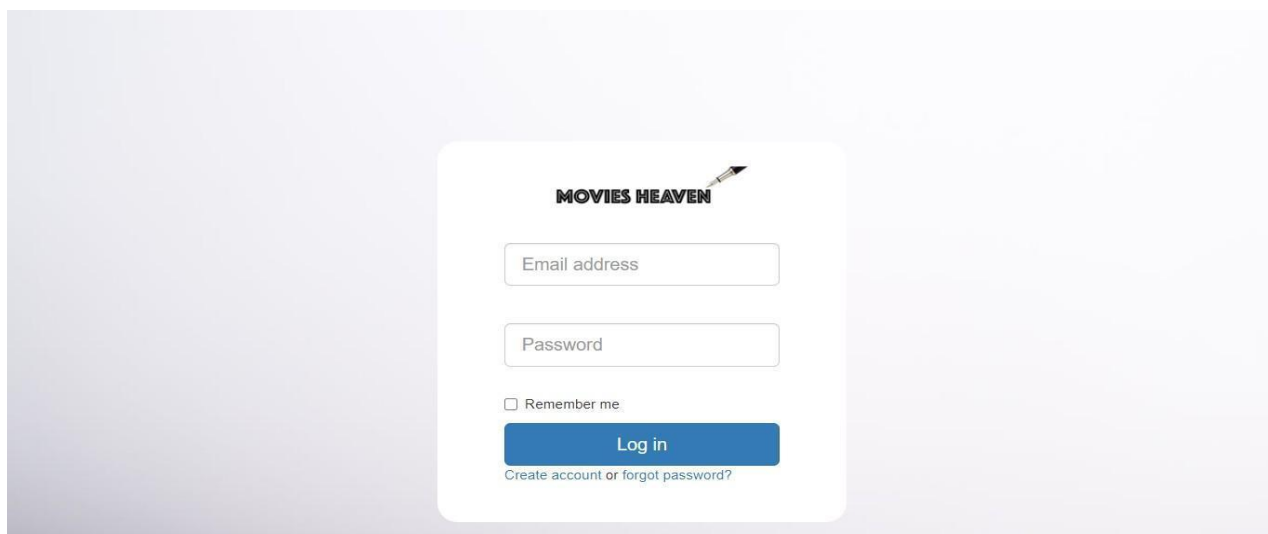


Рисунок 3.11 – Сторінка реєстрації на сайті кіномистецтва

Mont Day Year

Email Address

Username

Password

Confirm Password

I agree with the Terms and Conditions.

[Sign up](#)

Рисунок 3.12 – Надання інформації під час реєстрації

При натисканні на вкладку Critics буде здійснено перехід на сторінку з рецензіями (рис.3.13).

Movies Heaven HOME ABOUT US CRITICS CONTACT US User

All our critics





 <p>by Felix 25/08/2018 ★★★★★</p> <p>The House with a Clock in Its Walls</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p> <p>See the critic ></p>	 <p>by Carolina 25/08/2018 ★★★★★</p> <p>The predator</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p> <p>See the critic ></p>	 <p>by Arthur 25/08/2018 ★★★★★</p> <p>The Grinch</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p> <p>See the critic ></p>	 <p>by Audrey 25/08/2018 ★★★★★</p> <p>Alad'2</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit...</p> <p>See the critic ></p>
---	---	--	---

Рисунок 3.13 – Рецензії на сайті кіномистецтва

При відкритті певної рецензії буде такий загальний вид сторінки (рис.3.14).



Рисунок 3.14 – Розміщення критики на фільм «Інтерстеллар»

На рисунках 3.15-3.18 зображено процес проектування бази даних сайту. Рисунок 3.15 показує загальний CRUD-інструментарій при наявності чотирьох сутностей в БД.

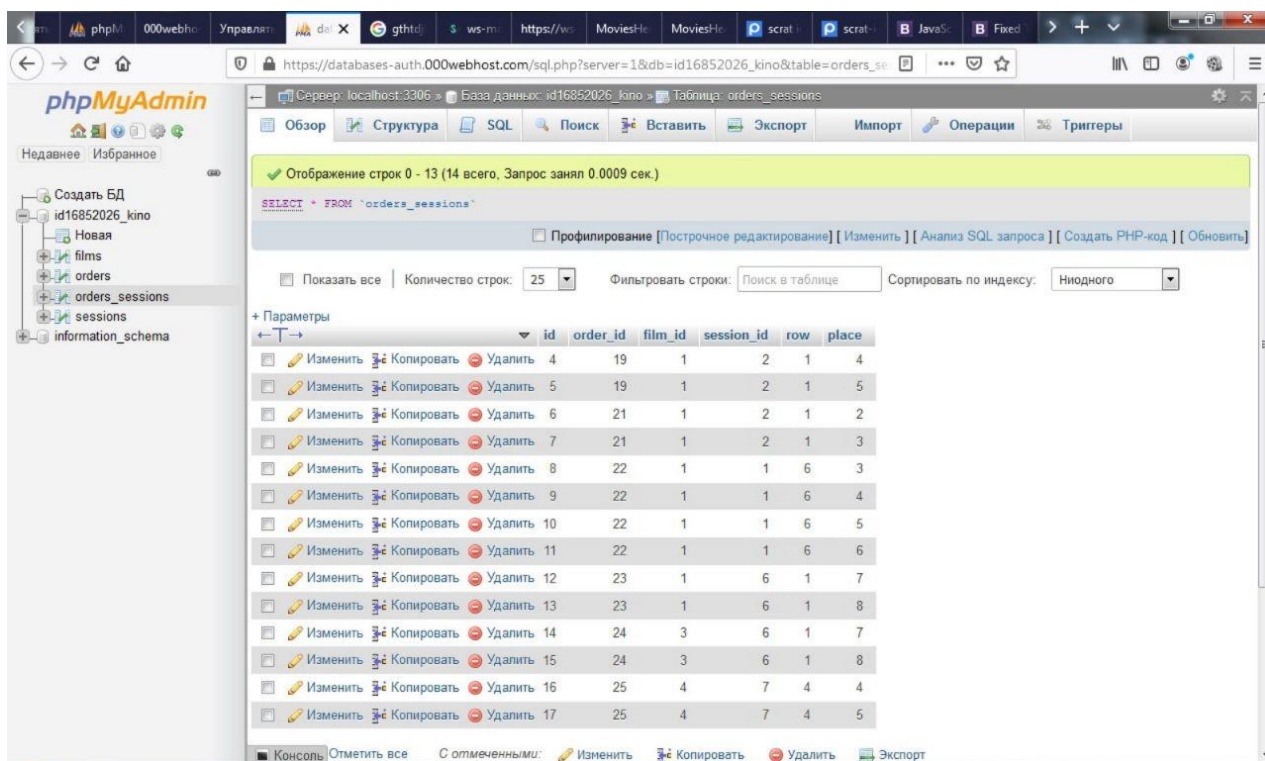


Рисунок 3.15 – Проектування бази даних

Рисунок 3.16 показує роботу з фільмами.

The screenshot shows the phpMyAdmin interface for a database named 'id16852026_kino'. The 'films' table is selected, and the following data is displayed:

id	name	status
1	ICE AGE	1
2	CLOCKWORK ORANGE	1
3	TOY STORY	1
4	INTERSTELAR	1

Рисунок 3.16 – Проектування бази даних

Рисунок 3.17 показує роботу з сесіями.

The screenshot shows the phpMyAdmin interface for the same database. The 'sessions' table is selected, and the following data is displayed:

id	film_id	s_time	status
1	1	18:00	1
2	1	20:30	1
3	2	15:00	1
4	2	22:00	1
5	3	13:00	1
6	3	15:30	1
7	4	10:00	1

Рисунок 3.17 – Проектування бази даних

Рисунок 3.18 показує роботу з замовленнями.

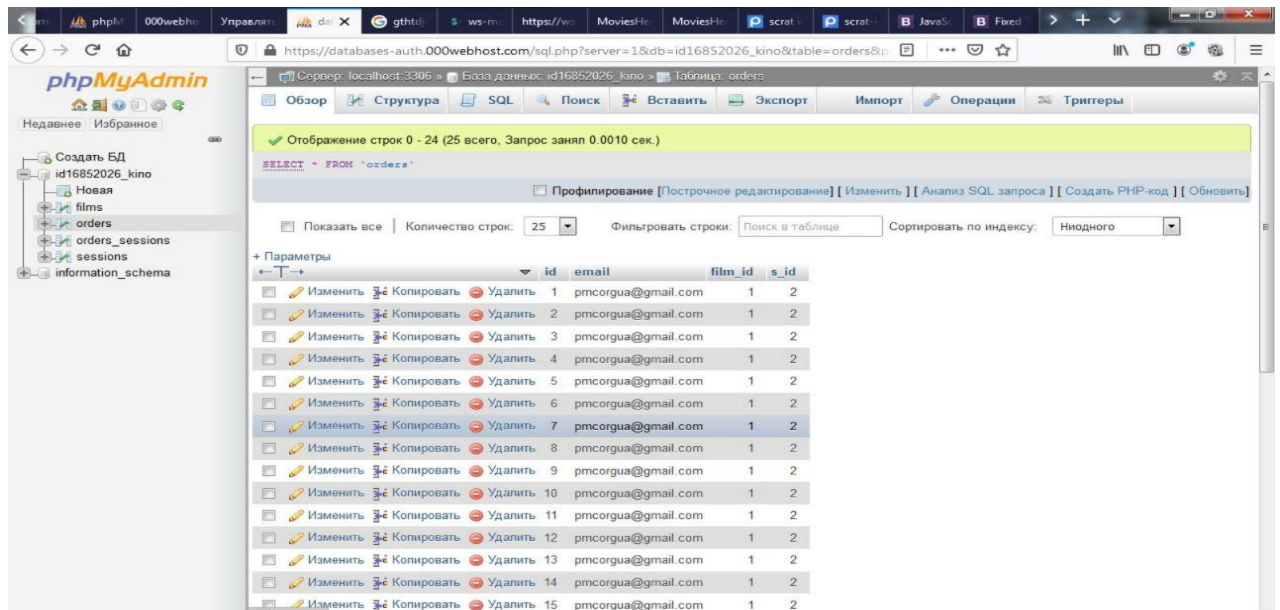


Рисунок 3.18 – Проектування бази даних

Проектування бази даних реалізується засобами phpMyAdmin.

В першу чергу phpMyAdmin дуже полегшує виконання «обивательських» завдань, які по класичному підходу, за допомогою введення SQL-запитів в командній консолі MySQL (для якої попередньо ще потрібно виконати авторизацію) займають невинувато багато часу. Візуальний стиль управління і редагування БД, що надається phpMyAdmin куди більш зрозумілий швидкий і наочний спосіб.

Найчастіше необхідно швидко відредагувати поля таблиці БД. В цьому випадку написання SQL-запитів є досить довгим. Або необхідно швидко створити, видалити або очистити БД phpMyAdmin дозволить це зробити в лічені секунди, буквально за кілька кліків.

Крім інструментів редагування phpMyAdmin надає також можливості для моніторингу швидкодії, продуктивності MySQL-сервера, а також аналізу його роботи.

Імпорт та експорт також не є для phpMyAdmin проблемою, проте з деякими обмеженнями, що стосуються розмірів файлів для імпорту / експорту, що

пов'язано з настройками PHP та веб-сервера. Все-таки phpMyAdmin - це веб-додаток. [27]

На рисунках 3.19-3.23 зображено процес взаємодії користувача з сервером. На рис. 3.19 бачимо список кінофільмів, на які можливо оформити замовлення. Рис. 3.20 ілюструє процес входу в систему і список замовлень певного користувача, а рис. 3.21 список замовлень у адміністратора ресурсу, рис. 3.22 ілюструє вибір місць на сеанс, рис. 3.23 – сторінку користувача після виконання замовлення.

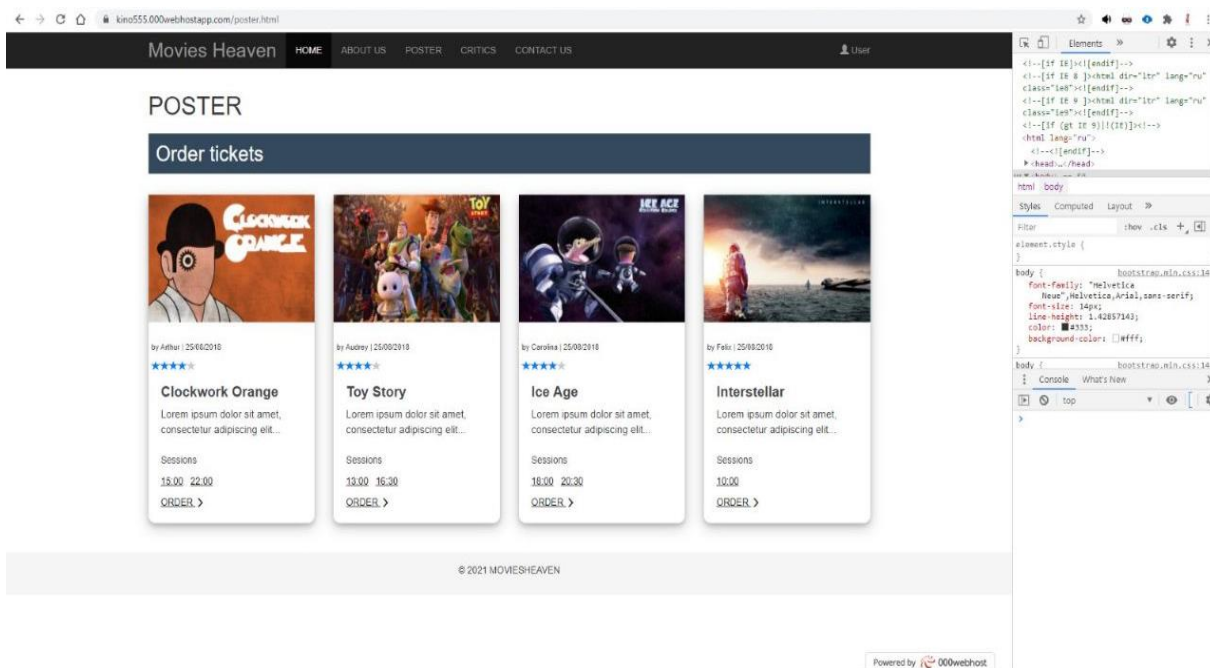


Рисунок 3.19 – Сторінка замовлення білетів

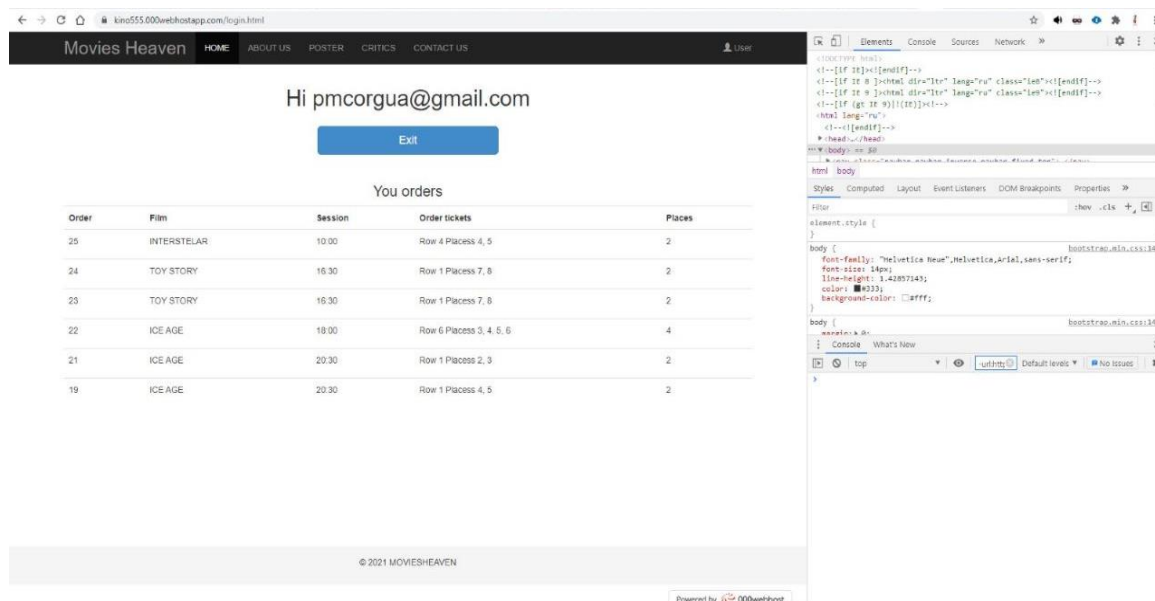


Рисунок 3.20 – Замовлення певного користувача

Hi admin@gmail.com

Exit

All orders

Order	User	Film	Session	Order tickets	Places
30	laneevav@gmail.com	CLOCKWORK ORANGE	22:00	Row 1 Places 5, 6	2
29	katyalominad@gmail.com	TOY STORY	15:00	Row 3 Places 6	1
28	katyalominad@gmail.com	TOY STORY	13:00	Row 5 Places 3, 4, 5	3
27	katyalominad@gmail.com	CLOCKWORK ORANGE	22:00	Row 5 Places 4	1
26	katyalominad@gmail.com	ICE AGE	16:00	Row 4 Places 4	1
25	pmcorgua@gmail.com	INTERSTELAR	10:00	Row 4 Places 4, 5	2
24	pmcorgua@gmail.com	TOY STORY	16:30	Row 1 Places 7, 8	2
23	pmcorgua@gmail.com	TOY STORY	16:30	Row 1 Places 7, 8	2
22	pmcorgua@gmail.com	ICE AGE	16:00	Row 6 Places 3, 4, 5, 6	4
21	pmcorgua@gmail.com	ICE AGE	20:30	Row 1 Places 2, 3	2
19	pmcorgua@gmail.com	ICE AGE	20:30	Row 1 Places 4, 5	2

Powered by 000webhost

Рисунок 3.21 – Особистий кабінет адміністратора

ORDER TICKETS 16:30

DISPLAY

Email address

ORDER

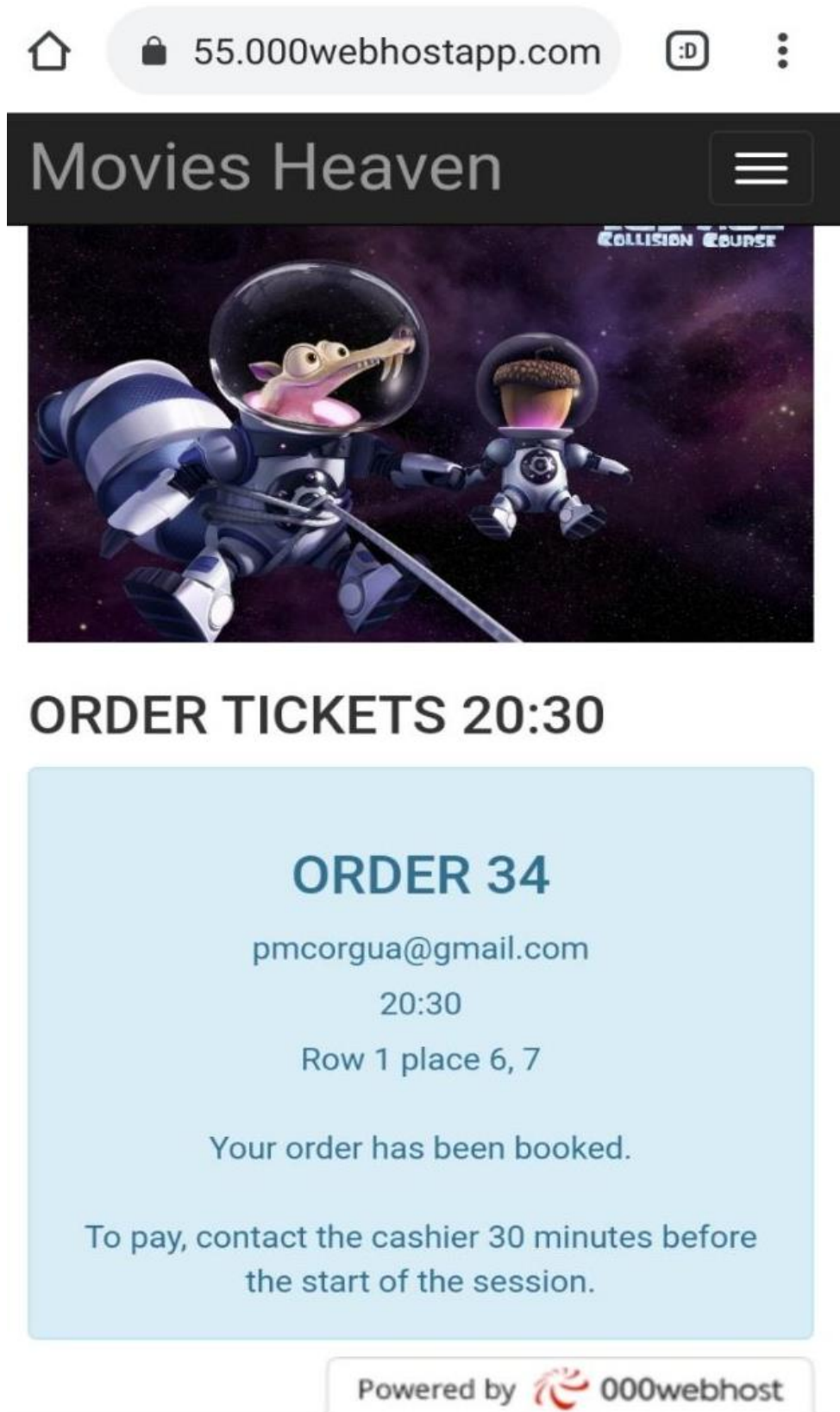
Ordered No ordered

© 2021 MOVIESHEAVEN

Powered by 000webhost

Рисунок 3.22 – Вибір місць на кіносеанс

Рисунок 3.23 показує сповіщення про виконане замовлення..



The screenshot shows a mobile browser interface. At the top, the address bar displays the URL "55.000webhostapp.com". Below the address bar is a dark navigation bar with the text "Movies Heaven" and a hamburger menu icon. The main content area features a movie poster for "COLLISION COURSE" showing two cartoon characters in space suits. Below the poster, the text "ORDER TICKETS 20:30" is displayed. A light blue box contains the following information: "ORDER 34", "pmcorgua@gmail.com", "20:30", "Row 1 place 6, 7", "Your order has been booked.", and "To pay, contact the cashier 30 minutes before the start of the session." At the bottom of the page, there is a footer that says "Powered by 000webhost" with a logo.

55.000webhostapp.com

Movies Heaven

COLLISION COURSE

ORDER TICKETS 20:30

ORDER 34

pmcorgua@gmail.com

20:30

Row 1 place 6, 7

Your order has been booked.

To pay, contact the cashier 30 minutes before the start of the session.

Powered by 000webhost

Рисунок 3.23 – Виконане замовлення

Таблиця 3.3 – Порівняння системи з аналогами

Назва	Переваги	Недоліки
MoviesHeaven	Адаптивність дизайну, гнучкість бази даних, швидкодія роботи	Недопрацювання в оформленні сайту
Kinopoisk	Простота, зручність роботи, якість інтерфейсу	Низька швидкодія БД
kinomania	Якість оформлення та стилістики	Повільність серверного модуля

Як бачимо, веб-сервіс MoviesHeaven є набагато кращим у клієнт-серверному плані за свої аналоги.

"Клієнт" використовується в середовищі клієнт-сервер для позначення програми, яку людина використовує для запуску програми. Середовище клієнт-сервер - це середовище, в якому кілька комп'ютерів обмінюються інформацією, наприклад введенням інформації в базу даних. "Клієнт" - це програма, яка використовується для введення інформації, а "сервер" - програма, що використовується для зберігання інформації.

Для розробки програми веб-додатки зазвичай використовують комбінацію сценарію на стороні сервера (ASP, PHP тощо) та сценарію на стороні клієнта (HTML, Javascript тощо).

Більшість веб-додатків базуються на архітектурі клієнт-сервер, куди клієнт вводить інформацію, а сервер зберігає та отримує інформацію. Прикладом цього є Інтернет-пошта, коли такі компанії, як Google, Gmail та Microsoft Outlook, пропонують веб-клієнти електронної пошти.

Веб-додатки можуть забезпечити однакову функціональність та отримати перевагу роботи на кількох платформах. Наприклад, веб-додаток може діяти як

текстовий процесор, зберігаючи інформацію в хмарі і дозволяючи "завантажувати" документ на свій особистий жорсткий диск.

Основна роль бази даних - зберігати та відображати оновлену інформацію у веб-програмі. Реєстраційні веб-сайти, дискусійні форуми та веб-сайти роздрібної торгівлі – це приклади веб-програм, які залежать від надійного компонента бази даних.

Додатки баз даних використовуються для пошуку, сортування, фільтрування та подання інформації на основі веб-запитів користувачів. Бази даних також можуть містити код для виконання математичних та статистичних обчислень даних, що підтримують запити, подані з веб-браузерів.

Бази даних надають і обмежують доступ до даних на основі таких критеріїв, як ім'я користувача, пароль, регіон або номер рахунку. Бази даних також забезпечують цілісність даних, забезпечуючи збір та подання даних у єдиному форматі.

Динамічний веб-сайт відображає оновлену інформацію на веб-сторінках, коли база даних оновлюється хостом або коли користувачі подають інформацію за допомогою веб-форм. База даних автоматично оновлює веб-сторінки, виключаючи вимогу вручну оновлювати HTML-код на окремих сторінках.

Розроблений веб-сервіс цілком відповідає даним вимогам до проектування сучасних клієнт-серверних веб-додатків.

Висновки

У даному розділі описано поняття алгоритму та алгоритм власне для даного веб-сервісу, проаналізовано підходящі технології та обрано найбільш відповідну мову програмування та середовище для розробки додатку, описано процес розробки та протестовано даний веб-сервіс.

ВИСНОВКИ

В ході написання бакалаврської дипломної роботи здійснено аналіз предметної області індустрії кінофільмів та кіномистецтва, здійснено порівняльний аналіз аналогів веб-систем з перегляду кінофільмів, виявлено їх основні переваги та недоліки.

Побудовано структурну схему та UML-діаграми роботи веб-сервісу кіномистецтва та перегляду фільмів.

Здійснено програмну реалізацію веб-системи кіномистецтва та перегляду фільмів.

Виконано тестування розробленої веб-системи кіномистецтва та перегляду фільмів.

Підтверджено, що розроблена веб-система працює коректно у всіх режимах та володіє 2 додатковими функціями – оцінюванням фільмів за системою «зірок» та надання усім користувачам ID.

Отже всі задачі, які було поставлено в дипломній роботі виконані, мету роботи досягнуто.

СПИСОК ЛІТЕРАТУРИ

1. Кіномистецтво. URL: <https://uk.wikipedia.org/wiki/%D0%9A%D1%96%D0%BD%D0%BE%D0%BC%D0%B8%D1%81%D1%82%D0%B5%D1%86%D1%82%D0%B2%D0%BE> (дата звернення: 20.04.2021).
2. 30 ЛУЧШИХ САЙТОВ О КИНО. URL: <https://infoselection.ru/infokatalog/kino-video/kino-sajty/item/71-8-samykh-populyarnykh-sajtov-o-kino> (дата звернення: 18.04.2021).
3. Кінематограф як особливий вид мистецтва. URL: <https://ru.osvita.ua/vnz/reports/culture/10727/> (дата звернення: 25.04.2021).
4. Unified Modeling Language. URL: https://uk.wikipedia.org/wiki/Unified_Modeling_Language (дата звернення: 13.05.2021).
5. UML для бизнес-моделирования: зачем нужны диаграммы процессов. URL: <https://evergreens.com.ua/ru/articles/uml-diagrams.html> (дата звернення: 10.04.2021).
6. UML-МОДЕЛИРОВАНИЕ. URL: <https://fingers.by/blog/uml> (дата звернення: 18.04.2021).
7. Діаграма прецедентів. URL: https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D1%96%D0%B2 (дата звернення: 20.04.2021).
8. Діаграма діяльності. URL: https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%B4%D1%96%D1%8F%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D1%96 (дата звернення: 28.04.2021).
9. Предметна область, моделювання предметної області. URL: <https://studfile.net/preview/5474325/page:2/> (дата звернення: 01.05.2021).
10. HTML. URL: <https://ru.wikipedia.org/wiki/HTML> (дата звернення: 01.05.2021).

11. Справочник по HTML. URL: <http://htmlbook.ru/html> (дата звернення: 08.05.2021).
12. Алгоритм. URL: <https://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC> (дата звернення: 08.04.2021).
13. Мова JavaScript URL: <https://learn.javascript.ru/js> (дата звернення: 15.04.2021).
14. алгоритмизация и программирование. URL: [http://csaa.ru/tema-5-algoritmizacija-i-programmirovanie/#:~:text=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F%20%E2%80%93%D1%8D%D1%82%D0%BE%20%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%20%D0%BE%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D0%B8%D1%8F%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC,\(%D0%93%D0%9E%D0%A1%D0%A2%2019.781%2D%2074\)](http://csaa.ru/tema-5-algoritmizacija-i-programmirovanie/#:~:text=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F%20%E2%80%93%D1%8D%D1%82%D0%BE%20%D0%BC%D0%B5%D1%82%D0%BE%D0%B4%20%D0%BE%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D0%B8%D1%8F%20%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC,(%D0%93%D0%9E%D0%A1%D0%A2%2019.781%2D%2074)). (дата звернення: 02.04.2021).
15. Бази даних URL: <http://lib.mdpu.org.ua/e-book/vstup/L5.htm> (дата звернення: 17.04.2021).
16. JavaScript - краща мова для програмування. Правда чи брехня? URL: <http://areps.kpi.ua/javascript-krashcha-mova-dlia-programuvania> (дата звернення: 27.04.2021).
17. Полный курс по JavaScript + React - с нуля до результата URL: https://www.udemy.com/course/javascript_full/ (дата звернення: 30.04.2021).
18. Введение в работу с MySQL в PHP URL: <https://metanit.com/php/mysql/1.1.php> (дата звернення: 15.05.2021).
19. Как получить данные для входа в phpMyAdmin? URL: <https://invs.ru/support/chastie-voprosy/kak-poluchit-dannye-dlya-vkhoda-v-phpmyadmin/> (дата звернення: 23.04.2021).
20. ЧТО ТАКОЕ PHPMYADMIN? URL: <https://freehost.com.ua/faq/wiki/chto-takoe-phpmyadmin/> (дата звернення: 17.04.2021).

21. Компоненты сетевого приложения. Клиент-серверное взаимодействие и роли серверов. URL: <http://www.4stud.info/networking/lecture5.html> (дата звернення: 21.04.2021).
22. ПРОЕКТУВАННЯ WEB-ДОДАТКІВ URL: https://stud.com.ua/97678/informatika/proektuvannya_dodatki (дата звернення: 24.04.2021).
23. Розробка WEB – додатків для бізнесу URL: <https://tqm.com.ua/ua/rozrobka-veb-dodatki-servisiv-web-application-development/rozrobka-veb-dodatki-dlia-biznesu> (дата звернення: 24.04.2021).
24. Сутність та поняття проекту URL: https://pidru4niki.com/1057011647752/informatika/sutnist_ponyattya_proektu (дата звернення: 26.04.2021).
25. Управление ИТ-проектами – 5 вызовов и их преодоление URL: <https://it-guild.com/info/blog/upravlenie-it-proektami-5-vyzovov-i-ikh-preodolenie/> (дата звернення: 02.04.2021).
26. Сервисное программное обеспечение ПК и основы алгоритмизации URL: <https://www.lessons-tva.info/edu/e-inf1/e-inf1-4-2.html> (дата звернення: 02.04.2021).
27. phpMyAdmin URL: <https://ru.wikipedia.org/wiki/PhpMyAdmin> (дата звернення: 17.04.2021).

ДОДАТКИ

Додаток А. Основний лістинг

[Index.html](#)

```
<!doctype html>

<html lang="en">

  <head>

    <!-- Required meta tags -->

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">

    <!-- Bootstrap CSS -->

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdkn
LPMO" crossorigin="anonymous" />

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

    <link rel="stylesheet" href="Styleheaven.css" />

    <title>MoviesHeaven - Toy Story</title>

  </head>

  <body>

    <!-- Navbar -->

    <div class="container">

      <div id="accueil">

        <div class="sticky-top">

          <nav class="navbar navbar-expand-lg navbar-light bg-light">
```

```
<a class="navbar-brand disabled" href="index.html"> Movies  
Heaven</a>
```

```
<button class="navbar-toggler" type="button" data-toggle="collapse"  
data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-  
label="Toggle navigation">
```

```
<span class="navbar-toggler-icon"></span>
```

```
</button>
```

```
<div class="collapse navbar-collapse" id="navbarNav">
```

```
<ul class="navbar-nav">
```

```
<li class="nav-item active">
```

```
<a class="nav-link" href="index.html">HOME</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="#about">ABOUT US</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="critics.html">CRITICS</a>
```

```
</li>
```

```
<li class="nav-item">
```

```
<a class="nav-link" href="contactus.html">CONTACT US</a>
```

```
</li>
```

```
</ul>
```

```
</div>
```

```

<a href="Log.html">
    <button type="button" class="btn btn-default btn-sm">
        <span class="glyphicon glyphicon-user"></span> User
    </button>
</a>
</nav>
</div>
<!-- space entre nav et carousel -->
<div class="vide"></div>
<!-- Carousel -->
<div id="carouselExampleIndicators" class="carousel slide" data-
ride="carousel">
    <ol class="carousel-indicators">
        <li data-target="#carouselIndicators" data-slide-to="0"
class="active"><a href="CLOCKWORK-ORANGE.html"></a></li>
        <li data-target="#carouselExample1Indicators" data-slide-to="1"></li>
        <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
        <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
    </ol>
    <div class="carousel-inner">
        <div class="carousel-item active">
            <a href="CLOCKWORK-ORANGE.html"></a>

```

```
</div>
```

```
<div class="carousel-item">
```

```
  <a href="TOY-STORY.html"> </a>
```

```
</div>
```

```
<div class="carousel-item">
```

```
  <a href="ICE-AGE.html"></a>
```

```
</div>
```

```
<div class="carousel-item">
```

```
  <a href="INTERSTELAR.html"></a>
```

```
</div>
```

```
</div>
```

```
<a class="carousel-control-prev" href="#carouselExampleIndicators"
role="button" data-slide="prev">
```

```
  <span class="carousel-control-prev-icon" aria-hidden="true"></span>
```

```
  <span class="sr-only">Previous</span>
```

```
</a>
```

```
<a class="carousel-control-next" href="#carouselExampleIndicators"
role="button" data-slide="next">
```

```
  <span class="carousel-control-next-icon" aria-hidden="true"></span>
```

Next

</div>

<div id="about">

<h1 class="principal">ABOUT US</h1>

<p class="description">Film, also called movie, motion picture or moving picture, is a visual art used to simulate experiences that communicate ideas, stories, perceptions, feelings, beauty, or atmosphere through the use of moving images. These images are generally accompanied by sound, and more rarely, other sensory stimulations. The word "cinema", short for cinematography, is often used to refer to filmmaking and the film industry, and to the art form that is the result of it.

The moving images of a film are created by photographing actual scenes with a motion-picture camera, by photographing drawings or miniature models using traditional animation techniques, by means of CGI and computer animation, or by a combination of some or all of these techniques, and other visual effects.

Traditionally, films were recorded onto celluloid film stock through a photochemical process and then shown through a movie projector onto a large screen. Contemporary films are often fully digital through the entire process of production, distribution, and exhibition, while films recorded in a photochemical form traditionally included an analogous optical soundtrack (a graphic recording of the spoken words, music and other sounds that accompany the images which runs along a portion of the film exclusively reserved for it, and is not projected).

Films are cultural artifacts created by specific cultures. They reflect those cultures, and, in turn, affect them. Film is considered to be an important art form, a source of popular entertainment, and a powerful medium for educating—or indoctrinating—citizens. The visual basis of film gives it a universal power of communication. Some

films have become popular worldwide attractions through the use of dubbing or subtitles to translate the dialog into other languages.

The individual images that make up a film are called frames. In the projection of traditional celluloid films, a rotating shutter causes intervals of darkness as each frame, in turn, is moved into position to be projected, but the viewer does not notice the interruptions because of an effect known as persistence of vision, whereby the eye retains a visual image for a fraction of a second after its source disappears. The perception of motion is partly due to a psychological effect called the phi phenomenon.

The name "film" originates from the fact that photographic film (also called film stock) has historically been the medium for recording and displaying motion pictures. Many other terms exist for an individual motion-picture, including picture, picture show, moving picture, photoplay, and flick. The most common term in the United States is movie, while in Europe film is preferred. Common terms for the field in general include the big screen, the silver screen, the movies, and cinema; the last of these is commonly used, as an overarching term, in scholarly texts and critical essays. In early years, the word sheet was sometimes used instead of screen.

</div>

</div>

<!-- Pied de page-->

<footer id="basdepage">

A propos

© 2020 MOVIESHEAVEN


```

</footer>

<!-- Optional JavaScript -->

<!-- jQuery first, then Popper.js, then Bootstrap JS -->

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>

<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
integrity="sha384-
ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnjAK/18WvCWPIP
m49" crossorigin="anonymous"></script>

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"
integrity="sha384-
ChfqquxZUCnJSK3+MXmPNlyE6ZbWh2IMqE241rYiqJxyMiZ6OW/JmZQ5stwE
ULTy" crossorigin="anonymous"></script>

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>

<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script
>

</div>

</body>

</html>

```

[Create_account.html](#)

```
<!doctype html>

<html lang="en">

  <head>

    <!-- Required meta tags -->

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">

    <!-- Bootstrap CSS -->

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdkn
LPMO" crossorigin="anonymous" />

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

    <link rel="stylesheet" href="Styleheaven.css" />

    <title>MoviesHeaven</title>

  </head>

  <body id="imagedefond">

    <section id="misenpage">

      <div>

        <form method="post" action="traitement.php">
```

```

<div>

    <a href="index.html"></a>

    <label for="f.name"> First Name </label>

    <input type="name" name="F.Name" class="form-control input-lg"
id="inputf.name" maxlength="18" placeholder="First name" required /><br/>

    <label for="l.name"> Last Name </label>

    <input type="name" name="L.Name" class="form-control input-lg"
id="inputl.name" maxlength="18" placeholder=" Last name" required /><br/>

    <label class="radio-inline">

        <input type="radio" name="sex" id="inlineCheckbox1"
value="male" />

        Male

    </label>

    <label class="radio-inline">

        <input type="radio" name="sex" id="inlineCheckbox2"
value="female" />

        Female

    </label><br/>

<br/><label for="">

    Birth Date</label>

<div class="row">

    <div class="col-xs-4 col-md-4">

```

```

        <select class="form-control">
            <option value="Month">Month</option>
        </select>
    </div>
    <div class="col-xs-4 col-md-4">
        <select class="form-control">
            <option value="Day">Day</option>
        </select>
    </div>
    <div class="col-xs-4 col-md-4">
        <select class="form-control">
            <option value="Year">Year</option>
        </select>
    </div>
</div><br/>
<br><label for="Email"> Email Address </label>
<input type="email" name="Email" class="form-control input-lg"
id="inputEmail" maxlength="18" placeholder="Email address" required /><br/>
<label for="username"> Username </label>
<input type="username" name="Username" class="form-control input-
lg" id="inputl.name" maxlength="18" placeholder="ID" required /><br/>
<label for="password"> Password</label>

```

```

        <input type="password" class="form-control input-lg"
id="inputPassword" placeholder="Password" required/><br/>

        <label for="c.password"> Confirm Password</label>

        <input type="password" class="form-control input-lg"
id="inputPassword" placeholder="Password" required/><br/>

        <div id="remember" class="checkbox">

            <label><input type="checkbox" name="terms"> I agree with the <a
href="#" >Terms and Conditions</a>.</label>

        </div>

        <button class="btn btn-lg btn-primary btn-block btn-signin"
type="submit"> Sign up </button>

    </div>

</form>

</div>

</section>

<!------- ----->

<link href="//netdna.bootstrapcdn.com/twitter-bootstrap/2.3.2/css/bootstrap-
combined.min.css" rel="stylesheet" id="bootstrap-css">

<script src="//netdna.bootstrapcdn.com/twitter-
bootstrap/2.3.2/js/bootstrap.min.js"></script>

<script src="//code.jquery.com/jquery-1.11.1.min.js"></script>

</body>

</html>

```

Critics.html

```
<!doctype html>
```

```
<html lang="en">
```

```
<head>
```

```
<!-- Required meta tags -->
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
<!-- Bootstrap CSS -->
```

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css"
integrity="sha384-
MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdkn
LPMO" crossorigin="anonymous" />
```

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
```

```
<link rel="stylesheet" href="Styleheaven.css" />
```

```
<title>MoviesHeaven - Critics</title>
```

```

</head>

<body>

  <!-- Navbar -->

  <div class="container">

    <div class="sticky-top">

      <nav class="navbar navbar-expand-lg navbar-light bg-light">

        <a class="navbar-brand disabled" href="index.html"> Movies
Heaven</a>

        <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-
expanded="false" aria-label="Toggle navigation">

          <span class="navbar-toggler-icon"></span>

        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

          <ul class="navbar-nav">

            <li class="nav-item active">

              <a class="nav-link" href="index.html">HOME</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="index.html#about">ABOUT US</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="critics.html">CRITICS</a>

```

```

    </li>

    <li class="nav-item">
        <a class="nav-link" href="contactus.html">CONTACT US</a>
    </li>
</ul>
</div>

<a href="Log.html">
    <button type="button" class="btn btn-default btn-sm">
        <span class="glyphicon glyphicon-user"></span> User
    </button>
</a>
</nav>
</div>

<!-- space entre nav et carousel -->
<div class="vide"></div>

<section id="Critics">
    <h1 id="titrepage">Critics</h1>

    <p id="descriptionpage">

        It is on this page that you will find all our reviews. Lorem ipsum dolor sit
        amet, consectetur adipiscing elit, sed do eiusmod tempor

        in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
        pariatur. </p>

```

```
<div class="movieslist liste1">  
  <h2>Our last critics</h2>  
  <div class="row flex-column-reverse flex-sm-row">  
    <div class="col-sm-12 col-md-6 col-lg-3">  
      <div class="card">  
        <div class="card-header">  
          <a href="CLOCKWORK-ORANGE.html"></a>  
        </div>
```