**Full Stack Development CS603**

**MINI PROJECT REPORT**

on

**MOVIE APP**

*Submitted by*

**D.VIJAY KUMAR REDDY (2021BCSE07AED102)**

**N. BHARATH KUMAR (2021BCSE07AED148)**

**G. RAJESH NAIDU (2021BCSE07AED146)**

*In partial fulfillment of the award of the degree of*

**BACHELOR OF TECHNOLOGY**

In

**COMPUTER SCIENCE AND ENGINEERING**

*Under the Supervision of*

**Prof.** **N Sengottaiyan**

****

**ALLIANCE COLLEGE OF ENGINEERING AND DESIGN**

**ALLIANCE UNIVERSITY, BENGALURU**

**MARCH - 2024**

****

# CERTIFICATE

This is to certify that the mini project work entitled “MOVIE APP” is the bonafide work done by D. VIJAY KUMAR REDDY (2021BCSE07AED102) N. BHARATH KUMAR (2021BCSE07AED148). G. RAJESH NAIDU (2021BCSE07AED146) submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology in Computer Science and Engineering during the year 2023-2024

Guide: Prof. N Sengottaiyan

Assistant Professor, CSE

**ABSTRACT**

The Movie App project aims to create a user-friendly web application using HTML, CSS, and JavaScript. Key features include user authentication, integration with a movie database API for comprehensive movie information, browsing and search functionality, movie details pages, user reviews and ratings, watchlist and favourites, responsive design, and suggestive features such as personalized recommendations. The implementation involves frontend development with frameworks like Bootstrap or React.js, backend development using Node.js or Python with Express.js or Django, database management, and API integration for real-time movie data retrieval.

The Movie App project endeavours to create an engaging web platform using HTML, CSS, and JavaScript, offering users a comprehensive movie exploration journey. Key components include user authentication for personalized experiences, seamless integration with a movie database API to fetch extensive movie details, robust search and browsing functionalities for effortless discovery, enriched movie pages with trailers and cast information, interactive user reviews and ratings fostering community engagement, personalized watchlists and favourites for user convenience, responsive design ensuring adaptability across devices, and intelligent recommendation features enhancing user satisfaction. The project's implementation encompasses frontend development leveraging frameworks like Bootstrap or React.js, backend support using technologies such as Node.js or Python with Express.js or Django, efficient database management, and API integration to ensure real-time data updates and a smooth user experience.

With a movie database API for up-to-date movie information, robust browsing and search capabilities, detailed movie pages with trailers and cast information. Implementation involves frontend development with potential frameworks like Bootstrap or React.js, backend support using technologies like Node.js or Python with Express.js or Django, efficient database management, and API integration to ensure a rich and interactive user experience.

**INTRODUCTION**

In an era dominated by digital entertainment, the Movie App project emerges as a beacon for cinephiles and casual viewers alike, aiming to revolutionize the movie-watching experience. With the fusion of HTML, CSS, and JavaScript, this web-based platform offers a gateway to a vast cinematic universe, replete with a plethora of features designed to captivate and delight users.

This introduction serves as a preamble to the intricate world of movie exploration that awaits within the confines of this application. From user authentication to seamless integration with a movie database API, every facet of the Movie App has been meticulously crafted to provide an immersive and intuitive experience. Through this project, users can embark on a journey of cinematic discovery, navigating through an extensive library of films, delving into detailed movie pages, engaging with fellow enthusiasts through reviews and ratings, and curating personalized watchlists tailored to their preferences.

As we delve deeper into the intricacies of this project, we unravel the layers of frontend and backend development, exploring the synergistic interplay of technologies such as Bootstrap, React.js, Node.js, Express.js, and Django. Furthermore, we delve into the realm of database management and API integration, essential components that underpin the seamless functionality and real-time data updates of the Movie App.

Through this project, we not only aspire to provide a platform for movie aficionados to indulge their passion but also to foster a vibrant community united by a shared love for cinema. With its responsive design and intelligent recommendation features, the Movie App endeavours to transcend boundaries, bringing the magic of the silver screen to users across devices and locations.In essence, the Movie App is more than just a digital repository of movies; it is a testament to the enduring allure of cinema and the boundless possibilities of technology.

**SYSTEM REQUIREMENTS**

**Hardware**

**Server:**

Processor: Powerful multi-core processor to handle user requests and database operations (Intel Xeon or AMD EPYC series recommended).

Memory (RAM): Ample RAM for caching, database operations, and running the web server (minimum 16GB, more is better for larger library collections).

Storage: High-capacity and fast storage (SSDs or a combination of SSD and HDD) for storing digital documents, metadata, search indexes, and user data. Consider redundancy options (RAID) for data protection.

**Client devices:**

Desktop/Laptop Computers: Modern web browsers support is essential. Processor and RAM requirements will depend on how resource-intensive the features of your digital library will be.

Mobile Devices (Tablets/Smartphones): The web interface should be responsive to provide a good user experience on smaller screens.

**Software:**

**Operating System (Server):**

Linux: Popular choice for web servers due to stability, security, and flexibility (e.g., Ubuntu, CentOS).

Windows Server: An option if you prefer a Windows-based environment.

**Web Server:**

Apache: Highly used, reliable, and open-source.

Nginx: Known for performance and ability to handle high traffic.

Database Management System (DBMS):

MySQL/MariaDB: Popular open-source relational databases.

PostgreSQL: Another powerful open-source option.

NoSQL options (MongoDB, Elasticsearch): Can be useful if you have a very large collection, need flexible document structures, and prioritize search performance.

**Programming Languages:**

Backend: Python (Django, Flask), PHP, Java, Ruby on Rails (for dynamic content, user management, database interactions).

Frontend: HTML, CSS, JavaScript (and potentially a JavaScript framework like React, Angular, or Vue.js for a more dynamic user interface).

Content Management System (CMS) (Optional): A CMS, like Drupal or WordPress, can simplify website structure and content management, especially if non-technical users will be involved.

**Search Engine:**

Solr/Elasticsearch: Powerful open-source options for advanced text-based searches.

**Device Compatibility**:

The application is accessible on various devices, including desktop computers, laptops, tablets, and smartphones. It is recommended to have a screen resolution of at least 1024x768 pixels for an optimal viewing experience.

**Internet Connection:**

A stable internet connection is required to access the Movie App and fetch real-time movie data from the integrated movie database API. A broadband connection is recommended for smooth streaming of movie trailers and other multimedia content.

**JavaScript Enabled:**

The Movie App relies heavily on JavaScript for interactive features and dynamic content loading. Therefore, users must ensure that JavaScript is enabled in their web browser settings for the application to function correctly.

**Optional Accounts:**

While users can access basic features of the Movie App without an account, creating an account enables access to personalized features such as watchlists, favorites, and user reviews. To create an account, users need a valid email address and password.

**API Key**:

If users plan to develop or deploy their instance of the Movie App, they may need to obtain an API key from the chosen movie database API provider. This key is used to authenticate requests to the API and fetch movie data.

By meeting these system requirements, users can enjoy a seamless and immersive movie browsing experience with the Movie App, unlocking a world of cinematic exploration at their fingertips.

**CODE**

**INDEX.HTML**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<meta charset="UTF-8" />**

**<meta name="viewport" content="width=device-width, initial-scale=1.0" />**

**<link rel="stylesheet" href="style.css" />**

**<title>Movie App</title>**

**</head>**

**<body>**

**<header>**

**<form id="form">**

**<input type="text" id="search" class="search" placeholder="Search">**

**</form>**

**</header>**

**<main id="main"></main>**

**<script src="script.js"></script>**

**</body>**

**</html>**

**STYLE.CSS**

**@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;400&display=swap');**

**:root {**

**--primary-color: #22254b;**

**--secondary-color: #373b69;**

**}**

**\* {**

**box-sizing: border-box;**

**}**

**body {**

**background-color: var(--primary-color);**

**font-family: 'Poppins', sans-serif;**

**margin: 0;**

**}**

**header {**

**padding: 1rem;**

**display: flex;**

**justify-content: flex-end;**

**background-color: var(--secondary-color);**

**}**

**.search {**

**background-color: transparent;**

**border: 2px solid var(--primary-color);**

**border-radius: 50px;**

**font-family: inherit;**

**font-size: 1rem;**

**padding: 0.5rem 1rem;**

**color: #fff;**

**}**

**.search::placeholder {**

**color: #7378c5;**

**}**

**.search:focus {**

**outline: none;**

**background-color: var(--primary-color);**

**}**

**main {**

**display: flex;**

**flex-wrap: wrap;**

**justify-content: center;**

**}**

**.movie {**

**width: 300px;**

**margin: 1rem;**

**background-color: var(--secondary-color);**

**box-shadow: 0 4px 5px rgba(0, 0, 0, 0.2);**

**position: relative;**

**overflow: hidden;**

**border-radius: 3px;**

**}**

**.movie img {**

**width: 100%;**

**}**

**.movie-info {**

**color: #eee;**

**display: flex;**

**align-items: center;**

**justify-content: space-between;**

**gap:0.2rem;**

**padding: 0.5rem 1rem 1rem;**

**letter-spacing: 0.5px;**

**}**

**.movie-info h3 {**

**margin-top: 0;**

**}**

**.movie-info span {**

**background-color: var(--primary-color);**

**padding: 0.25rem 0.5rem;**

**border-radius: 3px;**

**font-weight: bold;**

**}**

**.movie-info span.green {**

**color: lightgreen;**

**}**

**.movie-info span.orange {**

**color: orange;**

**}**

**.movie-info span.red {**

**color: red;**

**}**

**.overview {**

**background-color: #fff;**

**padding: 2rem;**

**position: absolute;**

**left: 0;**

**bottom: 0;**

**right: 0;**

**max-height: 100%;**

**transform: translateY(101%);**

**overflow-y: auto;**

**transition: transform 0.3s ease-in;**

**}**

**.movie:hover .overview {**

**transform: translateY(0);**

**}**

**SCRIPT.JS**

**const API\_URL = 'https://api.themoviedb.org/3/discover/movie?sort\_by=popularity.desc&api\_key=3fd2be6f0c70a2a598f084ddfb75487c&page=1'**

**const IMG\_PATH = 'https://image.tmdb.org/t/p/w1280'**

**const SEARCH\_API = 'https://api.themoviedb.org/3/search/movie?api\_key=3fd2be6f0c70a2a598f084ddfb75487c&query="'**

**const main = document.getElementById('main')**

**const form = document.getElementById('form')**

**const search = document.getElementById('search')**

**// Get initial movies**

**getMovies(API\_URL)**

**async function getMovies(url) {**

**const res = await fetch(url)**

**const data = await res.json()**

**showMovies(data.results)**

**}**

**function showMovies(movies) {**

**main.innerHTML = ''**

**movies.forEach((movie) => {**

**const { title, poster\_path, vote\_average, overview } = movie**

**const movieEl = document.createElement('div')**

**movieEl.classList.add('movie')**

**movieEl.innerHTML = `**

**<img src="${IMG\_PATH + poster\_path}" alt="${title}">**

**<div class="movie-info">**

**<h3>${title}</h3>**

**<span class="${getClassByRate(vote\_average)}">${vote\_average}</span>**

**</div>**

**<div class="overview">**

**<h3>Overview</h3>**

**${overview}**

**</div>**

**`**

**main.appendChild(movieEl)**

**})**

**}**

**function getClassByRate(vote) {**

**if(vote >= 8) {**

**return 'green'**

**} else if(vote >= 5) {**

**return 'orange'**

**} else {**

**return 'red'**

**}**

**}**

**form.addEventListener('submit', (e) => {**

**e.preventDefault()**

**const searchTerm = search.value**

**if(searchTerm && searchTerm !== '') {**

**getMovies(SEARCH\_API + searchTerm)**

**search.value = ''**

**} else {**

**window.location.reload()**

**}**

**})**

**SCREENSHOTS (OUTPUT)**

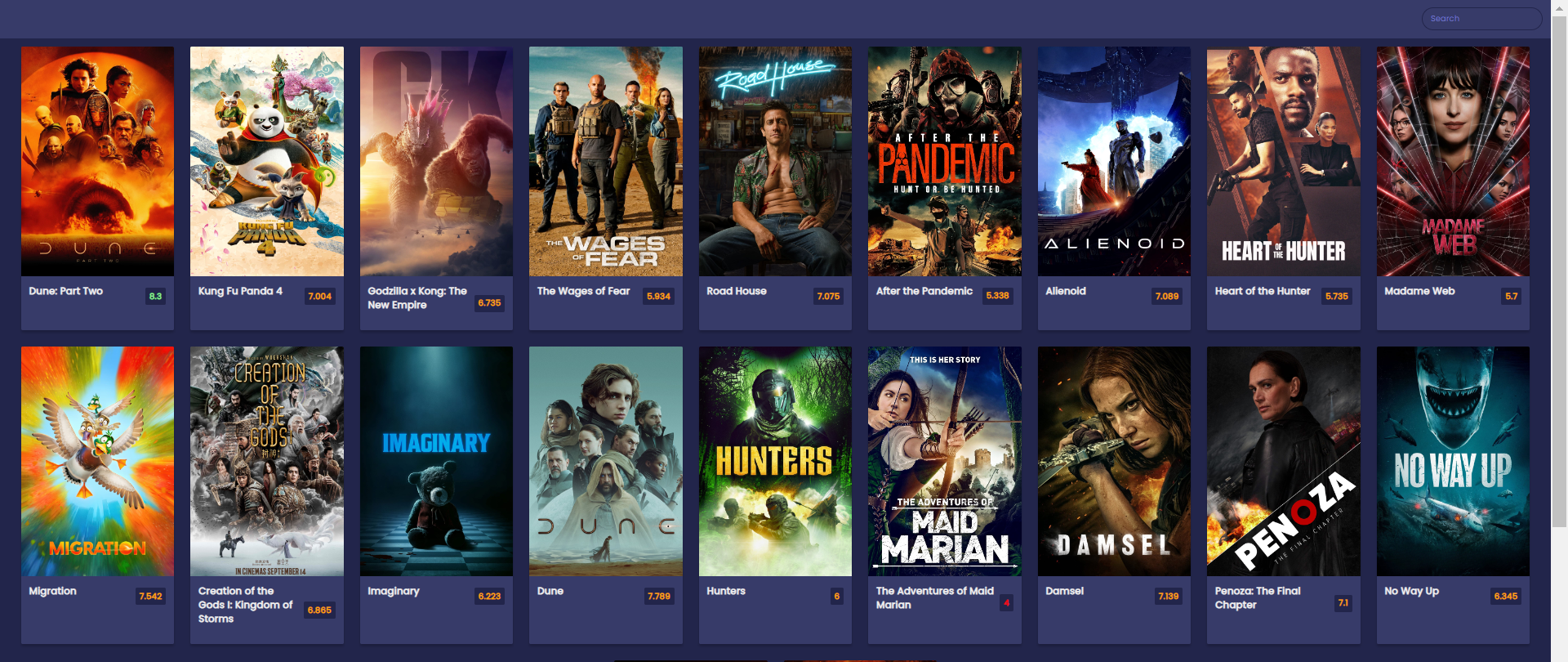


Fig-(i). Reviews of the movies

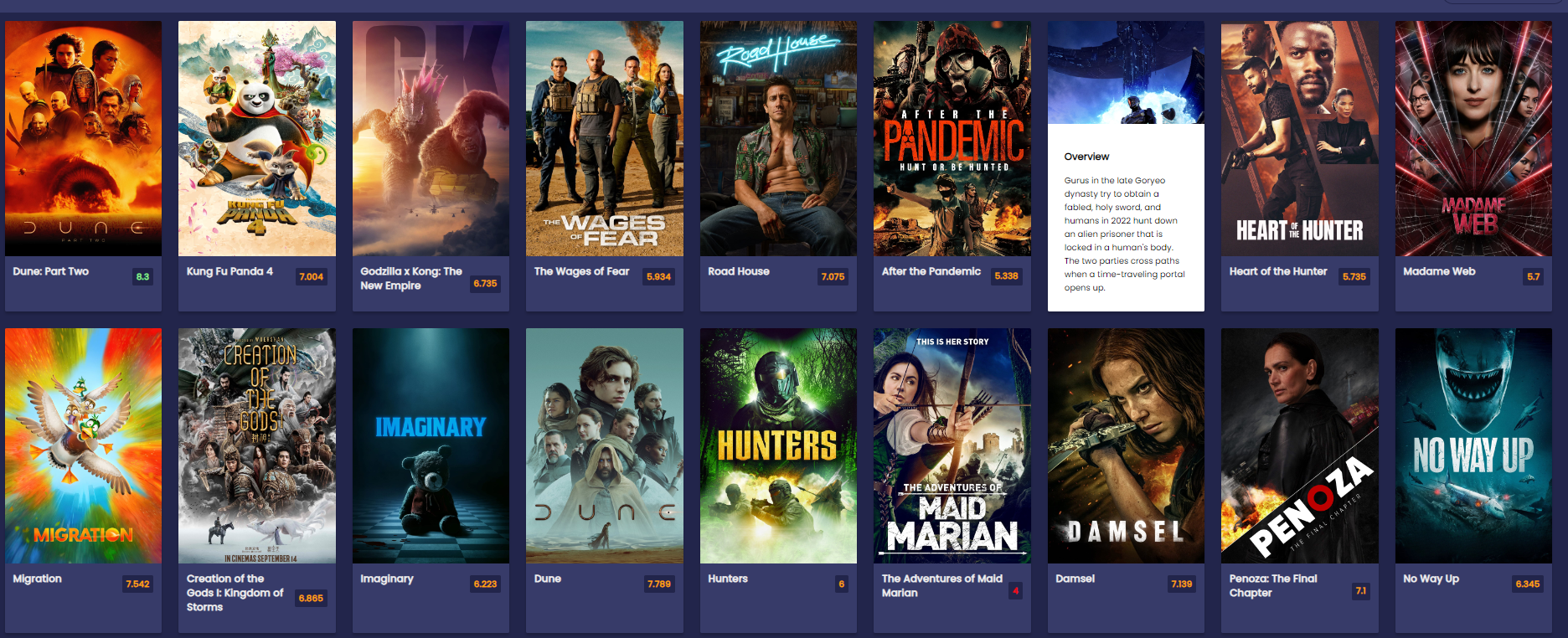


Fig-(ii). Overview of the movies

**Conclusion**

In conclusion, the "Movie Explorer" web application represents a culmination of innovation, accessibility, and user-centric design in the realm of movie exploration. Through the seamless integration of HTML, CSS, and JavaScript, this platform has redefined the way users engage with and discover movies, offering a dynamic and immersive experience that caters to the diverse tastes and preferences of movie enthusiasts worldwide.

From its visually captivating interface to its robust search functionality and real-time data integration, the "Movie Explorer" empowers users to explore and engage with a vast array of films effortlessly. Whether users are seeking the latest blockbusters, hidden gems, or timeless classics, the app provides a comprehensive platform for discovering, researching, and enjoying movies across genres, languages, and eras.

Moreover, the "Movie Explorer" prioritizes user experience above all else, with a responsive design that ensures seamless accessibility across devices and screen sizes. Whether users are browsing from the comfort of their desktop or on the go with their smartphone, the app adapts effortlessly to provide an immersive and enjoyable experience.

As users delve deeper into the "Movie Explorer" ecosystem, they discover a wealth of interactive features designed to enhance their journey through cinema. From detailed movie information and trailers to personalized recommendations and user reviews, the app equips users with everything they need to make informed decisions and curate their personalized movie-watching experience.

Ultimately, the "Movie Explorer" web application transcends the traditional boundaries of movie discovery, offering a dynamic and immersive platform that celebrates the art of cinema in all its diversity and splendor. Whether you're a seasoned film buff, a casual moviegoer, or someone just beginning to explore the world of movies, the "Movie Explorer" invites you to embark on a journey of cinematic exploration like never before. With its user-centric design, intuitive interface, and endless array of cinematic treasures, the "Movie Explorer" stands as a testament to the power of technology to enrich and enhance our lives through the magic of movies.