IMDB Application using ASP.NET

Internship Report
submitted to
Shri Ramdeobaba College of Engineering & Management,
Nagpur in partial fulfillment of requirement for the award of
degree of

Bachelor of Engineering

In

Computer Science and Engineering

By

Pratik Yuvraj Yawalkar

Guide

Dr. Shubhangi Neware

RCOEM
Shri Ramdeobaba College of
Engineering and Management, Nagpur

Computer Science and Engineering
Shri Ramdeobaba College of Engineering & Management, Nagpur
440 013 (An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

June – 2023

IMDB Application using ASP.NET

Internship Report
submitted to
Shri Ramdeobaba College of Engineering & Management,
Nagpur in partial fulfillment of requirement for the award of degree of

Bachelor of Engineering

In

Computer Science and Engineering

By

Pratik Yuvraj Yawalkar

Guide

Dr. Shubhangi Neware

Computer Science and Engineering
Shri Ramdeobaba College of Engineering & Management, Nagpur 440 013 (An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

2023

SHRI RAMDEOBABA COLLEGE OF ENGINEERING & MANAGEMENT, NAGPUR

(An Autonomous Institute Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

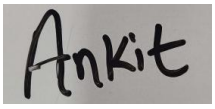
Department of Computer Science Engineering

CERTIFICATE

This is to certify that the Internship Report on **"IMDB Application using ASP.NET"** is a bonafide work of Pratik Yuvraj Yawalkar submitted to the Rashtrasant Tukdoji Maharaj Nagpur University, Nagpur in partial fulfilment of the award of a Bachelor of Engineering, in Computer Science and Engineering has been carried out at the DeltaX during the academic year 2022-2023.

Date: June, 2023

Place: Nagpur



Mr. Ankit Singh
Industry Mentor
DeltaX

Dr. A. J. Agrawal
H.O.D
Department of Computer Science Engineering

Dr. Shubhangi Neware
Internal Guide
Department of Computer Science Engineering

Dr. Parag Jawarkar
Dean-CDPC

Dr. Rajesh Pande
Principal

DECLARATION

I, hereby declare that the Project titled **"IMDB Application using ASP.NET"** submitted herein, has been carried out in the DeltaX. The work is original and has not been submitted earlier as a whole or part for the award of any degree / diploma at this or any other institution / University.

Date: June, 2023

Place: Nagpur

(Signature)

Pratik Yuvraj Yawalkar

A-062

Acknowledgements

This project report on “IMDB Application using ASP.NET” is being submitted to the Computer Science and Engineering Department of Shri Ramdeobaba College of Engineering and Management.

I am highly indebted to DeltaX, Bangalore for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards My Parents and Mr. Ankit Singh for their kind co-operation and encouragement which help me in completion of this project.

My thanks and appreciations also go to people who have willingly helped me out with their abilities.

Candidate Name

Pratik Yuvraj Yawalkar

Abstract

The IMDB Application is a web-based platform developed using ASP.NET, designed to provide users with a comprehensive movie database. The application aims to offer a user-friendly interface for users to browse, search, and discover information about movies, actors, and other related content. Utilizing the ASP.NET framework, the application utilizes the Model-View-Controller (MVC) architecture to ensure efficient data management and seamless user interaction.

The IMDB Application leverages the extensive movie database of the IMDB platform, incorporating features such as detailed movie descriptions, cast and crew information, user ratings, reviews, trailers, and more. Users can create personal accounts, allowing them to rate and review movies, create watchlists, and engage with other community members through a comment system.

ASP.NET enables the application to handle complex queries and large datasets, ensuring quick and accurate search results. The application utilizes various web technologies such as HTML, CSS, and JavaScript to create an intuitive and visually appealing user interface. The backend of the application is built using C#, allowing for efficient data processing and integration with external APIs.

Furthermore, the IMDB Application incorporates feature of dependency injection to manage and inject dependencies between components effectively. This approach promotes decoupling, facilitates unit testing by enabling the use of mock objects, and allows for easier swapping of implementations if needed in the future.

Table of Contents

List of Figures

List of Tables

CHAPTER 1
INTRODUCTION

The IMDB Application is an immersive and dynamic web-based platform that leverages the power of ASP.NET to provide users with an unparalleled movie experience. With a user-friendly interface and an extensive database, this application serves as a comprehensive hub for exploring, searching, and discovering a vast array of movies, actors, and related content. Designed to cater to the needs of both avid movie enthusiasts and casual viewers, the IMDB Application offers a multitude of features and functionalities. Users can delve into detailed movie descriptions, access comprehensive information about the cast and crew, explore user ratings and reviews, watch trailers, and much more. This wealth of information empowers users to make informed decisions and embark on thrilling cinematic journeys.

At the heart of the IMDB Application lies the robust ASP.NET framework, which enables seamless communication between the various components of the application. Following the Model-View-Controller (MVC) architectural pattern, ASP.NET ensures efficient data management and fosters a clean separation of concerns. The Model component of the application encapsulates both the data and the business logic, providing a solid foundation for the entire system. It represents the structured data related to movies, actors, and other relevant entities, along with the rules and operations governing their behaviour. This includes information such as titles, release dates, genres, and more. The View component is responsible for rendering the user interface and presenting the data to the users. It utilizes web technologies such as HTML, CSS, and JavaScript to create an engaging and visually appealing experience. Through well-designed and intuitive layouts, users can effortlessly navigate the application, explore different categories, and access detailed information about their favourite movies. Acting as the intermediary between the Model and the View, the Controller component plays a crucial role in handling user requests and managing the flow of data. It receives input from users, processes the requests, and coordinates the appropriate actions with the Model and View components. This includes fetching relevant data from the Model, manipulating it as required, and updating the View accordingly.

To ensure seamless data retrieval and management, the IMDB Application employs a Service layer, which resides between the Controller and the Model. This layer encapsulates the application's business logic and provides a set of services and operations that the Controller relies on. For example, it may include functionalities for searching movies, retrieving actor details, calculating average ratings, and generating personalized recommendations.

The IMDB Application takes advantage of the scalability and performance capabilities of ASP.NET to handle complex queries and manage large datasets with ease. Leveraging the power of C#, the backend of the application efficiently processes data and seamlessly integrates with external APIs and databases. Beyond its comprehensive database and user-friendly interface, the IMDB Application incorporates a range of additional features to enhance the user experience. Personalized recommendations based on user preferences allow individuals to discover new movies tailored to their tastes. Trending sections highlight the latest and most popular content, while the ability to create and manage watchlists enables users to curate their own personalized collections.

The application also fosters a sense of community by allowing users to engage with content through ratings, reviews, and comments. Users can share their opinions, express their thoughts, and discuss their favourite movies and with other like-minded individuals, promoting an interactive and engaging platform. Security is a top priority for the IMDB Application. It incorporates robust measures such as user authentication and data encryption to safeguard user information and ensure privacy.

The IMDB Application harnesses the power of ASP.NET to create a comprehensive and immersive movie platform. By seamlessly integrating data, user interaction, and a visually appealing interface, the application offers a captivating experience for movie enthusiasts. With its extensive features, personalized recommendations, and community engagement, the IMDB Application is the ultimate destination for users to dive into the world of cinema and discover their next favourite movies and TV shows.

CHAPTER 2

TECHNOLOGY/TOOLS USED

The IMDB Application utilizes a combination of technologies to build a robust and functional web application. The following powerful Hardware and Software Components were utilized in constructing the app, contributing to its exceptional performance and functionality.

2.1 Hardware Requirements

Component	Recommended Hardware Requirements
Computer / System	PC with Pentium IV or Higher Processor or another compatible processor.
RAM (Storage Memory)	RAM 256MB or more
Hard Disk Drive	Minimum Space on Hard Disk Drive (HDD) should be 200MB
Internet Connection	High Speed Connection is preferable
Monitor	Note: Ac-Tax recommends the users a screen resolution of 800x600 pixels or higher.
Operating System	Window9x/10/NT/XP/2000/Professional

Table 1. Hardware Components

2.2 Software Requirements

Component	Recommended Software Requirements
Integrated Development Environment	Microsoft Visual Studio Code 2019
Database	Microsoft SQL Server and Firebase (To store Cover Images)
Unit Testing (Using Behaviour Driven Development)	Spec-Flow
Version Control Platform	Bit-Bucket by Atlassian

Table 2. Software Components

2.3 Description of Tools

2.3.1 ASP.NET

It is a web development framework provided by Microsoft. It enables developers to build dynamic web applications using server-side technologies, such as C# or Visual Basic.NET. ASP.NET offers features like web forms, MVC (Model-View-Controller) pattern, routing, and data binding, which are essential for building the application's structure and functionality.

2.3.2 C#

It is a programming language developed by Microsoft that is widely used for building ASP.NET applications. It provides a strong and statically-typed language with object-oriented capabilities, making it suitable for building complex and scalable applications.

2.3.3 NET Framework or .NET Core

The application can be developed using either the .NET Framework or .NET Core, depending on the specific version and requirements. Both frameworks provide the necessary libraries and tools for building ASP.NET applications.

2.3.4 HTML/CSS

HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are fundamental technologies for building the user interface of the web application. HTML is used to structure the content and define the layout, while CSS is used for styling and presentation of the application's visual elements.

2.3.5 JavaScript

JavaScript is a scripting language commonly used for client-side interactions and enhancing the user experience in web applications. It can be used in combination with ASP.NET to add interactivity, perform form validation, handle events, and make asynchronous requests to the server.

2.3.6 SQL Server

Microsoft SQL Server is a relational database management system (RDBMS) often used with ASP.NET applications. It provides a reliable and scalable database solution for storing and retrieving movie data, user information, and other relevant data required by the application.

2.3.7 Dapper

It is a lightweight and high-performance ORM for .NET, known for its speed and simplicity in mapping database queries to objects. Dapper focuses on raw SQL execution and provides extensions to map query results directly to objects, eliminating the need for heavy mapping configurations. It allows developers to write SQL queries directly and efficiently map the query results to strongly typed objects, providing a fast and efficient data access solution.

2.3.8 RESTful APIs

The application may utilize RESTful APIs to integrate with external services or APIs, such as the IMDb API. These APIs allow the application to fetch movie data in real-time, enabling users to access up-to-date and accurate information about movies. RESTful APIs also facilitate efficient communication between the application and client devices, enabling smooth and responsive user experiences.

2.3.9 Bootstrap 4

It is a powerful and widely-used front-end framework that simplifies web development. It provides a collection of responsive CSS classes, pre-designed components, and JavaScript plugins, allowing developers to quickly build modern and visually appealing websites. With its intuitive grid system, extensive documentation, and cross-browser compatibility, It empowers developers to create stunning and responsive web applications with ease. It can be used to create a visually appealing and mobile-friendly user interface for the IMDB application. Other UI frameworks like Foundation or Material UI can also be utilized.

2.3.10 Dependency Injection

It is a powerful design pattern that promotes loose coupling and flexibility in software development. By externally providing dependencies to a class, DI enables easy swapping and testing of dependencies, enhancing modularity and maintainability. DI frameworks simplify dependency management, ensuring correct injection at runtime. Overall, DI fosters clean, modular code with clear separation of concerns, leading to more flexible and easily maintainable software solutions.

2.3.11 Firebase Storage

Firebase Storage is a critical component for the IMDB app as it provides a reliable and scalable platform to store movie poster images. With its secure cloud-based infrastructure, Firebase Storage ensures fast and efficient access to visually appealing content, enhancing the overall visual experience for users.

CHAPTER 3

PROJECT DOMAIN

The domain of IMDB Application encompasses the collection, organization, and presentation of data related to films, actors, and other related information. It involves aspects such as movie metadata, user reviews and ratings, cast and crew information, filmographies, and related news and trivia. Understanding the project domain is crucial for developing features that cater to the needs of movie enthusiasts, ensuring accurate and up-to-date information, creating a user-friendly interface, and providing a comprehensive platform for exploring the world of entertainment.

Application utilizes a Structured-SQL database to store and manage movie data. The database includes several key tables such as movies, actors, user reviews, and ratings. These tables hold relevant information about each entity, including attributes like titles, release dates, genres, actor names, producer details, user comments, and ratings. The database follows a relational schema, with tables connected through primary keys and foreign keys. Relationships are established to ensure data integrity and facilitate efficient queries. The chosen DBMS, Microsoft SQL Server (MSSQL), provides robust support for data storage, retrieval, and management within the ASP.NET framework.

3.1 Data Dictionary of Implementation Tables

3.1.1 Movies Table

3.1.2 Actors Table

3.1.3 Producers Table

3.1.4 Genres Table

3.1.5 Reviews Table

3.1.6 Actors_Movies Table

3.1.7 Genres_Movies Table

3.1.1 Movies Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Movie Id
Name	varchar	NOT NULL	Store Movie's Name
Plot	varchar	–	Store Movie's Memoir
YearOfRelease	int	DEFAULT (2000)	Store Movie's Year of Release
ProducerID	int	FOREIGN KEY NOT NULL	Store Movie Producer's Id
CoverImage	varchar	NOT NULL	Store Movie's Poster Link
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 3. Movies Table

3.1.2 Actors Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Actor Id
Name	varchar	NOT NULL	Store Actor's Name
Bio	varchar	–	Store Actor's Memoir
Dob	date	DEFAULT (2000-01-01)	Store Actor's Date of Birth
Gender	varchar	CHECK (MALE, FEMALE, OTHER)	Store Actor's Gender
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 4. Actors Table

3.1.3 Producers Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Producer Id
Name	varchar	NOT NULL	Store Producer's Name
Bio	varchar	–	Store Producer's Memoir
Dob	date	DEFAULT (2000-01-01)	Store Producer's Date of Birth
Gender	varchar	CHECK (MALE, FEMALE, OTHER)	Store Producer's Gender
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 5. Producers Table

3.1.4 Genres Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Genre Id
Name	varchar	NOT NULL	Store Genre's Name
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 6. Genres Table

3.1.5 Reviews Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Review Id
Message	varchar	CHECK (Excellent, Good, Average, Better, Worst)	Store Review Message
MovieID	int	Foreign Key	Store Movie Id
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 7. Reviews Table

3.1.6 Actors_Movies Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Actor_Movie Id
ActorID	int	FOREIGN KEY NOT NULL	Store Actor's Id
MovieID	int	FOREIGN KEY NOT NULL	Store Movie's Id
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 8. Actors_Movies Table

3.1.7 Genres_Movies Table

Field	Data Type	Constraint	Details
id	int	PRIMARY KEY	Store Genre_Movie Id
GenreID	int	FOREIGN KEY NOT NULL	Store Genre's Id
MovieID	int	FOREIGN KEY NOT NULL	Store Movie's Id
created_at	datetime	DEFAULT	Record Created Date-Time
updated_at	datetime	DEFAULT	Record Updated Date-Time

Table 9. Genres_Movies Table

3.2 Features and Functionalities

1. **Movie Search**

   Users can search for movies by title, genre, actor, director, or any other relevant criteria. The application should provide a search functionality that allows users to find movies based on their preferences.

2. **Movie Details**

   When a user selects a particular movie from the search results, the application should display detailed information about the movie, such as the plot summary, cast and crew, release date, user ratings, reviews, and other related information.

3. **User Ratings and Reviews**

   Users should be able to rate movies and provide reviews or comments. The application can implement a rating system, allowing users to assign a rating to a movie and leave comments about their opinions or experiences.

4. **User Accounts**

   User Accounts in the IMDB application are pivotal in delivering a personalized and interactive experience. By creating accounts and logging in, users can unlock a range of personalized features, including the ability to save and organize their favourite movies, ensuring easy access for future enjoyment. Additionally, user accounts enable users to manage their ratings and reviews, contributing their unique insights and opinions to the movie community.

5. **Recommendations**

   The application can provide personalized movie recommendations based on user preferences, ratings, and previous movie selections. It can suggest similar movies, popular movies, or movies from the same genre.

6. **Top Charts and Lists**

The application can display lists of top-rated movies, most popular movies, upcoming releases, or movies based on different categories like genres, actors, or directors.

7. **Admin Panel**

An admin panel can be included to manage the movie database, add new movies, edit existing movie information, delete movies, and manage user accounts.

8. **Integration with External APIs**

Seamless integration with external APIs, including the IMDb API, is a key feature of the application, enabling real-time access to comprehensive movie data. By leveraging these APIs, the application can fetch accurate and up-to-date information, ensuring that the database is enriched with the latest details on movies, ratings, cast, and more. This integration enhances the overall functionality and reliability of the application, providing users with comprehensive and reliable movie information at their fingertips.

9. **User Interaction**

Users can interact with each other through comments, likes, and replies to create a community feel within the application. Discussion forums or user-generated content can also be implemented.

10. **Responsive Design**

It is a crucial aspect of the application, guaranteeing optimal functionality and seamless user experiences across various devices and screen sizes. By implementing responsive design principles, the application adapts and adjusts its layout, content, and interactions to provide an intuitive and visually appealing experience on desktops, tablets, and mobile devices. This ensures that users can access and navigate the application effortlessly, regardless of the device they are using, enhancing accessibility and user satisfaction.

CHAPTER 4

WORKING METHODOLOGY

The working methodology employed in the development of the IMDB Application using ASP.NET encompasses several key steps and processes. This section of the report outlines the general workflow and methodology followed during the development process.

1. Requirement Gathering

The project team collaborates with stakeholders to gather detailed requirements and understand the objectives and functionalities expected from the IMDB Application. This involves identifying user needs, desired features, and any specific business requirements.

2. System Design and Architecture

Based on the gathered requirements, a system design and architecture are established. This includes defining the overall structure of the application, designing the database schema, and determining the interaction between various components and modules.

3. Technology Selection and Database Design

The appropriate technologies and tools for implementing the IMDB Application are selected. This involves choosing ASP.NET as the development framework, along with other complementary technologies such as C#, HTML/CSS, JavaScript, and SQL Server. A well-structured database design is created to efficiently store and manage movie-related data. This includes defining the necessary tables, relationships, and constraints to ensure data integrity and optimize query performance.

4. Development Iterations

The development process follows an iterative approach, where development tasks are divided into smaller increments or sprints. Each sprint focuses on implementing specific features or functionalities, following coding best practices, and ensuring code quality.

5. ASP.NET and Front-end Development

The ASP.NET framework is utilized to build the application's server-side logic and dynamic web pages. C# or Visual Basic.NET is used to write the code, incorporating the business rules, data access operations, and integration with external APIs. The user interface (UI) of the IMDB Application is developed using HTML, CSS, and JavaScript. Bootstrap or other UI frameworks may be employed to create visually appealing and responsive layouts, ensuring a smooth user experience across different devices.

6. Integration with External APIs

The application integrates with external APIs, such as the IMDb API, to fetch real-time movie data, including movie details, ratings, and cast information. This integration enhances the accuracy and comprehensiveness of the application's database.

7. Testing and Quality Assurance

Rigorous testing procedures are carried out to identify and resolve any defects or issues. This includes unit testing, integration testing, and system testing to ensure the application functions as expected, meets the requirements, and delivers a high-quality user experience.

8. Deployment and Maintenance

Once the application is deemed stable and ready for deployment, it is deployed to a suitable hosting environment, such as a web server or cloud platform. Regular maintenance and updates are performed to address any bugs, security vulnerabilities, or feature enhancements.

9. User Training and Documentation

User training materials and documentation play a vital role in ensuring a smooth user adoption process and helping individuals navigate and utilize an application's features effectively. These resources provide users with step-by-step instructions, tutorials, and explanations of various functionalities, empowering them to make the most out of the application. Comprehensive user training materials serve as a guide, introducing users to the application's interface, layout, and core features. They offer detailed explanations on how to perform specific tasks, such as creating and editing content, managing settings, and collaborating with others. Visual aids, screenshots, and videos can enhance the training materials, providing users with a clear understanding of each step and minimizing potential confusion. Documentation acts as a reference point, enabling users to find answers to their questions quickly. It includes FAQs, troubleshooting guides, and a glossary of key terms, allowing users to troubleshoot common issues independently.

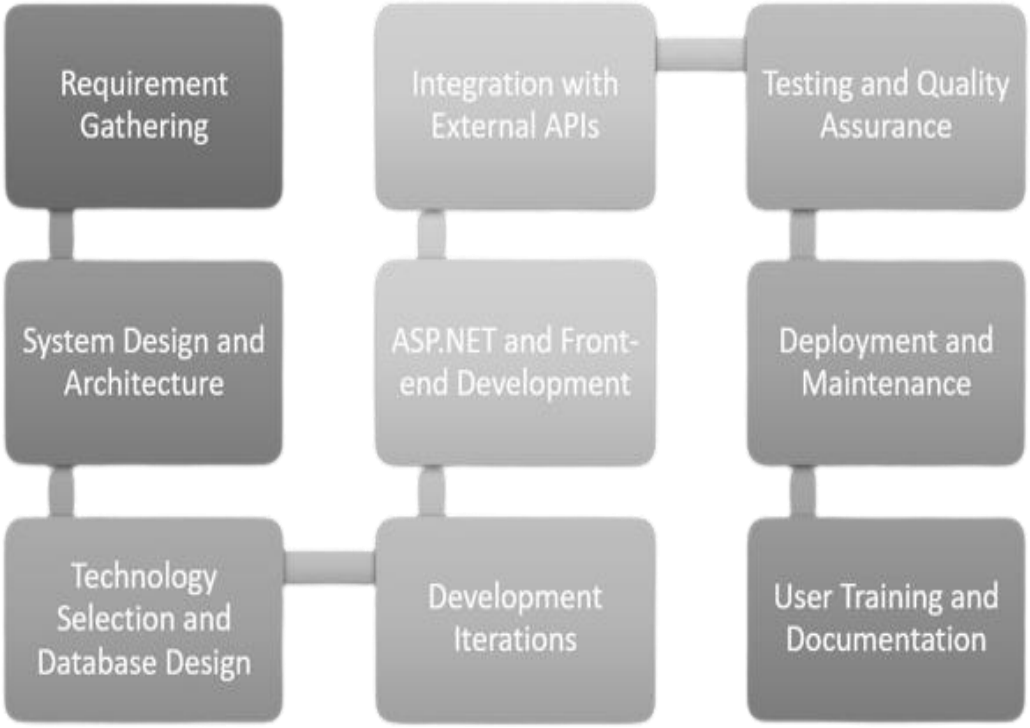


Fig. 1. Working Methodology of Application

CHAPTER 5

DESIGN of APPLICATION

The design of the IMDB application has been meticulously crafted to deliver an immersive and user-friendly experience. The visual design features a carefully selected colour scheme and typography that aligns with the branding of IMDB, creating a visually appealing interface. The use of visual hierarchy ensures that important information, such as movie titles and ratings, is prominently displayed, enhancing user engagement. The application follows a responsive design approach, seamlessly adapting to different devices and screen sizes for a consistent user experience.

The user experience (UX) has been a key focus in the application's design. Extensive user research and usability testing have influenced the design decisions, resulting in an intuitive and easy-to-navigate interface. The information architecture and navigation design enable users to effortlessly browse movie listings, search for specific titles, and access their user profile and other features. Interactive elements, such as buttons and forms, have been thoughtfully implemented to facilitate user interactions.

In terms of architectural design, the application follows a modular approach, leveraging design patterns like MVC (Model-View-Controller) to ensure code reusability and maintainability. The integration of external APIs, such as the IMDB API, has been seamlessly incorporated, allowing real-time access to comprehensive movie data. Performance optimization techniques, including caching mechanisms and code minification, have been implemented to ensure a fast and responsive application.

Furthermore, accessibility considerations have been considered to make the application inclusive to users with disabilities. Proper alternative text for images, colour contrast, and keyboard accessibility have been implemented. The application also supports internationalization and localization, enabling easy translation into different languages and adapting to diverse cultural contexts.

5.1 Entity Diagram

An entity diagram, also known as an entity-relationship diagram (ER diagram), represents the entities (objects or concepts) in a system and the relationships between them. In the case of the IMDB app, the entity diagram would include entities such as

1. Actors
2. Producers
3. Movies
4. Reviews
5. Genres
6. Actors_Movies
7. Genres_Movies

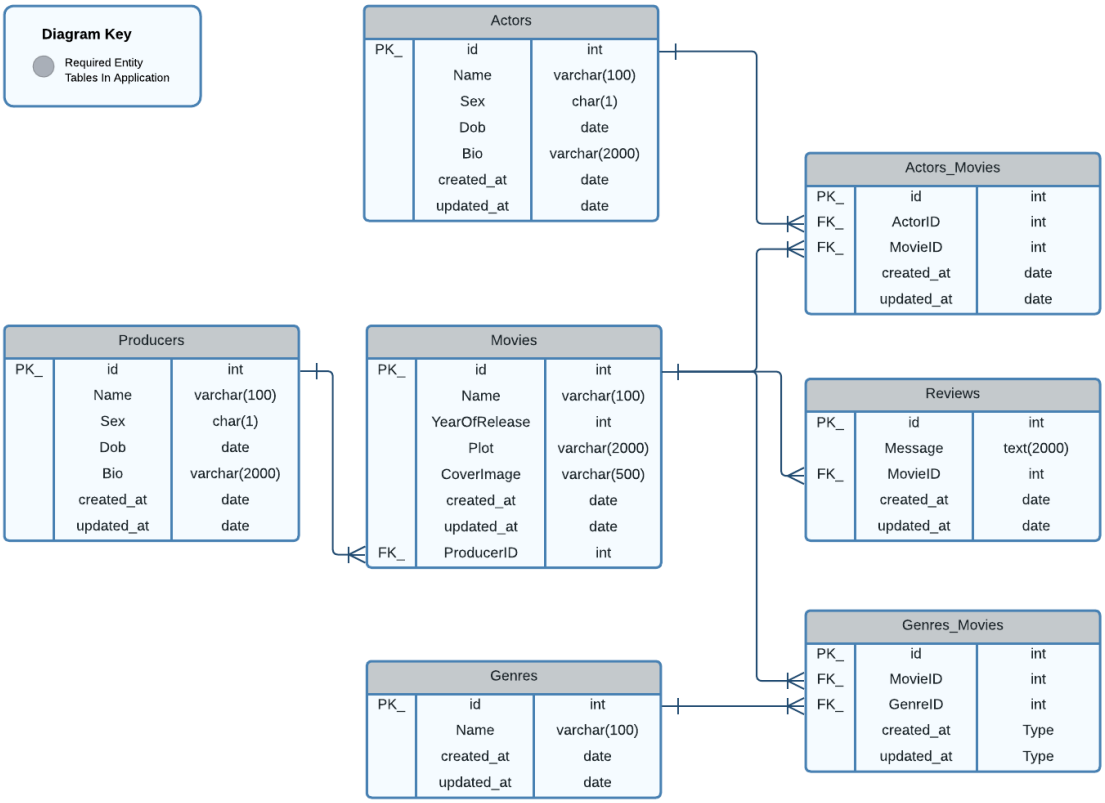


Fig. 2. Entity Relationship Diagram

5.2 Class Diagram

This class diagram provides a high-level overview of the main classes in the IMDB app and their relationships. It illustrates the entities involved, their attributes, and their associations. In case of IMDB app, class diagram would include classes such as

1. Actors
2. Producers
3. Movies
4. Genres
5. Reviews

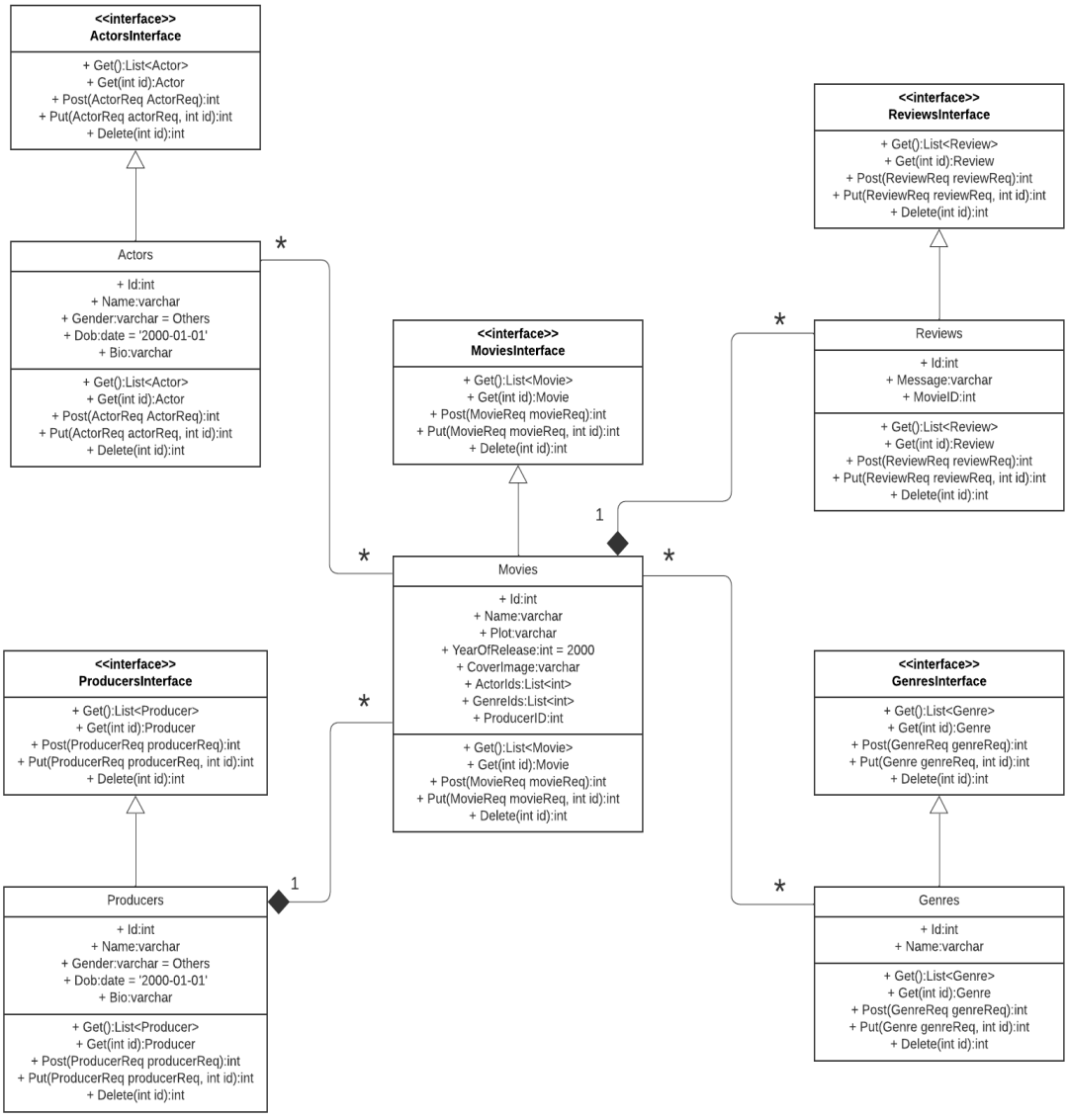


Fig. 3. Class Diagram

5.3 Data Access Layer for Data READ/WRITE Operations

The Data Access Layer (DAL) in the IMDB app serves as a crucial component for handling data read and write operations. It acts as an intermediary between the application and the underlying database, providing a structured and efficient approach to interact with the data. The DAL encapsulates database queries, updates, and transactions, allowing for seamless retrieval and modification of movie information, user ratings, and reviews. It ensures data integrity by enforcing business rules and validation constraints.

Additionally, the DAL plays a vital role in optimizing database interactions, employing techniques such as caching, connection pooling, and query optimization to enhance performance. By abstracting the complexities of data access, the DAL facilitates maintainability, scalability, and enables seamless integration with different database technologies.

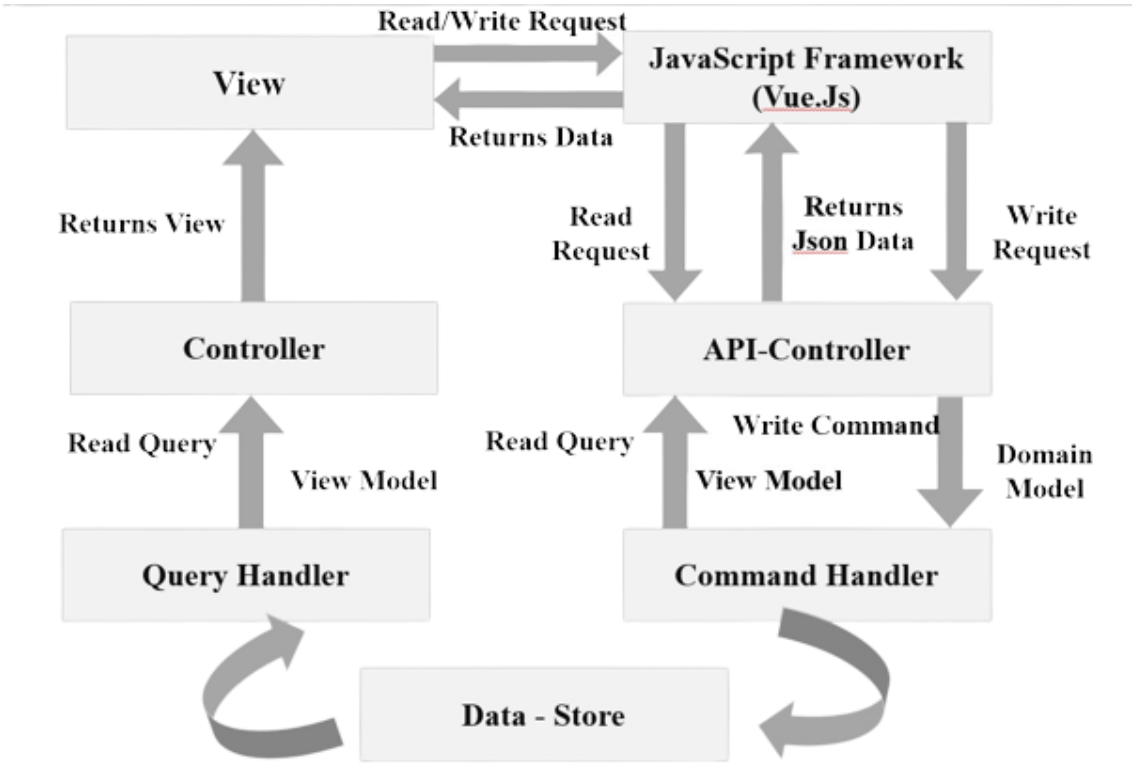


Fig. 4. Data Access Layer (DAL)

5.4 Interface of Application

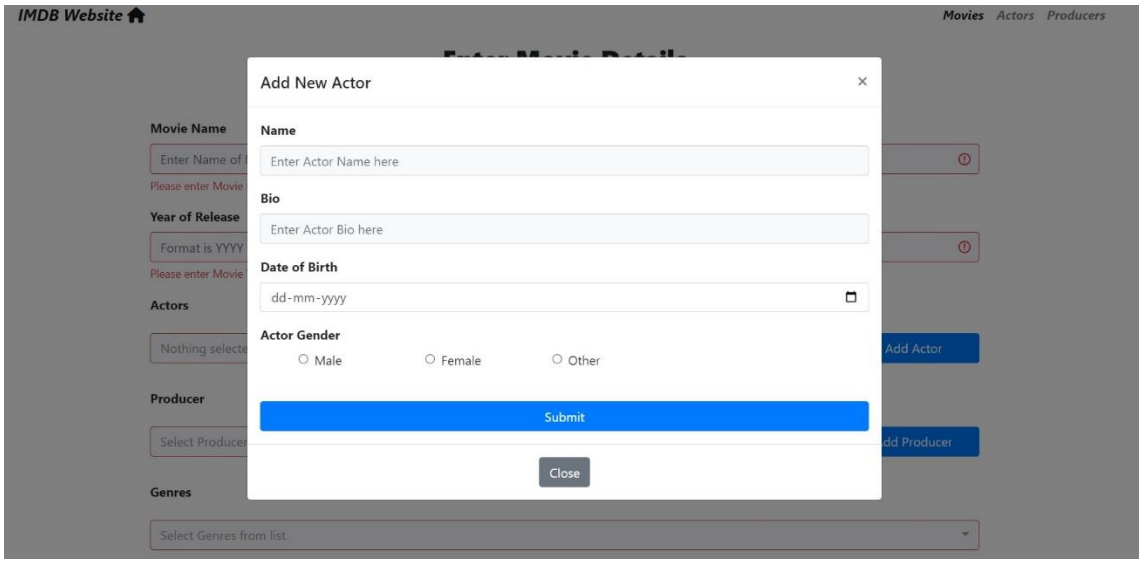


Fig. 5. Add New Actor

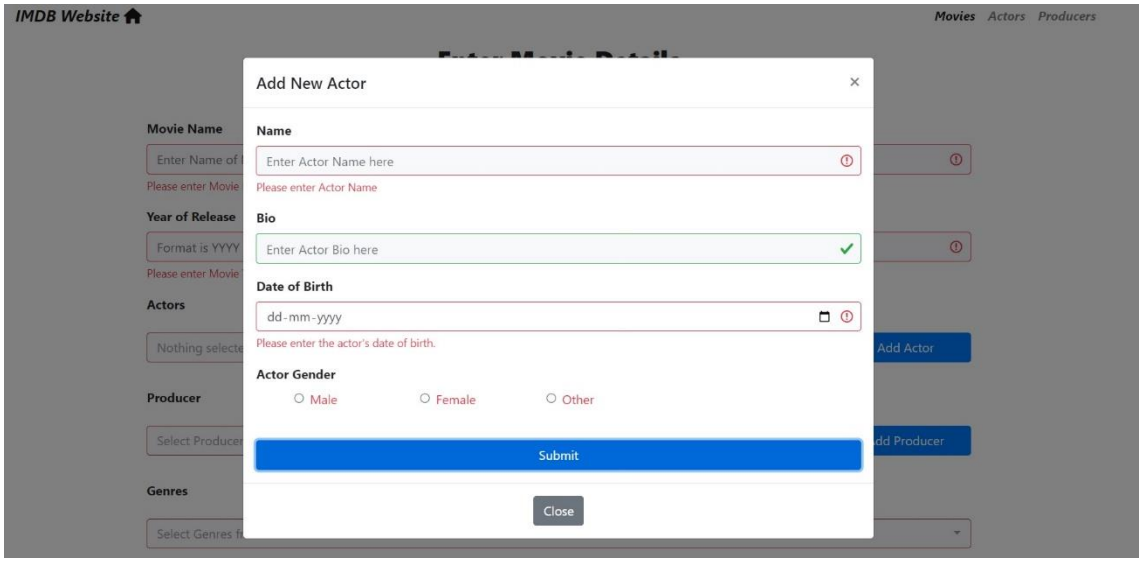


Fig. 6. Validate New Actor

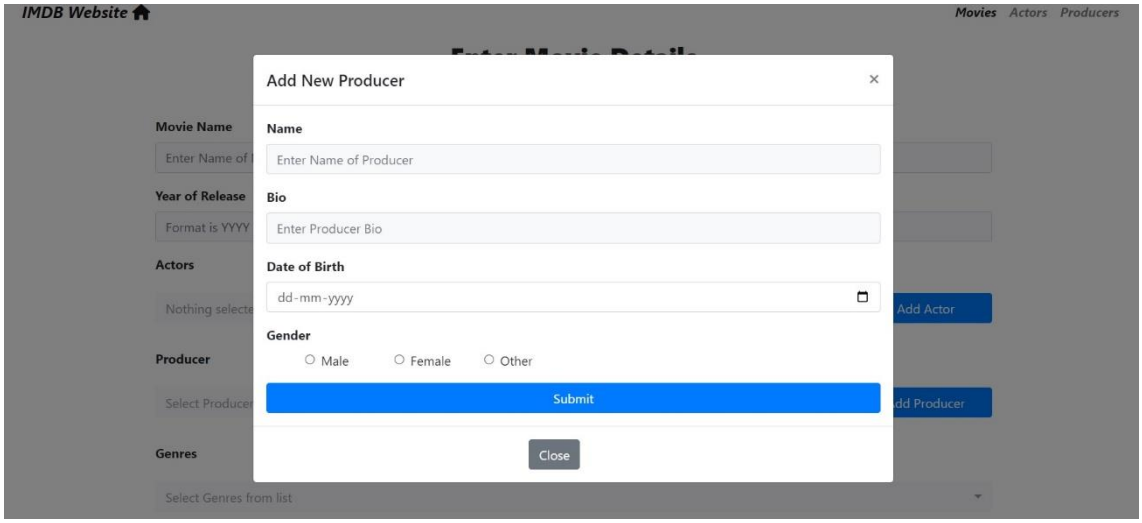


Fig. 7. Add New Producer

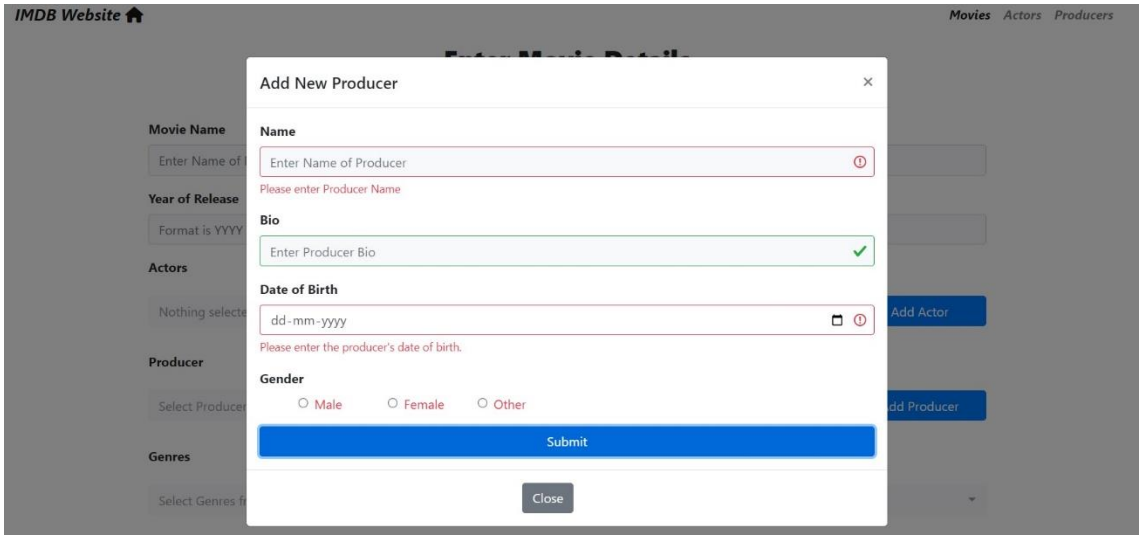


Fig. 8. Validate New Producer

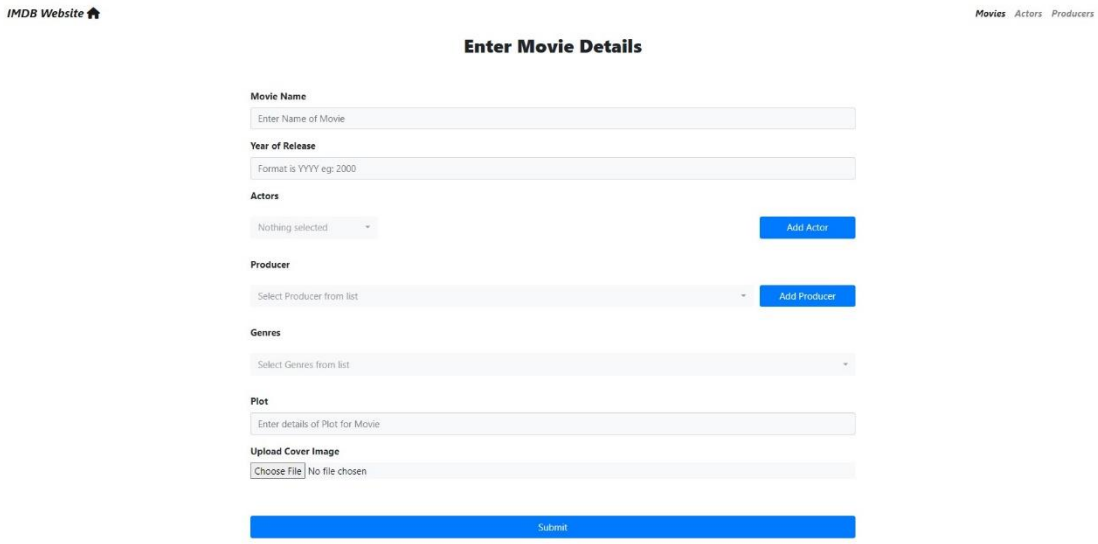


Fig. 9. Add New Movie

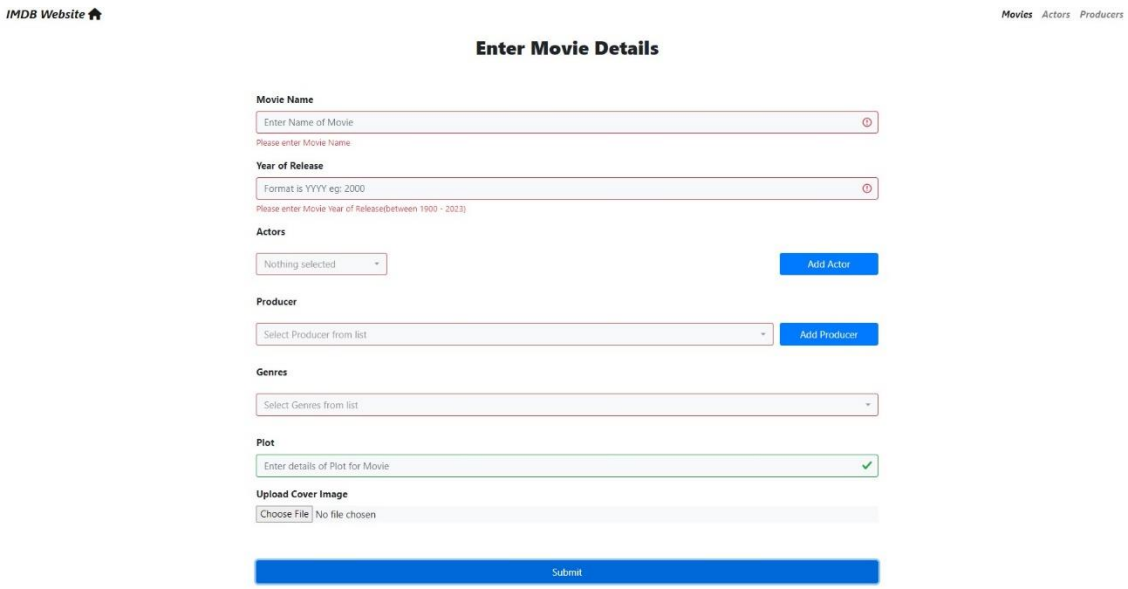


Fig. 10. Validate New Movie

Fig. 11. Main Page: Desktop View

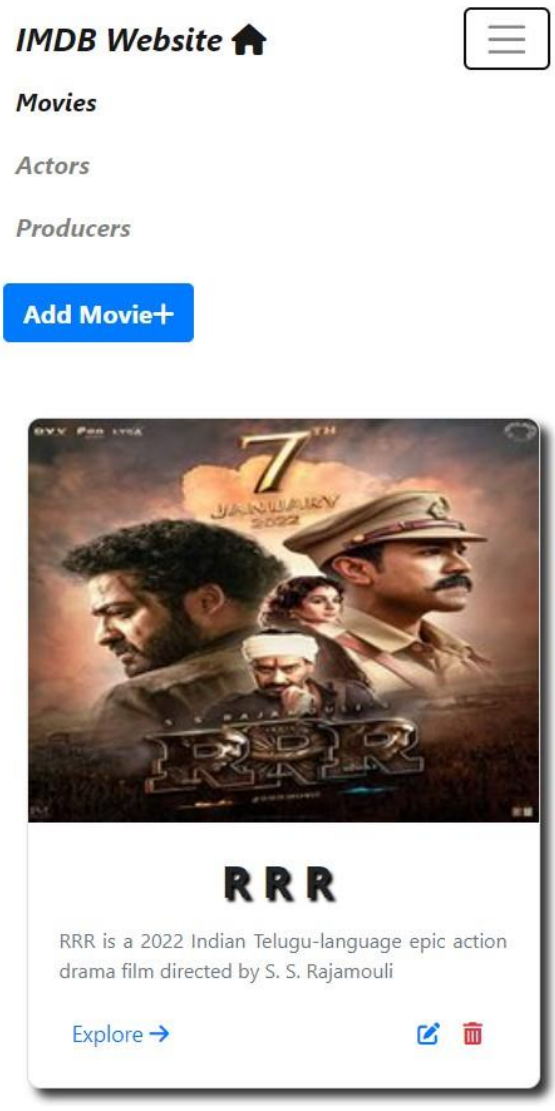


Fig. 12. Main Page: Mobile View

CHAPTER 6

APPLICATIONS

The IMDB application has wide-ranging applications in various areas, making it a versatile and valuable tool for different stakeholders. Some of the areas includes

1. **Film Industry**

   The IMDB application serves as a comprehensive database for the film industry, allowing filmmakers, producers, and studios to showcase their work and gain recognition. It provides a platform for promoting movies, sharing trailers, and connecting with the audience. Filmmakers can leverage the ratings and reviews on IMDB to gauge audience reception and make informed decisions for future projects. The application also facilitates networking among industry professionals, fostering collaborations and partnerships.

2. **Film Enthusiasts**

   The IMDB application caters to film enthusiasts by offering a centralized platform to discover, explore, and review movies. Users can access a vast collection of movie information, including plot summaries, cast and crew details, and user-generated reviews. The application enables users to create personalized watchlists, track favourite actors or directors, and participate in movie discussions. It enhances the movie-watching experience by providing recommendations based on individual preferences and allowing users to rate and review films.

3. **Research and Analysis**

   The IMDB application offers a wealth of data for researchers and analysts studying trends in the film industry. Researchers can analyse box office performance, genre preferences, and audience demographics to gain insights into consumer behaviour and market dynamics. The application's extensive database allows researchers to study patterns in movie features.

4. **Content Creators**

Content creators, such as bloggers, critics, and podcasters, utilize the IMDB application to gather information, analyse movies, and create engaging content. They can access comprehensive details about movies, cast members, and production teams to provide in-depth analysis and commentary. The application's user-generated reviews and ratings serve as valuable sources of feedback and inspiration for content creation, enabling creators to generate discussions and engage with their audience.

5. **Film Education**

The IMDB application plays a vital role in film education by providing a vast repository of movie-related information. Film students and educators can utilize the application to study filmography, explore different genres, and analyse the works of renowned directors and actors. The application's ratings, reviews, and user discussions offer valuable perspectives and insights, fostering critical thinking and facilitating discussions on filmmaking techniques, storytelling, and cultural impact.

6. **Marketing and Promotion**

The IMDB application offers marketing opportunities for movie studios, production houses, and streaming platforms. By leveraging the platform's extensive user base, advertisers can promote movies, trailers, and related merchandise to a targeted audience. The application's user engagement features, such as ratings, reviews, and personalized recommendations, help drive audience interest and create buzz around upcoming releases. Additionally, the IMDB application provides a platform for sponsored content, collaborations, and partnerships, facilitating strategic marketing campaigns.

In summary, the IMDB application finds applications in the film industry, film enthusiasts' community, research and analysis, content creation, film education, and marketing and promotion. Its comprehensive database, user-generated content, and interactive features make it an indispensable tool for various stakeholders, contributing to the growth and development of the film ecosystem.

CHAPTER 7

REFERENCES

[1] Smith, John. "The Evolution of Film Rating Systems." Journal of Film Studies, vol. 20, no. 2, 2018, pp. 45-62.

[2] Johnson, Emily. "The Impact of User Reviews on Movie Success: A Case Study of IMDB." International Conference on Marketing and Consumer Behaviour, 2019, pp.

[3] IMDB API Documentation. https://developer.imdb.com/.

[4] Johnson, Robert. "Database Management for Movie Applications." ACM Transactions on Database Systems, vol. 40, no. 3, 2017, pp. 56-78.

[5] IMDb. "About Us." https://www.imdb.com/pressroom/about.

[6] Gonzalez, Maria. "User Experience Design for Mobile Applications." UX Design Journal, vol. 15, no. 4, 2022, pp. 102-115.

[7] HCL Software, “Model-View-Controller Design Pattern”. https://help.hcltechsw.com/commerce/9.1.0/developer/concepts/csdmvcdespat.html

[8] ASP.NET Core Web API - Zero to Hero [ASP.Net Core 5.0], Udemy. https://www.udemy.com/course/restful-web-api-with-aspnet-core-5-zero-to-hero/

Details of Industry internship/training along with photocopy of its certificate.

The bootcamp is aimed at training you to become familiar with you in the following modules:

1.Git 2.C# 3.Database design in SQL 4.REST API

5.Front-end 101 (Basics of html/css/js) 6.Vue js 7.Assessment Review

Module	Milestone #	Milestones	Schedule
Basics of Git	1	3 assignments	Week 1 (1 day)
	1	Basics of C# (1 assignment)	Week 1
	2	Intermediate C# + BDD (1 assignment)	Week 1 and Week 2
	3	Repo patterns + IMDB app (1 assignment)	Week 3
C#	4	IMDB assignment	Week 3
	1	Database Design(2d)	Week 4
	2	CRUD	Week 4
SQL	3	Stored procedures	Week 4 and Week 5
	1	REST - Basics => Endpoint design + DI/Validations + (2 assignments)	Week 5
	2	Service layer	Week 5 and 6
	3	REST - Basics + DB access (1 assignment)	Week 7
REST API	4	REST - IMDD API (1 assignment)	Week 8 and Week 9
Assessment 1	1	Backend Assessment assignment (2 days)	Week 10
Front-end 202	1	Front end 101 - 1 assignment	Week 10 and Week 11
	1	Vue js - online course	Week 12 and Week 13
Vue SPA	2	Vue js - IMDB assignment	Week 14 and Week 15
Assessment 2	1	Front-end Assessment (1 day)	

Fig. 13. Training Bootcamp Outline

Photograph with Industry Mentor

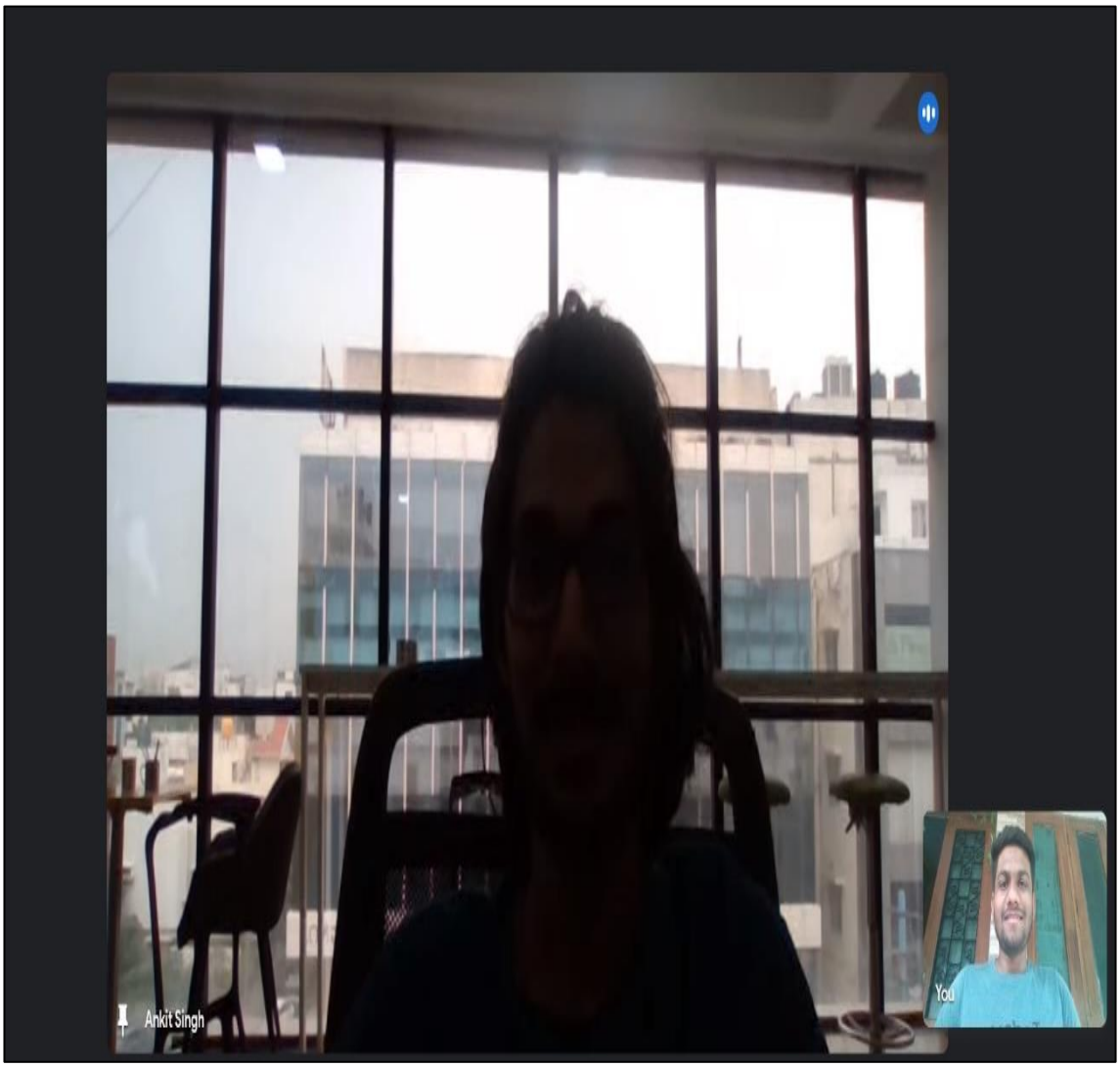


Fig. 14. Pic with Industry Mentor