MEGHNAD SAHA INSTITUTE OF TECHNOLOGY

Behind Urbana Complex, Near Ruby General Hospital Anandapur Rd, Ucchepota Kolkata, West Bengal 700150



MINOR PROJECT REPORT

### OTT PLATFORM

**MAKAUT ODD SEMESTER 2023-24**

[BACHELOR OF COMPUTER APPLICATION]

**MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY**

(Formerly known as WEST BENGAL UNIVERSITY OF TECHNOLOGY)

**UNDER THE SUPERVISION OF**

**Mr. Soumya Chakravarty**

Acknowledgement

**I would like to extend my sincere and heartfelt thanks towards all those who have helped me in making this. project. Without their active guidance, help, cooperation and encouragement, I would not have been able to present the project on time.**

**I extend my sincere gratitude to Aniruddha Sir and Subject teacher for their moral support and guidance during the tenure of my project.**

**I also acknowledge with a deep sense of reverence, my gratitude towards my parents and other faculty members of the school for their valuable suggestions given to me in completing the project.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Abhiraj Sardar** | 31001221027 |  |  |
| **Michael Gomes** | 31001221039 |  |  |
| **Rupak Sarkar** | 31001221010 |  |  |
| **Suman Roy** | 31001221018 |  |  |
| **Snehasish Bhattacharjee** | 31001221045 |  |  |



**MEGHNAD SAHA INSTITUTE OF TECHNOLOGY**

Behind Urbana Complex, Near Ruby General Hospital Anandapur Rd, Ucchepota Kolkata, West

Bengal 700150

### CERTIFICATE

**“OTT PLATFORM**” project that is being presented as BCA MINOR PROJECT 2023 of the requirement for the award of the **Bachelor of Computer Application** and submitted to the Department of BCA of Meghnad Saha Institute of Technology is an authentic record of my own work carried out during a period from **July 2023 to September 2023(5th semester)** under the supervision of “Mr. Soumya Chakravarty “, BCA Department.

**Signature Of Students**

Abhiraj Sardar Michael Gomes

Rupak Sarkar Suman Roy

Snehasish Bhattacharjee

**Date: 6/12/23**

**Signature Of Guide Mr.Soumya Chakravarty**

**Department of BCA**

**Head of the Department DR. Aparna Dutta**

**HOD-BCA Signature of External Examiner**

**CONTENTS**

**Page No.**

**Chapter:** About java, jsp and servlet 6-8

**Chapter:** Introduction of Project…………………………………………………………….8-10

|  |  |
| --- | --- |
| **Chapter** | Scope of Project 10-12 |
| **Chapter** | System and Software Requirements 12 |
| **Chapter** | Software Development Life–cycle 12-14 |
| **Chapter** | Data Flow Diagram and ERD 14-22 |
| **Chapter** | Tables used in Database 23-24 |
| **Chapter**  **Chapter** | UI Screenshots and Source code 25-40  Testcases of project 40-42 |
| **Chapter** | Conclusion 43 |

### ABOUT OF JAVA AND JSP:

**Java and JSP** (Java Server Pages) are two important technologies used in web development. They are often used together to create dynamic and interactive web applications.

### FEATURES:

###### Object-Oriented Programming (OOP):

Java is a fully object-oriented programming language. Everything in Java is an object, and it follows OOP principles such as encapsulation, inheritance, and polymorphism.

###### Strongly Typed and Statically Typed:

Java is strongly typed, which means that variable types are strictly defined and enforced. It's also statically typed, meaning variable types are checked at compile-time, reducing the chances of type-related errors at runtime.

###### Rich Standard Library (Java API):

Java provides a vast standard library known as the Java API (Application Programming Interface). This library includes classes and methods for various tasks like I/O, networking, data structures, and more. Developers can leverage these APIs to streamline development.

###### Garbage Collection:

Java includes an automatic garbage collection mechanism that manages memory by deallocating objects that are no longer in use. This helps developers avoid memory leaks and makes Java programs more reliable.

### *JSP AND SERVLET

**JSP (Java Server Pages) and Servlets** are both Java technologies used for developing web applications, and they often work together to create dynamic web content. Here's an overview of each technology and how they relate to one another:

**JSP is a technology** that simplifies the process of creating dynamic web content by allowing Java code to be embedded within HTML pages.

###### Mixing Java and HTML:

In a JSP, you can include Java code within special tags, typically enclosed in

<% and %>. This allows you to generate dynamic content and execute Java logic directly within the HTML page.

###### View Layer:

JSPs are often used as the presentation or view layer in a web application. They are responsible for rendering the HTML that is sent to the client's browser.

###### Simplifying Dynamic Content:

JSPs make it easier to generate dynamic content without having to write extensive Java code within servlets. This separation of presentation and logic improves code readability and maintainability.

###### Expression Language (EL):

JSP includes an Expression Language (EL) that simplifies the embedding of dynamic data into HTML templates. EL expressions are enclosed in ${} and can be used to retrieve and display data from Java objects.

###### SERVLET:

Servlets are Java classes that extend the capabilities of a web server. They receive and respond to HTTP requests from clients (usually web browsers) and are a part of the server-side processing in a web application.

###### Lifecycle:

Servlets have a well-defined lifecycle consisting of methods like init(), service(), and destroy(). The service() method is particularly important, as it handles incoming requests and generates responses.

###### Processing Requests and Responses:

Servlets are responsible for handling the core logic of a web application. They can generate dynamic content, interact with databases, and perform various operations based on the incoming request parameters.

###### Flexibility and Control:

Servlets provide developers with fine-grained control over the HTTP request and response cycle. They can be used to implement custom request handling and complex business logic.

### HOW IT WORKS:

Servlets and JSPs are often used together in a web application. Servlets handle the core logic, such as processing requests, interacting with databases, and managing session data. JSPs, on the other hand, focus on the presentation layer by rendering the

HTML content.

When a client sends an HTTP request to a web application, a servlet can process the request, perform business logic, and set attributes in the request or session. The servlet can then forward or redirect the request to a JSP for rendering the response.

### INTRODUCTION:

Traditionally, the consumption of movies and other audio and video content has always been in the form of mediums like theatre and television. As the technology developed, it was easily accessible at home and whenever required with the introduction of VHS, DVDs, Blue-rays and disc rental services. Further, cable television brought the content through Co-axial cables and fibre optic cables. Another better service emerged as Direct-to-

home (DTH) technology through satellite and dish connectivity that brought high-quality broadcast and on-demand content directly to the consumer. Recently, technological advancements have made the movie or TV watching more convenient through online streaming or Video on Demand (VoD)services. VoD refers to streaming of video content over the Internet, through applications typically referred to as Over-The-Top (OTT). Viewers can access video content through OTT apps in any Internet-connected device like a Smartphone, smart TV, tablet, desktop computer, laptop, etc. Unlike traditional media, streaming services tell varied stories that are not restricted by censors, box office or demographic. It gives a viewing experience with greatly improved sound and visual quality, provided consumers have a stable Internet.

**The motivation and objectives** of an OTT (Over-The- Top) platform project can vary depending on the specific goals and market dynamics, but here are some common motivations and objectives:

**Access to a Global Audience:** One of the primary motivations is to reach a global audience. OTT platforms can deliver content to users anywhere in the world, enabling content creators to expand their reach beyond geographical limitations.

**Content Monetization**: OTT platforms aim to generate revenue through various monetization models such as subscription fees, advertising, or pay-per-view. The objective is to create a sustainable revenue stream from content consumption.

**Diverse Content Catalog:** To cater to a wide range of audience preferences, OTT platforms often aim to offer a diverse catalog of content, including movies, TV shows, original productions, documentaries, and more.

**Personalization and Recommendation**: Many OTT platforms focus on providing personalized content recommendations to enhance user engagement. The objective is to keep users engaged and returning for more content.

**High-Quality User Experience**: Creating a user-friendly interface and ensuring high-quality streaming experiences are common objectives.

Users expect seamless playback, excellent video and audio quality, and easy navigation.

### OUTLINE:

#### User and Admin Login System

* 1. **User Authentication 1**. User registration process - Collect user details (name, email, password). - 2. User login process - Secure authentication (hashing and salting passwords). - "Remember me" functionality. - Password reset mechanism.
  2. **Admin Authentication** 1. Admin account setup - Admin accounts with unique credentials. - Secure admin login credentials. 2. Admin login process

- Authentication with username and password. - Role-based access control (RBAC) for admin.

###### User and Admin Dashboards

* 1. **User Dashboard** 1. User profile management - Edit user information (name, email, password). - Subscription status and billing details.
  2. **Admin Dashboard** 1. Content management - Upload and categorize

content. - Schedule content releases. - Monitor content performance. 2. User management - View and manage user accounts. - User analytics and demographics. 3. Monetization control - Pricing adjustments for subscription plans.

###### Security and Privacy

* 1. **User Data Protection** 1. Data encryption in transit (SSL/TLS). 2. Data encryption at rest (user data storage). 3. Compliance with data protection regulations.
  2. **Admin Security**1. Strong password policies. 2. Audit logs for admin actions. 3. Regular security assessments and updates.

This outline covers the key aspects of implementing user and admin login systems for an OTT platform project, including authentication, dashboards, security, testing.

## SCOPE OF OTT PLATFORM:

**The future scope** of an OTT (Over-The-Top) platform developed using Java EE can be quite extensive, offering numerous opportunities for enhancement and expansion. Below are some potential areas for future development and improvement:

#### Content Expansion:

* + Adding more content categories such as documentaries, sports, and news. Expanding the library with regional and international content to reach a wider audience.

#### Personalization and Recommendation:

* + Implementing advanced recommendation algorithms to provide personalized content suggestions.Incorporating user feedback and preferences into the recommendation engine.

#### User Engagement:

* + Developing interactive features like live chat during shows, polls, and user-generated content sharing.Implementing gamification elements to boost user engagement, such as badges, rewards, or contests.

#### Monetization Strategies:

Exploring different monetization models, including subscription tiers, ad-supported content, and pay-per- view. Integrating payment gateways and analytics to optimize revenue generation.

#### Content Delivery Optimization:

Enhancing video streaming quality with adaptive bitrate streaming and improved buffering. Implementing Content Delivery Network (CDN) integration for better global reach.

#### Mobile and Smart TV Apps:

Developing dedicated mobile apps for Android and iOS platforms.Creating applications for popular Smart TV platforms like Roku, Apple TV, and Fire TV.

#### Offline Viewing:

Enabling users to download content for offline viewing, considering factors like content expiry and storage management.

#### Security and DRM:

Strengthening security measures to protect against piracy and unauthorized access. Implementing Digital Rights Management (DRM) solutions to safeguard content.

#### User Analytics:

Collecting and analyzing user behavior data to improve content recommendations and platform performance. Using data insights for targeted marketing and content acquisition.

**Required Languages:**

|  |  |
| --- | --- |
| **O/S** | WINDOWS 10 |
| **LOCAL SERVER** | TOMCAT |
| **DATABASE** | MYSQL |
| **USER INTERFACE** | HTML, CSS, JS,  BOOTSTRAP |
| **PROGRAMMING**  **LANGUAGE** | Java, JSP |

## SOFTWARE DEVLOPMENT LIFE CYCLE:

**SDLC** is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

1. Identify the Current Problems

This stage of the SDLC means getting input from all stakeholders, including customers, salespeople, industry experts, and programmers.

1. Plan

In this stage of the SDLC, the team determines the cost and resources required for implementing the analyzed requirements. It also details the risks involved and provides sub-plans for softening those risks.

1. Design

This phase of the SDLC starts by turning the software specifications into a design plan called the Design Specification. All stakeholders then review this plan and offer feedback and suggestions.

1. Build

At this stage, the actual development starts. It’s important that every developer sticks to the agreed blueprint. Also, make sure you have proper guidelines in place about the code style and practices.

1. Code Test

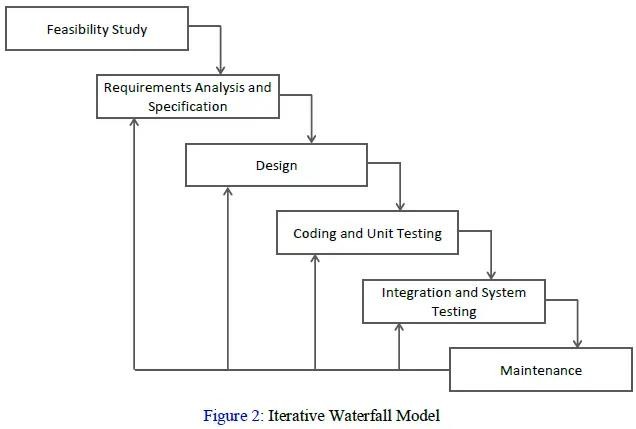
In this stage, we test for defects and deficiencies. We fix those issues until the product meets the original specifications.

1. Software Deployment

At this stage, the goal is to deploy the software to the production environment so users can start using the product.

#### Iterative Model:

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).



FEASIBILITY

A feasibility study is part of the initial design stage of any proposed project/plan. It is carried out to evaluate the feasibility of a proposed project or an existing software used by the business. It can assist in identifying and assessing the opportunities and threats present in the natural environment, the resources needed for the project, and the chances of success.

**Technical Feasibility:**

Technical Feasibility analyses and evaluates the project’s current resources, including hardware and software along with the technical requirements of the proposed system.

# Economic Feasibility**:**

When someone wants to start a new project, the cost factor is the first thing that comes to mind. Cost does not only refer to how much money is available in the company, but also where the company can seek assistance when they are in need. Economic feasibility in terms of software development helps companies to examine the development costs and financial gains.

# Legal Feasibility**:**

Legal feasibility is one of the most important types of feasibility studies in project management because, even if money is abundant, each country’s laws must allow the legal implementation of the project.

**DATA FLOW DIAGRAM:**

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually

“say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That’s why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.

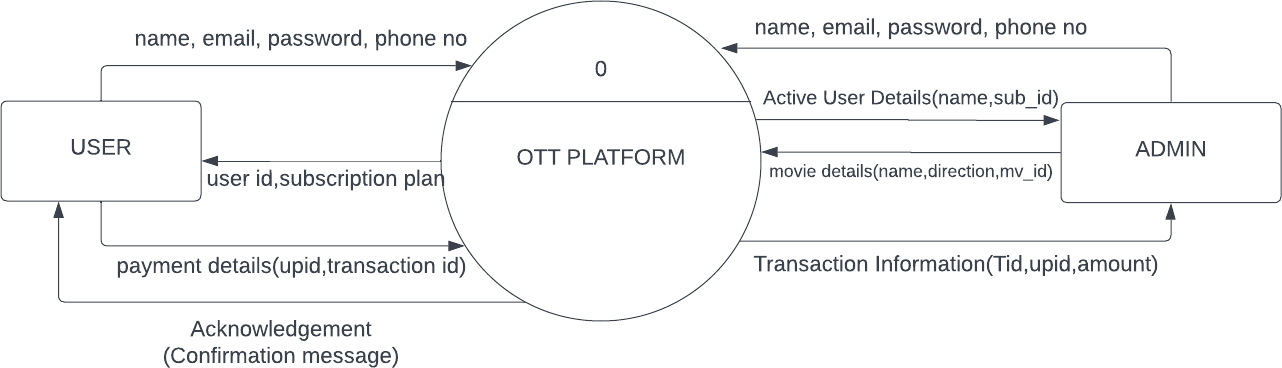
Using any convention’s DFD rules or guidelines, the symbols depict the four components of data flow diagrams.

* 1. External entity: an outside system that sends or receives data, communicating with the system being

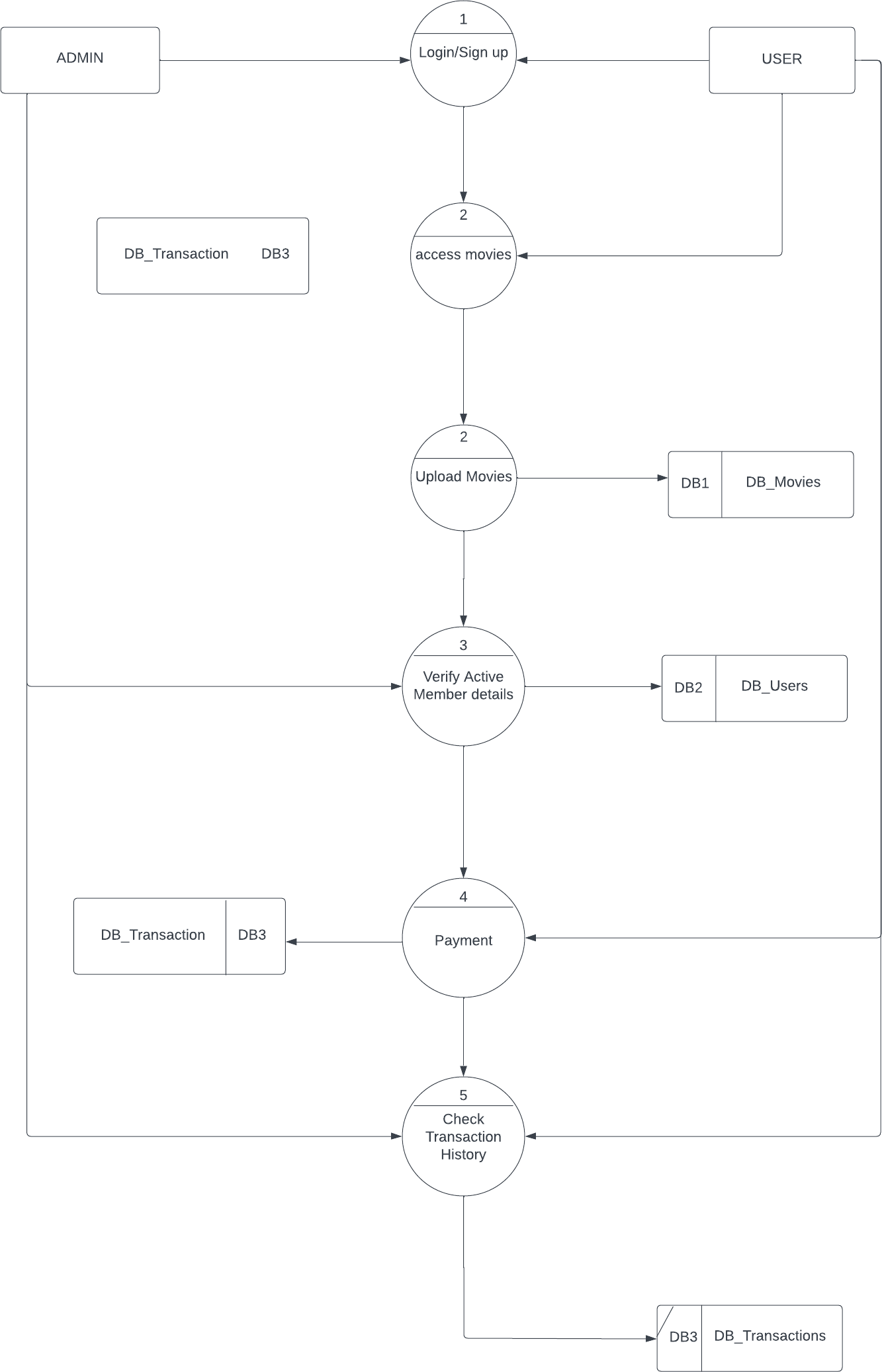
diagrammed. They are the sources and destinations of information entering or leaving the system.

* 1. Process: any process that changes the data, producing an output. It might perform computations, or sort data based on logic, or direct the data flow based on business rules. A short label is used to describe the process, such as “Submit payment.”
  2. Data store: files or repositories that hold information for later use, such as a database table or a membership form.
  3. Data flow: the route that data takes between the external entities, processes and data stores. It portrays the interface between the other components and is shown with arrows, typically labeled with a short data name, like “Billing details.”

**Context Level Diagram:** A context diagram is a high-level diagram that shows the relationship between a system and its external entities. It provides a clear and simple overview of a system and its interactions with other systems or actors.



**DFD level 1**: A level 1 DFD (Data Flow Diagram) is a diagram that shows the main sub-processes that make up a complete system. It's a more detailed version of a context diagram, which represents the entire system as a single process.



###### Entity Relationship Diagram:

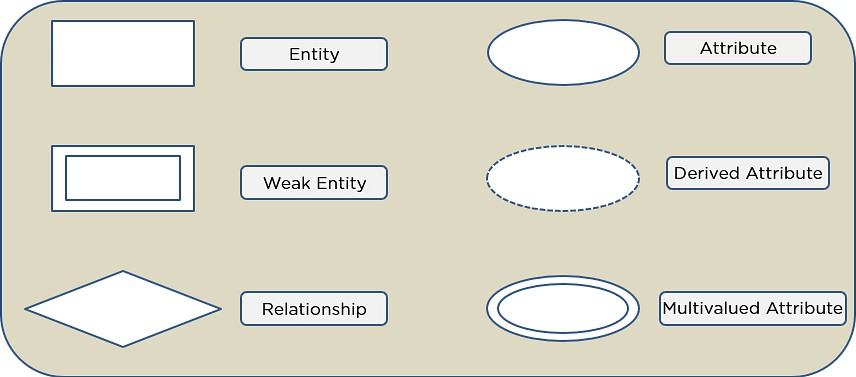
An Entity Relationship Diagram (ERD) is a visual representation

of **different entities within a system and how they relate to each other**. It is a tool used to design and model relational databases, and shows the logical structure of the database. ER diagrams use symbols to represent entities, attributes, and relationships, which help to illustrate the relationships between the entities in the database. ER

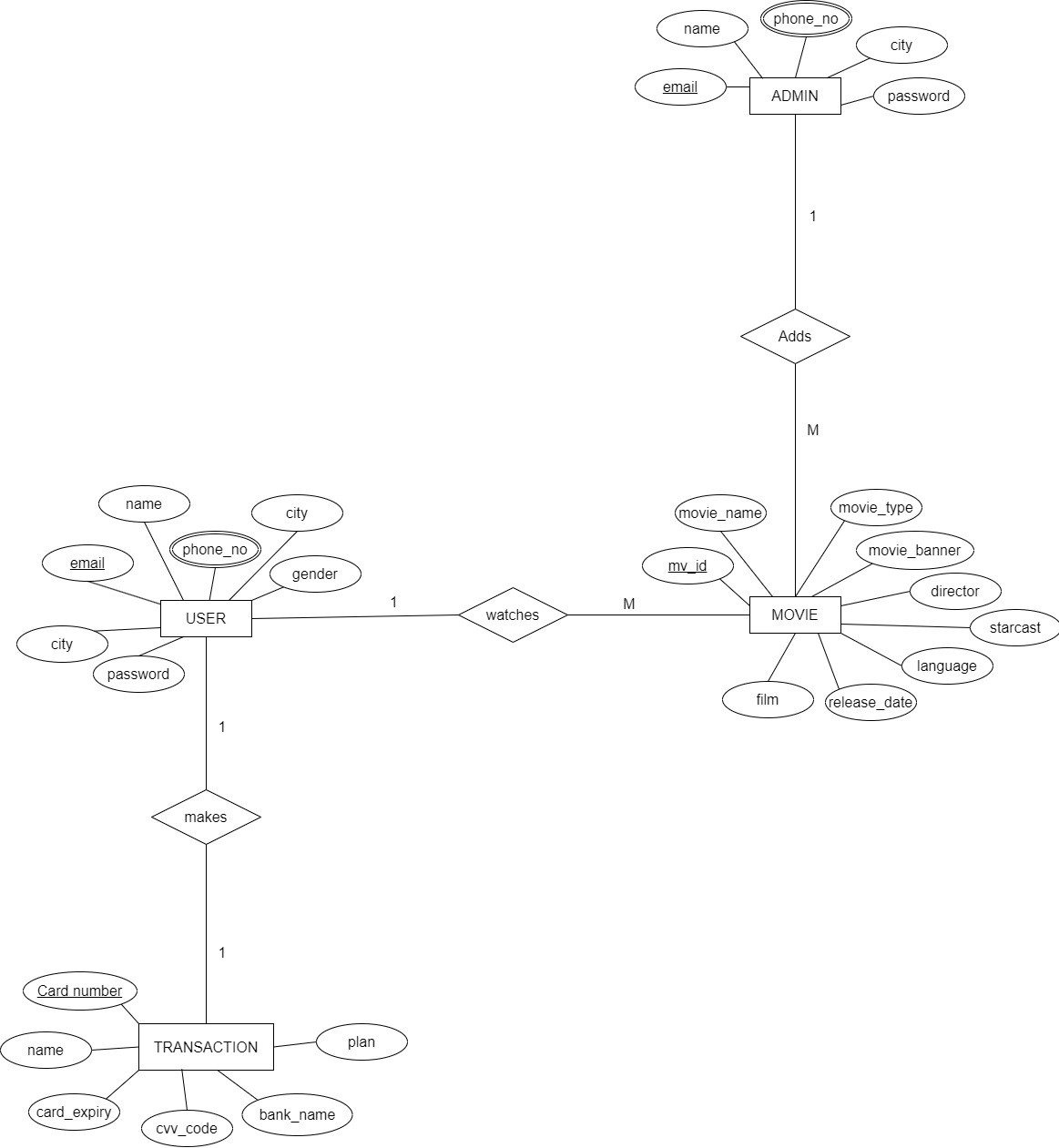
diagrams are commonly used in software engineering and database design to help developers and stakeholders understand and design complex databases.

**Symbols Used in ER Diagrams:**

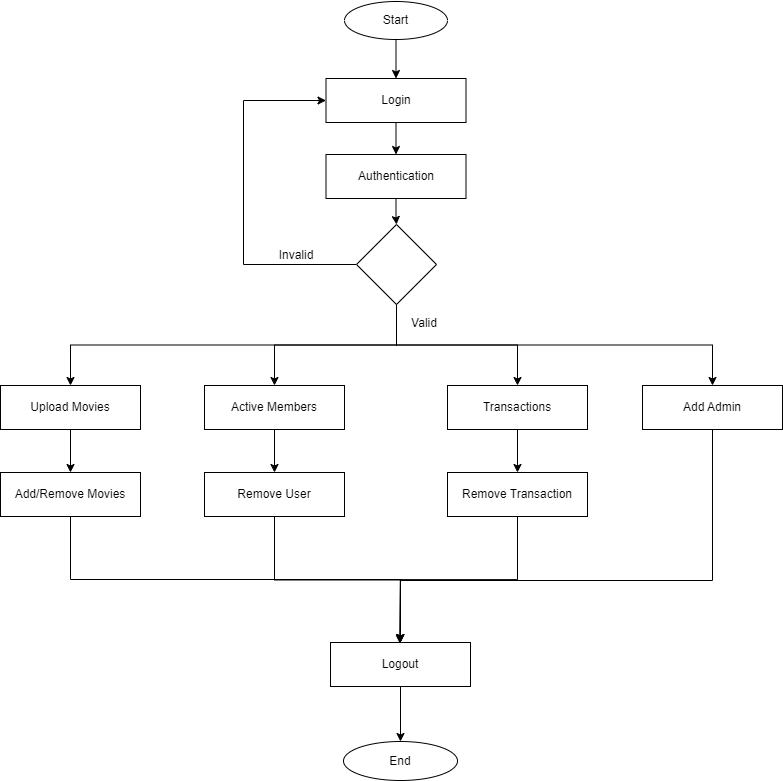
* Rectangles: This Entity Relationship Diagram symbol represents entity types
* Ellipses: This symbol represents attributes
* Diamonds: This symbol represents relationship types
* Lines: It links attributes to entity types and entity types with other relationship types
* Primary key: Here, it underlines the attributes
* Double Ellipses: Represents multi-valued attributes



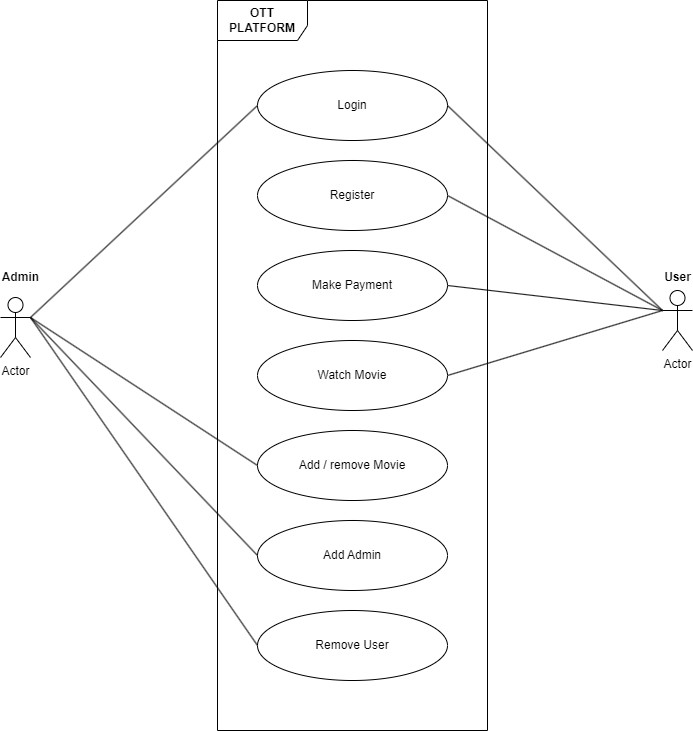
##### ERD Diagram of Project:



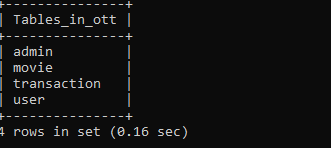
**ADMIN UML DIAGRAM:**



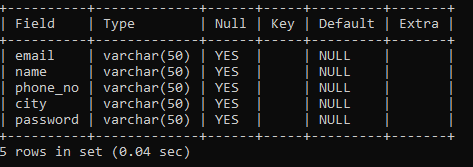
### USER UML DIAGRAM:



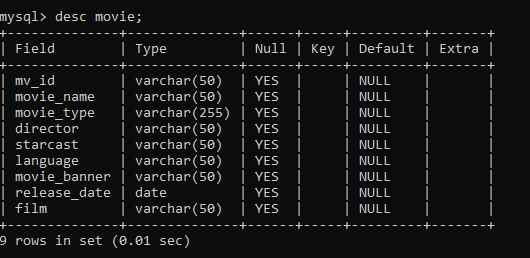
#### Database Schemas:



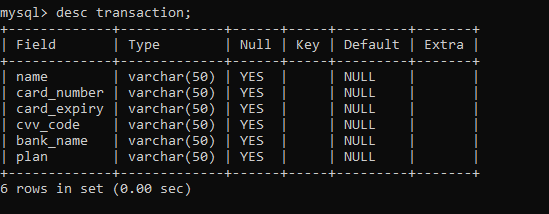
Admin Table:



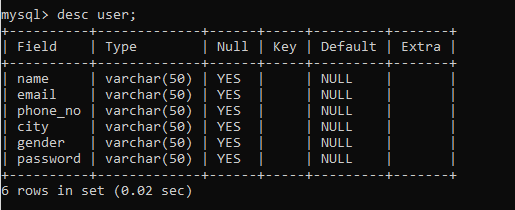
Movie Table:



Transaction Table:

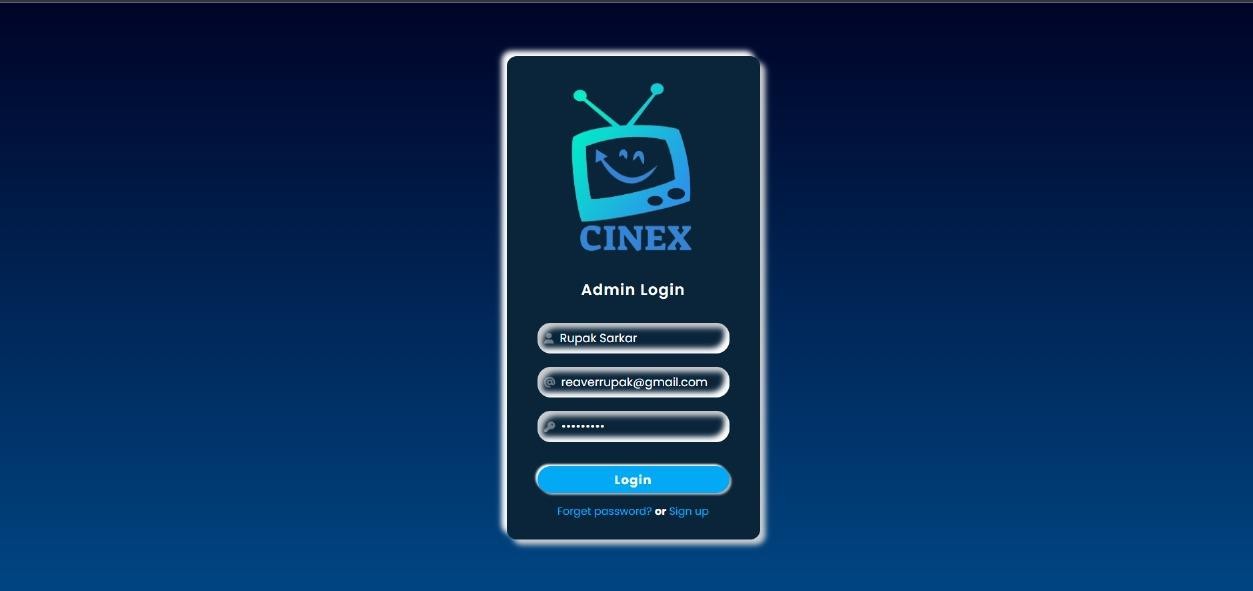


User Table:

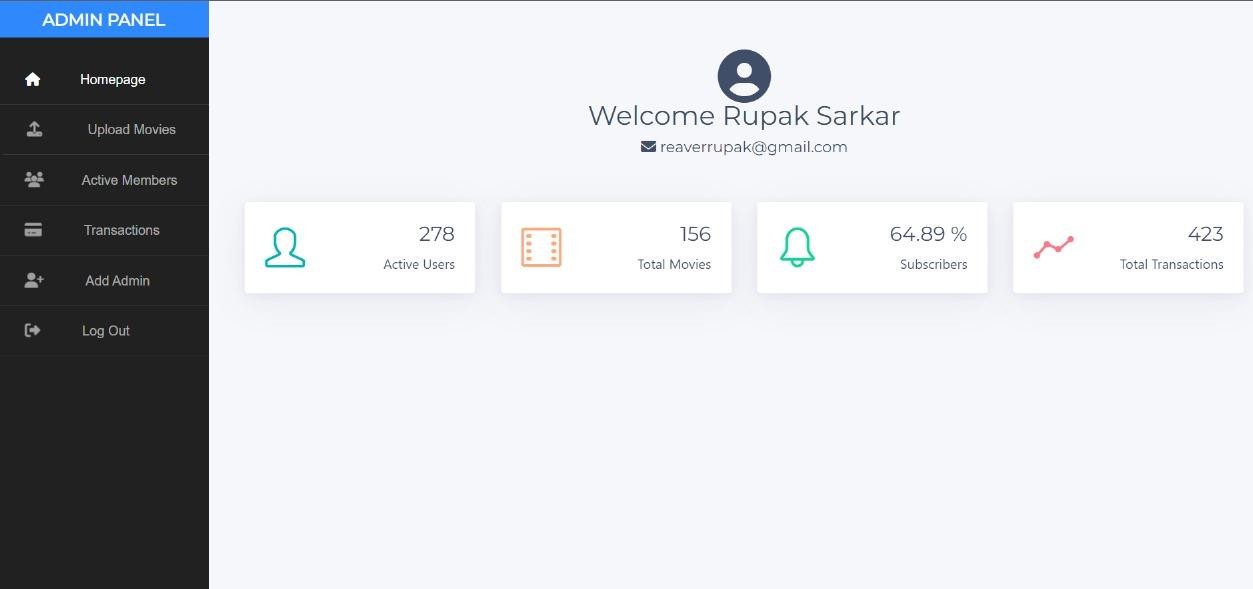


UI and Source Code:

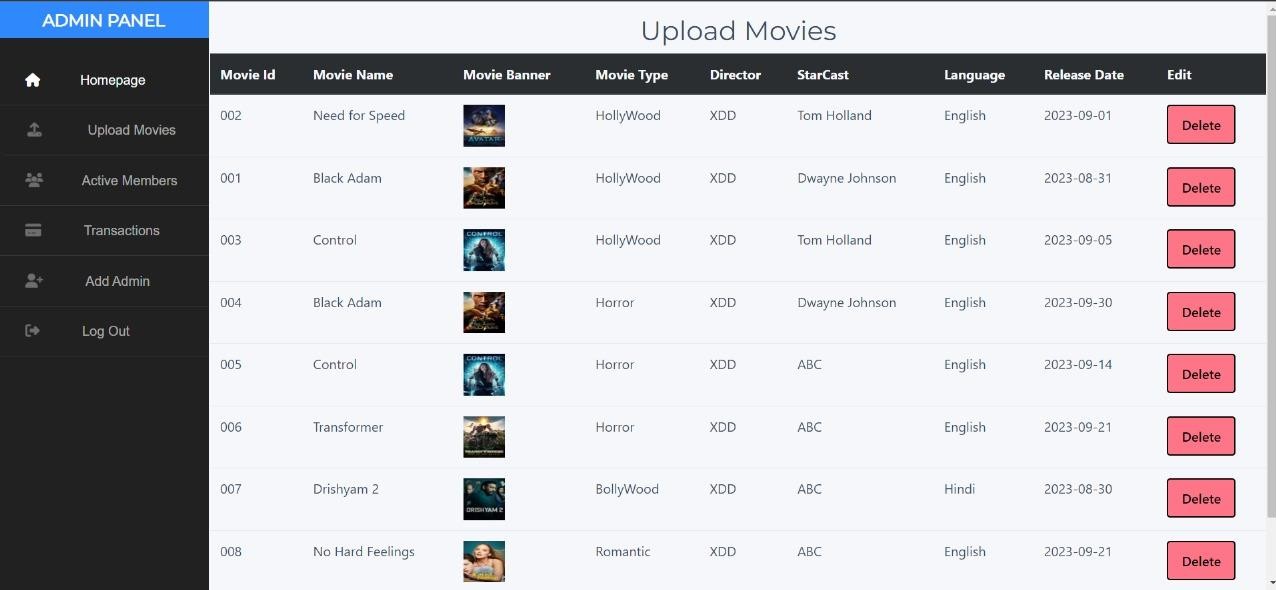
* Admin



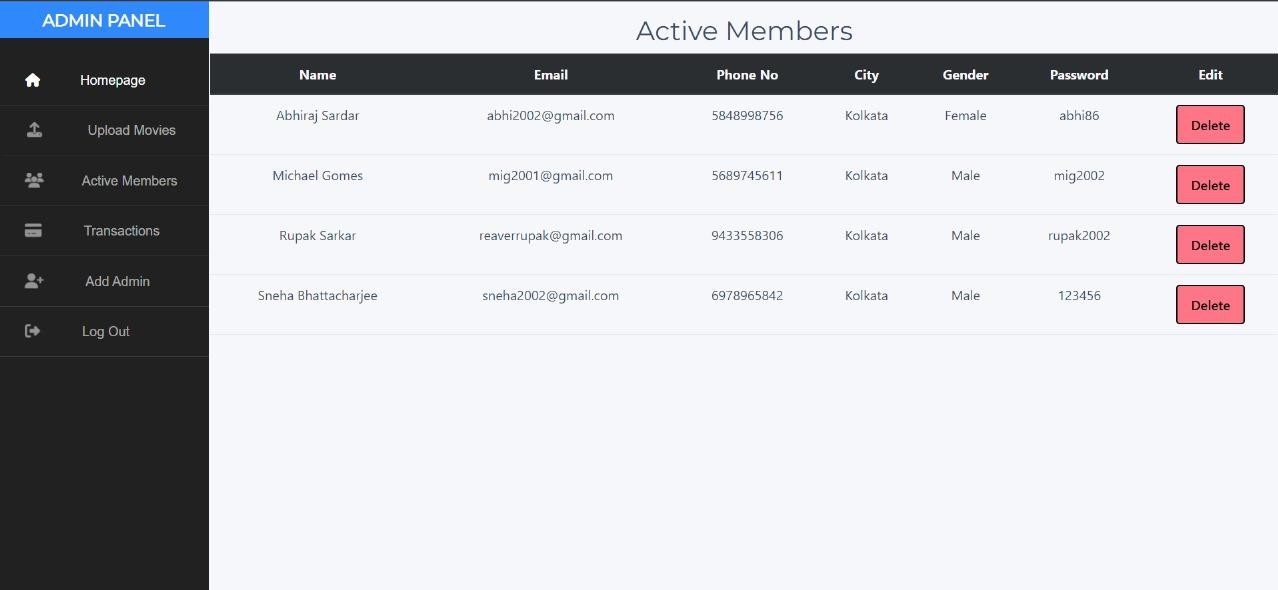
##### * Admin Dashboard



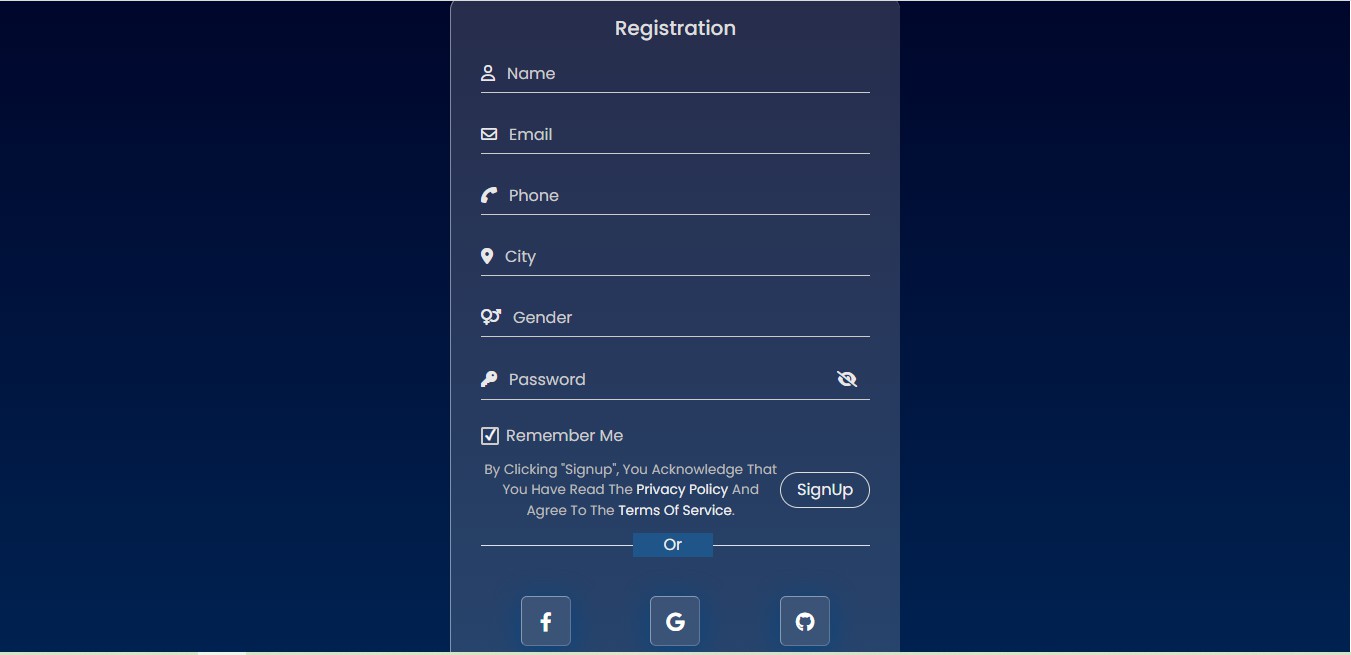
* Upload Movies



* Active Members

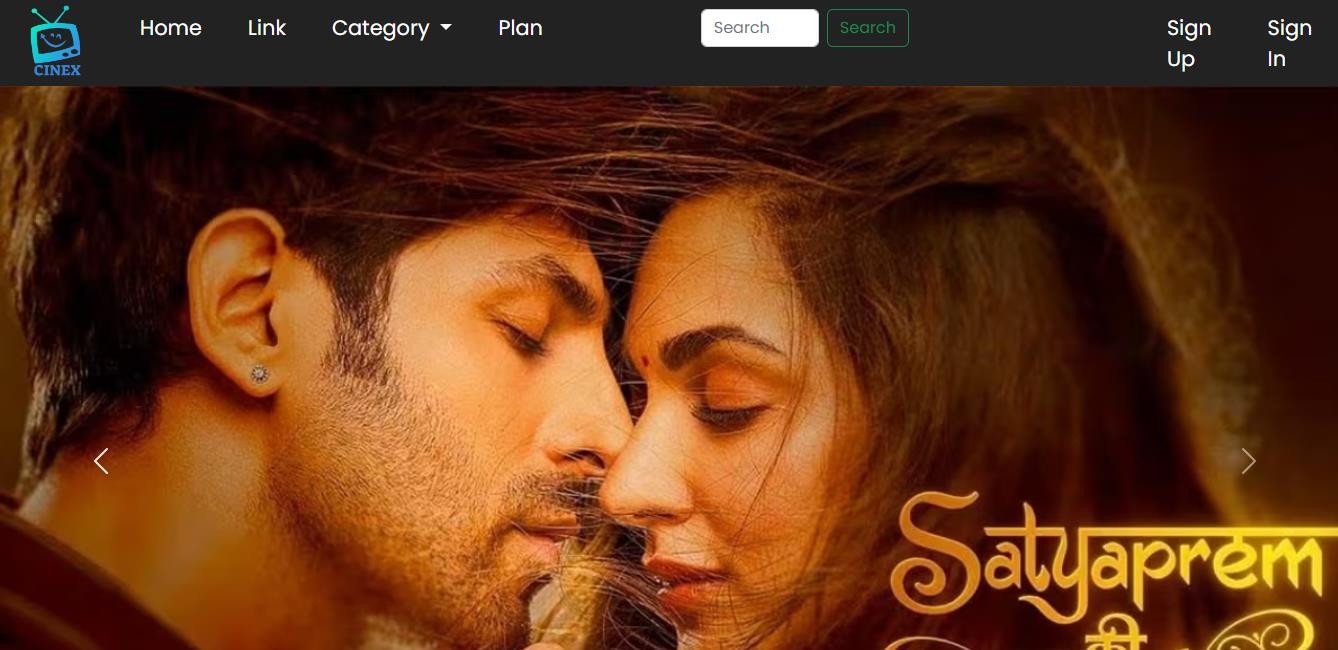


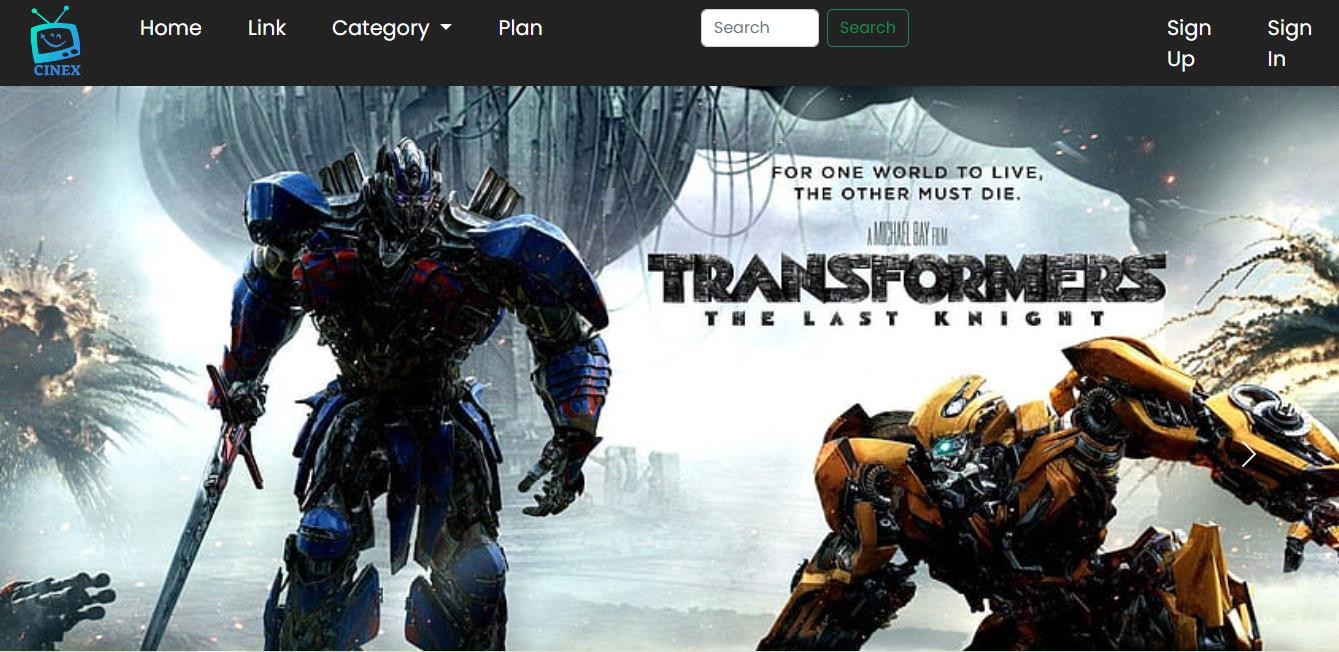
User Sign up



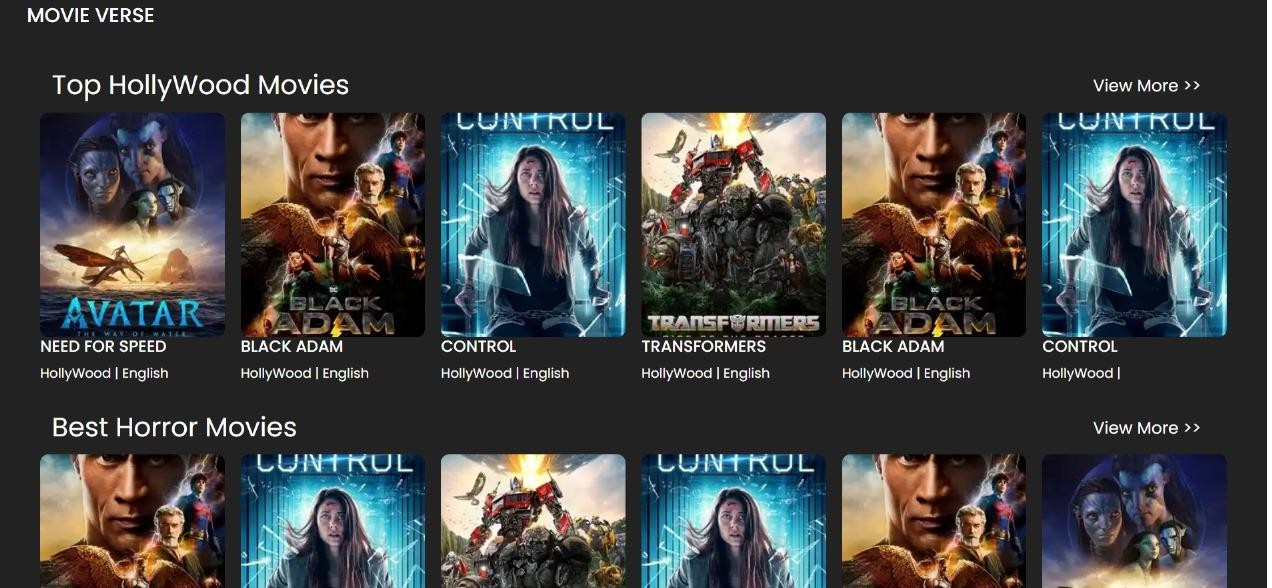
Index Page

* Carousel





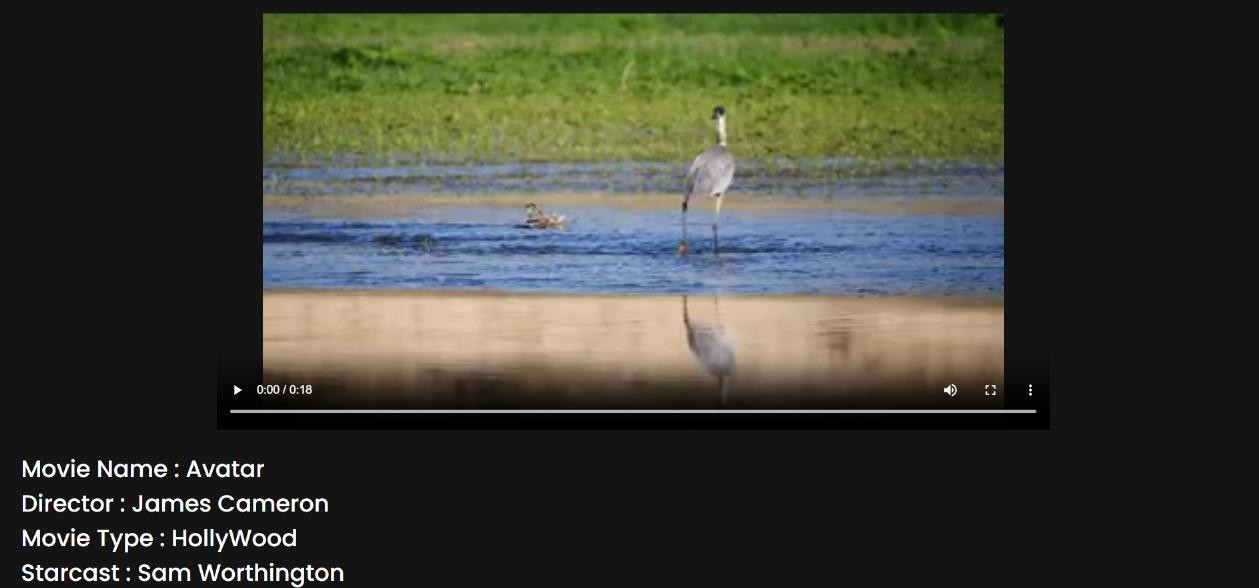
* Card Section

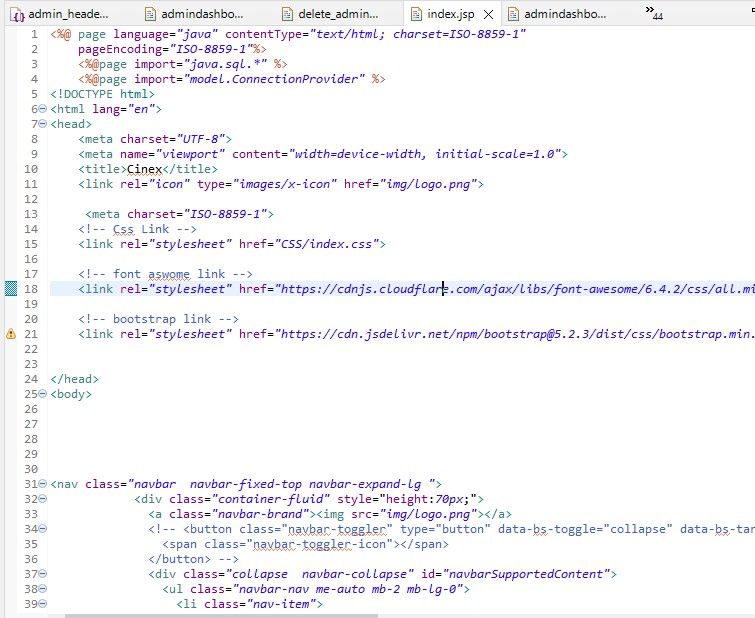


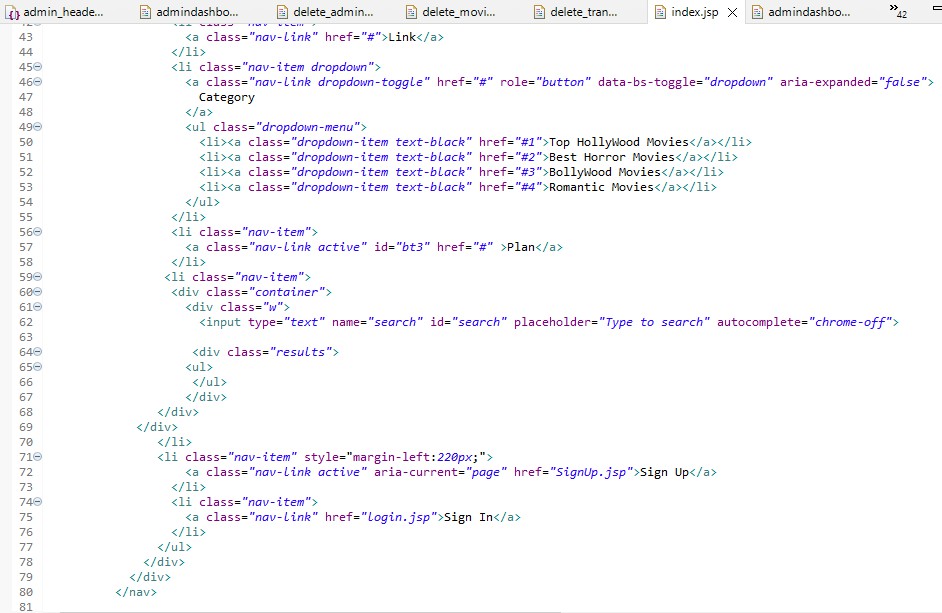
* Footer



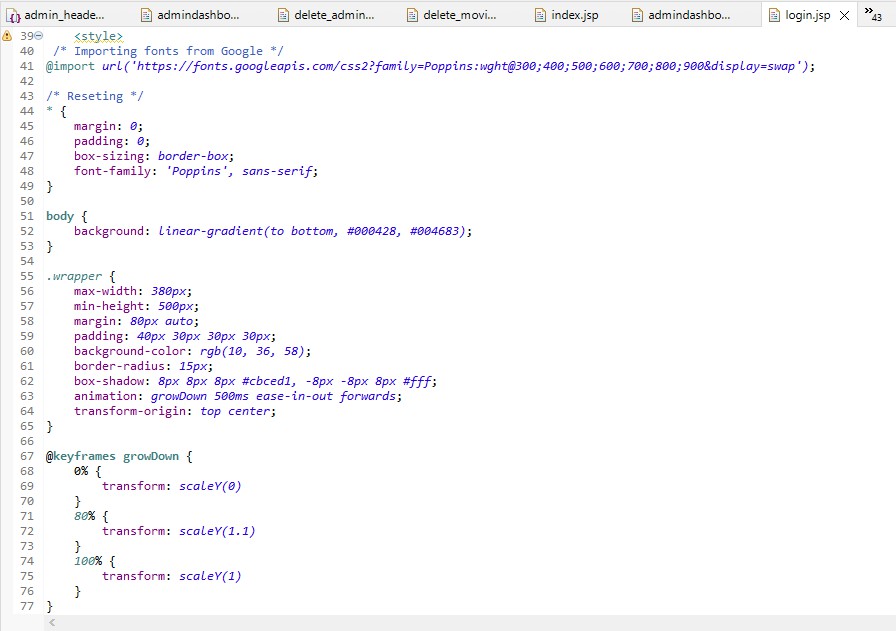
* Video



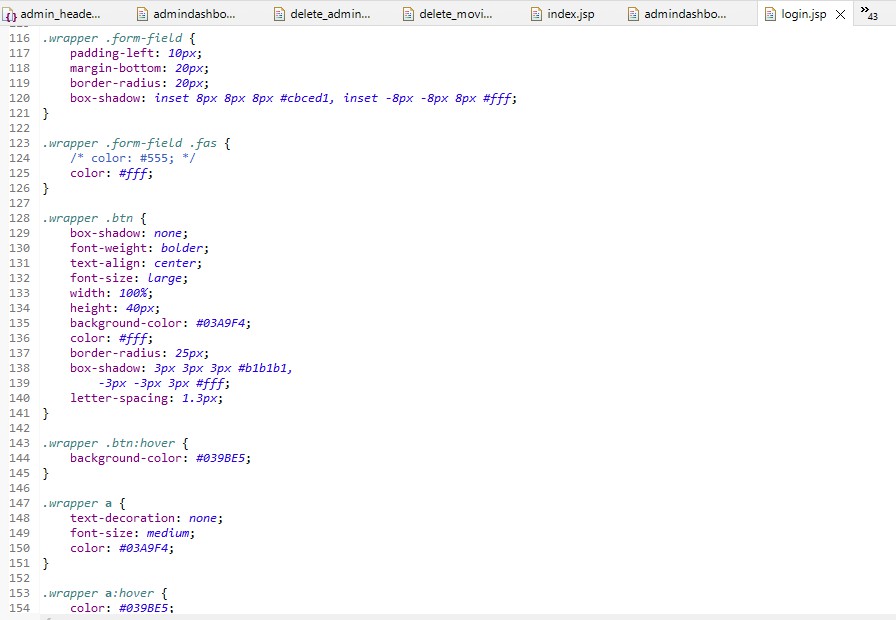




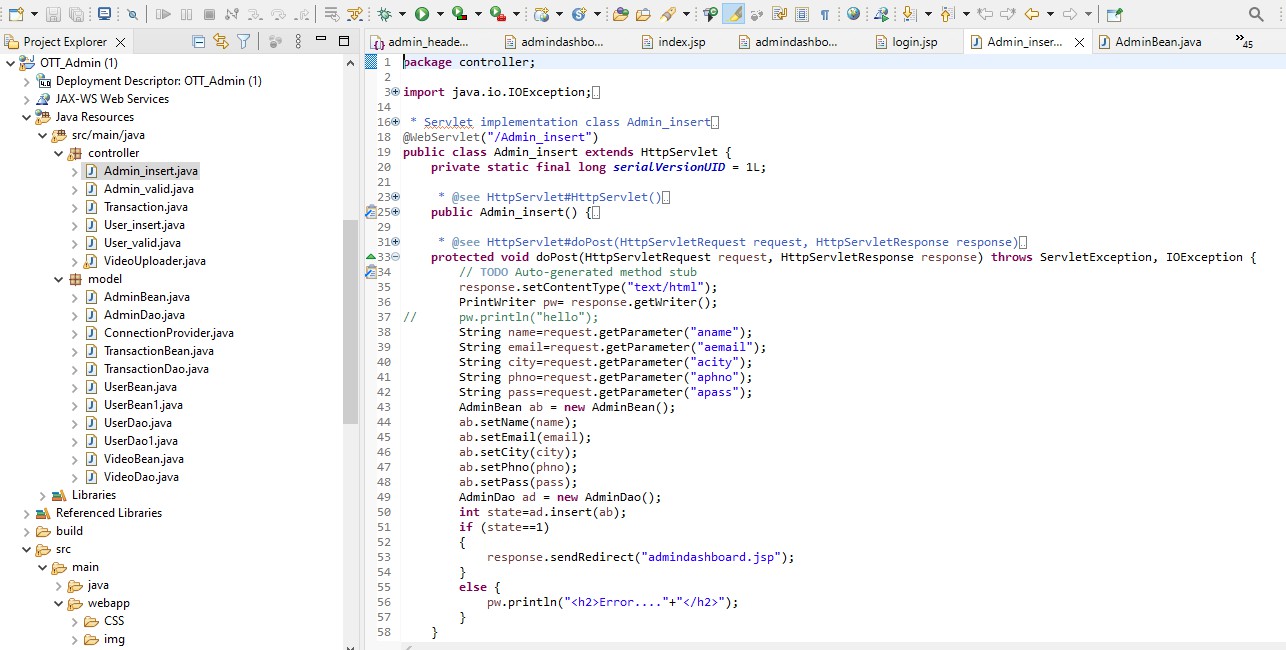


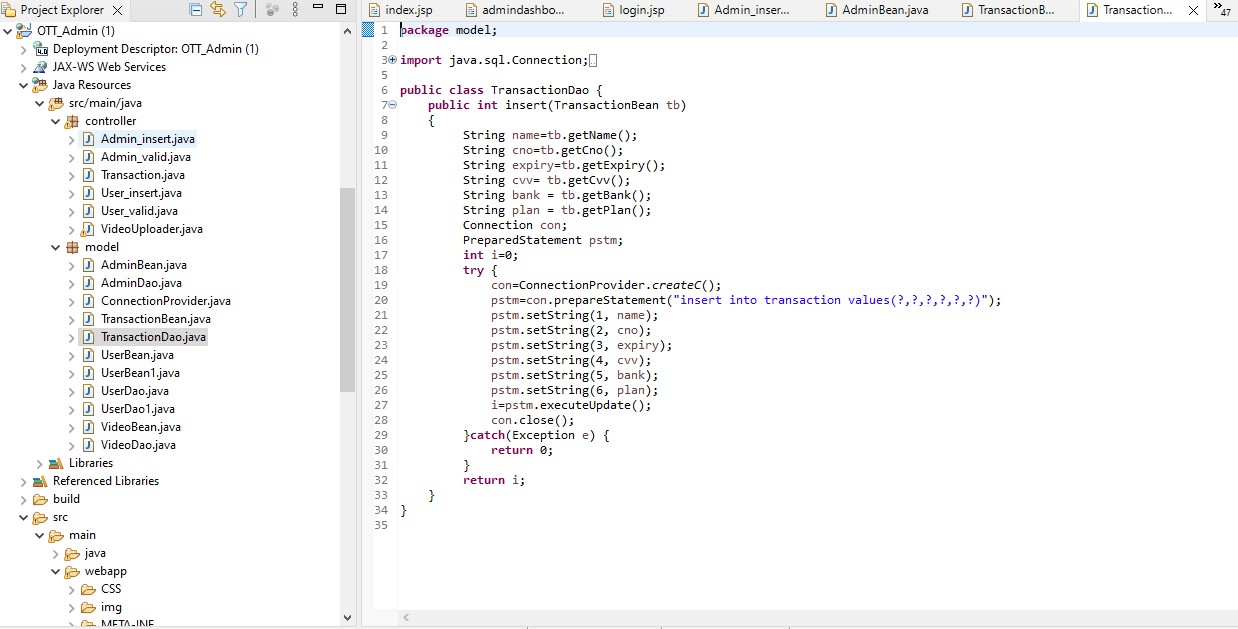


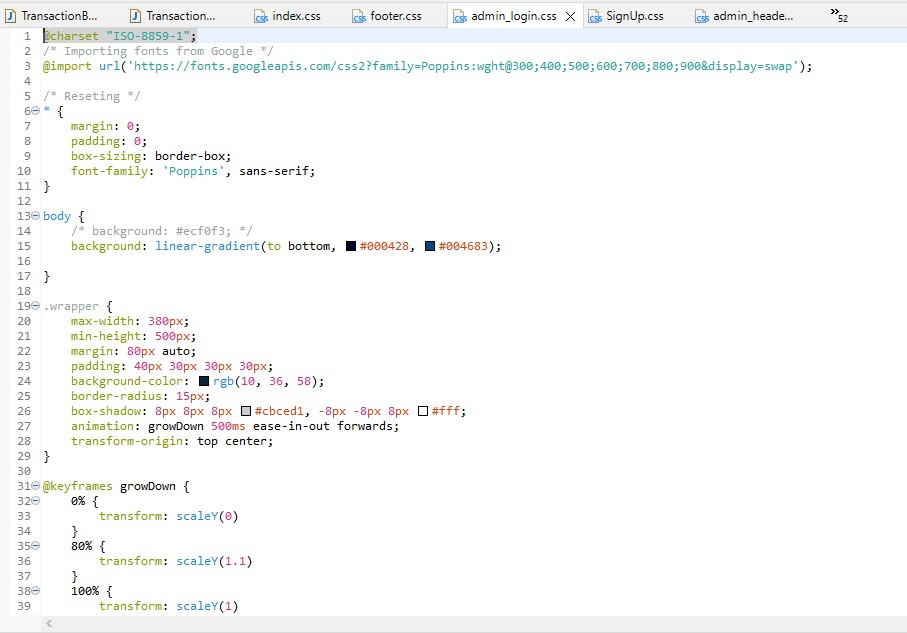


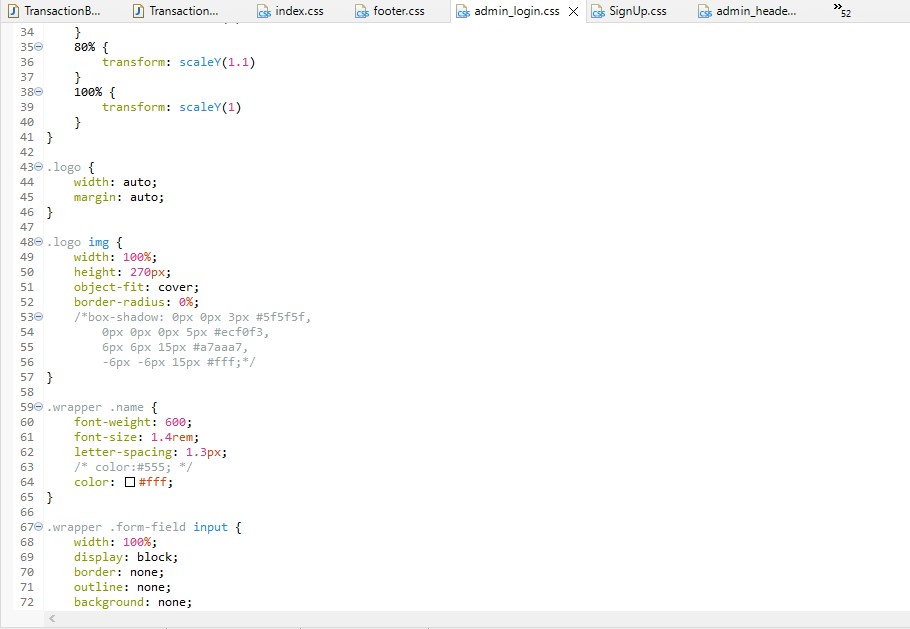


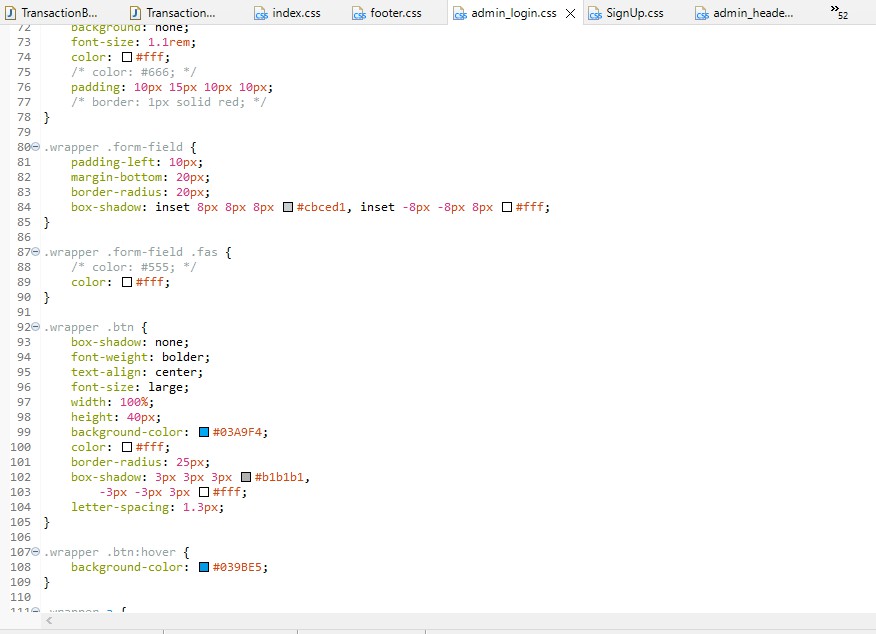


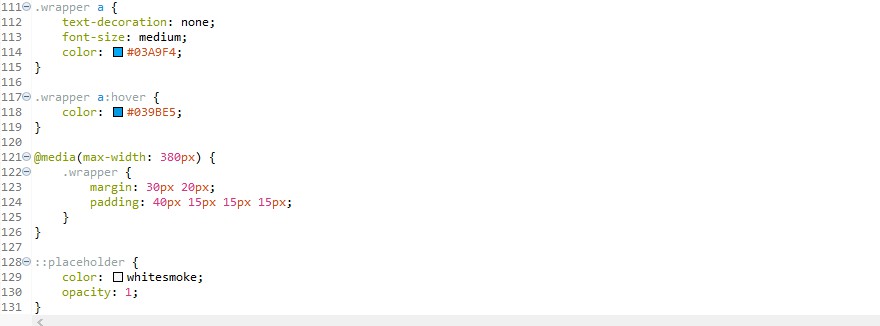


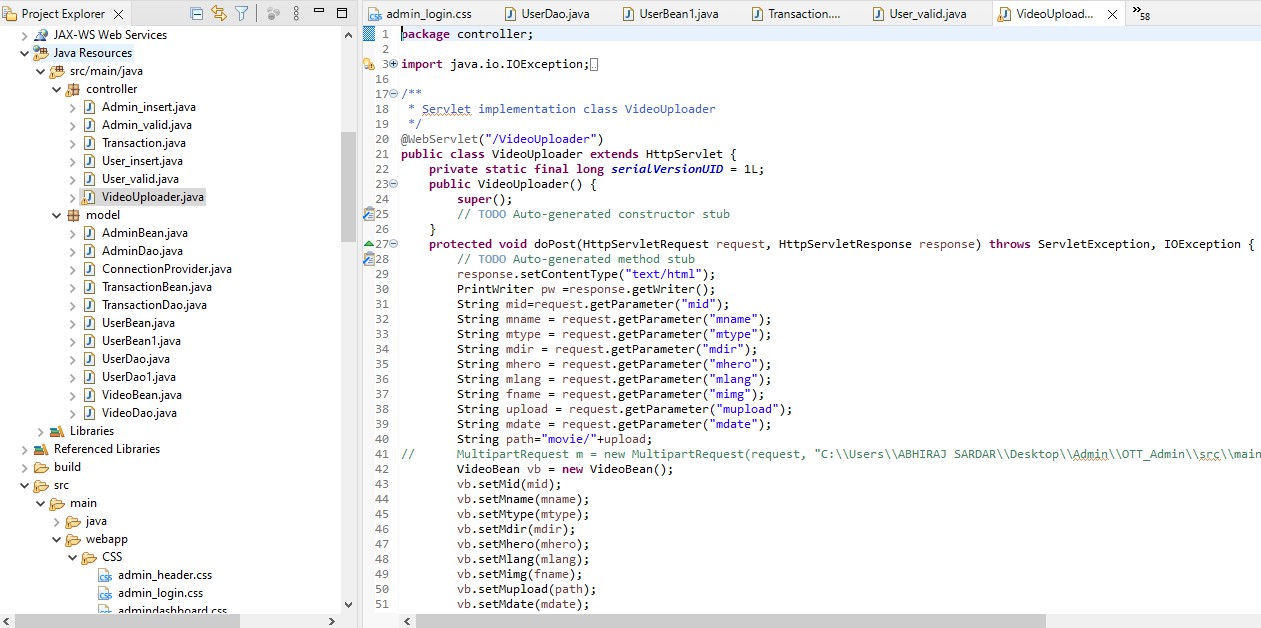


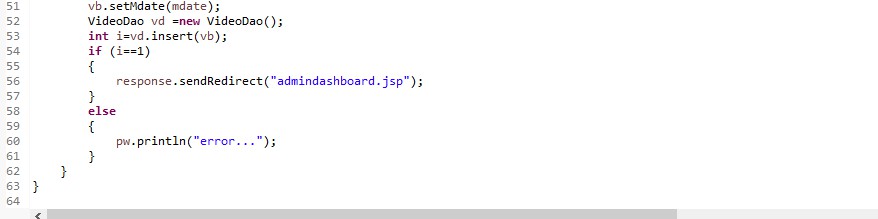


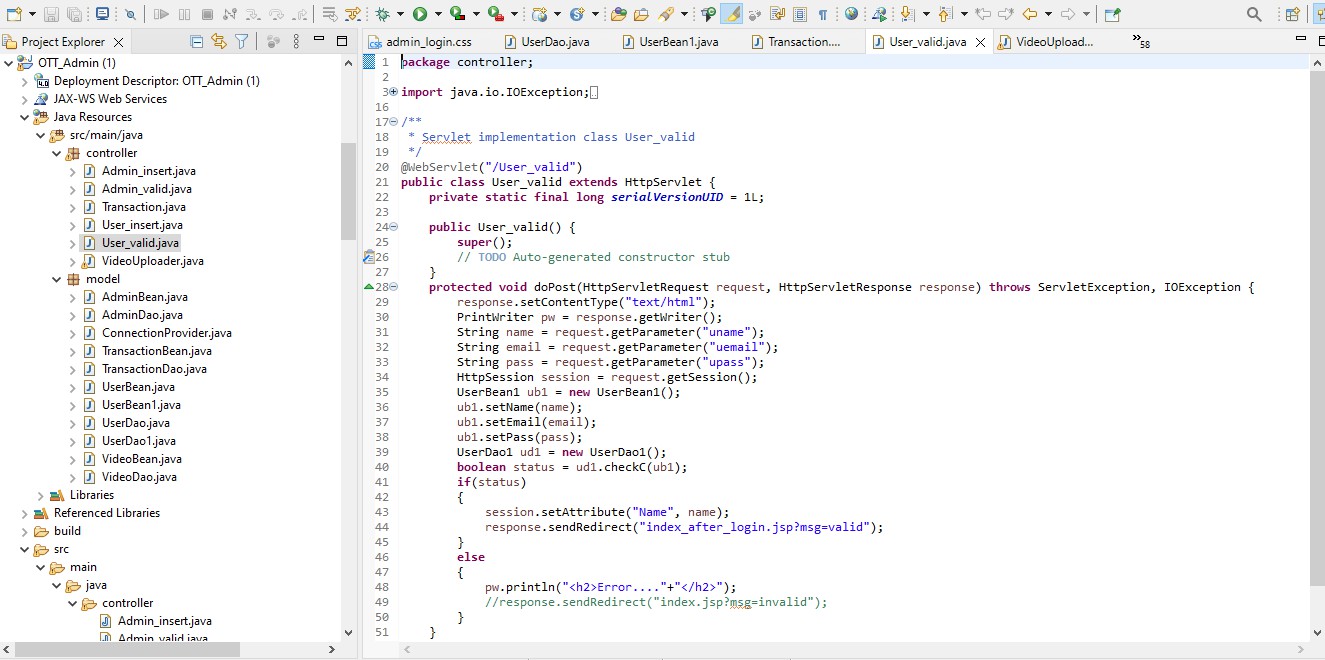


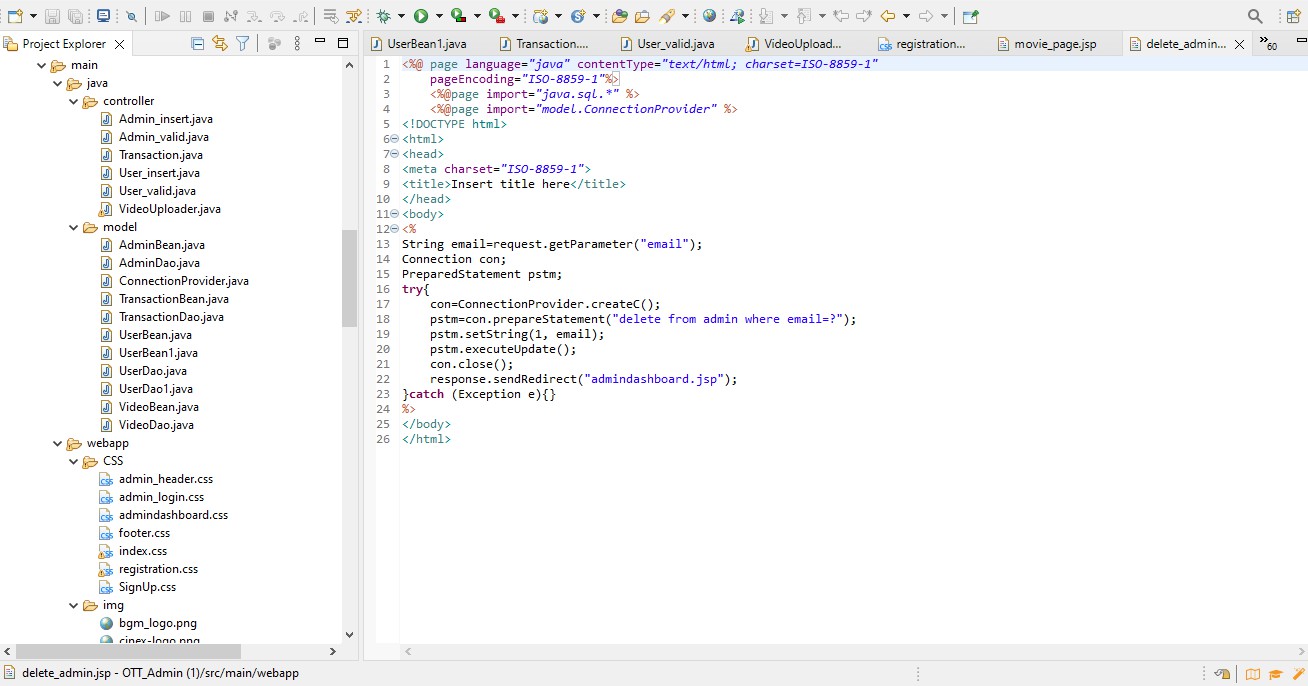


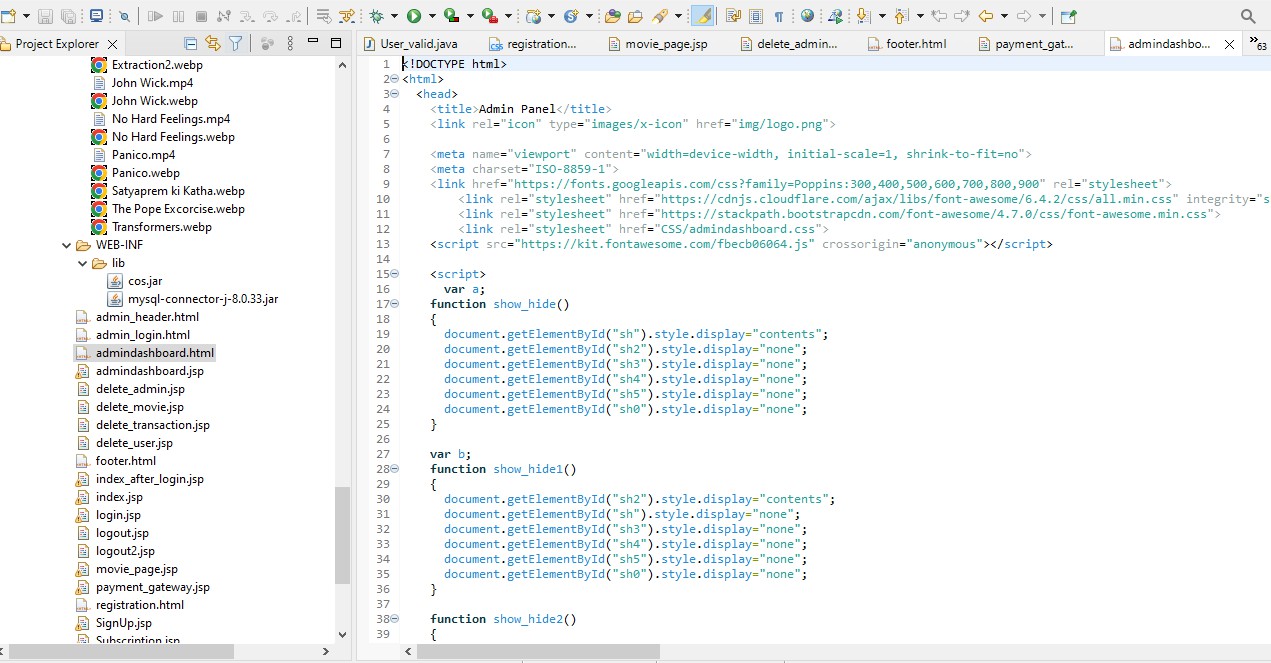












###### PROJECT TESTING:

Online Movie Platform is the place with the huge collection of Movies. It is place from where the students and publics watching the online contents. But the maintenance and add new movies in the record and update the new movies into movie section. It will be very useful. The test cases for Online over the top movies are an application that explains the test cases for OTT. Software testing is a critical part that is involved in the overall development of the application. This will be one of the interesting projects that one can work on and implement in real time world. Quality assurance is the review of the software product that checks for the correctness, reliability, completeness and maintainability. The different sections under quality assurance are unit testing, integrated testing, validation testing, output testing and user acceptance testing. Test cases gives an idea like on perform of some tasks what will be the predicted output or result. It will help in predicting the result on perform of certain tasks. The test cases below gives an idea of what **Login form**: The test cases involved are whether **password** is valid, the result must be obtained on performing a particular task.

LOGIN FORM:

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **Test Case** | **Excepted Result** | **Test Result** |
| 1 | Enter valid name and password & click on login button | Software should display main window | Successful |
| 2 | Enter invalid | Software should not  display main window | Successful |

Movie Entry Form:

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **Test Case** | **Excepted Result** | **Test Result** |
| 1 | On the Click of ADD Button | At first user have to fill all fields with proper data, if any Error like entering text data instead of number or entering number instead of text is found then it gives proper message otherwise Adds  Record To the Database | Successful |
| 2 | On the Click of DELETE Button | Movies records are  deleted by clicking of delete button. | Successful |
| 3 | On the Click of UPDATE Button | Modified records are Updated in database by clicking UPDATE  button | Successful |
|  |  |  |  |
|  |  |  |  |

Transaction Form:

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **Test Case** | **Excepted Result** | **Test Result** |
| 1 | On the click of delete button | Users Records of transactions are deleted by clicking  of delete button. | Successful |

User Admin Form:

|  |  |  |  |
| --- | --- | --- | --- |
| **SL.NO** | **Test Case** | **Excepted Result** | **Test Result** |
| 1 | On the Click of ADD Button | At first user have to fill all fields with proper data, that’s shows in admin panel by clicking of add  button. | Successful |
| 2 | On the Click of DELETE Button | Users account records are deleted by clicking of delete  button. | Successful |

**CONCLUSION:**

In conclusion, the development of our OTT (Over-The-Top) platform using Java EE has been a significant achievement in our minor project journey. This endeavor allowed us to apply various principles and technologies of enterprise-level Java programming to create a robust and feature-rich platform for streaming and delivering digital content.