ShowTime – Ott Platform

A project report submitted in partial fulfillment of the requirement for degree of

Bachelor Of Technology

In

Computer Science and Engineering

By

Narayana Vishnu kumar(R171066)
Bujjigalla Vijay(R170659)

Under the Guidance Of
V.Sravani

Asst.Prof. In Department of Computer Science & Engineering

Ap IIIT,RGUKT Rk Valley
Vempalli,Kadapa(Dt),Andhra Pradesh
Academic Year 2022-2023

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P.Government Act 18 of 2008) RGUKT-RK Valley

Vempalli,Kadapa,Andhrapradesh-516330.

CERTIFICATE OF EXAMINATION

This is to certify that we have examined the thesis entitled **ShowTime-Ott Platform,** submitted by Narayana Vishnu kumar(R171066) and Bujjigalla Vijay(R170659) hereby accord my approval of it as a study carried out and presented in a manner required for its acceptance in partial fulfillment for the award of Bachelor of Technology degree for which it has been submitted. This approval does not necessarily endorse or accept every statement made, opinion expressed or conclusions drawn, as recorded in this thesis. It only signifies the acceptance of this thesis for the purpose for which it has been submitted.

EXAMINER

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P.Government Act 18 of 2008) RGUKT-RK Valley

Vempalli,Kadapa,Andhrapradesh-516330.

CERTIFICATE OF PROJECT COMPLETION

This is to certify that we have examined the thesis entitled **ShowTime-Ott Platform** submitted by Narayana Vishnu kumar(R171066) and BujjigallaVijay(R170659) under our guidance and supervision for the partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic year 2022 – 2023 at RGUKT-RK Valley

To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma

V.Sravani
Asst.Proffesor,Computer Science and Engg
RGUKT Rk Valley

N.Satyanandaram
Head of the Department
Computer Science and Engg
RGUKT Rk Valley

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P.Government Act 18 of 2008)

RGUKT-RK Valley

Vempalli,Kadapa,Andhrapradesh-516330.

DECLARATION

We , Narayana Vishnu kumar(R171066) and Bujjigalla Vijay(R170659) hereby declare that the project report entitled “ShowTime – ott platform” done by is under guidance of V.Sravani is submitted in partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic year 2022 – 2023 at RGUKT-RK Valley. we also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. To the best of my knowledge , the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma.

N.Vishnu kumar(R171066)
B.Vijay(R170659)

Acknowledgement

We would like to express our deep sense of gratitude & respect to all those people behind the screen who guided, inspired and helped us crown all our efforts with success. We wish to express our gratitude to **V.Sravani** for his valuable guidance at all stages of study, advice, constructive suggestions, supportive attitude and continuous encouragement, without which it would not be possible to complete this project.

We would also like to extend our deepest gratitude & reverence to the Director of RGUKT,RK Valley **Prof. K. Sandyarani** and HOD of Computer Science and Engineering **Mr. N. Satyanandaram** for their constant support and encouragement.

Last but not least we express our gratitude to our parents for their constant source of encouragement and inspiration for us to keep our morals high.

Table of Contents

Abstract

Main aim of the project is to develop an OTT platform,where users can watch their favourite shows,movies etc., in a flexible way through laptop,mobile phone.We are not only providing movies but also webisodes.Within no time users get latest updates through our OTT platform and this makes the users not to miss his/her favourite things.

One of the best feature added to out platform is chatting feature,where user will get great experience with his friends by sharing their views,trolls on the movie,show etc., they are watching.This will help them communicate each other even they are watching different shows,movies etc., and even they are remote places.

Introduction

1.1 Why Web Applications?

Web applications are easy to design, deploy and maintain and more than that it has a great advantage that web applications are platform independent hence, unlike mobile or desktop applications there is no need of developing separate applications for different platforms. So, the cost of development, maintains and upgradation becomes very less and greatly easier.Web applications can also reach numerous audiences and in case of our E-Learning Platform it is having many more reasons why we focus more on web application only, because our app is for providing courses for different categories of learners especially engineering students and many undergraduate students use laptops and hence web applications are much accessible for them.

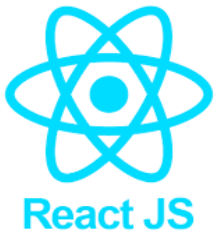
1.2 Web Frameworks for Building Dynamic Application

A web framework is a software framework that provides support for development of web applications like web services, web APIs, web resources, etc. Web framework provides a standard way to build, deploy and maintain web applications. Web framework also provides tools and libraries that simplify the common operations like data base querying, world wide web interaction. Especially REST APIs play crucial role in simplifying and establishing a reliable and fast connection between client and server. Client – Server communication happens via HTTP where client initiates the communication by sending a HTTP request to the server and server send a HTTP response to client.

1.3 Why Responsive Web Design

Responsive web design is the approach that suggests that design and development should respond to the user’s behavior and environment based on screen size,platform and orientation.The practice consists of a mix of flexible grids and layouts,images and an intelligent use of CSS media queries.

Tools and Technologies



React.js

React is a declarative, efficient, and flexible JavaScript library for building user interfaces. It is an open-source, component-based front-end library that is responsible only for the view layer of the application. ReactJS is not a framework, it is just a library developed by Facebook to solve some problems that we were facing earlier

Tailwindcss

Tailwind CSS can be used to style websites in the fastest and easiest way.Tailwind CSS is basically a utility-first CSS framework for rapidly building custom user interfaces. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override.

Antd

Ant Design is a React UI library that contains easy-to-use components that are useful for building interactive user interfaces. It is very easy to use as well as integrate. It is one of the smart options to design web applications using react. It provides us with high-quality components which can be used with ease.

Nodejs

NodeJS is an open-source and cross-platform runtime environment built on Chrome's V8 JavaScript engine for executing JavaScript code outside of a browser. You need to recollect that NodeJS isn't a framework, and it's not a programing language. It provides an event-driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side applications using JavaScript.

Expressjs

Express.js is a small framework that works on top of Node.js web server functionality to simplify its APIs and add helpful new features. It makes it easier to organize your application's functionality with middleware and routing. It adds helpful utilities to Node.js HTTP objects and facilitates the rendering of dynamic HTTP objects.

Socket.io

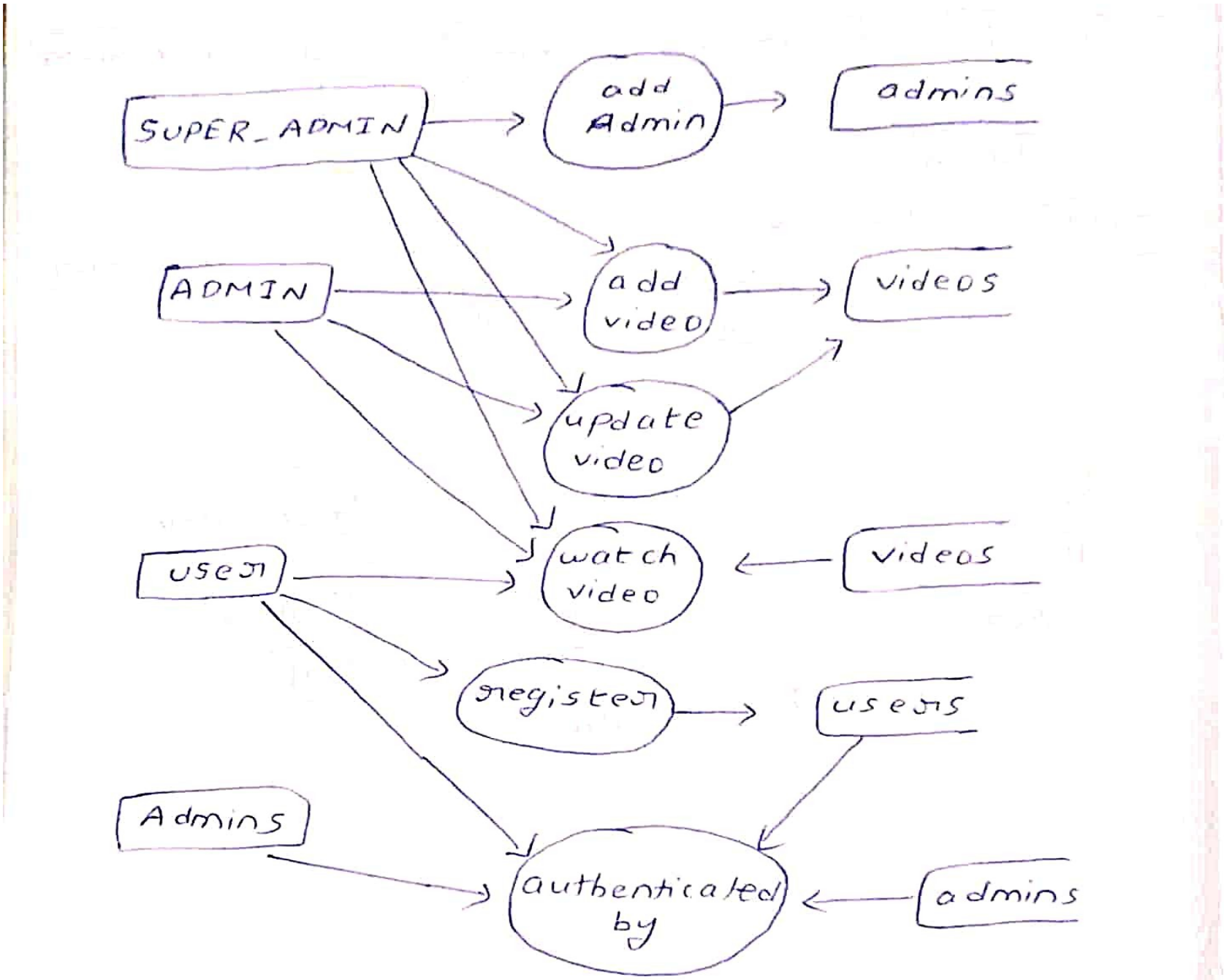
Socket.IO relies on Engine.IO, which is the implementation of the transport-based cross-browser/cross-device bi-directional communication layer. It can establishes connection even in the presence of proxies, load-balancers, personal firewalls, and antivirus softwares.

MySql

MySQL server is a open-source relational database management system which is a major support for web based applications. Databases and related tables are the main component of many websites and applications as the data is stored and exchanged over the web. Even all social networking websites mainly Facebook, Twitter, and Google depends on MySQL data which are designed and optimized for such purpose. For all these reasons, MySQL server becomes the default choice for web applications.

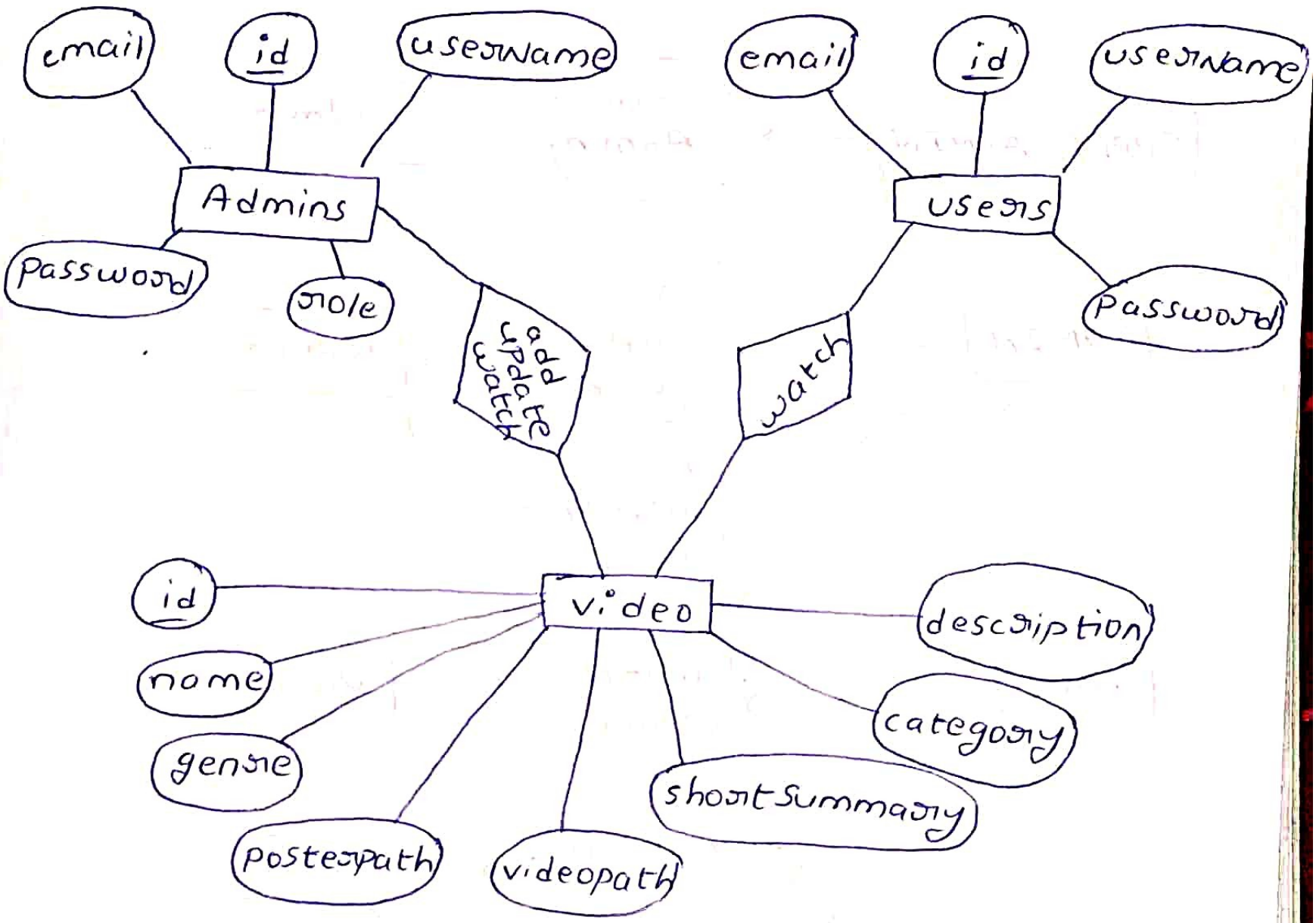
System Design

DFD Diagram



Database Design

ER Diagram



Sample Code

Frontend:

signInUser.js

```
import { Form, Input, Button, Checkbox, Card, message } from "antd";
import axios from "axios";
import { useCookies } from 'react-cookie';
import { useEffect } from "react"
const formItemLayout = {
    labelCol: {
        span: 4,
    },
    wrapperCo: {
        span: 8,
    },
};
const onFinish = async (values, setCookie, cookies) => {
    let url = "";
    if (values.isAdmin) {
        url = "http://localhost:8080/admin/verifyAdmin"
    }
    else {
        url = "http://localhost:8080/user/verifyUser"
    }
    await axios({
        method: "post",
        url: url,
        headers: { "Content-Type": "application/json" },
        data: values
    }).then((ret) => {
        console.log(ret.data)
        if (ret.status == 200) {
            message.success(ret.data.data, 2)
            const { data, ...cookieData } = ret.data
            console.log(cookieData)
            setCookie("authorization", cookieData, { secure: true })
            setTimeout(() => {
                window.open("/video?category=movie", "_self")
```

```
            }, 300)
          }
          else { message.error(ret.data.data, 2) }
      }).catch((ret) => {
          message.error(ret.response.data.data, 2)
      })
}
export default function SignInUser() {
    const [cookies, setCookie] = useCookies()
    return (
        <div className="flex flex-col items-center justify-center h-full bg-black">
            <Card
                title="SignIn"
                className="w-2/5"
                hoverable={true}
                headStyle={{ fontSize: "30px", color: "#e50914" }}
            >
                <Form
                    onFinish={(values) => onFinish(values, setCookie, cookies)}
                    className="w-full border-0 mt-5"
                >
                    <Form.Item
                        {...formItemLayout}
                        name="email"
                        label="Email"
                        rules={[
                            {
                                required: true,
                                message: 'Please enter your email',
                            },
                        ]}
                    >
                        <Input type="Email" placeholder="Please enter your email" />
                    </Form.Item>
                    <Form.Item
                        {...formItemLayout}
                        name="password"
                        label="Password"
                        rules={[
```

```
                {
                    required: true,
                    message: 'Please enter password',
                },
              ]}
            >
              <Input.Password placeholder="Please enter password" />
            </Form.Item>
            <Form.Item name="isAdmin" valuePropName="checked" wrapperCol={{ offset: 10, span:
12 }}>
              <Checkbox>Admin?</Checkbox>
            </Form.Item>
            <Form.Item wrapperCol={{ span: 12, offset: 10 }}>
              <Button type="primary" style={{ background: "#E50914" }} className="rounded-md text-
white" htmlType="submit" danger>
                Submit
              </Button>
            </Form.Item>
          </Form>
          <div className="w-full flex justify-center underline text-appRedColor"><a
href="/addUser">Register?</a></div>

        </Card>
      </div>
  );
}
```

Backend

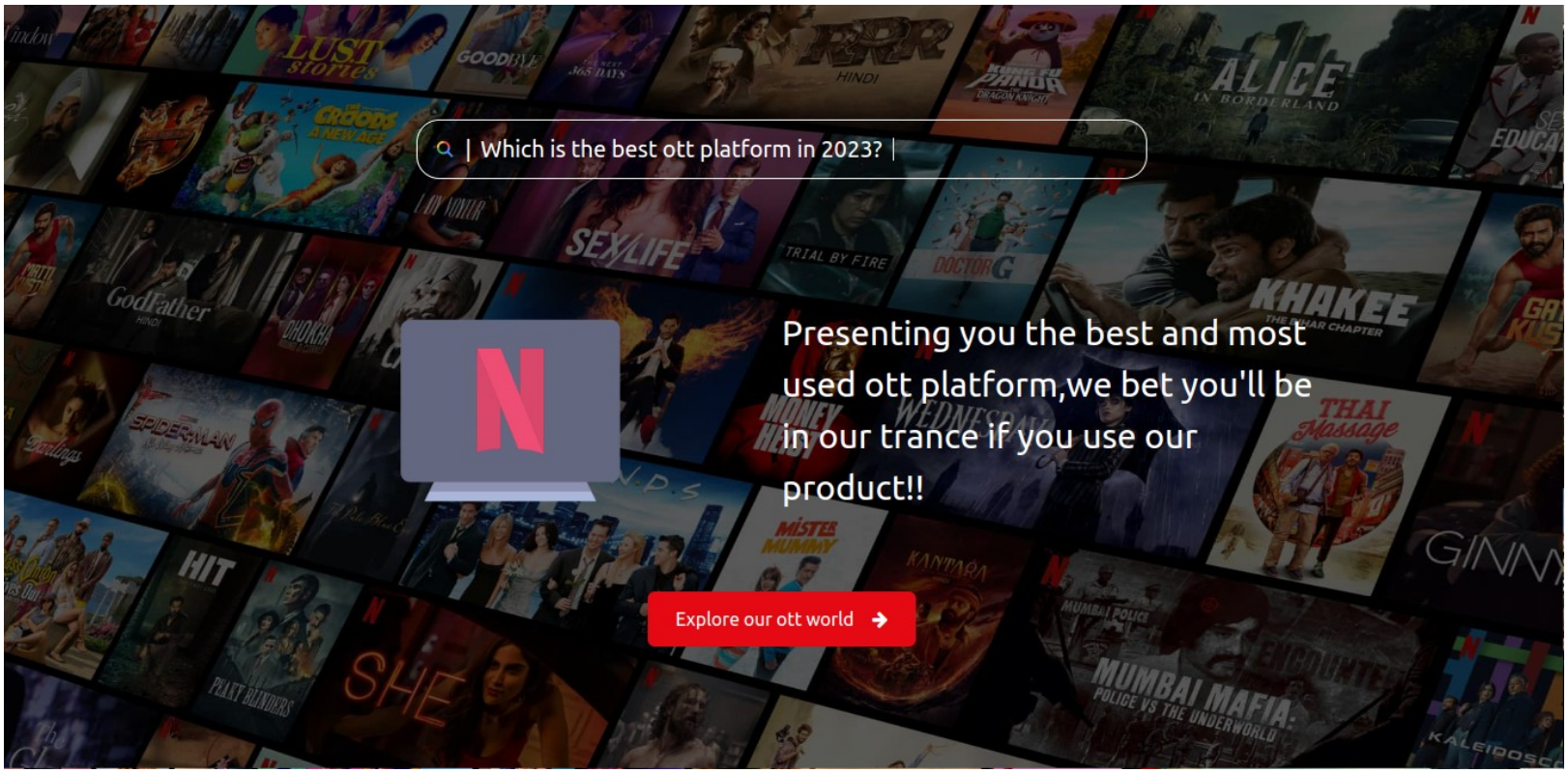
userModule.js

```
const user = require("../models/userModel.js")
const express = require("express")
const router = express.Router()
const { StatusCodes } = require("http-status-codes")
const bcrypt = require("bcrypt")
module.exports = router
async function userExisted(email) {
  const userData = JSON.parse(JSON.stringify(await user.findOne({ where: { email: email } })))
  if (userData == null) {
    return false
```

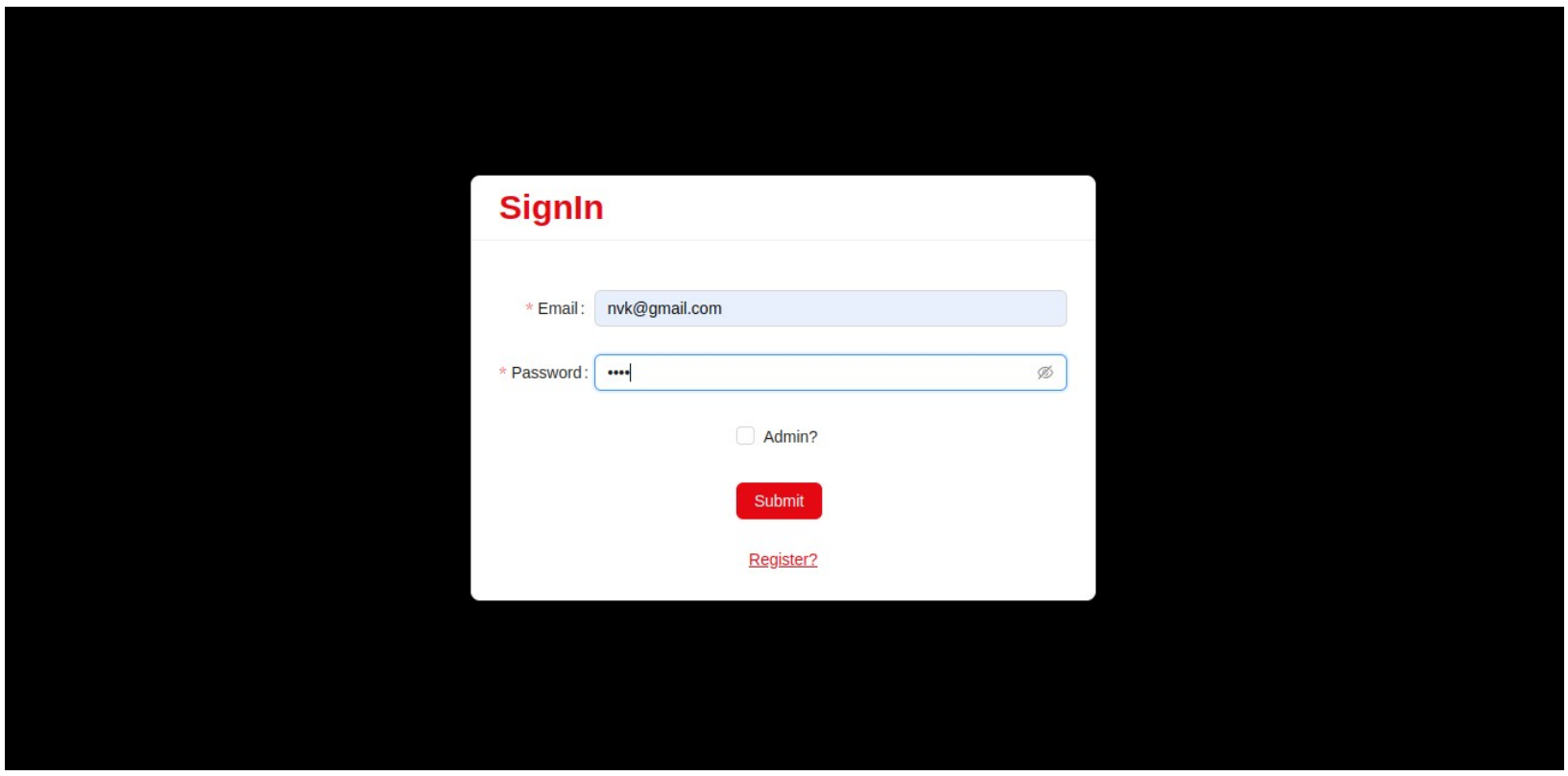
```
    }
    else {
        return userData
    }
}
router.post("/addUser", async function (req, res) {
    try {
        const data = req.body
        data.password = bcrypt.hashSync(data.password, 10)
        const userData = await user.create(data)
        if (userData instanceof user) {
            return res.status(StatusCodes.OK).json({ "data": "user added successfully" })
        }
        else {
            return res.status(StatusCodes.INTERNAL_SERVER_ERROR).json({ "data": "internal server
error" })
        }

    }
    catch (err) {
        return res.status(StatusCodes.PARTIAL_CONTENT).json({
            "data": "please enter all the required data"
        })
    }
})
router.post("/verifyUser", async function (req, res) {
    const userData = await userExisted(req.body.email)
    if (!userData) {
        return res.status(StatusCodes.INTERNAL_SERVER_ERROR).json({ "data": "user unauthorized" })
    }
    else if (bcrypt.compareSync(req.body.password, userData.password)) {
        return res.status(StatusCodes.OK).json({ "data": "user verified successfully", "id": userData["id"],
"role": "USER" })
    }
    else {
        return res.status(StatusCodes.UNAUTHORIZED).json({ "data": "user unauthorized" })
    }

})
```

Output

Home page

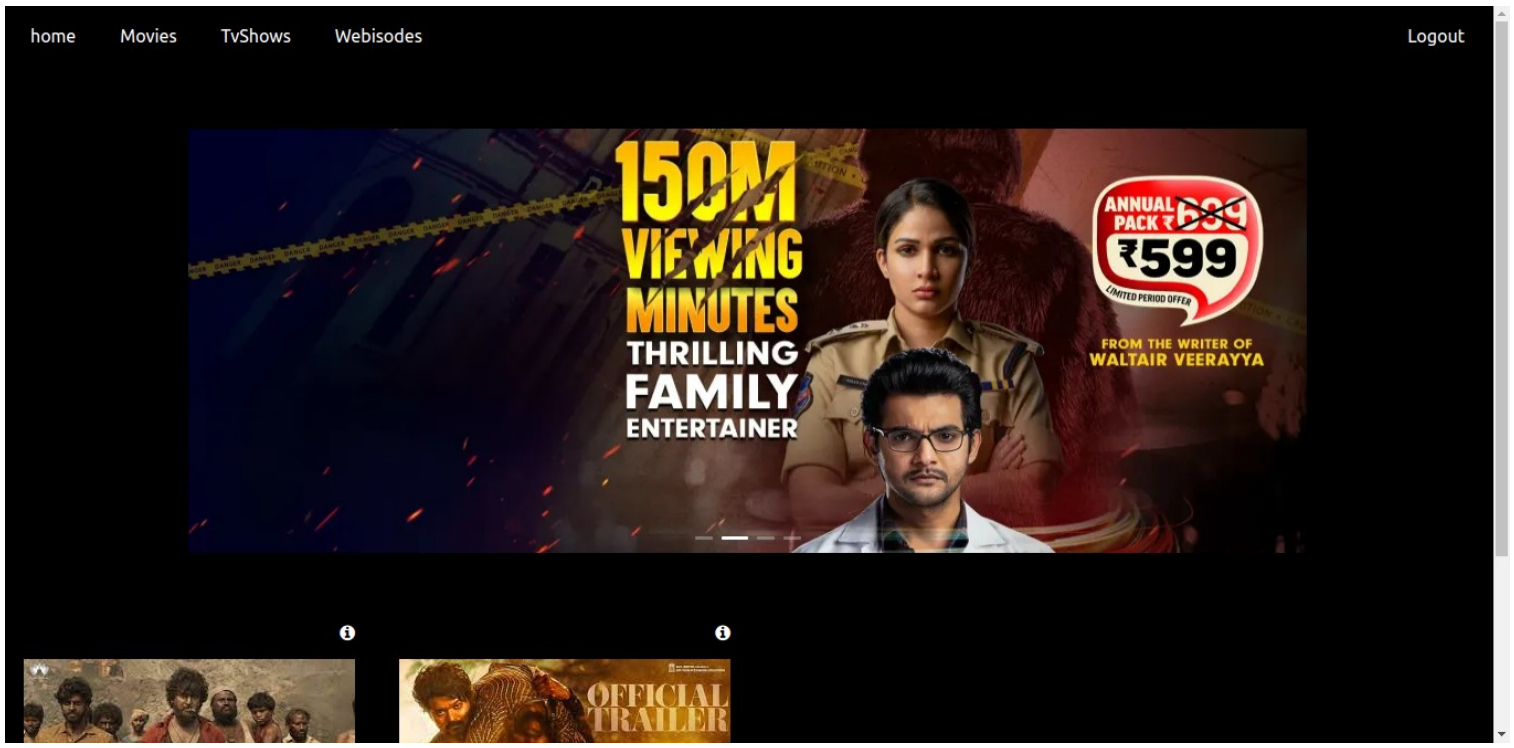


Login Page



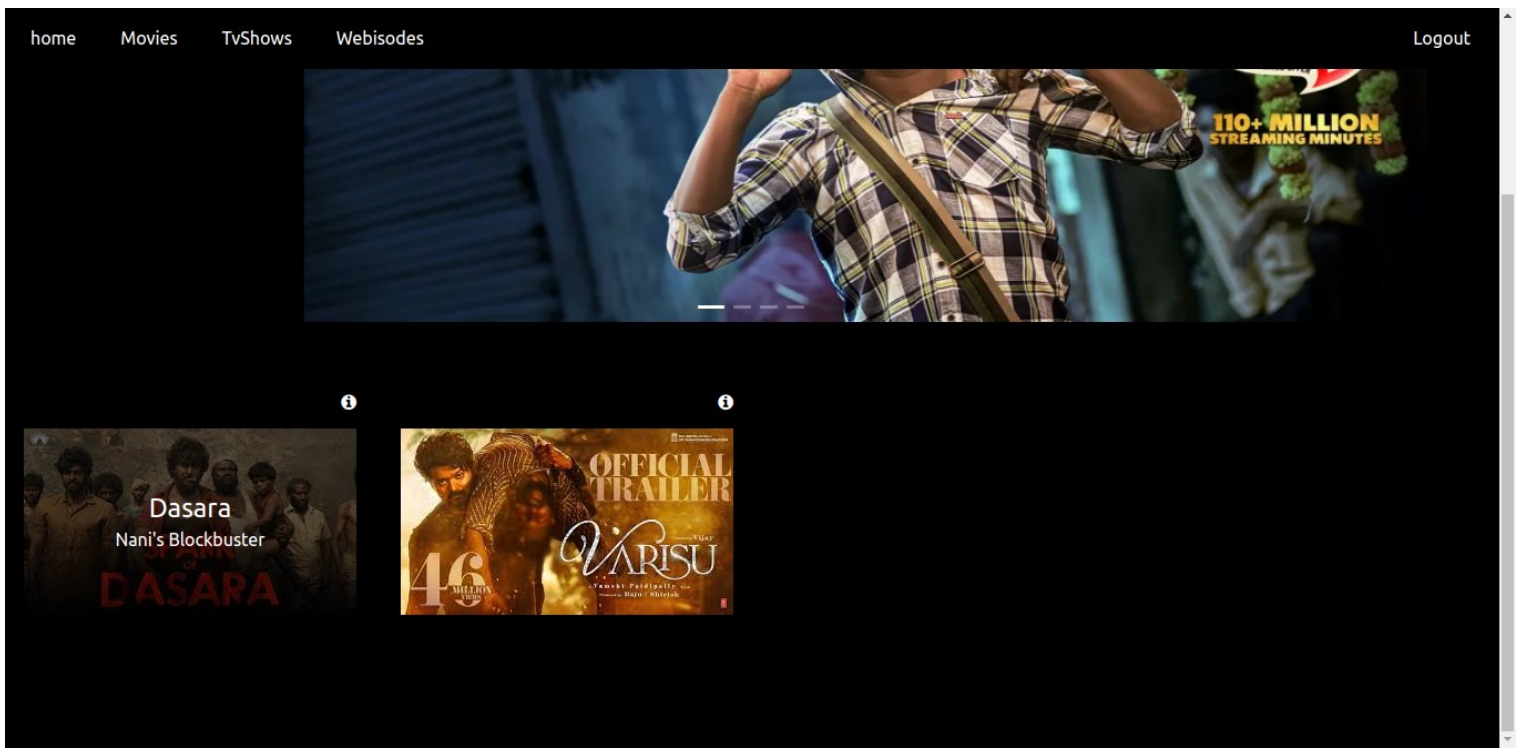
Platform Main Screen

This screen contains all the video categories like movies,webisodes,tvshows



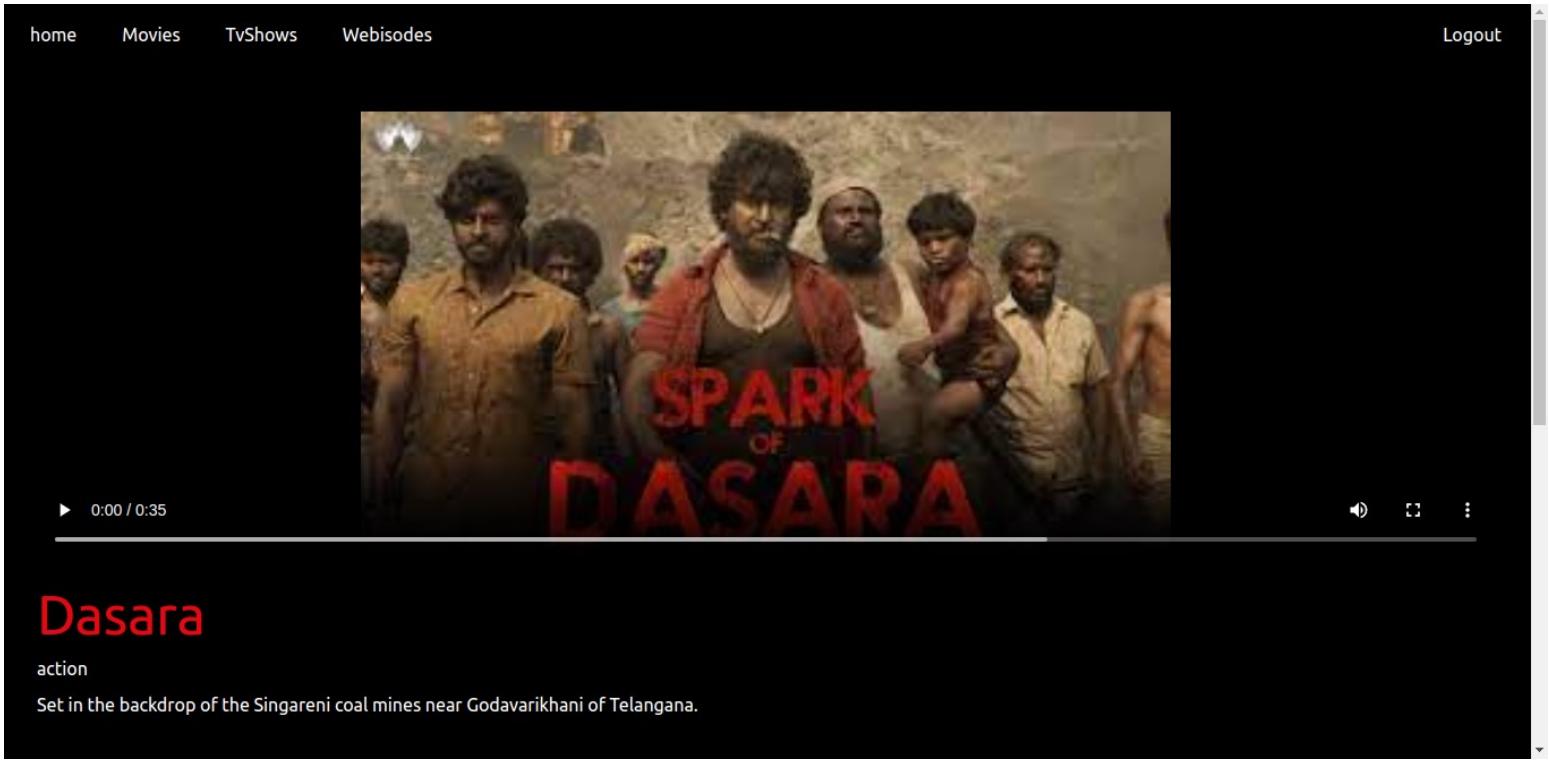
VideoCards

Design of video card

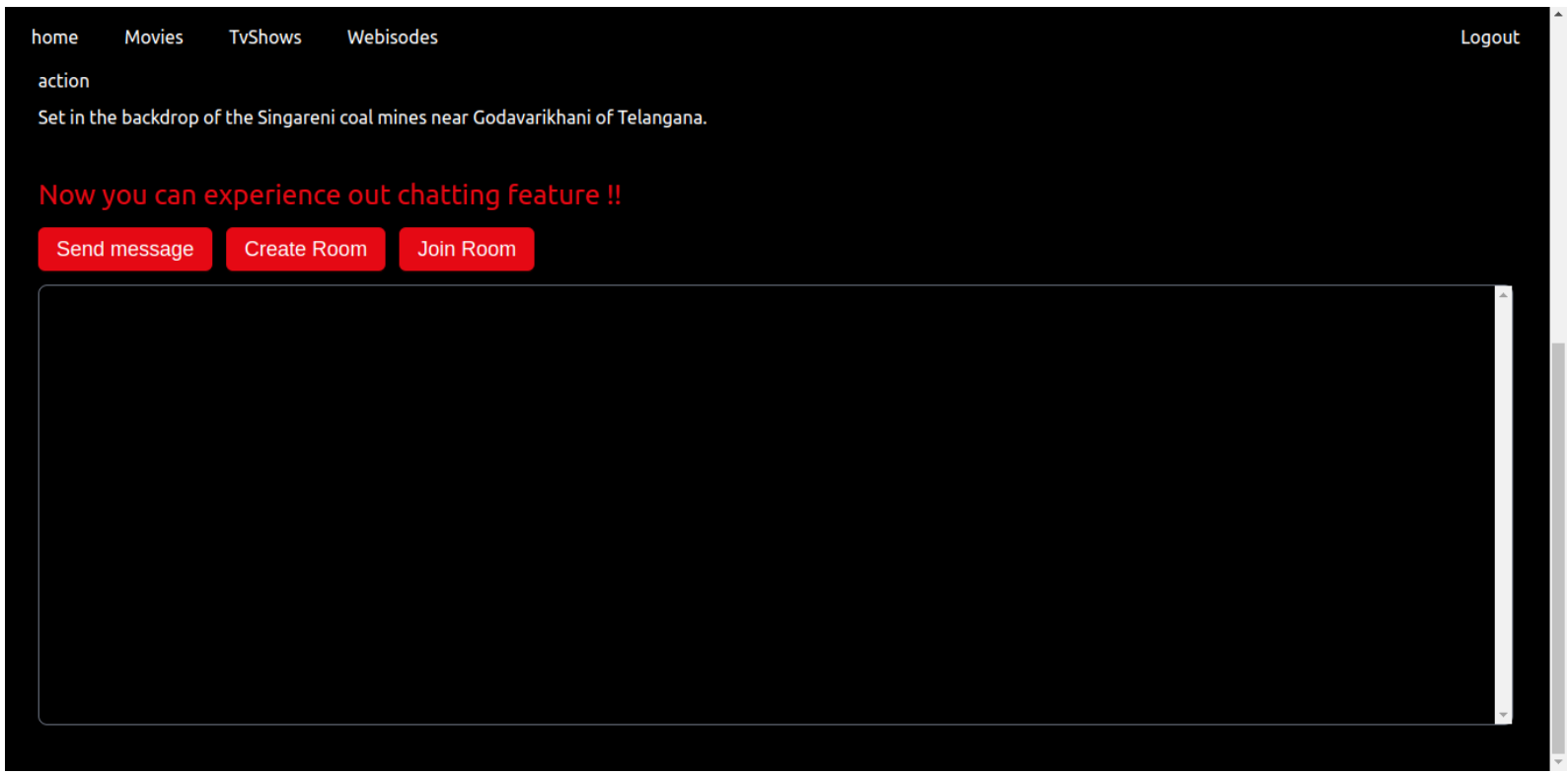


VideoDisplay Page

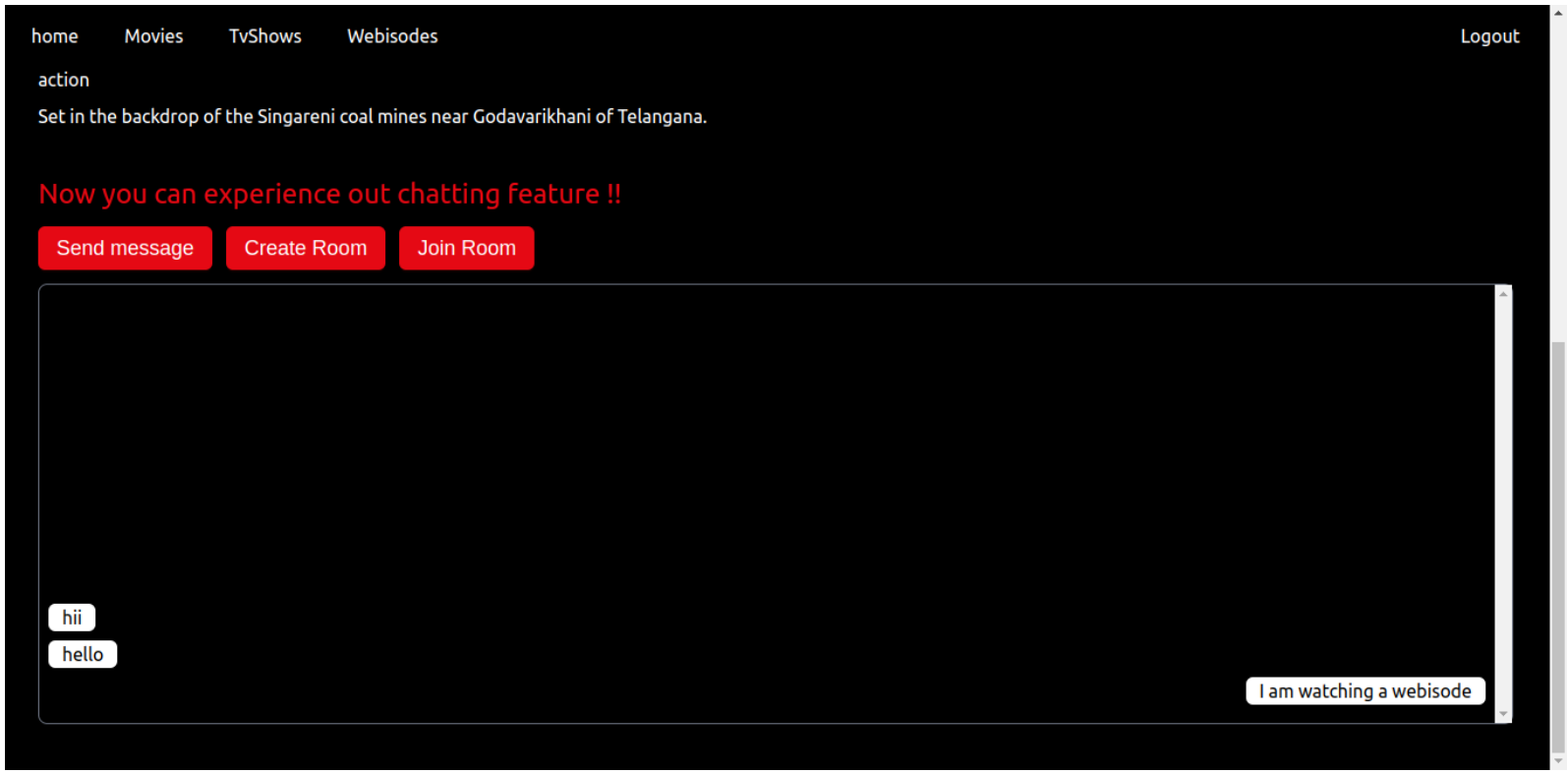
Whenever user clicks on the video card he redirected to the page where the video plays



Chatting Feature



UserChat Page



Conclusion

During the Project we learnt different concepts like

- ORM implementation using Sequelize
- Using Tailwindcss to increase productivity
- For predefined components we used Andtd library
- Bi communication implementation using Socket.io

Now a days OTT platforms are high popular.We always try to improve user experience by implementing new features.The Project’s chatting feature helps users by providing a very good movie experience and this must attract users very much.