



IT 309 SOFTWARE ENGINEERING

PROJECT DOCUMENTATION

MOVEEZ

Prepared by:

Ema Jamaković

Almina Šehić

Proposed to:

Nermina Durmić, Assist. Prof. Dr.
Aldin Kovačević, Teaching Assistant

21/06/2023

Table of Contents

1. Introduction.....	2
1.1. About the Project.....	2
1.2. Project Functionalities and Screenshots	2
2. Project Structure.....	4
2.1. Technologies	4
2.2. Database Entities.....	4
2.3. Design Patterns	5
2.4. Tests.....	5
3. Conclusion	5

1. Introduction

1.1. About the Project

The Online Cinema Reservation System is a website that allows users to easily reserve movie tickets online. It will enable users to book their tickets in advance and avoid long queues and waiting times at the cinema. The website will feature a user-friendly interface providing a seamless experience when booking movie tickets.

<https://moveez-project.netlify.app/>

1.2. Project Functionalities and Screenshots

1. User Registration and Login: Users will be required to register and create an account on the website to access the movie reservation system.

Log In

Email address:

Password:

Moveez Software Engineering Project built using MERN stack!

[Contact us](#)[Our Services](#)[Privacy Policy](#)[Terms & Conditions](#)[Career](#)

MOVEEZ Copyright © 2023

Sign Up

Email address:

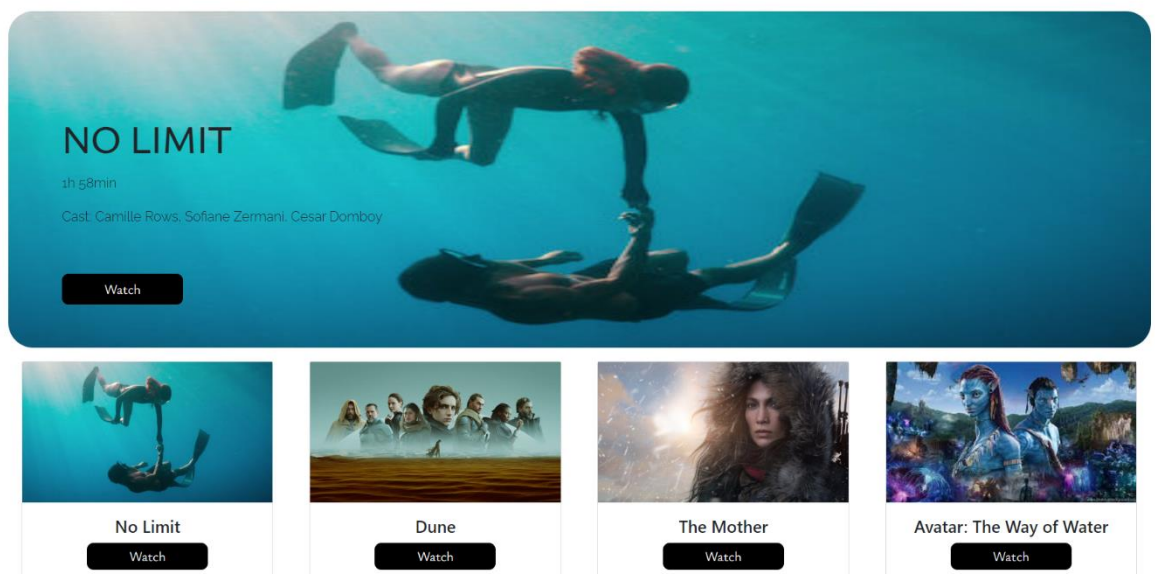
Password:

Moveez Software Engineering Project built using MERN stack!

[Contact us](#)[Our Services](#)[Privacy Policy](#)[Terms & Conditions](#)[Career](#)

MOVEEZ Copyright © 2023

2. **Movie Selection:** The website will provide a list of movies currently playing at the cinema, including showtimes, movie cast, director and runtime.



Dune**Director:** Denis Villeneuve**Cast:** Timothée Chalamet, Rebecca Ferguson, Oscar Isaac**Duration:** 1h 55min**Projection:** Mon Sep 25
2023 18:00:00 GMT+0200
(Central European Summer

Moveez Software Engineering Project built using MERN stack!

[Contact us](#)[Our Services](#)[Privacy Policy](#)[Terms & Conditions](#)[Career](#)

MOVEEZ Copyright © 2023

3. Order page with option to choose number of tickets and snacks.

**DUNE**

Timothée Chalamet, Rebecca Ferguson, Oscar Isaac

1h 55m

- 1 +

Price 6\$

Food and Drink

Popcorn-M x

x

v

[Order now](#)

Moveez Software Engineering Project built using MERN stack!

[Contact us](#)[Our Services](#)[Privacy Policy](#)[Terms & Conditions](#)[Career](#)

MOVEEZ Copyright © 2023

2. Project Structure

2.1. Technologies

In a typical MERN stack project, MongoDB is used to store and manage data, Express.js handles server-side routing and middleware, React builds the frontend user interface, and Node.js powers the backend server logic. Together, these technologies enable full-stack JavaScript development and facilitate the creation of modern, real-time web applications.

We implemented the Airbnb coding standards for JavaScript and React and Google HTML/CSS Style Guide.

2.2. Database Entities

- Users – storing registered users
- Movies – storing all the movies that are available to book
- Orders - storing orders

2.3. Design Patterns

MVC (Model-View-Controller): The MVC pattern separates an application into three interconnected components: the Model (manages data and business logic), the View (handles presentation and user interface), and the Controller (mediates between the Model and the View, handling user input and updating the Model).

Middleware: Middleware is a design pattern commonly used in Express.js, the backend framework in the MERN stack. Middleware functions are executed sequentially in the request-response cycle, allowing you to add reusable and modular components for tasks such as logging, authentication, error handling, and more. We used it for authorizing users when signing up, including JWT token authorization.

Render Props:

According to official React documentation, render props are defined as a 'technique for sharing code between React components using a prop whose value is a function'.

Render props prove really handy as they allow us to share the same state across different components. Instead of hardcoding the logic inside each component, we used a function prop to determine what to render.

React Hooks: These functions allow developers to use React without classes. There are a number of different pre-built hooks available like the Effect Hook (useEffect) and the State Hook. We used the Effect Hook for fetching data from the database and the State Hook for adding stateful behavior to functional components. It provides a way to declare and update state variables. The useState function returns an array with two elements: the current state value and a function to update the state.

2.4. Tests

We wrote Selenium tests for our application and they are located in a separate directory called selenium-testing. To run the tests, open terminal, cd to selenium-testing and run the following command: 'node testName'.

We have 5 tests and they are the following:

- testLoginPage = checking if a login button, located in the navbar section, when clicked leads to the login page;
- testSignupPage = checking if a signup button, located in the navbar section, when clicked leads to the signup page;
- testOrderPage = checking if the buttons clicked successfully lead to Order page and check if the displayed text matches the expected one;
- testSingleMovie = checking whether the single-movie component is loaded and displayed on the Home page;
- testOrderTicketNumber = checking if the buttons clicked successfully lead to Order page and if the adding more tickets button is updating the ticket number when clicked.

3. Conclusion

First of all, we are not satisfied with our project. We didn't manage to implement everything we wanted to due to the problem of connecting API calls with the routes in the backend. For user registration, signup function is working on the backend part and also in Postman (saving new users in the database) but for some unknown reason we are constantly getting the same error when trying to signup in the browser (error: "secretOrPrivateKey must have a value"), and the same goes for login. We tried solving it many ways but nothing was successful.

Sign Up

Email address:

Password:

secretOrPrivateKey must have a value

Moveez Software Engineering Project built using MERN stack!

[Contact us](#)[Our Services](#)[Privacy Policy](#)[Terms & Conditions](#)[Career](#)

```
_id: ObjectId('6492f5089e3bd0b8156582eb')
email: "user17@gmail.com"
password: "$2b$10$6RApD7ZW6Fe6Z0gzlVeIVe2L0Ro/cJt0qaMuYBTD.wbee4rb95mJq"
__v: 0
```

The most challenging part was trying to implement saving orders in the database but we managed to solve it in the end. Since it is our first time building MERN stack application and using these technologies, we are proud that we managed to implement at least some functionalities in this project but we are eager to learn more and do better in the future projects.