

INTEGRATED PROJECT REPORT

On

MOVIEMANIA

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CS 203) of

**COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2019**

in

JUNE-2022



Under the Guidance of
Dr. Ambuj Kumar Agarwal
Associate Professor, Department of
Computer Science and Engineering

Submitted By:
Tanishq Sehra
1910991409
Anmol
1910990705
Gautam Gaur
1910990670
Anush Rawat
1910991267
Akshit
1910990741

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHITKARA UNIVERSITY
PUNJAB**

INTEGRATED PROJECT REPORT

On

MOVIEMANIA

Submitted in partial fulfillment of the requirement for the
Course Integrated Project (CS 203) of

COMPUTER SCIENCE AND ENGINEERING
Batch-2019

in

JUNE-2022



Under the Guidance of

Dr. Ambuj Kumar Agarwal
Associate Professor, Department of
Computer Science and Engineering

Submitted By:

Tanishq Sehra
1910991409
Anmol
1910990705
Gautam Gaur
1910990670
Anush Rawat
1910991267
Akshit
1910990741

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHITKARA UNIVERSITY
PUNJAB

CERTIFICATE

This is to be certified that the project entitled “**MOVIEMANIA**” has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester January 2022-May-2022 is a bona fide piece of project work carried out by **Tanishq Sehra-1910991409, Anmol-1910990705, Gautam gaur-1910990670, Anush Rawat-1910991267, Akshit-1910990741** towards the partial fulfillment for the award of the course Integrated Project (CS 203) under the guidance of **Dr. Ambuj Kumar Agarwal** and supervision.

**Sign. of Project
Guide :**

Dr. Ambuj Kumar Agarwal
Associate Professor, Department of
Computer Science and Engineering

CANDIDATE'S DECLARATION

We, **Tanishq Sehra-1910991409, Anmol-1910990705, Gautam gaur-1910990670, Anush**

Rawat-1910991267, Akshit-1910990741, B.E.-2019 of the Chitkara University, Punjab hereby declare that the

Integrated Project Report entitled "**MOVIEMANIA**" is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

**Tanishq Sehra
1910991409**

**Anmol
1910990705**

**Gautam Gaur
1910990670**

**Anush Rawat
1910991267**

**Akshit
1910990741**

Place: Chitkara University

Date: 24th June 2022

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behavior and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to “**Dr. Ambuj Kumar Agarwal**” for his support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

We also extend our sincere appreciation to **Dr. Ambuj Kumar Agarwal** who provided his valuable suggestions and precious time in accomplishing our Integrated project report.

Lastly, We would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to-day experience and received lots of suggestions that improve our quality of work.

Tanishq Sehra
19190991409

Anmol
1910990705

Gautam Gaur
1910990670

Anush Rawat
1910991267

Akshit
1910990741

KEYWORDS

HTML : Hyper Text Markup Language

CSS : Cascading Style Sheets

DB : DataBase

SPA : Single Page Application

JS : Java Script

OTT : Over The Top

API : Application Programming Interface

UI : User Interface

MVC : Model-View-Controller

INTRODUCTION TO THE PROJECT

1. Problem Statement

What's the one thing that you all remember doing when the whole world was shut down? All of you might have guessed right. Enjoying movies with your loved ones. The transition from Movie theaters to the OTT platform was so smooth that we didn't know how much we have become dependent on them. So, we present to you our movie app Moviemania, a beautiful goto platform for all movie lovers. What if you saw a movie for half an hour only to find it boring for you? Your mood might take a toll and you would no longer want to watch a movie, right? But we have a solution for you: every user can share their valuable feedback with others which is visible just below the movie. Are your parents movie lovers but find it difficult to use movie apps? Not an issue with MOVIEMANIA its user interface is both attractive and simple to use. Found a movie worth watching again and again? No problem, just add it to your favorites and you are good to go. So, let's make your watching experience smooth and beautiful with MOVIEMANIA.

2. Background

Movie mania is a web application where people can enjoy movies, series etc.. Through a web browser the viewers can search for latest shows and also add them to their favorite list. The user can login using his account details or new customer can setup account very quickly. They must provide with the details like name, contact number, email address. They can set the password for their account instantly.

This project has the following functionalities:-

- Starting with the Signup Page, the signup page provides the user some functionalities and also provides option to signup for the Movie mania app. Already signed up users can directly Login through login button which is also

provided at the Signup page.

- If the user does not have an account created and is trying to login then an alertbox will appear. On the signup page we provide with the proper validation and so with the login page also. For example if the user provide incorrect details the warning will be displayed on the screen.
- The Home Page, the home page consists of a grid of all varieties of movies latest, old, action every genre , type. It also have a “Favorite” button clicking on which the user can access to all its favorite marked movies.
- The Favorite Movies page consists lists of all the movies that have been marked favorite by the user. In each row of that particular list the user also get remove button, by clicking on it the movie will be removed from the favorite list.
- When the user clicks on a movie, a new page appears. It consists of information about the movie(Title, status, runtime, release date etc.). It also consists of toggle actor view button, on clicking that button the user can see the cast of that particular movie. It also has a “ADD TO FAVORITE” button. By clicking add to favorite button the user can add that movie to their favorite movie’s list. As soon as that button is clicked, it gets toggled to “REMOVE FROM FAVORITE” button.
- The user can also share his/her views to all the other users by adding his/ her comments about the movie under the share your opinion textbox. we also provide with the like and dislike button (thumbs up/thumbs down)icon just above the opinion textbox.
- The user also get specific details about the particular movie clicked. The details of the movie is shown in a poster layout. The details of the movie is texted on the poster.

SOFTWARE REQUIREMENT SPECIFICATION

1. Methods

Our project follows the MVC approach, that is the Model View Controller Approach. The application is divided into three primary logical components using the Model-View-Controller (MVC) architectural pattern: the model, the view, and the controller. Each of these parts is designed to handle particular application development facets. One of the most popular and widely accepted web development frameworks for building scalable and flexible projects is the Model View Controller Approach.

All of the user's data-related logic is represented by the Model component. This could be any other data related to business logic or the data that is being passed between the View and Controller components.

All of the application's UI logic is implemented using the View component. For instance, the Customer view will have every UI element that the end user interacts with, such as text boxes, dropdown menus, etc.

In order to handle the business logic and incoming requests, manipulate data using the Model component, and interact with Views to generate the output, controllers serve as an interface between the Model and View components.

LOGIN PAGE

The screenshot shows a login form titled "Log In". At the top left is a logo for "MOVIE APP" with a small icon. To its right are links for "Favorite", "Signin", and "Signup". Below the title are two input fields: one for "Enter your email" with a magnifying glass icon, and another for "Enter your password" with a lock icon. Underneath these fields are two buttons: a small checkbox labeled "Remember me" and a link "forgot password". A large blue button labeled "Log In" is centered below the password field. At the bottom of the form, there is a link "Or register now!".

To begin with, like in every application we have a login page in our project as well. Login pages are extremely important in every project, especially in this day and age. In addition to granting the user access to the website, logging in enables the website to monitor the user's behavior and activity. When certain circumstances (such as closing the page, turning off the computer, a lengthy delay, etc.) take place, logging off of a website or webpage may take place manually by the user or automatically.

We used the React JS language to create the Login page.

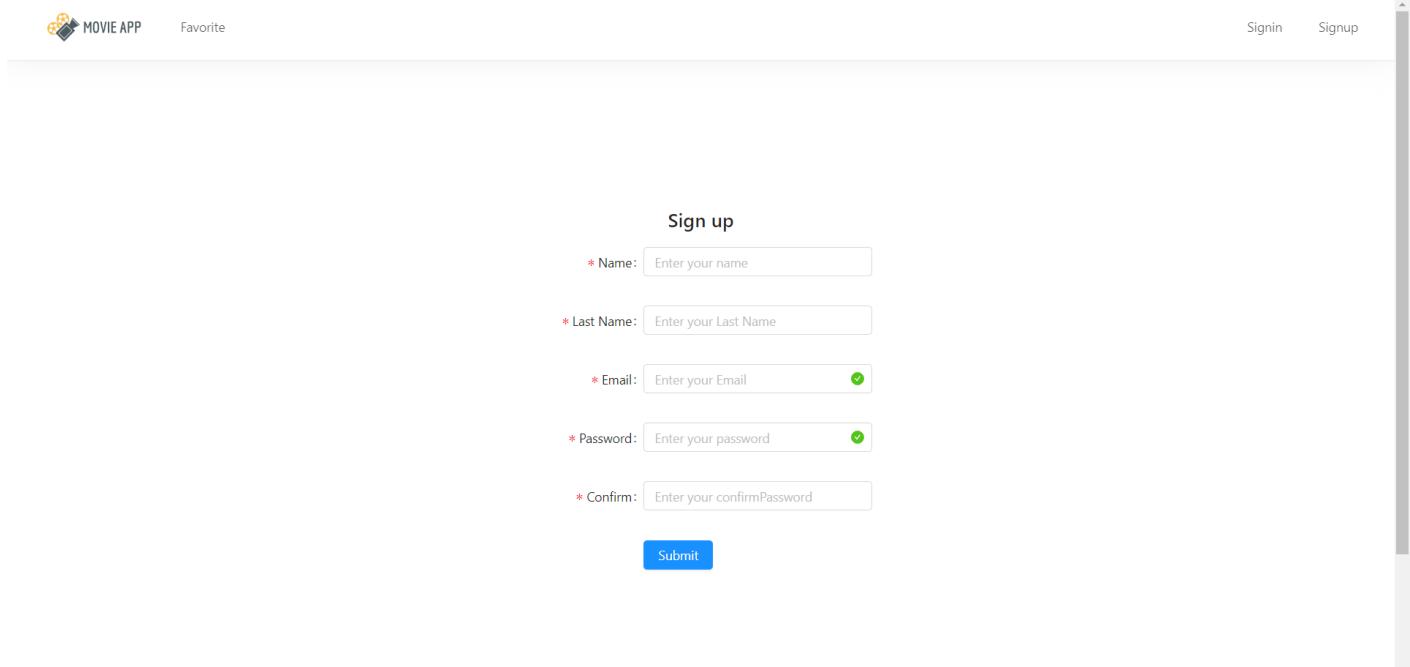
React is a front-end JavaScript toolkit that is free and open-source for creating user interfaces based on UI components. It is kept up-to-date by Meta and a group of independent programmers and businesses.

The use of React Js in this project has made many functionalities possible and easy to implement.

As soon as we begin to login, a page pops up which asks for the following information-

- User ID
- Name
- Password
- Last Name
- Email Address

All the data provided is saved in the MongoDB database.
Document-oriented database application MongoDB is cross-platform and open source.
MongoDB, a NoSQL database application, employs documents that resemble JSON
and may or may not include schemas.
Once successfully logged in, we arrive at the HOME PAGE.



A screenshot of a web browser showing a sign-up form for a "MOVIE APP". The form is titled "Sign up" and contains six input fields: "Name", "Last Name", "Email", "Password", and "Confirm". Each field has a placeholder text and a small green checkmark icon indicating validation status. A "Submit" button is located below the fields. The top navigation bar includes links for "Signin", "Signup", and "Favorite".

Sign up

* Name: Enter your name

* Last Name: Enter your Last Name

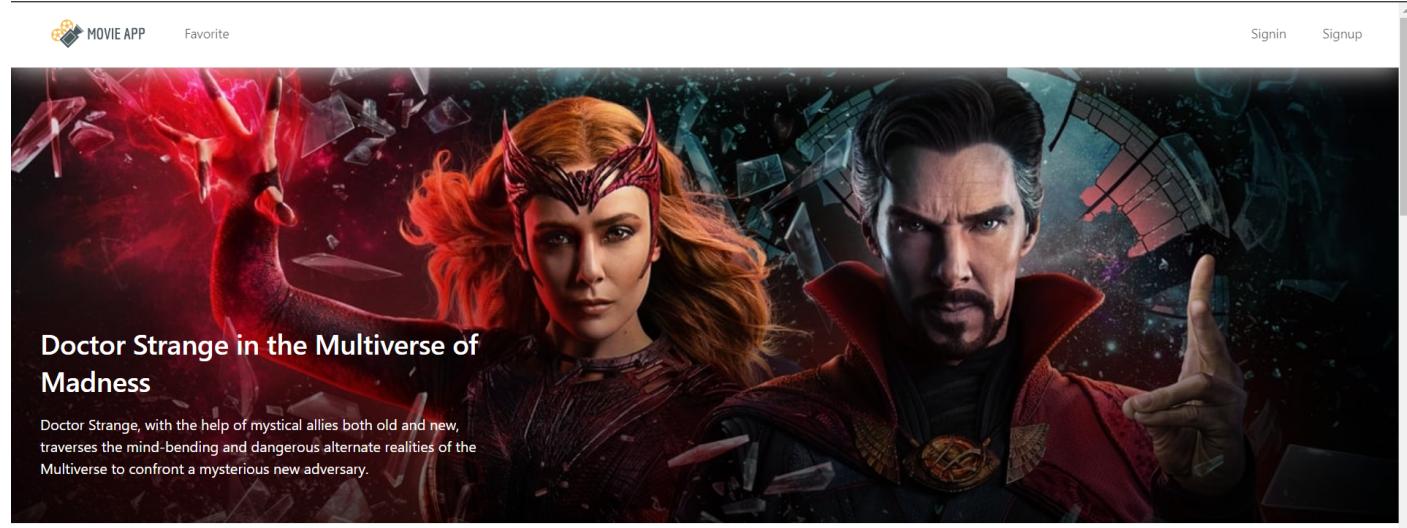
* Email: Enter your Email ✓

* Password: Enter your password ✓

* Confirm: Enter your confirmPassword

Submit

HOME PAGE

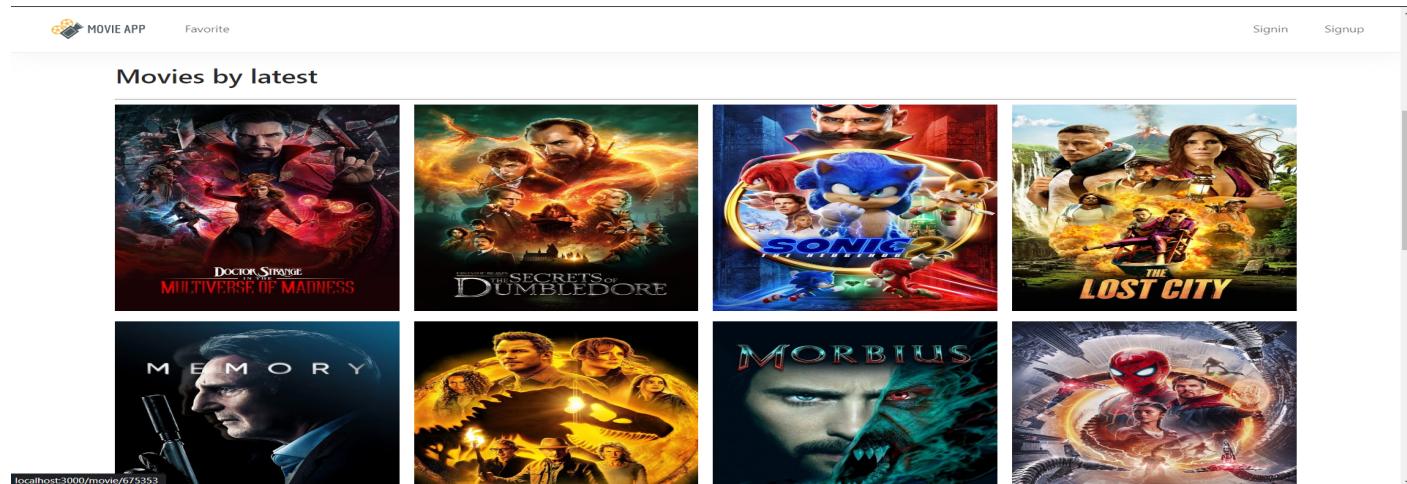


The banner features a large image of Doctor Strange and Wanda Maximoff from the movie. Below the image, the title "Doctor Strange in the Multiverse of Madness" is displayed in bold white text. A descriptive subtitle follows: "Doctor Strange, with the help of mystical allies both old and new, traverses the mind-bending and dangerous alternate realities of the Multiverse to confront a mysterious new adversary." The top navigation bar includes a "MOVIE APP" logo, a "Favorite" button, and "Signin" and "Signup" links.

Movies by latest

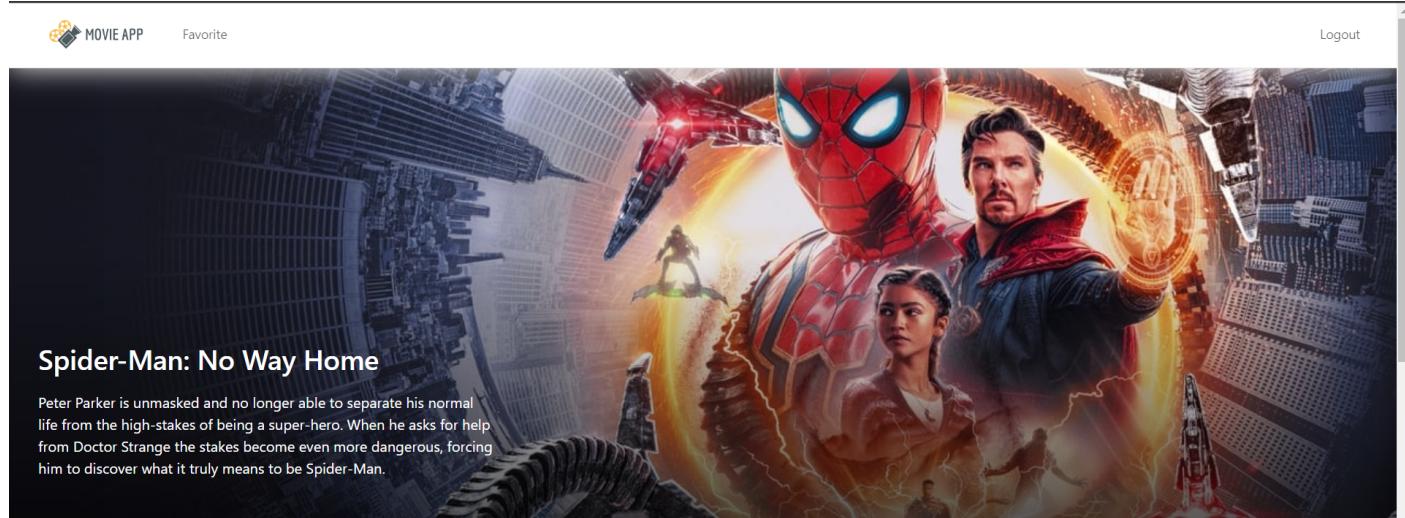


In the home page we have implemented the grid design layout. On the top of the page there is a banner which defines the home page. Every box in the grid shows a unique movie/series poster. These movies/ series are fetched from moviedb API. The movies are displayed in sorted order by recent release date. The home page consists of a topbar which provides the access to the user for all the functionalities. On the top left corner of the homepage, there is Home Page button which refreshes the page or takes you to the home page if you are on any other page. On clicking the particular Movie poster this will take you to the Movie Info Page. The grid is designed by CSS and Javascript.



The grid displays eight movie posters arranged in two rows of four. The top row includes Doctor Strange in the Multiverse of Madness, Fantastic Beasts: The Secrets of Dumbledore, Sonic the Hedgehog 2, and The Lost City. The bottom row includes Memory, Morbius, and two other posters whose titles are not clearly legible. The top navigation bar is identical to the one in the first screenshot.

MOVIE INFO PAGE



The movie poster for "Spider-Man: No Way Home" features Spider-Man in the foreground, looking intensely at the viewer. Behind him is Doctor Strange, and further back is Mjolnir. The background is a cityscape with various Marvel characters and vehicles like Iron Man's suit and a Hulk Hogan figure.

Spider-Man: No Way Home

Peter Parker is unmasked and no longer able to separate his normal life from the high-stakes of being a super-hero. When he asks for help from Doctor Strange the stakes become even more dangerous, forcing him to discover what it truly means to be Spider-Man.

Not Favorite 1

Movie Info

Title	Spider-Man: No Way Home	release_date	2021-12-15	revenue	1892000000
runtime	4 hr	vote_average	0.4		

The Movie Info Page will give you all the specific details about the particular movie the user has clicked. The poster will be enlarged to the full screen using CSS. Movie Info Page has a chart which gives you details about the movie like title, runtime, release date etc. These details are fetched from moviedb API. The “Toggle Actor View” button shows / hides all the cast photos in the similar grid view designed via CSS. This page also consists of Like and Dislike buttons. Clicking on like and dislike button the count will be stored in “MongoDB” database. The user can also share his/her views in the “share your opinion” Text Box. All the details provided by each and every particular user is stored in “MongoDB” database.

Movie Info

Title	Spider-Man: No Way Home	release_date	2021-12-15	revenue	1892000000
runtime	148	vote_average	8.1		
vote_count	13683	status	Released	popularity	1894.244

[Toggle Actor View](#)

Movie Info

[Not Favorite 1](#)

Title	Spider-Man: No Way Home	release_date	2021-12-15	revenue	1892000000
runtime	148	vote_average	8.1		
vote_count	13683	status	Released	popularity	1894.244

[Toggle Actor View](#)

0 0 0

Share your opinions about Spider-Man: No Way Home



Tanishq

Best movie of all time

Like 0 Reply 0

write some comments

[Submit](#)

FAVORITE'S PAGE

The screenshot shows a web application interface. At the top left is a logo with a movie camera icon and the text "MOVIE APP". To its right are links for "Favorite", "Signin", and "Signup". A vertical scrollbar is on the far right. The main content area has a header "Favorite Movies By Me". Below it is a message "Please Log in first...". A blue link "Go to Login page" is present. The page has a light gray background.

Favorite page contains the list of all the movies that has been selected as favorites by the user. The list is stored in MongoDB for that particular user. This list consists of Movie Title, Movie RunTime, Remove From Favorites. On hovering on any The Remove From Favorite Button will remove the movie from the list and also will remove that movie from the database.

The screenshot shows a web application interface. At the top left is a logo with a movie camera icon and the text "MOVIE APP". To its right are links for "Favorite" and "Logout". A vertical scrollbar is on the far right. The main content area has a header "Favorite Movies By Me". Below it is a table with three columns: "Movie Title", "Movie RunTime", and "Remove from favorites". The table has one row with data: "Memory" and "114 mins" under the first two columns, and a "Remove" button under the third column. The page has a light gray background.

PROGRAMMING/WORKING ENVIRONMENT

→ **Frontend Framework:** React.JS

→ **Backend:** Node.Js, Express.Js

→ **Development:** VS Code

→ **Database:** MongoDB

→ **Web Browser:** Google Chrome

→ **API:** The MovieDB Api

STEPS TO RUN

- *Open the terminal of your choice.*
- *Change to Client directory*
- *Run command “npm install”*
- *Open another terminal*
- *Change to Server directory*
- *Run command “npm install”*
- *Open another terminal*
- *Run command “npm run dev”*

CONCLUSION

The project statement used multiple concepts to achieve the outcome.

- The project helped us to understand React topics like routing, states, props, components and storage.
- We learn database connectivity.
- Gain Experience working with NodeJs with the combination of Express and MongoDB.

FUTURE SCOPE

The current application is a small replica of the product/application which we are looking forward to building in future. Our future application will have advance features and complex database storage system. The advance features like :-

- To store the personal chats between two users.
- Parental control will be provided for the parents, to guide their children with good educational and meaningful content.
- Renting movies.
- Can also browse movies online.
- Can also buy goodies related to movies.
- Users can host a movie for family or friends.
- This application can also be customized for every individual user.

This application can be evolved to the next generation movie application.

BIBLIOGRAPHY/REFERENCES

- <https://stackoverflow.com/>
- <https://reactjs.org/>
- <https://nodejs.org/en/>
- <https://cloud.mongodb.com/>
- <https://expressjs.com/>
- <https://www.themoviedb.org/>