

Technical Coding Research Innovation, Navi Mumbai, Maharashtra, India-410206

(**MOVIE COLLECTION WEBSITE (R U B Y**)

A Case-Study Submitted for the requirement of

**Technical Coding Research Innovation**

For the Internship Project work done during

**FRONT-END WEB DEVELOPMENT INTERNSHIP PROGRAM**

by

PRATIK NAWKAR(TCRIA5R129)

Date:27/06/22

Grade: Third Year(Entc)

Rutuja Doiphode

CO-FOUNDER &CEO

TCR innovation.

**About Organization**

TCR Innovation stands for Technical Coding & Research Innovation, is a best institute for industrial training and internship programs and believes in "bridging the gap between students and their knowledge in the industrial field & bringing them a step closer to their Dream. TCR Innovation focus on industrial training and offer Internship Programs where you can learn the in-demand professional skills to advance your career in Artificial Intelligence, Machine Learning, Deep Learning, Python with Automation, Computer Vision, Data Science, Digital Marketing, Python Programming, Software development, Database and more.

TCR INNOVATION is founded in Feb 2020 by Rutuja Doiphode and Saheel Ramji. Their Internship Programs are specially designed so Interns can learn new skills or enhance their skills in the most efficient way. Interns are First trained by the Mentors in their applied Domain through Live Lectures. Comprehensive Notes & Study Materials are provided along with Intensive Training in the domain. After the Training Period is Complete, Interns are given the Opportunity to work on an Industrial-Level Project which will be the Project Phase and one of the most Important Phase of their Internship. The Training completion certificate along with the Internships Certificate are awarded after the Successful Completion of your Project with Letter of Recommendation.

Email of TCR Innovation is tcrinnovation@tcrinnovation.co.in and their website is <https://tcrinnovation.co.in/>

The current status of TCR Innovation is active.

**Abstract**

*Front-end web development, also known as client-side development is the practice of producing HTML, CSS and JavaScript for a website or Web Application so that a user can see and interact with them directly. The challenge associated with front end development is that the tools and techniques used to create the front end of a website change constantly and so the developer needs to constantly be aware of how the field is developing.*

*The objective of designing a site is to ensure that when the users open up the site, they see the information in a format that is easy to read and relevant. This is further complicated by the fact that users now use a large variety of devices with varying screen sizes and resolutions thus forcing the designer to take into consideration these aspects when designing the site. They need to ensure that their site comes up correctly in different browsers (cross-browser), different operating systems (cross-platform) and different devices (cross-device), which requires careful planning on the side of the developer.*

*A front-end developer architects and develops websites and applications using web technologies (i.e., HTML, CSS, DOM, and JavaScript), which run on the Open Web Platform or act as compilation input for non-web platform environments (i.e., React Native).*

* **Introduction**

front-end web development, also called client-side development, refers to writing Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript code for a website or web application so that users can see and interact directly with them. Everything you see on a website, like buttons, links, animations, and more, were created by a front-end web developer. It is the front-end developer's job to take the vision and design concept from the client and implement it through code. Everything on the page from the logo to the search bar, buttons, overall layout and how the user interacts with the page was created by a front-end developer. Front end developers are in charge of the look and feel of the website.

Front end developers also have to make sure the website looks good on all devices (phones, tablets, and computer screens).



 **What a Front-End Developer Does:**

A front-end developer has one general responsibility: to ensure that website visitors can easily interact with the page. They do this through the combination of design, technology and programming to code a website’s appearance, as well as taking care of debugging. Whenever you visit a website, anything that you see, click on or otherwise use is the work of a front-end developer.

Common Tasks of a Front-End Developer

Although there is some variation across companies, you can generally expect a front-end developer role to include some or all of the following:

1. Optimizing the user experience.
2. Using HTML, JavaScript and CSS to bring concepts to life.
3. Developing and maintaining the user interface.
4. Implementing design on mobile websites.
5. Creating tools that improve site interaction regardless of the browser.
6. Managing software workflow.
7. Following SEO best practices.
8. Fixing bugs and testing for usability.

**** Front End Development: Commonly Used Programming Languages

Most front-end developers spend a great deal of their time working in HTML, CSS and JavaScript, making proficiency in each key to their success.

**** How Developers Use Each Programming Language

Front end developers use HTML to lay out a document’s general structure and content, CSS for styling and JavaScript for situations that require advanced interactivity. Additionally, they might use AJAX (a combination of JavaScript and XML) to update specific areas of a website without having to refresh the entire page.

**** Libraries and Frameworks

Front end developers also commonly use the libraries built on these programming languages like AngularJS, jQuery and React; and design frameworks including Foundation and Bootstrap. CSS extensions, such as SASS, provide improved modularity and power.

**** Additional Front End Development Languages

While they are less common, front-end developers may also use Python, Ruby or PHP to easily connect data with the back end of their website.

**** Common Tools Used in Front End Development

Because front end developers use a combination of design and web development in their work, the tools they use span across these areas of focus.

**** Graphic Design Tools

Before a front-end developer begins coding, they typically use graphic design tools to create a prototype of their website, which lets them test and experiment with the user interface before working out the actual code.

**** Code Editing Tools

A code editing tool is simply the program a front end developer chooses to use to write the code for their website. Some developers prefer to use a lightweight editor like Notepad, while others opt for something more feature-rich like Visual Studio or Eclipse.

* **HTML5 and CSS**

**•** HTML

HTML stands for Hyper Text Markup Language. It is used to design web pages using markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. Markup language is used to define the text document within tag which defines the structure of web pages. HTML 5 is the fifth and current version of HTML. It has improved the markup available for documents and has introduced application programming interfaces (API) and Document Object Model (DOM).

**•** Features

1. It has introduced new multimedia features which supports audio and video controls by using <audio> and <video> tags.
2. There are new graphics elements including vector graphics and tags.
3. Enrich semantic content by including <header> <footer>, <article>, <section> and <figure> are added.
4. Drag and Drop- The user can grab an object and drag it further dropping it on a new location. 5) Geo-location services- It helps to locate the geographical location of a client.
5. Web storage facility which provides web application methods to store data on web browser.
6. Uses SQL database to store data offline.
7. Allows to draw various shapes like triangle, rectangle, circle, etc.
8. Capable of handling incorrect syntax.
9. Easy DOCTYPE declaration i.e. <!doctype html>
10. Easy character encoding i.e. <meta charset=”UTF-8″>

HTML is the skeleton of all web pages and its most basic building block. It’s often the first language learned by developers, marketers, and designers and is core to front-end development work.

HTML provides structure to the content appearing on a website, such as images, text, or videos. Right-click on any page on the internet, choose “Inspect,” and you’ll see HTML in a panel of your screen.

A markup language is a computer language that defines the structure and presentation of raw text.

In HTML, the computer can interpret raw text that is wrapped in HTML elements.

Hypertext is the text displayed on a computer or device that provides access to other text through links, also known as hyperlinks.

Learning HTML is the first step in creating websites, but even a bit of knowledge can help you inject code snippets into newsletter, blog, or website templates. As you continue learning, you can layer HTML with CSS and JavaScript to create visually compelling and dynamic websites.

**•** CSS (Cascading Style Sheets)

Cascading Style Sheets (CSS) is used to format the layout of a webpage.

With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colours are to be used, different displays for different devices and screen sizes, and much more!

**•** CSS can be added to HTML documents in 3 ways**:**

1)Inline - by using the style attribute inside HTML elements

2)Internal - by using a <style> element in the <head> section

3)External - by using a <link> element to link to an external CSS file

**•** Inline CSS

An inline CSS is used to apply a unique style to a single HTML element. An inline CSS uses the style attribute of an HTML element.

The following example sets the text color of the <h1> element to blue, and the text color of the <p> element to red:

<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>

**•** Internal CSS

An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

The following example sets the text color of ALL the <h1> elements (on that page) to blue, and the text color of ALL the <p> elements to red. In addition, the page will be displayed with a "powderblue" background color:

| <!DOCTYPE html>  <html>  <head> <style>  body {background-color: powderblue;}  h1{color: blue;} p{color: red;}  </style>  </head>  <body>  <h1>This is a heading</h1>  <p>This is a paragraph. </p>  </body>  </html> |
| --- |

**•** External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page.

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<h1>This is a heading</h1>

<p>This is a paragra

ph.</p>

| body {  background-color: powderblue;  } h1 {  color: blue;  }  p {  color: red;  } |
| --- |

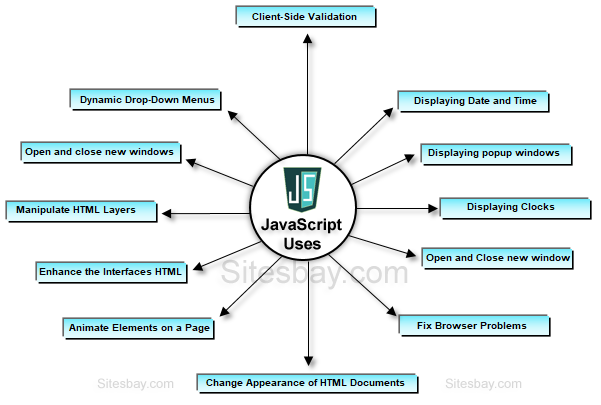
</body> </html>

Styles.css file =>

* **JavaScript**

JavaScript is a light-weight object-oriented programming language that is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language. JavaScript enables dynamic interactivity on websites when it is applied to an HTML document.

JavaScript helps the users to build modern web applications to interact directly without reloading the page every time. JavaScript is commonly used to dynamically modify HTML and CSS to update a user interface by the DOM API. It is mainly used in web applications.



JavaScript is commonly used for creating web pages. It allows us to add dynamic behaviour to the webpage and add special effects to the webpage. On websites, it is mainly used for validation purposes. JavaScript helps us to execute complex actions and also enables the interaction of websites with visitors. Using JavaScript, it is also possible to load the content in a document without reloading the webpage.

* **Frameworks and Libraries**

A web framework (WF) or web application framework (WAF) is a software framework that is designed to support the development of web applications including web services, web resources, and web APIs. Web frameworks provide a standard way to build and deploy web applications on the World Wide Web. Web frameworks aim to automate the overhead associated with common activities performed in web development. For example, many web frameworks provide libraries for database access, templating frameworks, and session management, and they often promote code reuse. Although they often target development of dynamic web sites, they are also applicable to static websites.



* **Bootstrap**

1. Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.
2. It is absolutely free to download and use.
3. It is a front-end framework used for easier and faster web development.
4. It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others.
5. It can also use JavaScript plug-ins.

It facilitates you to create responsive designs

**** History of Bootstrap

Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter. It was released as an open-source product in August 2011 on GitHub.

In June 2014 Bootstrap was the No.1 project on GitHub.

**** Why use Bootstrap

Following are the main advantage of Bootstrap:

1. It is very easy to use. Anybody having basic knowledge of HTML and CSS can use

Bootstrap.

1. It facilitates users to develop a responsive website.
2. It is compatible on most of browsers like Chrome, Firefox, Internet Explorer, Safari and Opera etc.

**** What Bootstrap package contains

Scaffolding: Bootstrap provides a basic structure with Grid System, link styles, and background. CSS: Bootstrap comes with the feature of global CSS settings, fundamental HTML elements style and an advanced grid system.

Components: Bootstrap contains a lot of reusable components built to provide iconography, dropdowns, navigation, alerts, pop-overs, and much more.

JavaScript Plugins: Bootstrap also contains a lot of custom jQuery plugins. You can easily include them all, or one by one.

Customize: Bootstrap components are customizable and you can customize Bootstrap's components, LESS variables, and jQuery plugins to get your own style.

* **ReactJS**

ReactJS is JavaScript library used for building reusable UI components. According to React official documentation, following is the definition −

React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

**React Features**

* JSX − JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
* Components − React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
* Unidirectional data flow and Flux − React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.
* License − React is licensed under the Facebook Inc. Documentation is licensed under CC BY

4.0.

**React Advantages**

* Uses virtual DOM which is a JavaScript object. This will improve apps performance, since JavaScript virtual DOM is faster than the regular DOM.
* Can be used on client and server side as well as with other frameworks.
* Component and data patterns improve readability, which helps to maintain larger apps.

**React Limitations**

* Covers only the view layer of the app, hence you still need to choose other technologies to get a complete tooling set for development.
* Uses inline templating and JSX, which might seem awkward to some developers.
* *Cross-Browser and Device Testing*

Cross Browser testing is a type of non-functional testing that lets you check whether your website works as intended when accessed through:

Different Browser-OS combinations i.e., on popular browsers like Firefox, Chrome, Edge, Safari—on any of the popular operating systems like Windows, macOS, iOS and Android.

Different devices i.e., users can view and interact with your website on popular devices— smartphones, tablets, desktops and laptops etc.

Assistive Tools i.e., the website is compatible with assistive technologies like screen readers for individuals who are differently abled.

**** What Features are Analyzed in a Browser Test?

Compatibility testing includes everything, but you may not always have the time for that. To do it right, product teams constrain their testing with a test specification document (test specs) which outlines broad essentials—a list of features to test, what browsers/versions/ platforms to test on in order to meet the compatibility benchmark, test scenarios, timelines, and budget.

You can categorize the features that will undergo testing like this:

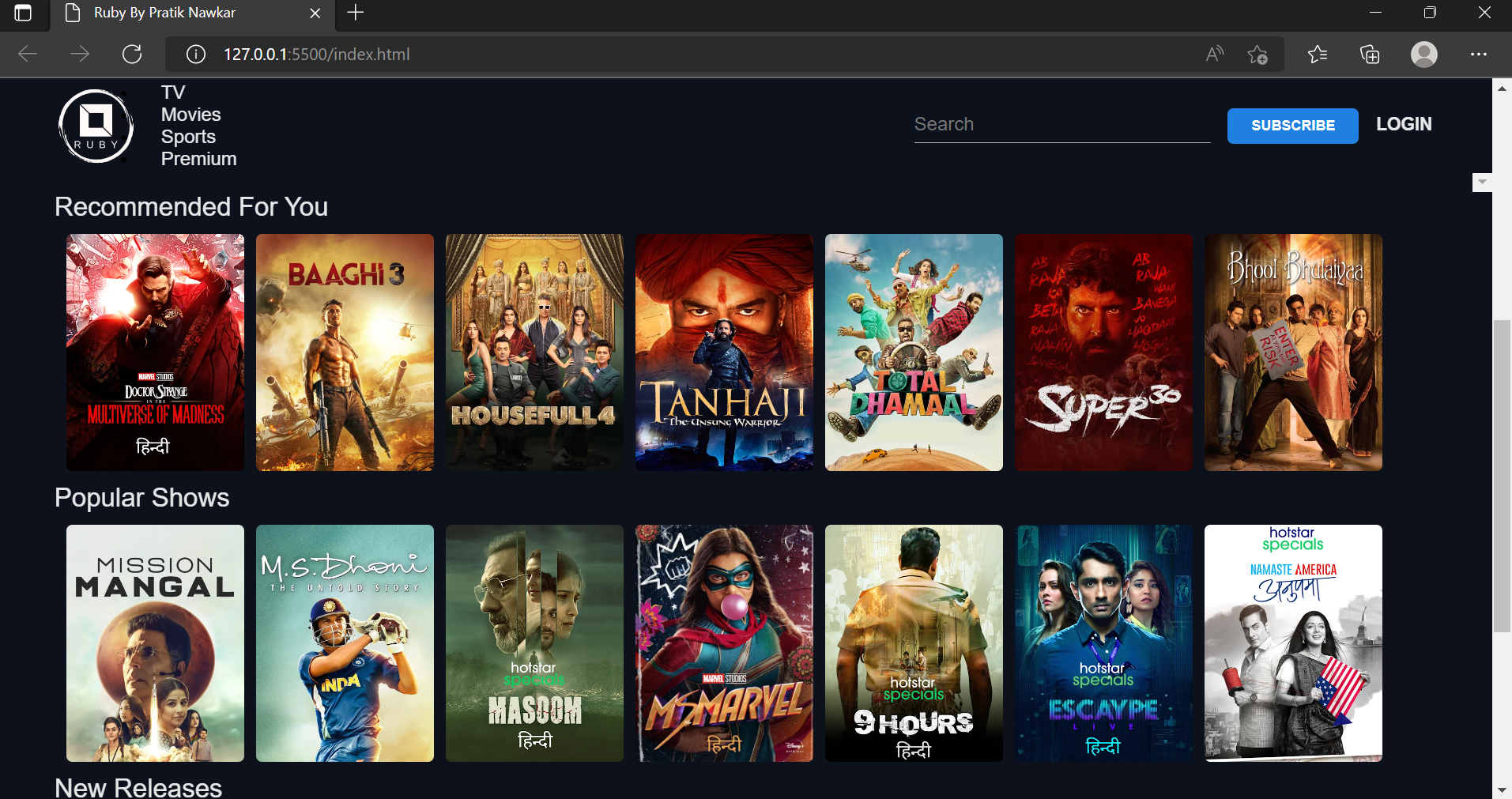
1. Base Functionality: To ensure that basic functionality works on most browser-OS combinations. For example, you could be testing to verify that:
2. All dialogs boxes and menus are working as intended
3. All form fields accept inputs after validating them correctly
4. Website handles first-party cookies (and features like personalization that are dependent on them) correctly
5. Seamless touch input for mobiles or tablets
6. Design: This ensures that the website’s appearance—fonts, images, and layout—matches the specifications shared by the Design team.
7. Accessibility: Accounts for compliance with Web Content Accessibility Guidelines (WCAG) to enable differently-abled users to access the website.
8. Responsiveness: Verifies that design is fluid and fits different screen sizes/orientations.

**Final Project**

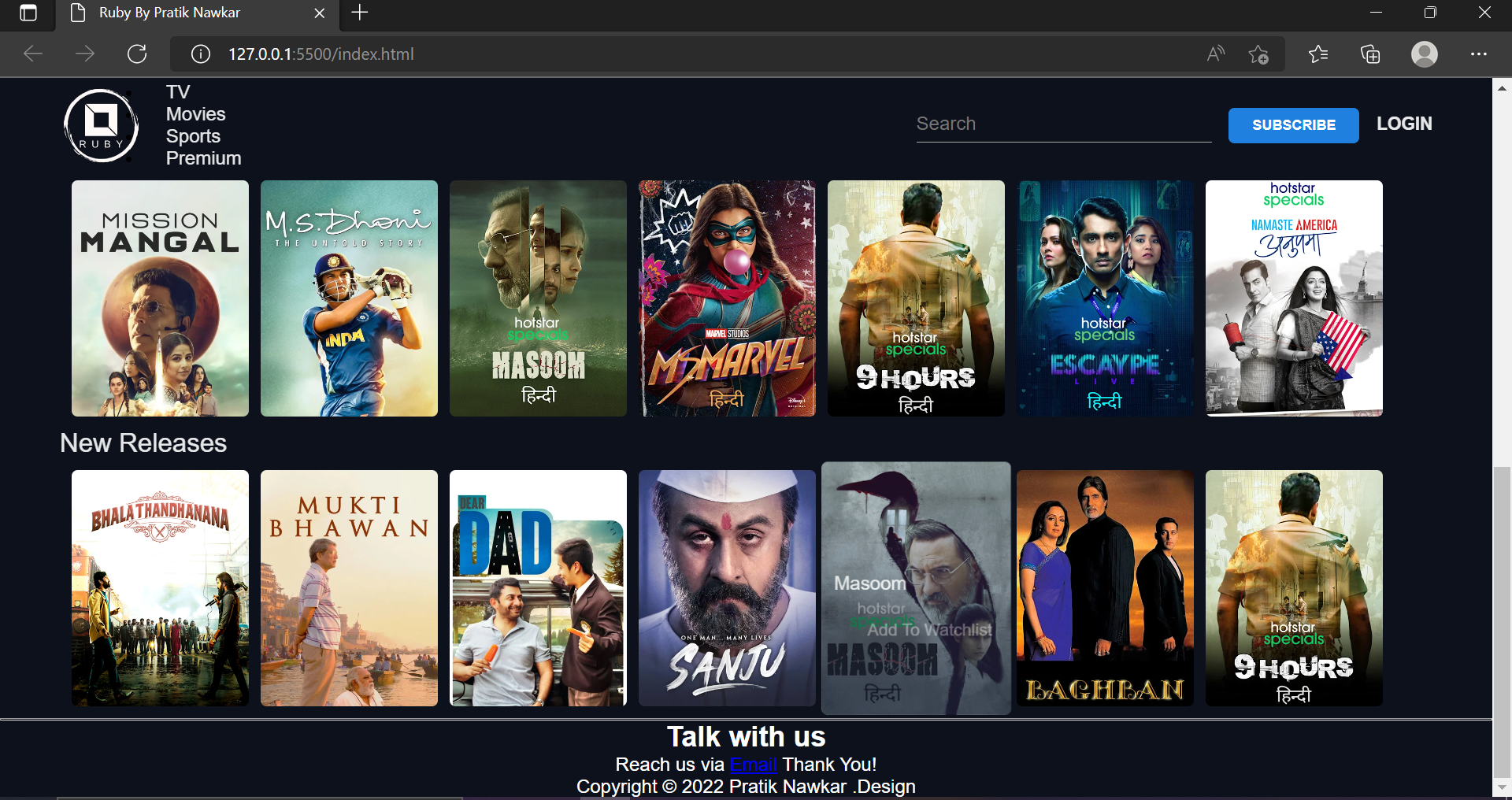
**HOMEPAGE**



**ABOUT PAGE**

****

**New Release Movie**

****

**Conclusion:**

In the [**S**tarting a Movie Website](https://promovieblogger.com/tag/starting-a-movie-website/) series, you have learned how to start, setup, and bring online a movie website. Popular online publishing platforms, found at the bottom of, have made it extremely easy for anyone to get into the game and bring their own movie website to market. Starting a movie website is an exciting adventure. I hope this series has given you the tools to make it more exciting, easier, and rewarding. Its up to you to use the steps outlined in this series in the order and the way you see fit. Maybe your movie website is already established and you have used the info in the latter parts of the series to augment your site offerings and/or its back-end mechanics. That is fine by us. We never intended for this series to be just for green horns. We believe the order in which we have outlined the steps within [Starting a Movie Website: Introduction](https://promovieblogger.com/starting-a-movie-website-introduction/) is best. If you think you have found a better order for the steps, let us know. There is always room for improvement. If you have used the [Starting a Movie Website](https://promovieblogger.com/tag/starting-a-movie-website/) steps to create a movie website or to augment an existing one, please post the name of the site in the comments section below. We look forward to seeing what we have helped others to achieve. Good luck and be persistent.

* **CONCLUSION**

My objective was to manufacture the front end of the site venture given by TCR Innovation. I have taken every necessary step utilizing HTML and CSS and adapted well about these subjects. The coursework appropriately set me up for the expert substance of the temporary job. I confronted a few difficulties while doing the CSS cushioning and situating the site substance. After work, I've figured out how to work in an expert way

In a nutshell, this internship has been an excellent and rewarding experience. I can conclude that there have been a lot I’ve learnt from my work at the research centre. Needless to say, the technical aspects of the work I’ve done are not flawless and could be improved provided enough time. As someone with no prior experience in JavaScript whatsoever I believe my time spent in research and discovering new languages was well worth it and contributed to finding an acceptable solution to an important aspect of web design and development. Two main things that I’ve learned the importance of are time-management skills and self-motivation. Although I have often stumbled upon these problems at University, they had to be approached differently in a working environment.

* **REFERENCE**

1. developer.mozilla.org

For front-end ( HTML, CSS & JavaScript ) reference and learning.

1. w3schools.com and w3resource.com

For HTML, CSS, JavaScript reference and learning.

1. htmlreference.io and cssreference.io

For HTML and CSS visual guide & reference from the creator of Bulma.

1. javascript.info

Amazing resource to learning JavaScript.

1. css-tricks.com

Learn everything about CSS (they've got some seriously awesome tricks).