TextTalker

A PROJECT REPORT
for
Mini Project (KCA353)
Session (2023-24)

Submitted by

Priyanshu Singh
2200290140118
Parth Gupta
2200290140105

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Dr. Amit Kumar Gupta
Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206

MARCH 2024

DECLARATION

I hereby declare that the work presented in report entitled "TextTalker " was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University of Institute. I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, that are not my original contribution. I have used quotation marks to identify verbatim sentences and give credit to the original authors/sources. I affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name: Priyanshu Singh

Roll No.: 2200290140118

Name: Parth Gupta

Roll No: 2200290140105

CERTIFICATE

Certified that **Priyanshu Singh (2200290140118), Parth Gupta (2200290140105)** has/ have carried out the project work having "**TextTalker**" (**Mini Project-KCA353**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Priyanshu Singh
(2200290140118)
Parth Gupta
(2200290140105)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Dr. Amit Kumar Gupta	**Dr. Arun Tripathi**
Professor	**Head**
Department of Computer Applications	**Department of Computer Applications**
KIET Group of Institutions, Ghaziabad	**KIET Group of Institutions, Ghaziabad**

PRIYANSHU SINGH , PARTH GUPTA

ABSTRACT

In this we will build a web-based Zoom Clone application using HTML, CSS, JavaScript, Node.js, Express.js, Socket.io, npm, nodemon. It is Social Networking digital communication web application that provides video chat between computers, tablets and mobile devices through web browser. HTML describes the structure of a Web page. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. JavaScript is used mainly for enhancing the interaction of a user with the webpage. Node.js is primarily used for non-blocking, event-driven servers, due to its single-threaded nature. It's used for traditional websites and back-end API services but was designed with real-time, push-based architectures in mind. This project is to create a application with a server and client that enables video conferencing.

ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Dr. Amit Kumar Gupta**, Professor for his guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to Dr. Arun Kumar Tripathi, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Priyanshu Singh
(2200290140118)

Parth Gupta
(2200290140105)

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. OVERVIEW

Our project is an example of a client server application. It is a real-time and multi-platform application and can be used by many users. It provides video telephony and online chat service through a peer-to-peer service platform and it uses tele-conferencing and tele-commuting. It is made up of 2 applications the client application, which runs on the users' Pc and server application, which runs on any pc on the network. To start Video chatting client should get connected to the server.

1.2 PROBLEM STATEMENT

In the evolving landscape of remote work and digital communication, there is a pressing need for a comprehensive and reliable Real-Time Video Conferencing and Messaging Application. The current solutions available in the market lack a seamless integration of both video conferencing and instant messaging features within a single application, often requiring users to switch between platforms for different modes of communication. Additionally, security and privacy concerns have become paramount, highlighting the necessity for a robust server-client architecture that ensures end-to-end encryption and data protection.

1.3 OBJECTIVE

The primary goal of this project is to design and implement a Real-Time Video Conferencing and Messaging Application that addresses the aforementioned challenges.

1.4 SCOPE

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of passed year perfectly and vividly. It also helps in current all works relative to Online E-commerce System. It will be also reduced the cost of collecting the management & collection procedure will go on smoothly. Our project aims at Business process automation, i.e. we have tried to computerize various processes of Online E-commerce System.

- In computer system the person has to fill the various forms & number of copies of the forms can be easily generated at a time.
- In computer system, it is not necessary to create the manifest but we can directly print it, which saves our time.
- To assist the staff in capturing the effort spent on their respective working areas.
- To utilize resources in an efficient manner by increasing their productivity through automation.
- The system generates types of information that can be used for various purposes.
- It satisfy the user requirement
- Be easy to understand by the user and operator
- Be easy to operate
- Have a good user interface
- Be expandable
- Delivered on schedule within the budget.

1.5 **FEATURE**

- Product and Component based
- Creating & Changing Issues at ease
- Query Issue List to any depth
- Reporting & Charting in more comprehensive way
- User Accounts to control the access and maintain security
- Simple Status & Resolutions
- Multi-level Priorities & Severities.
- Targets & Milestones for guiding the programmers
- Attachments & Additional Comments for more information
- Robust database back-end
- Various level of reports available with a lot of filter criteria's • It contain better storage capacity.
- Accuracy in work.
- Easy & fast retrieval of information.
- Well-designed reports.
- Decrease the load of the person involve in existing manual system.
- Access of any information individually.
- Work becomes very speedy.
- Easy to update information

1.6 HARDWARE/ SOFTWARE REQUERIMENT

S. N.	**Description**
1	PC with 5 GB or more Hard disk.
2	PC with 2 GB RAM.
3	PC with core i3 or above processor.

Table 1.1 Hardware Requirements

S. N.	Description	Type
1	Operating System	Windows 10 or 11 or Ubuntu 18.04 or above
2	Language	Python 3
3	Front End	React 17
4	IDE	Google Colab, VS Code
5	Browser	Chrome, Firefox, Edge

Table 1.2 Software Requirements

CHAPTER 2

FEASIBILITY STUDY

After doing the project Online E-commerce System, study and analyzingall the existing or required functionalities of the system, the next task is todo the feasibility study for the project. All projects are feasible - given unlimited resources and infinite time. Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements.

2.1 ECONOMICAL FEASIBILITY

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible costfactor.

- All hardware and software cost has to be borne by the organization.
- Overall we have estimated that the benefits the organization is goingto receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

2.2 TECHINICAL FEASIBILITY

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in thesystem, as described in the System Requirement Specification (SRS), and checked if everything was possible using different type of frontendand backend platforms.

2.3 OPERATIONAL FEASIBILITY

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essenceof the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing.

2.4 BEHAVIOURAL FEASIBILITY

The behavioural feasibility of our proposed e-commerce project is highly promising, as it aligns seamlessly with the contemporary attitudes and preferences of potential users. Market research indicates a widespread acceptance of online shopping, and our platform caters to this trend by offering a user-friendly interface, responsive design for mobile devices, and robust security measures. The project addresses user concerns through clear communication of security protocols and trust-building elements such as secure payment gateways. Additionally, our commitment to providing excellent customer support and feedback mechanisms ensures a positive user experience, building trust and credibility. With a focus on cultural adaptability, intuitive on boarding processes, and integration with social media platforms, the project aims to not only meet but exceed user expectations, fostering a strong and loyal customer base in the competitive e-commerce landscape.

CHAPTER 3

BASIC INTRODUCTION OF TECHNOLOGY

3.1. JAVASCRIPT

JavaScript is an interpreted, lightweight programming language. It is intended to use in the development of network-centric software. It works in conjunction with Java and is also compatible with Java. It is extremely simple to incorporate due to its integration with HTML. It is cross-platform and open source. JavaScript has been referred to as a scripting language since 1995. However, its applications in web creation are numerous nowadays. NodeJS sparked the development of various JS frameworks and libraries by web professionals. JS was initially developed as an in-browser scripting language with the aim of adding interactivity to a static web page. However, it has recently expanded beyond its initial boundaries. A variety of JavaScript frameworks have appeared in a comparatively short amount of time, enabling programmers to create web apps of greater sophistication and quality in less time. Additionally, the programming language has gained attention with non-frontend developers. A growing number of app developers are beginning to understand its utility in desktop and mobile growth. As a result, businesses in the information technology and web creation industries also have a broader range of opportunities for designing various types of applications. One of them is designing software exclusively in JS for various platforms. It is not possible to list all the JavaScript frameworks and libraries available as JavaScript's universe is too wide and too dynamic. Some useful JavaScript frameworks are Angular, React, jQuery, Vue.js, Ext.js, Ember.js, Meteor, Mithril, Node.js, Polymer, Aurelia, Backbone.js etc. [1] 2.1 Brief history on JavaScript Brendan Eich developed JavaScript in 1995 working at Netscape Communications Corporation, the company that produced the venerable Netscape Navigator web browser. Java was gaining popularity at the time, and 8 Netscape Communications attempted to integrate it into Netscape Communicator. Mocha was the name given to the early implementation of JavaScript. It was renamed Live Script not long after a version of mocha was integrated into Netscape communicator. It was changed to LiveScript in May 1995. In December of the same year, it was renamed again, this time to JavaScript. This initial iteration of JavaScript described many of the great characteristics (such as its object-model) for which this coding language is now famous. Additionally, it possessed plenty of advanced features that allowed it to gradually outgrow its original intent.

These characteristics include the following:

- First-class object functions
- Similar Syntax to Java
- Prototype-based object model.

3.2. PROS AND CONS OF JAVASCRIPT

Majority of developers believe JavaScript to be the most common programming language, according to the 2019 Stack Overflow Survey. This describes its widespread usage by developers working in a variety of product domains. Although it is no secret that JavaScript is a particularly preferred among most developers, there are certain advantages and disadvantages to JavaScript. These JavaScript benefits and drawbacks are crucial when choosing JavaScript. Without diminishing the value and utility of JavaScript, it is important to recognize there can be instances where a different language is a better fit. A side-by-side comparison of the advantages an disadvantages of JavaScript is undoubtedly the perfect way to determine if this is the correct programming language for your next project

3.3. ADVANTAGES OF JAVASCRIPT

- Despite of where you host JavaScript, it is often run on the client side to save bandwidth and speed up the production phase.
- XMLHttpRequest is a critical entity in JavaScript that was created by Microsoft. The entity call provided by XMLHttpRequest serves as an asynchronous HTTP request to the server to transfer data between the two parties without reloading the page.
- The primary benefit of JavaScript is its potential to help and deliver similar results in all modern browsers.
- Global corporations contribute to community growth by launching vital programs. Google (which invented the Angular framework) or Facebook are two examples (created the React.js framework).
- JavaScript is a scripting language that works well with other languages and can be used in a wide range of applications.
- Numerous open-source programs assist developers in adding JavaScript.
- There are numerous courses available in the field of JavaScript, that provide knowledge and resources to learn it rapidly and easily.
- There are many methods for using JavaScript through Node.js servers. It is possible to create a complete JavaScript application utilizing just JavaScript.

3.4. LIMITATION

- Since JavaScript is directly used in web pages and client browsers, it can exploit the user's mechanism and executing malicious code on the client computer.
- Different browsers view JavaScript differently at times. Different layout engines can make JavaScript different, resulting in functionality and interface inconsistency. The majority of JavaScript is based on manipulating the browser's DOM components. Additionally, various 10 browsers, specifically Internet Explorer, provide different types of access to objects.

• JavaScript is a very old machine-based scripting language, and there is other technology that perform the same function (for example, JQuery) more efficiently and easily.

• If JavaScript is disabled in the window, the whole JavaScript code is not executed.

• The JavaScript file is downloaded to the client computer, allowing others to translate and reuse the text

3.5. REACT JS

It is a free and open-source JavaScript library for developing user interfaces for single-page apps. React enables developers to build massive web apps that dynamically update data without requiring the user to refresh the website.

Reacts primary goal is to be fast, scalable, and basic.

3.5.1 History

In 2011, Facebook developers were having issues maintaining their code. Facebook was growing expeditiously and with increasing amounts of features it was getting impossible for their team to keep up with all the updates and maintenance for their website. They needed a more efficient and fast way of maintaining their website. Jordan Walke, a software engineer working for Facebook, created a library which in later days came to be known as React.js. React is heavily influenced by XHP which is a HTML component framework for PHP. React was first implemented in Facebook's newsfeed in 2011. After seeing its results in Facebook, Instagram also implemented React in their system. In due 11 course React became popular and grew exponentially and in May 2013 it was made Open Source at JSConf US.

3.5.2 Features

React consists of many useful features to make life easier for developers. Some of the features of react are below. JSX JSX is a syntax extension to JavaScript. JSX specifies how the user interface appears. The primary purpose of JSX is to write HTML structures inside the same file as JavaScript code, which simplifies the code's comprehension and debugging. This avoids the usage of complicated JavaScript DOM frameworks.

3.6. WEB SOCKET

WebSocket is a stateful mechanism the ensures that communication between the client and server can continue before one of the parties wishes to close it (client or server). After the server or client closes the connection, it is terminated at both ends. The following sections discuss the main characteristics of web sockets.

- Real-time web application Real-time web application allows use of a web socket to view data that is continuously sent by the server at the client end. WebSocket’s allow continuous data transmission over an already-established network, which speeds up the application.
- Gaming application The server receives data in gaming applications without requiring the user interface to be refreshed. In gaming apps, the server collects data without refreshing the user interface; the UI is automatically reset without initiating a new request, which is incredibly advantageous in a gaming program.
- Chat program The chat application makes use of WebSocket to establish a connection only once for the purpose of exchanging, posting, and broadcasting messages between subscribers. It makes use of the same WebSocket connection to send and receive messages, as well as one-to-one message transfer

CHAPTER 4

TECHNOLOGY USED

4.1. HTML

HTML is an abbreviation for Hypertext Markup Language. It enables users to build and organize web pages and applications by allowing them to create and structure sections, paragraphs, heading etc. HTML does not support complex features. Other than that, it enables the organization and formatting of documents in a manner comparable to Microsoft Word. As we deal with HTML, we mark up a website page using basic code structures (tags and attributes). For instance, we can construct a heading by enclosing the text within a beginning

In 1991, Tim Berners-Lee, a scientist at Switzerland's CERN science centre, released the first edition of HTML. Since then, each modern edition of the HTML language has introduced new markup tags and attributes (tags modifiers). There are currently 140 HTML tags, several of which do not support modern browsers. The language's most significant update occurred with the release on HTML5 in 2014. It added many new semantic tags to the markup, such as , , , that disclose the context of their own material.

4.2. CSS

CSS is an abbreviation for cascading style sheets. In a nutshell, CSS is a programming language that enhances the visual appeal of a website over bland or uninspiring text. If HTML is primarily concerned with textual material, CSS is concerned with visual structure, layout, and aesthetics. HTML is a markup language, and CSS is a language for creating style sheets. [13] CSS is a rule-based code, which ensures that you define rules specifying style classes to be applied to individual elements or groups of elements on your web page. 17 To begin the rule, a selector is used. This option specifies the HTML attribute to be styled. And there are the curly braces. Among such would be one or more statements in the context of property and value pairs. Each pair specifies a property of the chosen element(s), accompanied by the value to be added to the property. Before the colon, we have the property and after the colon, we have the value. Allowable values for CSS properties differ due to the property being described. In the following example we are styling the paragraph or tag. Hence, p is a selector. We have one declaration inside the curly braces which has the property of color, and value is given red. Hence, it will change the color of all paragraphs in the HTML document to blue. p { color: blue; } Both web technology, such as HTML and CSS, are specified in lengthy documentation referred to as specifications (or referred to as "specs" by the W3C, ECMA, or Khron). CSS was developed by the CSS Working Group of the World Wide Web Consortium. This group consists of organisations and individuals committed to CSS. The CSS Working Group creates new CSS rules in response to web developer requests for new functionality or browser requests for enhanced capabilities... CSS features are being introduced at a rapid rate.

4.3 NODE.JS

One of the most common areas of confusion for newcomers to Node.js is determining what it is. Is it a distinct language, a structure built on top of another, or something else? Node.js is not a modern language, nor is it merely a JavaScript framework. It can be thought of as a JavaScript runtime framework 18 designed on top of Google’s V8 engine. Thus, it offers a framework in which we can write JavaScript code on any platform that supports node.js. Now a little history. In 2009, Ryan Dahl delivered a lecture at JSConf that permanently altered the course of JavaScript. He added that Node.js to the JavaScript world during his introduction. He ended his nearly 45-min speech with a standing ovation from the crowd. He was motivated to create Node.js after seeing a basics progress bar for file uploads on Flickr, an image sharing website. Recognizing that the site was approaching the procedure incorrectly, he determined that there had to be a better way.

4.4. EXPRESS JS

Express is a framework for Node.js that makes it significantly simpler and cleaner to setup and use APIs and web servers. Being lightweight and containing most of the NodeJS features makes it very popular among the developers. 5.5 Material UI Google created a UI component library focused on the company's material design guidelines. It is composed of numerous open and configurable UI widgets, and the components are self-contained, injecting only the styles required for show. With a large community behind it, it is one of the most common component libraries. The primary selling points are the user-friendly architecture and usability elements derived from Google's familiarity with user interface interfaces. We can install Material UI by using a simple command, npm install inside the terminal.

4.5 VISUAL STUDIO

In April 2015 at Build developer conference, Microsoft launched Visual Code, which runs on OS X, Linux, and Windows. The code editor got updated and enhanced over the years and now is the one of the most popular code editors in the world. [16] The main features of Visual Studio Code include • Integration of Git Although using Git can be difficult at times, Visual Studio Code contains embedded Git, which allows quick GUI-based addition, commit, pull, and push modifications to a remote Git repository.

• Debugging Configuring debugging software in Visual Studio Code is a breeze thanks to the comprehensive API. Debugging is a wide topic that often varies by language/stack. Debugging plugins are required for the language you are using, and you will be able to debug the code using breakpoints when creating.

- Integrated Terminal While a terminal is typically accessible to the side or elsewhere on the computer while running code, Visual Studio Code has an Integrated Terminal that simplifies production.

- Plugins and Themes Visual Studio Code has a robust plugin API, which enables developers to create truly amazing plugins. I will mention a couple of the more popular 20 ones I have seen, but for a more detailed list, we can visit the Visual Studio Code Marketplace.

4.6. SOCKET.IO

In 2010, Socket.IO was created. It was created to promote real-time connection, which was already a relatively new concept at the time. Socket.IO enables bi-directional connection between the client and the server. Bi-directional communication is allowed when a client's browser includes the Socket.IO package and a server also includes the Socket.IO package. Though data may be transmitted in a variety of formats, JSON is the most straightforward. Socket.IO makes use of Engine.IO to create the connection and share data between the client and server. This is a low-level implementation that is used internally. Engine.IO is used for the
Server implementation, while Engine. IO-client is used.

4.7. WEB RTC

WebRTC is an HTML5 standard that enables the addition of real-time media contact between the browser and the user. WebRTC allows the integration of voice and video contact within websites without installing any additional plugins on the website. WebRTC was introduced in 2011 and has slowly increased in prominence and acceptance since then.

4.8. NPM

Npm is a Node.js package manager that has hundreds of thousands of packages. Though it does contribute to the creation of the directory structure/organization, 21 this is not the primary objective. The primary objective is automatic dependency and package management. This ensures that you can specify any of the dependencies for your project within the package. json file, so if you (or someone else) want to get started on your project, they may just run npm install and all the dependencies will be enabled automatically. Additionally, it is possible to define which variants your project relies on to avoid patches destroying your project. Manually downloading the libraries, copying them to the appropriate folders, and using them is possible. However, as the project (and its list of dependencies) expands, this becomes more time-consuming and messier. Additionally, it complicates collaboration and discussing the project. Basically, npm is an invaluable element in workflow as a JavaScript developer (both client-side and serve

CHAPTER 5

DESIGN

5.1. FLOW CHART

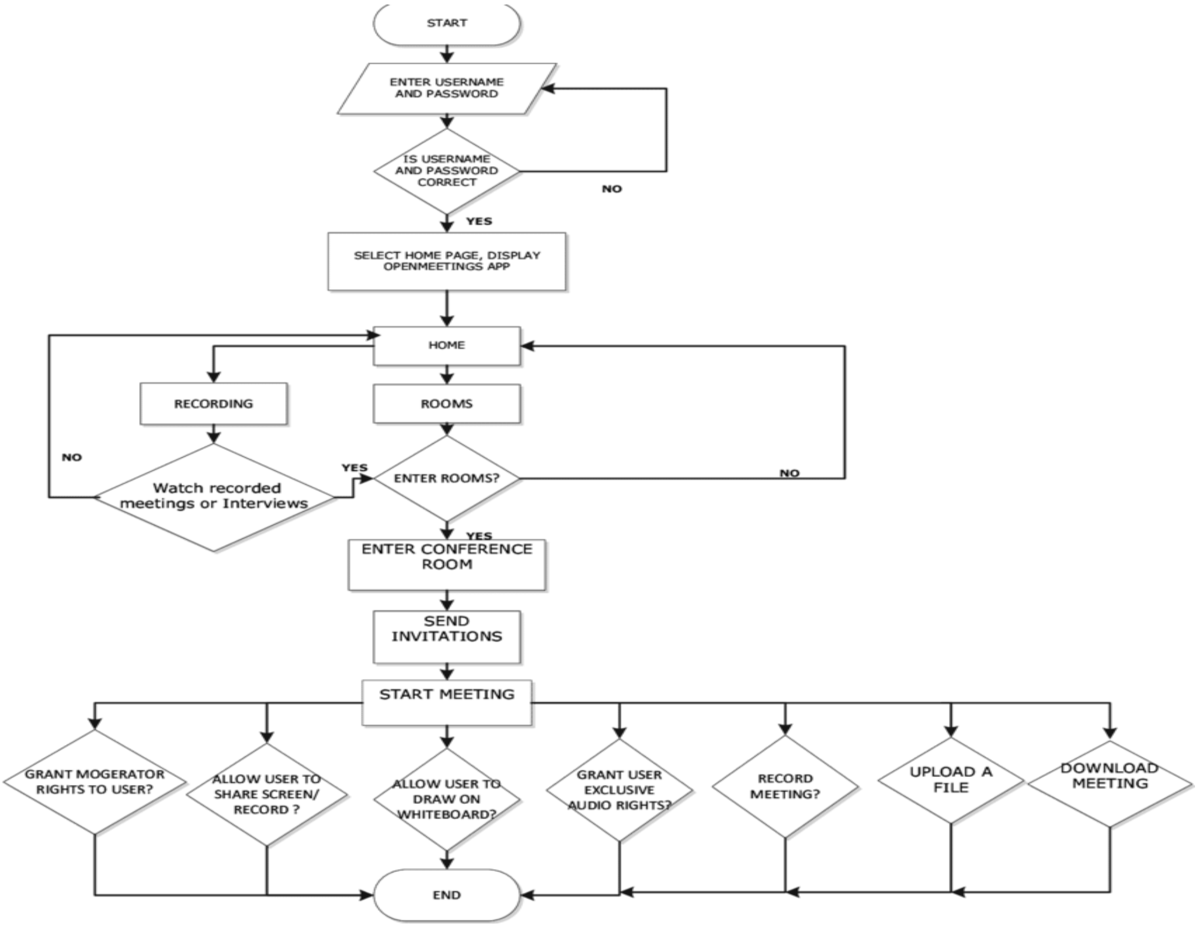

START
ENTER USERNAME AND PASSWORD
IS USERNAME AND PASSWORD CORRECT
NO
YES
SELECT HOME PAGE, DISPLAY OPENMEETINGS APP
HOME
RECORDING
ROOMS
NO
Watch recorded meetings or Interviews
YES
ENTER ROOMS?
NO
YES
ENTER CONFERENCE ROOM
SEND INVITATIONS
START MEETING
GRANT MOGERATOR RIGHTS TO USER?
ALLOW USER TO SHARE SCREEN/ RECORD ?
ALLOW USER TO DRAW ON WHITEBOARD?
GRANT USER EXCLUSIVE AUDIO RIGHTS?
RECORD MEETING?
UPLOAD A FILE
DOWNLOAD MEETING
END

Figure 5.1 flow Diagram

5.2. DATA FLOW DIAGRAM

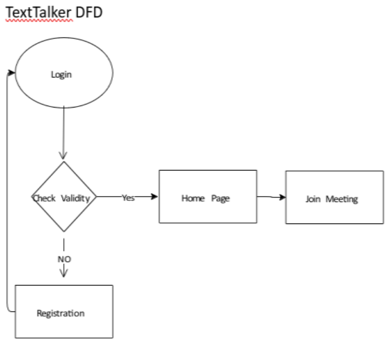


Figure 5.2 Data flow Diagram

5.3. USED CASED DIAGRAM

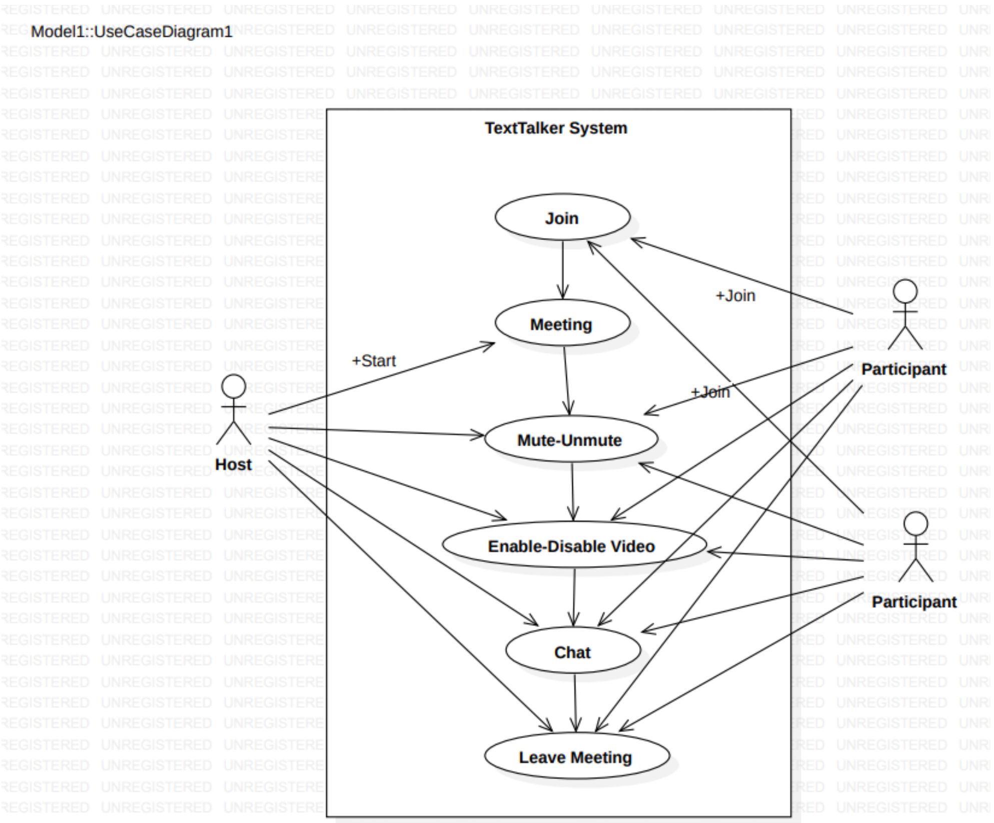


Figure 5.3 Used Case

CHAPTER 6

TESTING CASE AND ITS TYPES

6.1. TESTING TABLE AND TYPE WITH STEPS

Test Case ID	Test Scenario	Test Steps	Expected Result	Status
1.	User Account Creation	Navigate to the sign-up page. Enter valid user details. Click on 'Sign Up'.	User account is created successfully.	
2.	User Login	Navigate to the login page. Enter valid credentials. Click on Login.	User is successfully logged into the application	

3.	Invalid Login Attempt	Navigate to the login page. Enter invalid credentials. Click on 'Login'.	User receives an appropriate error message unable to log in.	
4.	UI Navigation and Responsiveness	Open the application on different devices. Navigate through the UI.	UI is intuitive, and the application is responsive on various devices	
5.	Camera and Microphone Access	Start a video call. Verify access to the camera and microphone	Camera and microphone are accessible, and video call starts successfully.	

6.	Audio and Video Quality	Initiate a video call. Check the quality of audio and video.	Audio and video quality are clear and synchronized.	
7.	Network Conditions	Test under 3G, 4G, and Wi-Fi conditions. Simulate low bandwidth.	Application performs well under different network conditions.	

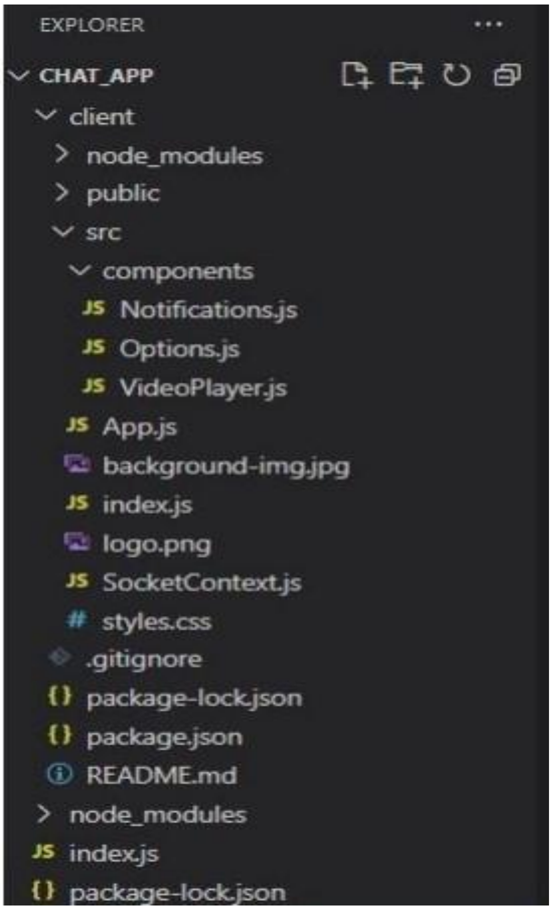
Table 6.1 Testing and its Types

CHAPTER 7

IMPLEMENTATION

7.1. IMPLEMENTATION WITH CODE

In this project we used different dependencies such as material UI, react, simple peer and socket IO client. We used the npm command to install these dependencies. We created three component Notifications, options, video player. In this project we have two important packets. Json file which contains all the important dependencies. The folder structure for this project is shown.

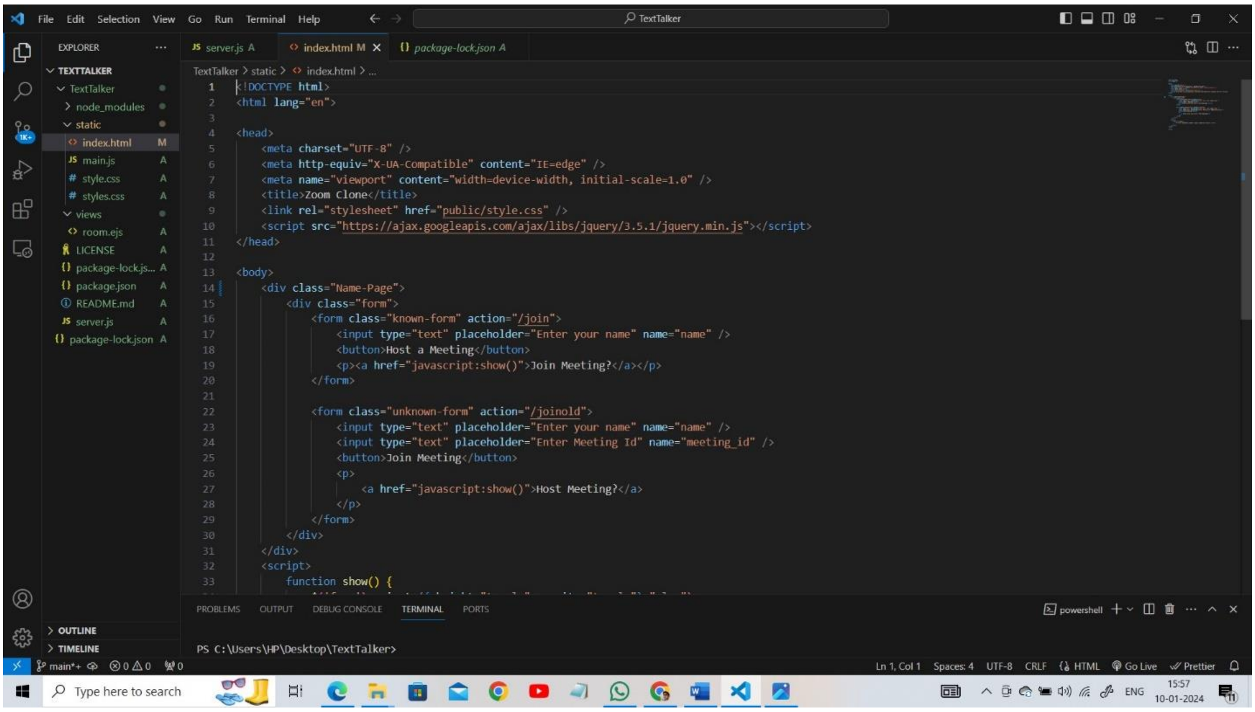


```
EXPLORER ...
v CHAT_APP
  v client
    > node_modules
    > public
    v src
      v components
        JS Notifications.js
        JS Options.js
        JS VideoPlayer.js
      JS App.js
      background-img.jpg
      JS index.js
      logo.png
      JS SocketContext.js
      # styles.css
    .gitignore
    {} package-lock.json
    {} package.json
    README.md
  > node_modules
  JS index.js
  {} package-lock.json
```

Figure 7.1 Components

7.2. FRONTEND

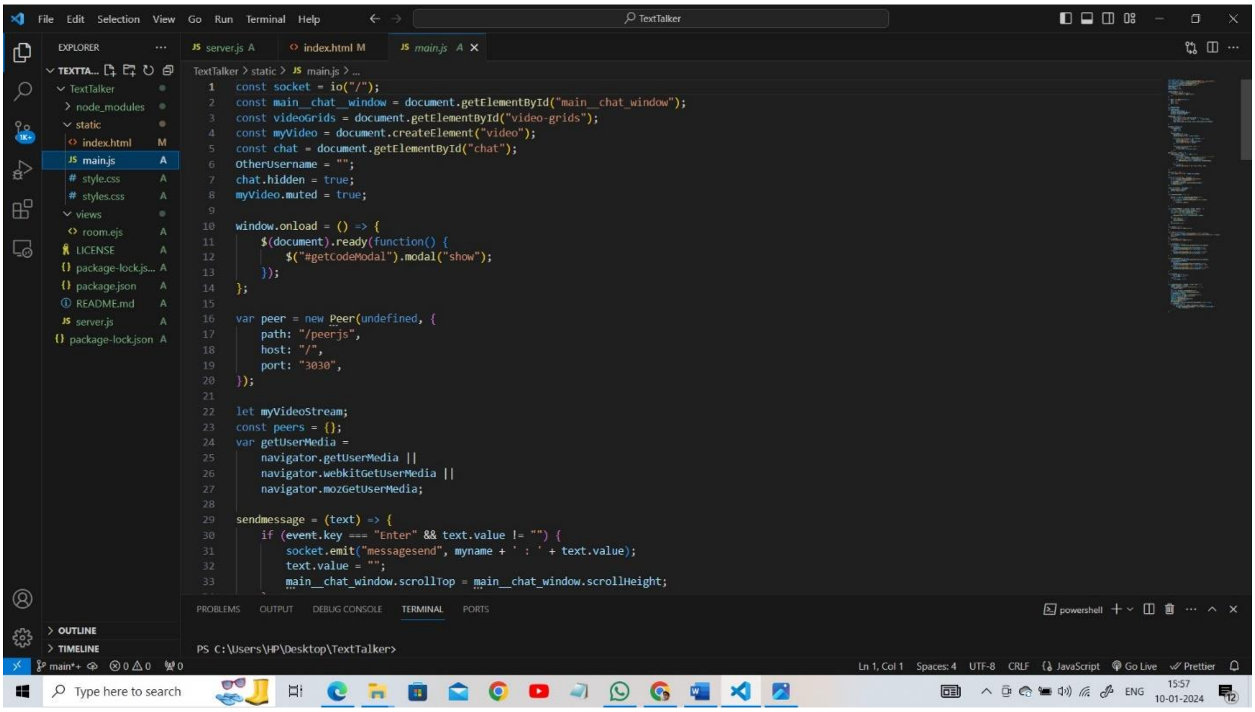
The front of the application was developed using Material UI, HTML, CSS. Material UI is the most preferred React UI library, as there is no other React UI library that can be even suggested. In the figure below we can see that the main content is in the centre of the page. This is done by using grid Container. We created a logo from third parties and inserted it on the top of the page. The width of the video player for the desktop is set to (490) pixel and for the mobile devices it is set to (300) pixel. After that we made two buttons to copy user’s id and another button for calling.



```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Zoom Clone</title>
    <link rel="stylesheet" href="public/style.css" />
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
</head>

<body>
    <div class="Name-Page">
        <div class="form">
            <form class="known-form" action="/join">
                <input type="text" placeholder="Enter your name" name="name" />
                <button>Host a Meeting</button>
                <p><a href="javascript:show()">Join Meeting?</a></p>
            </form>

            <form class="unknown-form" action="/joinold">
                <input type="text" placeholder="Enter your name" name="name" />
                <input type="text" placeholder="Enter Meeting Id" name="meeting_id" />
                <button>Join Meeting</button>
                <p>
                    <a href="javascript:show()">Host Meeting?</a>
                </p>
            </form>
        </div>
    </div>
    <script>
        function show() {
```

```
PS C:\Users\HP\Desktop\TextTalker>
```

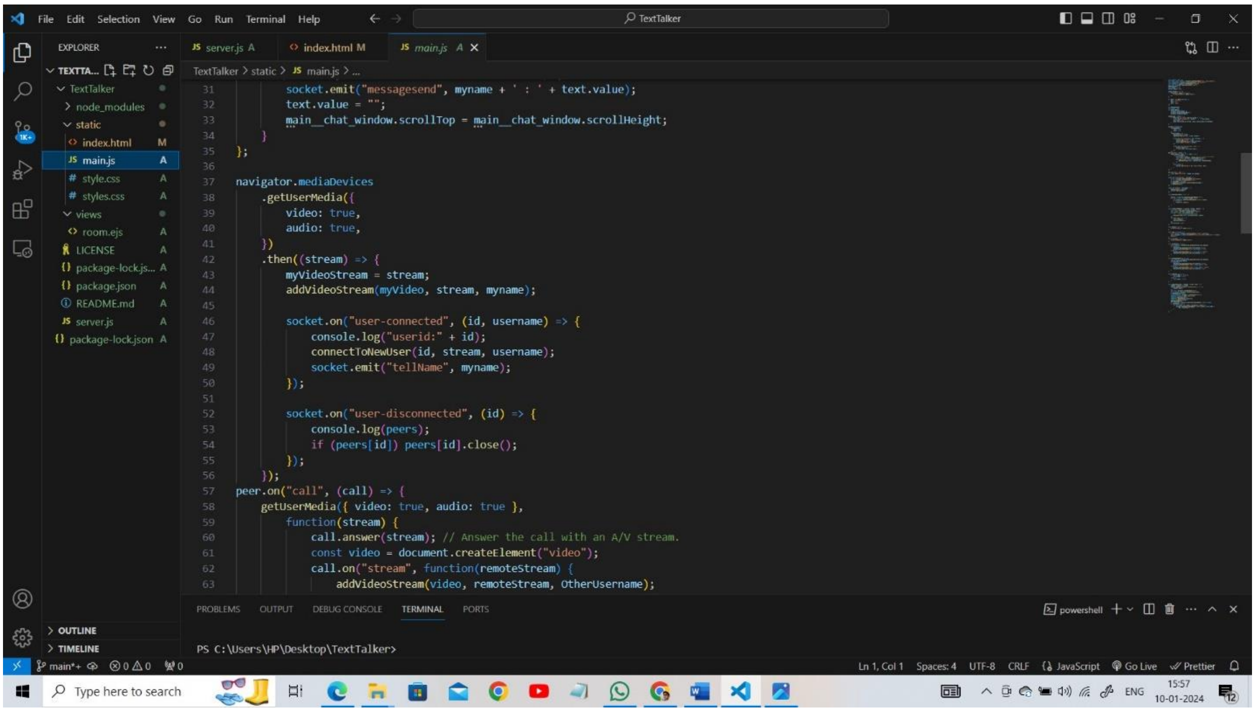
Figure 7.2.1. Html Code



```javascript
const socket = io("/");
const main__chat__window = document.getElementById("main__chat_window");
const videoGrids = document.getElementById("video-grids");
const myVideo = document.createElement("video");
const chat = document.getElementById("chat");
OtherUsername = "";
chat.hidden = true;
myVideo.muted = true;

window.onload = () => {
    $(document).ready(function() {
        $("#getCodeModal").modal("show");
    });
};

var peer = new Peer(undefined, {
    path: "/peerjs",
    host: "/",
    port: "3030",
});

let myVideoStream;
const peers = {};
var getUserMedia =
    navigator.getUserMedia ||
    navigator.webkitGetUserMedia ||
    navigator.mozGetUserMedia;

sendmessage = (text) => {
    if (event.key === "Enter" && text.value != "") {
        socket.emit("messagesend", myname + ' : ' + text.value);
        text.value = "";
        main__chat_window.scrollTop = main__chat_window.scrollHeight;
```

```
PS C:\Users\HP\Desktop\TextTalker>
```

Figure 7.2.2 Html Code

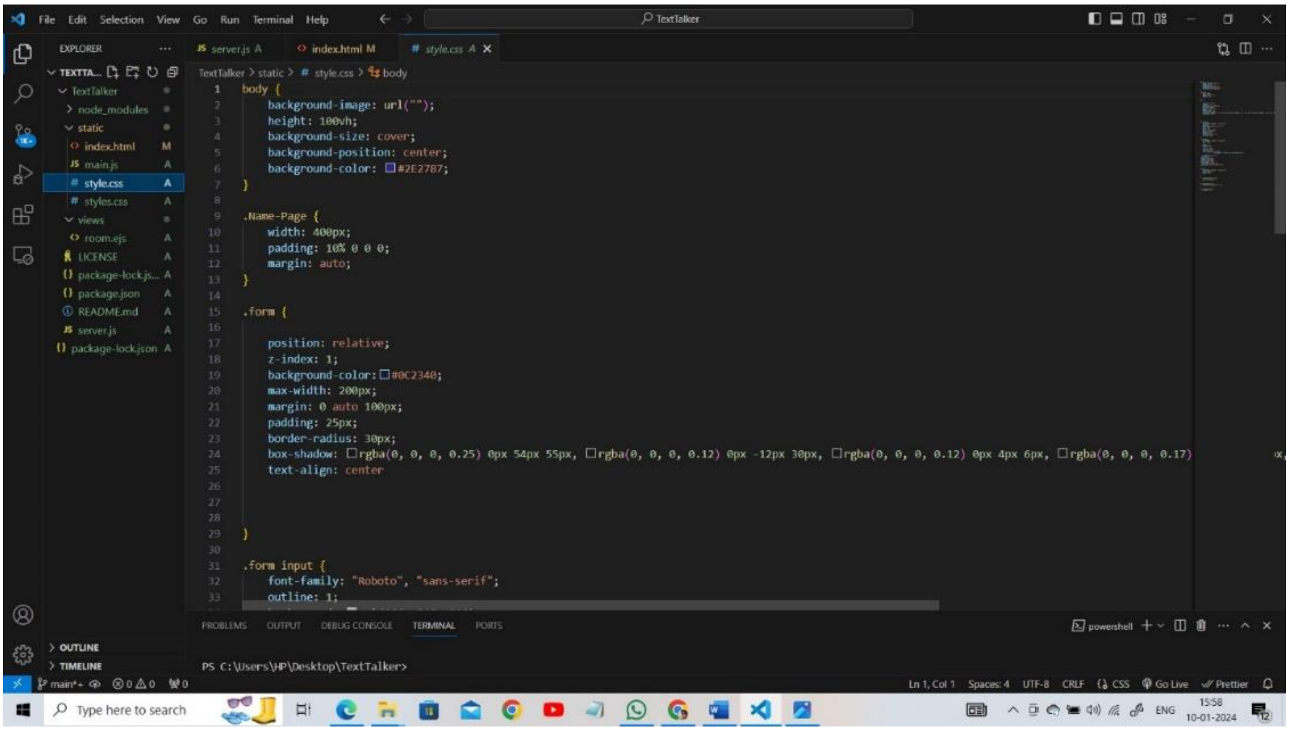


```
        socket.emit("messagesend", myname + ' : ' + text.value);
        text.value = "";
        main__chat_window.scrollTop = main__chat_window.scrollHeight;
    }
};

navigator.mediaDevices
    .getUserMedia({
        video: true,
        audio: true,
    })
    .then((stream) => {
        myVideoStream = stream;
        addVideoStream(myVideo, stream, myname);

        socket.on("user-connected", (id, username) => {
            console.log("userid:" + id);
            connectToNewUser(id, stream, username);
            socket.emit("tellName", myname);
        });

        socket.on("user-disconnected", (id) => {
            console.log(peers);
            if (peers[id]) peers[id].close();
        });
    });
peer.on("call", (call) => {
    getUserMedia({ video: true, audio: true },
        function(stream) {
            call.answer(stream); // Answer the call with an A/V stream.
            const video = document.createElement("video");
            call.on("stream", function(remoteStream) {
                addVideoStream(video, remoteStream, OtherUsername);
```

```
PS C:\Users\HP\Desktop\TextTalker>
```

Figure 7.2.3 JavaScript code

Style.CSS

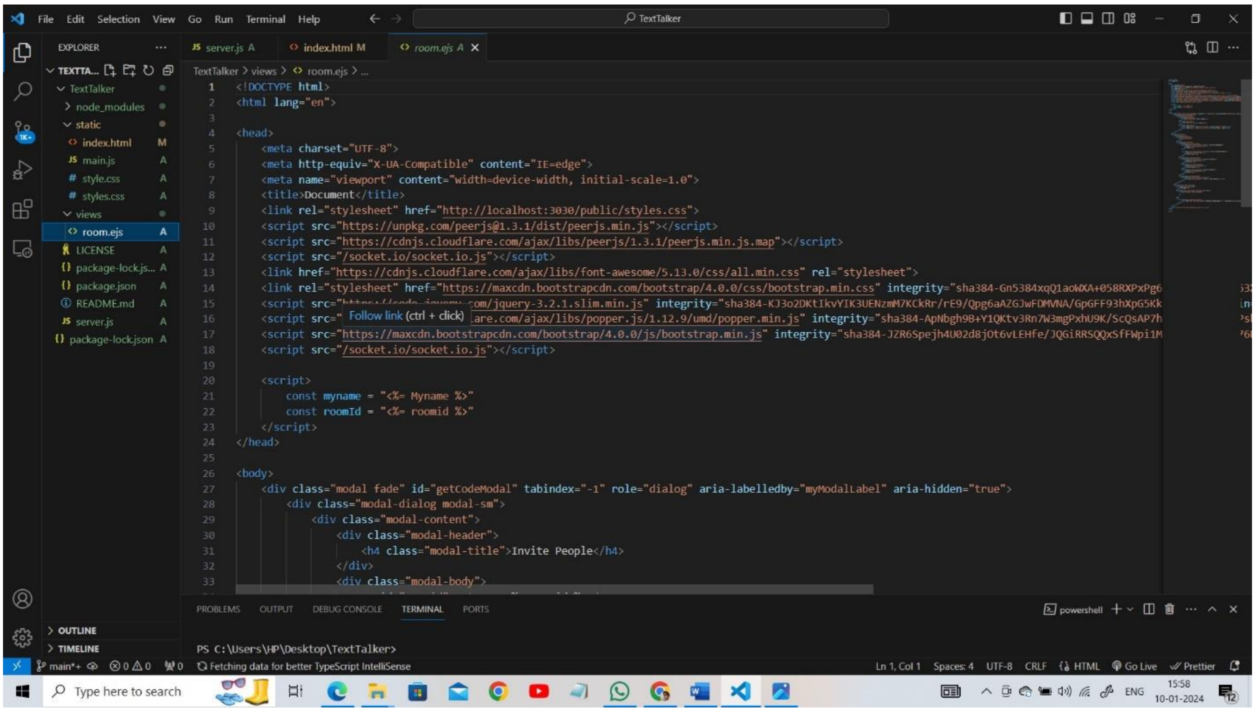


```
body {
    background-image: url("");
    height: 100vh;
    background-size: cover;
    background-position: center;
    background-color: ■#2E2787;
}

.Name-Page {
    width: 400px;
    padding: 10% 0 0 0;
    margin: auto;
}

.form {

    position: relative;
    z-index: 1;
    background-color:□#0C2340;
    max-width: 200px;
    margin: 0 auto 100px;
    padding: 25px;
    border-radius: 30px;
    box-shadow: □rgba(0, 0, 0, 0.25) 0px 54px 55px, □rgba(0, 0, 0, 0.12) 0px -12px 30px, □rgba(0, 0, 0, 0.12) 0px 4px 6px, □rgba(0, 0, 0, 0.17)
    text-align: center



}

.form input {
    font-family: "Roboto", "sans-serif";
    outline: 1;
```

```
PS C:\Users\HP\Desktop\TextTalker>
```

Figure 7.2.4 Style CSS

Room.ejs-



```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet" href="http://localhost:3030/public/styles.css">
    <script src="https://unpkg.com/peerjs@1.3.1/dist/peerjs.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/peerjs/1.3.1/peerjs.min.js.map"></script>
    <script src="/socket.io/socket.io.js"></script>
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.13.0/css/all.min.css" rel="stylesheet">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6
    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5Kk
    <script src=" Follow link (ctrl + click) .are.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7h
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1M
    <script src="/socket.io/socket.io.js"></script>

    <script>
        const myname = "<%= Myname %>"
        const roomId = "<%= roomid %>"
    </script>
</head>

<body>
    <div class="modal fade" id="getCodeModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">
        <div class="modal-dialog modal-sm">
            <div class="modal-content">
                <div class="modal-header">
                    <h4 class="modal-title">Invite People</h4>
                </div>
                <div class="modal-body">
```

```
PS C:\Users\HP\Desktop\TextTalker>
```

Figure 7.2.5 Room. ejs

The initial phase of the working website without styling was developed which is shown in figure 3 below-

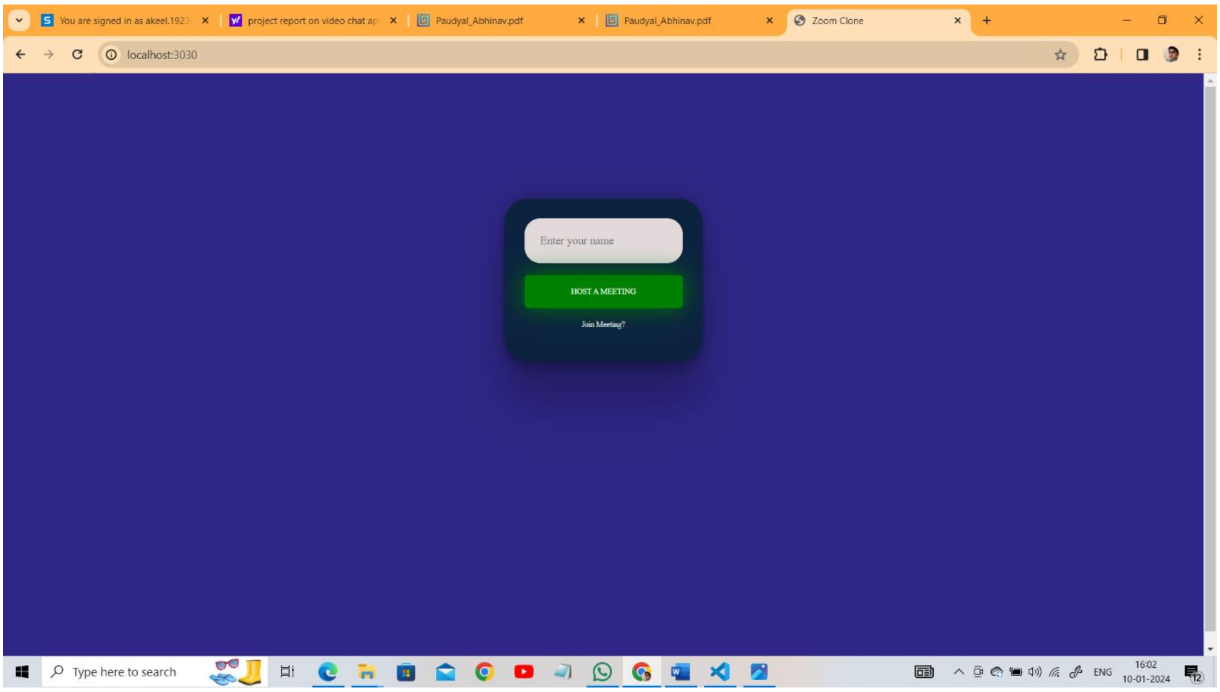


Figure.7.2.6 Initial Interface

User Hosting the Meeting –

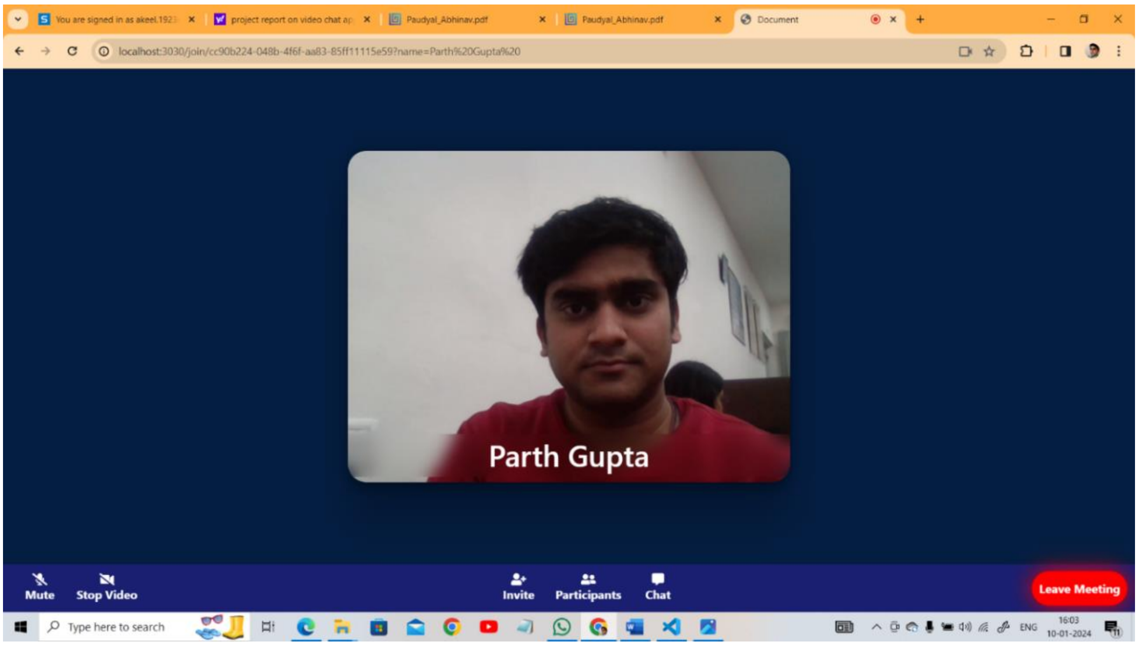
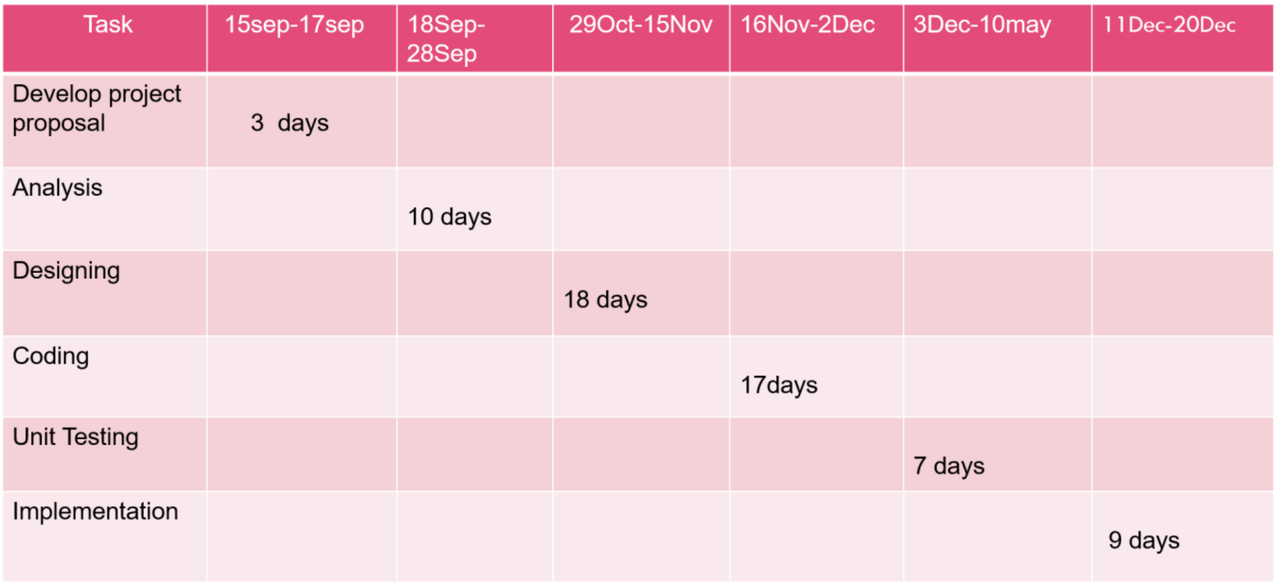


Figure.7.2.7 User Login

7.3 Project Duration:



Task	15sep-17sep	18Sep-28Sep	29Oct-15Nov	16Nov-2Dec	3Dec-10may	11Dec-20Dec
Develop project proposal	3 days					
Analysis		10 days				
Designing			18 days			
Coding				17days		
Unit Testing					7 days	
Implementation						9 days

Figure 7.3 Time management

CHAPTER 8

RESUTLS AND CONCLUSION

8.1. RESULTS

Finally, a working video chat app was created where you can call your friends and family. The following are the steps to use the web app.

Step 1: As shown in the below the web application will ask for users' permission to use the microphone and camera.

Step 2: As user Should fill their Name or he and she can join the meeting or Host the Meeting by Both Options

Step 3: After joining the meeting User get Unique Id for join other participates in the call

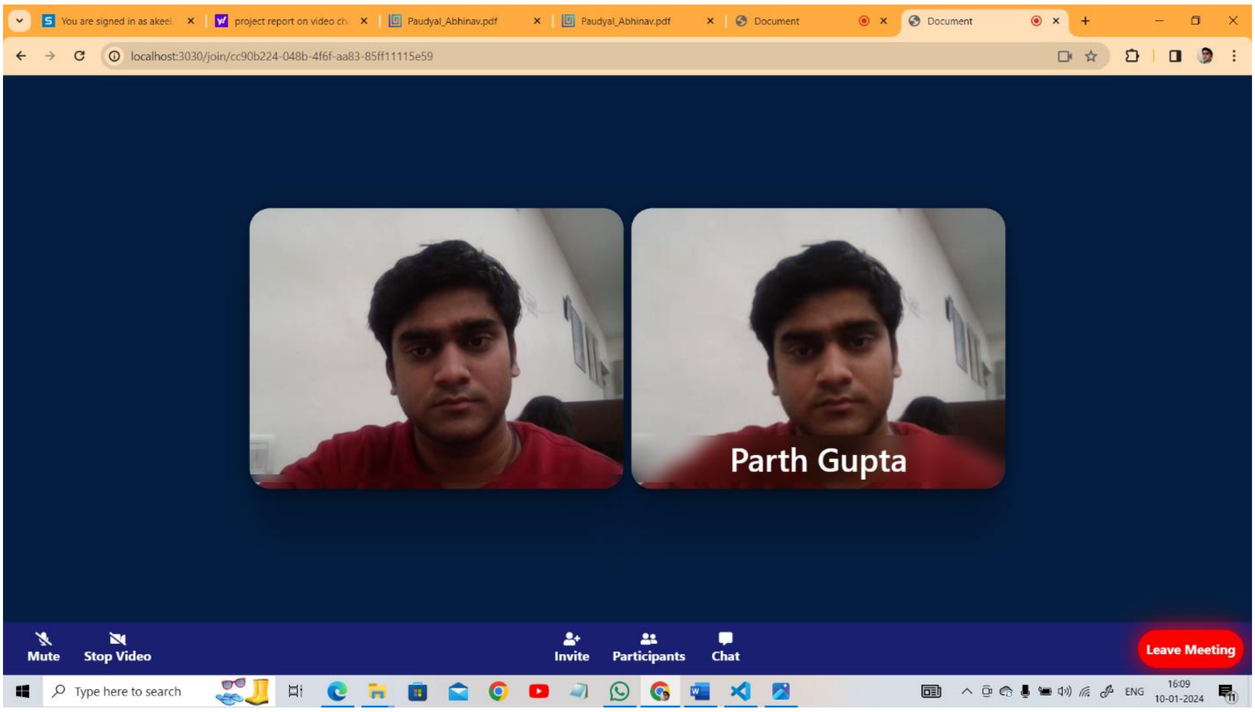


Figure 8.1 final Result

8.2. CONCLUSION

Today, React Js is one of the most common and efficient front-end technologies available. React Js is the only library that can be implemented anywhere. It performs admirably in all applications. Additionally, React Js is highly compatible with a variety of formats, browsers, and applications. Following the creation of the Chat App, it was obvious that React Js is an easy to learn and simple to implement framework. In conclusion this project was a simple project in which you can video call your friends and family, but future development is very much possible by adding much more features in the application

CHAPTER 9

REFRENCES

[1] “https://www.tutorialspoint.com/,” Tutorials Point, [Online]. Available: https://www.tutorialspoint.com/javascript/index.htm. [Accessed 07 05 2021].

[2] T. Kidder, “https://www.learnacademy.org/,” LEARN ACADEMY, 10 02 2020. [Online]. Available: https://www.learnacademy.org/blog/whointroduced-javascript-originally-called-why-created/. [Accessed 02 05 2020].

[3] “https://insights.stackoverflow.com/,” StackOverflow, 2019. [Online]. Available: https://insights.stackoverflow.com/survey/2019. [Accessed 02 05 2021].

[4] “https://www.geeksforgeeks.org/,” GeeksforGeeks, 25 11 2020. [Online]. Available: https://www.geeksforgeeks.org/advantages-anddisadvantages-of-javascript/. [Accessed 04 05 2021]. [5] “http://net-informations.com/,” net-informations, [Online]. Available: http://net-informations.com/js/iq/advan.htm. [Accessed 01 05 2021].

[6] F. Hámori, “https://blog.risingstack.com,” RingStack, 04 04 2018. [Online]. Available: https://blog.risingstack.com/the-history-of-react-js-on-atimeline/. [Accessed 06 05 2021].

[7] T. Sufiyan, “https://www.simplilearn.com,” Simplilearn, 09 04 2021. [Online]. Available: https://www.simplilearn.com/tutorials/reactjstutorial/what-is-reactjs. [Accessed 06 05 2021].

[8] C. Eygi, “https://www.freecodecamp.org,” freeCodeCamp, 09 02 2020. [Online]. Available: https://www.freecodecamp.org/news/react-js-forbeginners-props-state-explained/. [Accessed 21 04 2021].

[9] E. Lvova, “https://dzone.com,” DZone, 2021 03 19. [Online]. Available: https://dzone.com/articles/react-vs-vue-in-2021-best-javascriptframework. [Accessed 22 04 2021]. [10] “https://www.guru99.com,” Guru99, [Online]. Available: https://www.guru99.com/react-vs-angular-key-difference.html#9. [Accessed 28 04 2021].

[11] “https://www.geeksforgeeks.org/,” GeeksforGeeks, 04 12 2019. [Online]. Available: https://www.geeksforgeeks.org/what-is-web-socket-and-how-itis-different-from-the-http/. [Accessed 02 05 2021].

[12] D. G, “www.hostinger.com,” Hostinger, 25 11 2019. [Online]. Available: https://www.hostinger.com/tutorials/what-is-html. [Accessed 08 05 2021].

[13] “https://blog.devmountain.com,” Devmountain, [Online]. Available: https://blog.devmountain.com/what-is-css-and-why-use-it/. [Accessed 25 04 2021].

[14] “https://developer.mozilla.org/,” MDN Web Docs, [Online]. Available: https://developer.mozilla.org/enUS/docs/Learn/CSS/First_steps/What_is_CSS. [Accessed 07 05 2021].

[15] B. J. D. J. K. Mithun Satheesh, “Web Development with MongoDB and NodeJS,” Packt, Birmingham, 2015.

[16] F. Lardinois, “https://techcrunch.com/,” TechCrunch, 29 04 2015. [Online]. Available: https://techcrunch.com/2015/04/29/microsoft-shocks-the-worldwith-visual-studio-code-a-free-code-editor-for-os-x-linux-andwindows/?guce_referrer=aHR0cHM6Ly93d3cud2lraXdhbmQuY29tLw&guce_referrer_sig=AQAAAHAoQuS6Qm61BXPY87sdFQPZky83wVXY1iiicEp3EwfHpRTNWzJ. [Accessed 14 04 2021].

[17] C. Ganga, “https://scotch.io/,” Scotch.io, 25 07 2018. [Online]. Available: https://scotch.io/bar-talk/my-top-8-visual-studio-code-tips-and-features. [Accessed 17 04 2021].

[18] G. Lewington, “https://ably.com/,” Alby, [Online]. Available: https://ably.com/topic/socketio#:~:text=Socket.IO%20allows%20bi%2Ddirectional,integrated%20the%20Socket.IO%20package.&text=To%20establish%20the%20connection%2C%20and,.IO%20uses%20Engine.IO.. [Accessed 25 04 2021].

[19] “https://bloggeek.me/,” Blogeek.me, 15 12 2019. [Online]. Available: https://bloggeek.me/what-is-webrtc. [Accessed 02 05 2021].

[20] “https://www.education-ecosystem.com,” Education Ecosystem, 2019. [Online]. Available: https://www.educationecosystem.com/guides/programming/react-js/history. [Accessed 27 04 2021]. [21] L. Kalshteyn, “https://dev.to,” DEV Community , 21 10 2020. [Online]. Available: https://dev.to/gorgutzz/material-ui-intro-50md. [Accessed 03 05 2021]