**GLA University, Mathura -2020**

***Mini Project***

***Mid-Term Report***



**TEAM DETAILS**

**Name Git-Hub Usernames**

1. AMAN GUPTA Aman9306
2. RISHINDRA MISHRA rishindramishra11
3. SWATI SAXENA swatisaxena23

***PROJECT NAME: “I-CHAT”***

***Supervised By:-***

***Mandeep Singh***

***Content***

* Abstract
* Introduction
* Problem Statement
* Objectives
* Implementation Details
* Progress till Date & The Remaining Work
* Some Screenshots
* References

***Abstract***

We are creating “A Web-browser based application” named “**I-chat**”.

As we know, every Website has a front-end as well as a back-end. So, **Front-end** is a practice of converting data to a graphical interface, through the use of HTML, CSS, and JavaScript, so the users can view and interact with that data and,

**Back-end** development refers to server-side development. It focuses on databases, scripting, and architecture of websites.

Code written by back-end developers helps to communicate the database information to the browser.

So in this mini-project we are using:-

* CSS
* Javascript
* Html

As a front- end for our real time-chat web-page.

And for back-end:-

* Node.Js

Another technology that will be used in this project is named as Socket.io(will be explained below).

**Introduction**

In today’s era, we all are aware of the need to communicate with each other. However we the people are so busy in our own lives, we forget and sometimes ignore that *the world beyond us exists.*

One solution to this is **Virtual Communication**.

Communicating virtually with people is when individuals interact without being face-to-face but through words and texts that are sent at a higher speed than letters. We can share information, comment, and ask questions, develop social etiquette.

Communicating virtually make us elevate from being digitally literate to digitally fluent, collaborate using appropriate virtual tools and system and they are cost-effective too.

So putting our efforts into this application, we are going to create I-chat.

In that, we will be using Socket.io and many more technologies.

**Q. What Socket.io is?**

* Socket.io is the two-way connection between the server and the client.
* As we know Http, Https protocol is a one-way connection between server and client. So in this, we are going to use ws(web socket), wss(web socket secure) protocol to establish the connection between the server and client too.

As we know a web-page contains front-end and back-end.

Let's describe the technologies used to build our front-end.

**Q. What CSS is?**

* CSS is used for styling the Webpages.
* CSS is of three types i.e. internal CSS, external CSS, inline CSS.
* It describes how Html elements are to be displayed on the screen.
* It involves many designing for the bare Mark-up to create it attractive.

**Q.** **What is HTML?**

* The first version of Html was written by Tim Berners-Lee in 1993. Since then, there have been many different versions of Html. We are using Html 5 in our project.
* Html is a programming language used to describe the structure of information on a webpage.
* Front end developer uses Html as to make the structure of their ideas or we can say a raw website. In that, design our idea by using CSS, JS.

**Q. What is JavaScript?**

* JavaScript was created by Brendan Eich in 1995.
* It is the programming, Scripting language that allows implementing complex features on web pages.
* JS helps the web developer to make a dynamic and interactive webpage by implementing a custom client-side script.

Now, the technology used in the back-end is:-

**Q. What Node.Js is?**

* NodeJs was developed by many developers but the original author was Ryan Dahl in 2009.
* NodeJs is an open-source, cross-platform that executes JavaScript code outside a web browser.
* It also works as a backend. And we are going to use it as a backend.

**Area of Computer Science**

This real-time chat web-browser is been developed for web-technologies on various devices like iPad, mobile phones, desktop, laptops, etc by many users. Multiple users can interact at the same time on this web-page and can chit-chat there.

Web-Technologies like:-

* Firefox
* Windows
* Mac-Os
* Crome
* Opera

***Hardware Requirements***

**Hardware requirements**

In hardware requirement, we require all those components which will provide us the platform for the development of the project. The minimum hardware required for the development of this project is as follows-

**RAM - a minimum of 128 MB**

**Hard disk- minimum 4GB**

**Processor - core i3**

These all are the minimum hardware required for our project. We want to make our project to be used in any type of computer therefore, we have taken a minimum configuration to a large extent. 128 MB ram is used so that we can execute our project in the least possible ram. 500 GB hard disk is used because the project takes less space to be executed or stored. Therefore minimum hard disk is used.

**Software Requirement**

The software can be defined as a program that runs on our computer. It acts as petrol in the vehicle. It provides the relationship between the human and computer. Various software is needed in this project for its development.

Which are as follows-

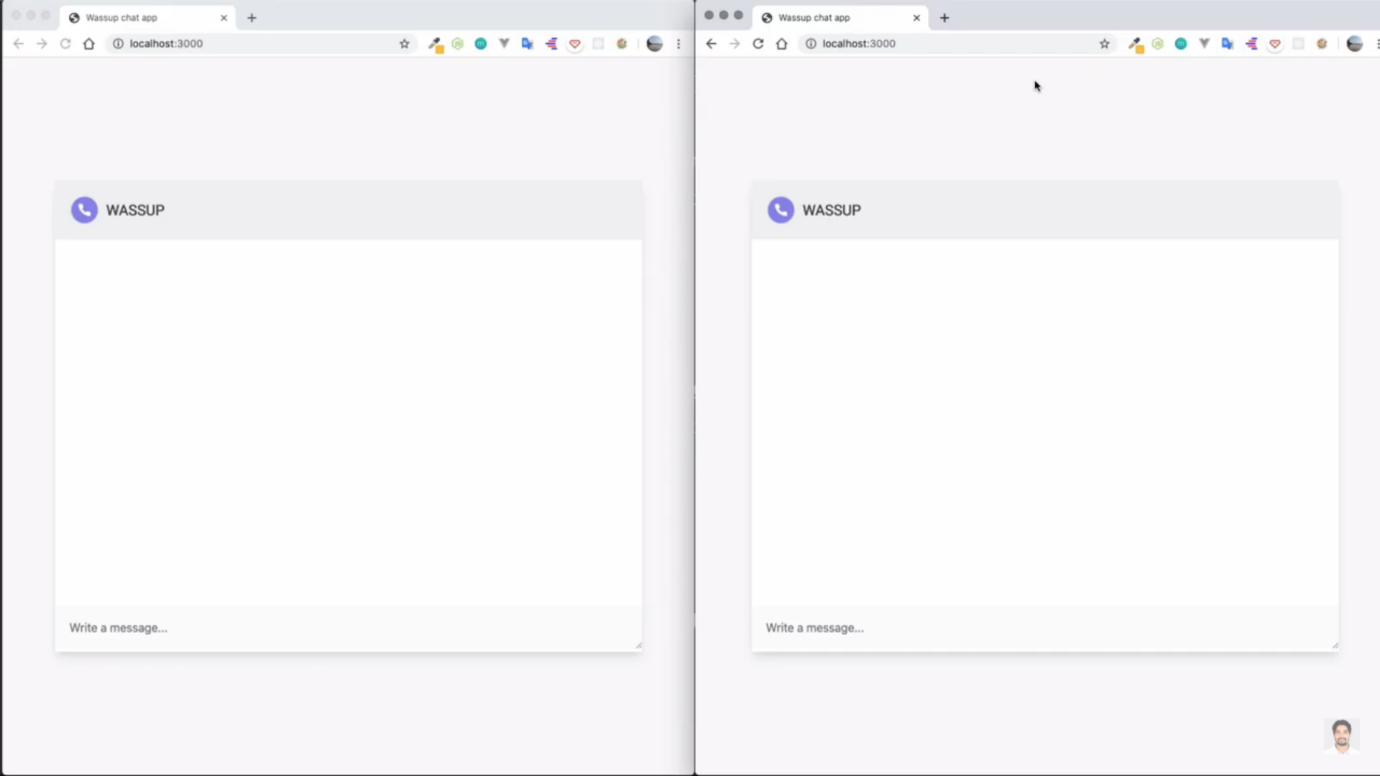
**Operating system- windows7**

**Others- visual studio**

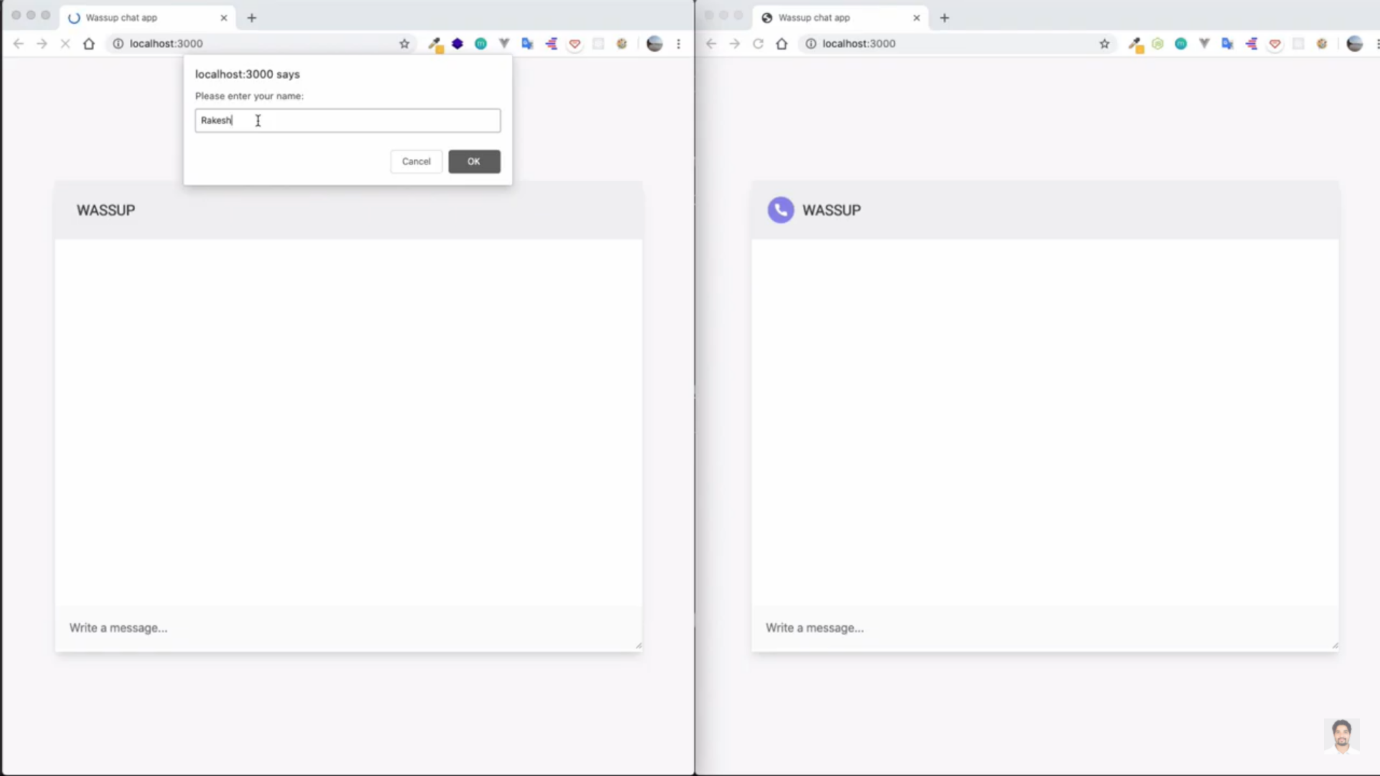
**Objective**

According to **our idea of creating a Real-Time Chatting Application named “I-chat” to establish productively and trusting virtual working relationships.**

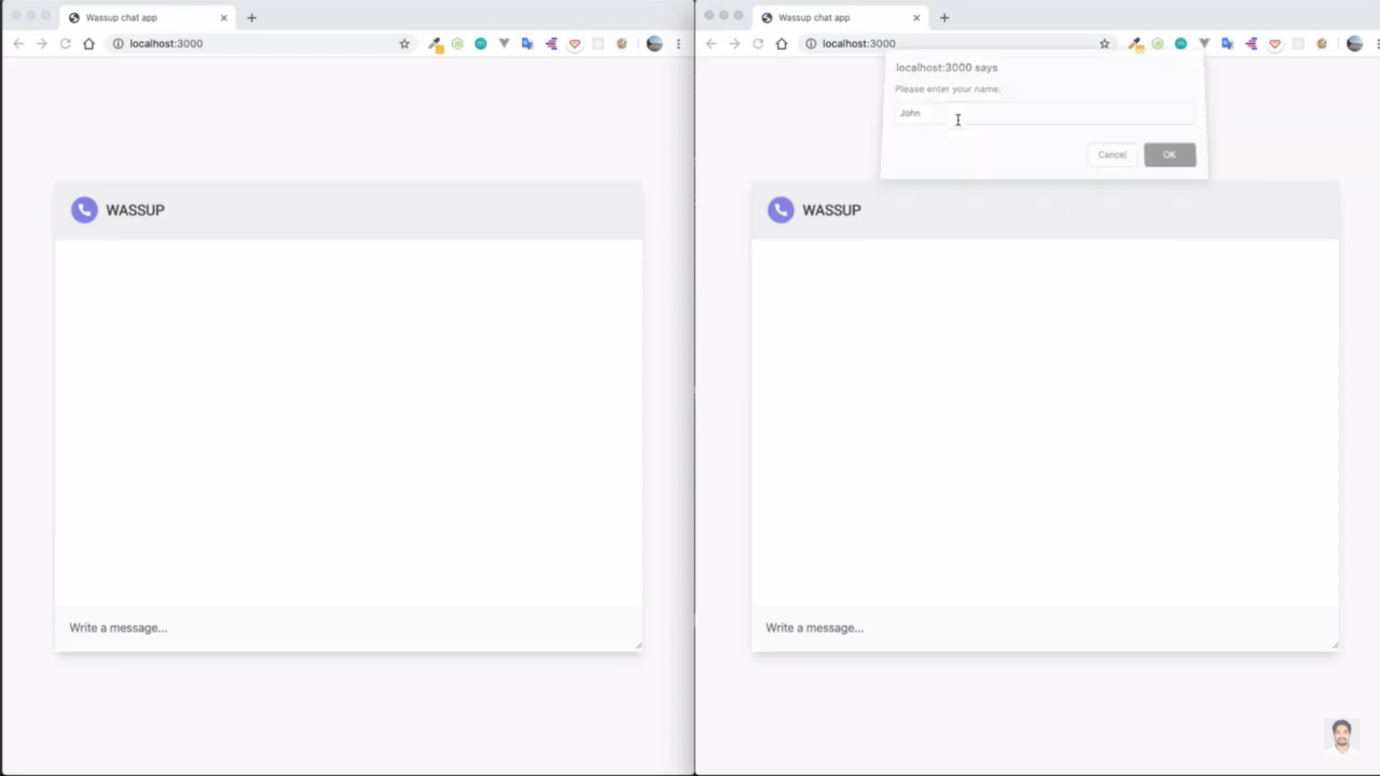
Firstly, we got many points regarding this project. We are using the basic mark-up language (for creating a webpage) that is HTML (HyperText Markup Language) and the basic styling requiring with CSS (Cascading Styling Sheets) to give it a gesture of a real-time application as shown below:



Then as soon as the user enters, there will be a prompt box asking the user to enter his/her credential (here the Name) and the same will happen to the second user who enters the chat area.



Likewise,



After the person enters the chat there will be a notification that the person has entered the chat. With the help of JavaScript, the webpage is interactive. Socket.io is used to link the client and server-side.

So this application is fully responsive. In this, they have created a room so as multiple users can chat at a time in a particular room. For example, two guys are chatting at a time named “Rakesh” and “John”. So firstly Rakesh texts John “Hello John”. John will get that message with a sweet notification sound as “Rakesh: Hello“, then John replies with “Hello Rakesh ” which again Rakesh receives. For instance, if a new user named “Riti” joins the room there would be an update on the room as “Riti joins the chat”. And if the user left the chat then the notification be like “< user-name> left the chat”.

Concerning this context, we will create our project with full enthusiasm and many more features indulged in it.

Some of the features are:-

1. Scrolling bar at the right:-

Whenever there are a lot of messages on the screen, to see them all, a scrolling bar on the right-hand side will appear on its own.

1. Lots of designing by using CSS:-

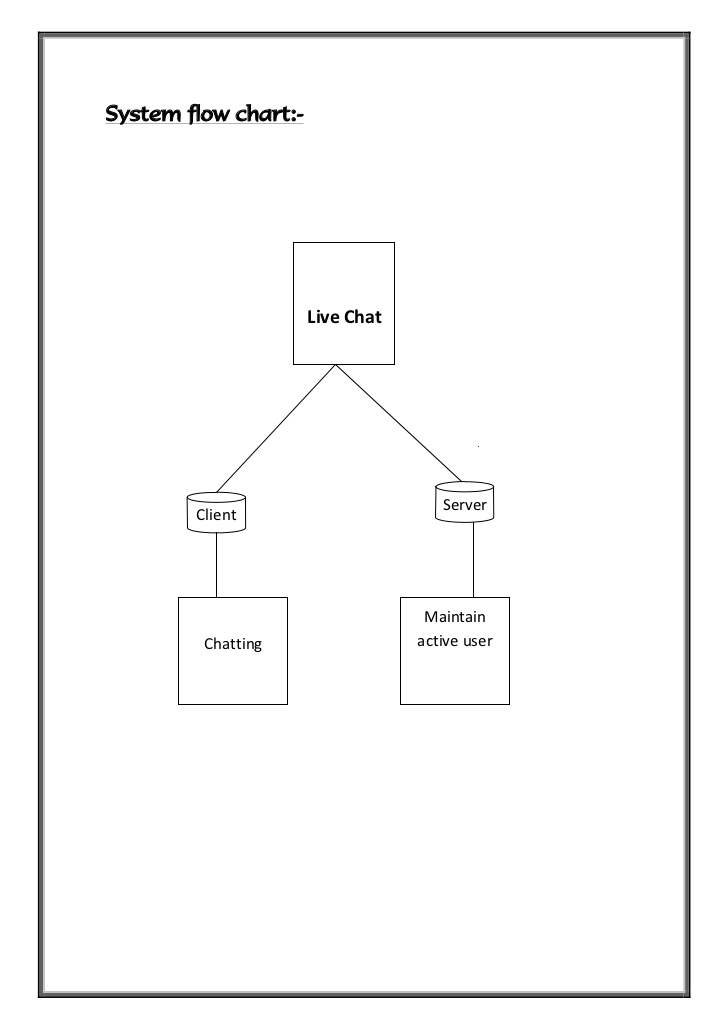
We will be using a nice lot of RGB colors, and many more CSS to enhance our webpage.

1. Message notification sound:-

With every message that arrives, there is a notification alert with a minute ringtone noise.

This is all we will be doing in this project, and apart from using the basic CSS, we will put all our efforts to add more features to this application.

**Implementation Details**



A live chat contains two parts:-

* Client
* Server

So we have divided our mini project into two parts as shown.

As per the implementation progress, we have completed our part 1 with full enthusiasm. And part 2 is in progress.

**Part 1:- Client-side includes front-end.**

* Basic Mark-up (HTML)
* Designing HTML(CSS)
* Building the logic of webpage(JavaScript)

**Part 2:- Server-side includes backend.**

* Connecting frontend with backend.
* Connection of socket.io from front-end.
* Connection of node.js from front-end.

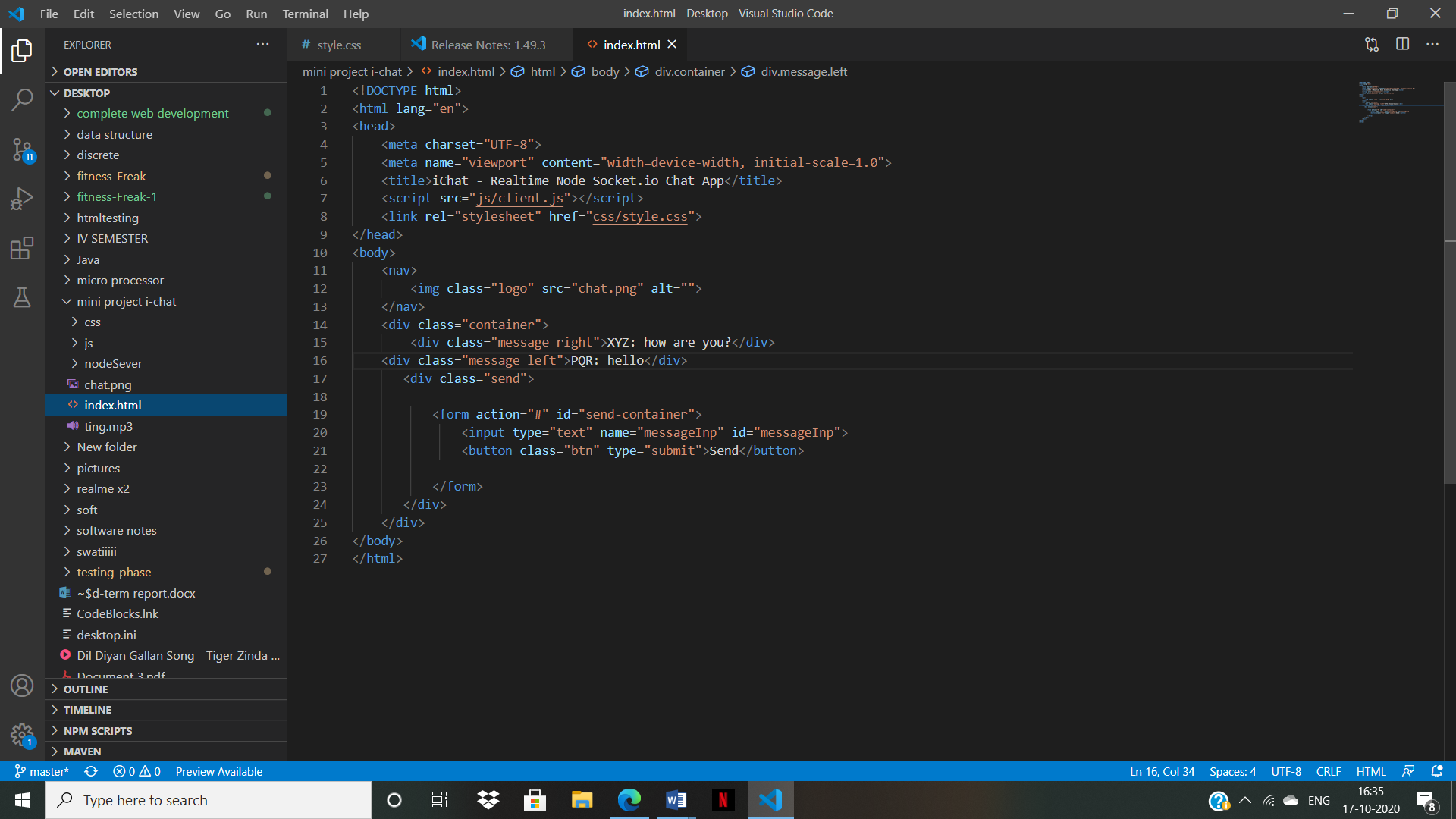
**Part 3:- Final Submission**

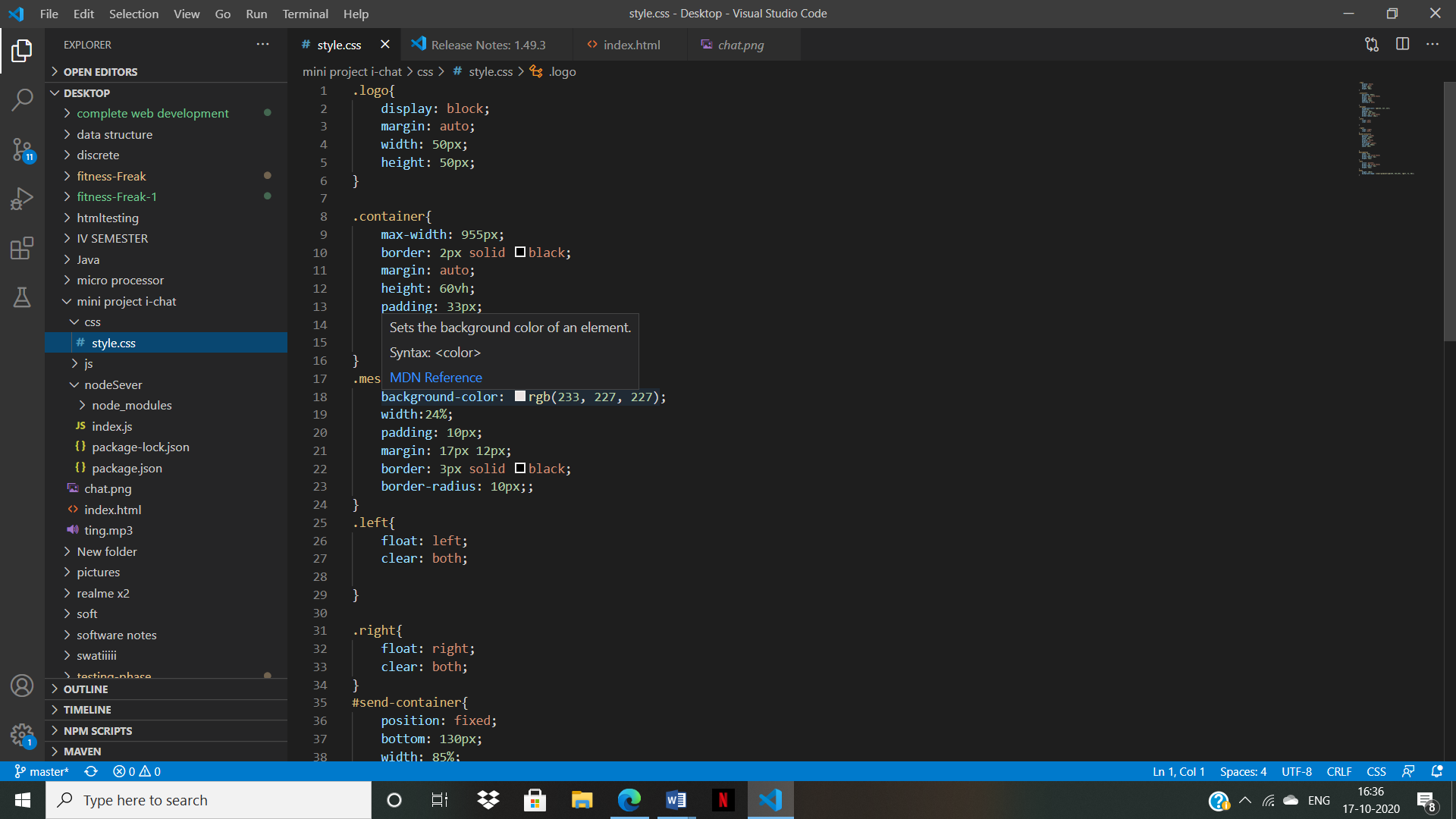
* It includes part1+ part2.

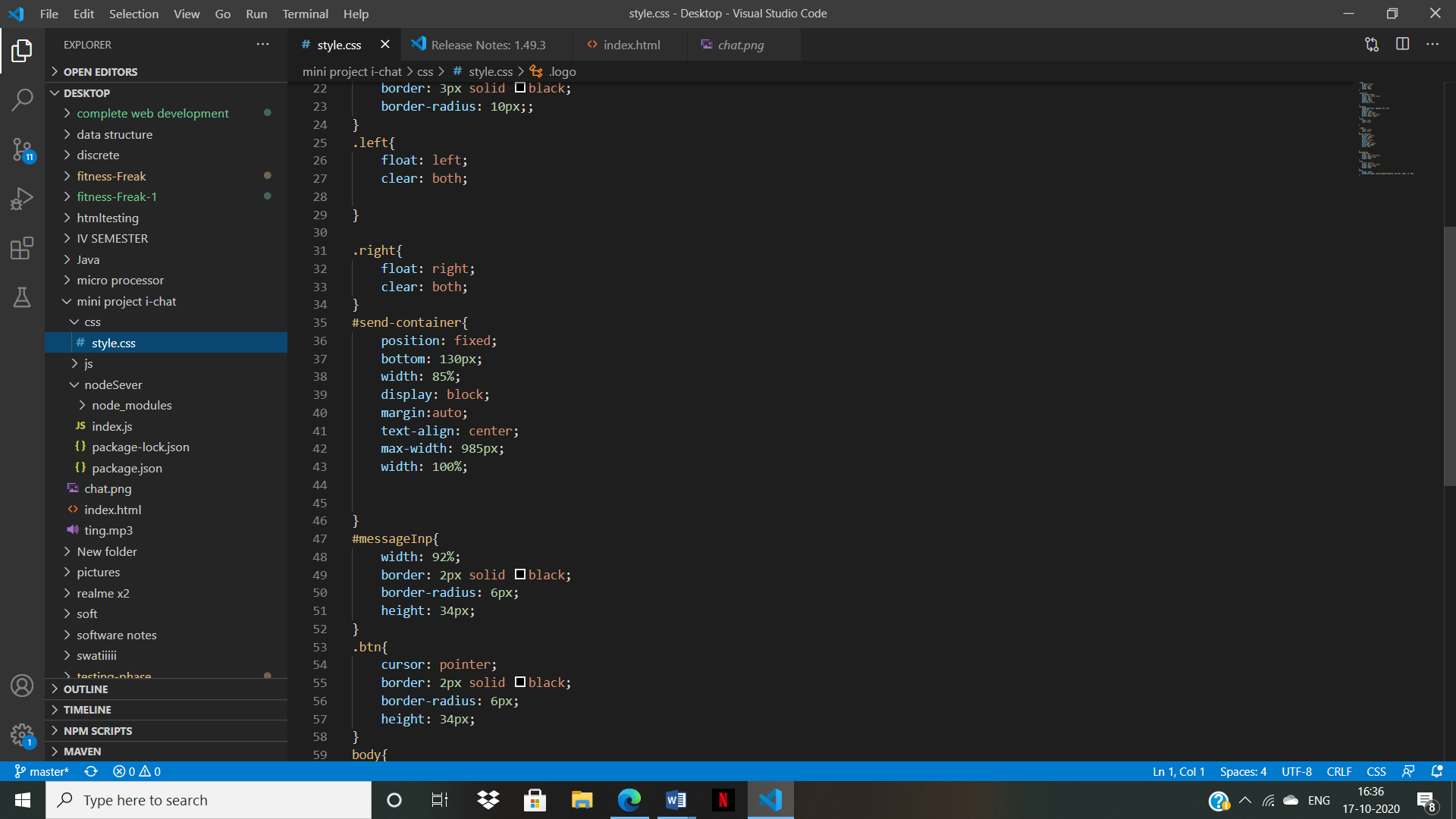
**Progress Report**

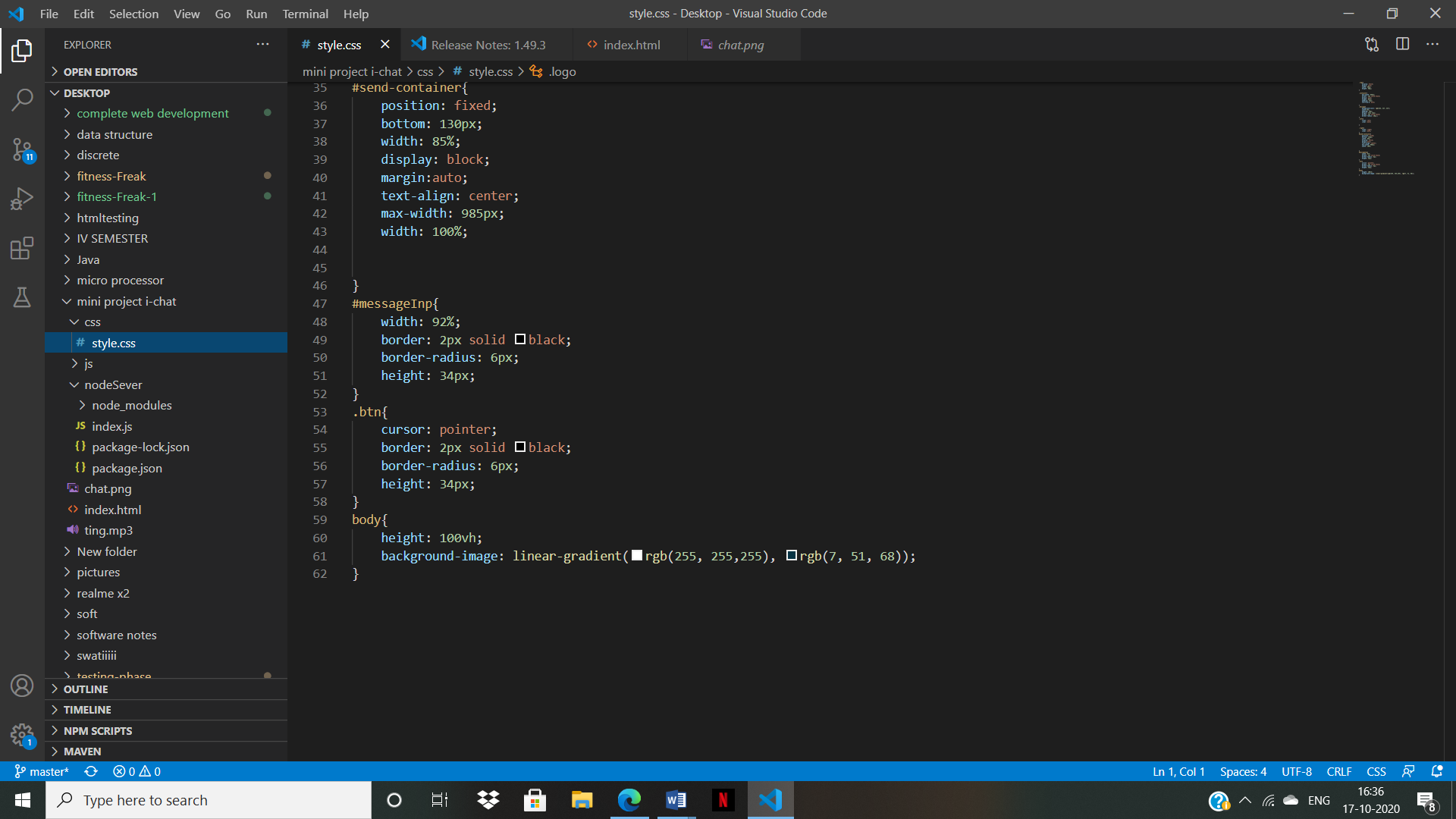
|  |  |
| --- | --- |
| **Work** | **Progress** |
| Part 1 | completed |
| Part 2 | Remaining(partially completed) |
| Part 3 | Remaining(on due date) |

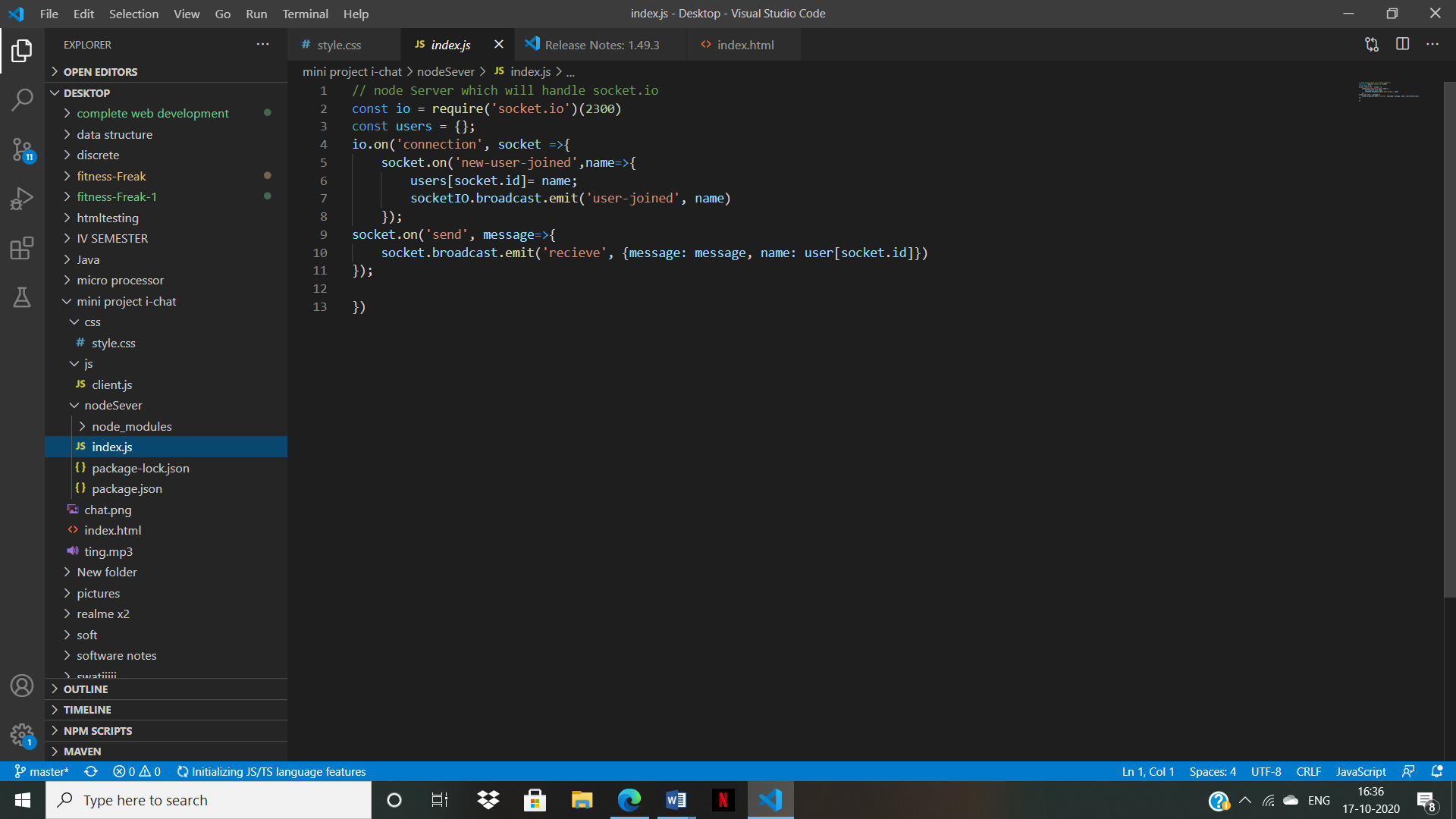
**ScreenShots**

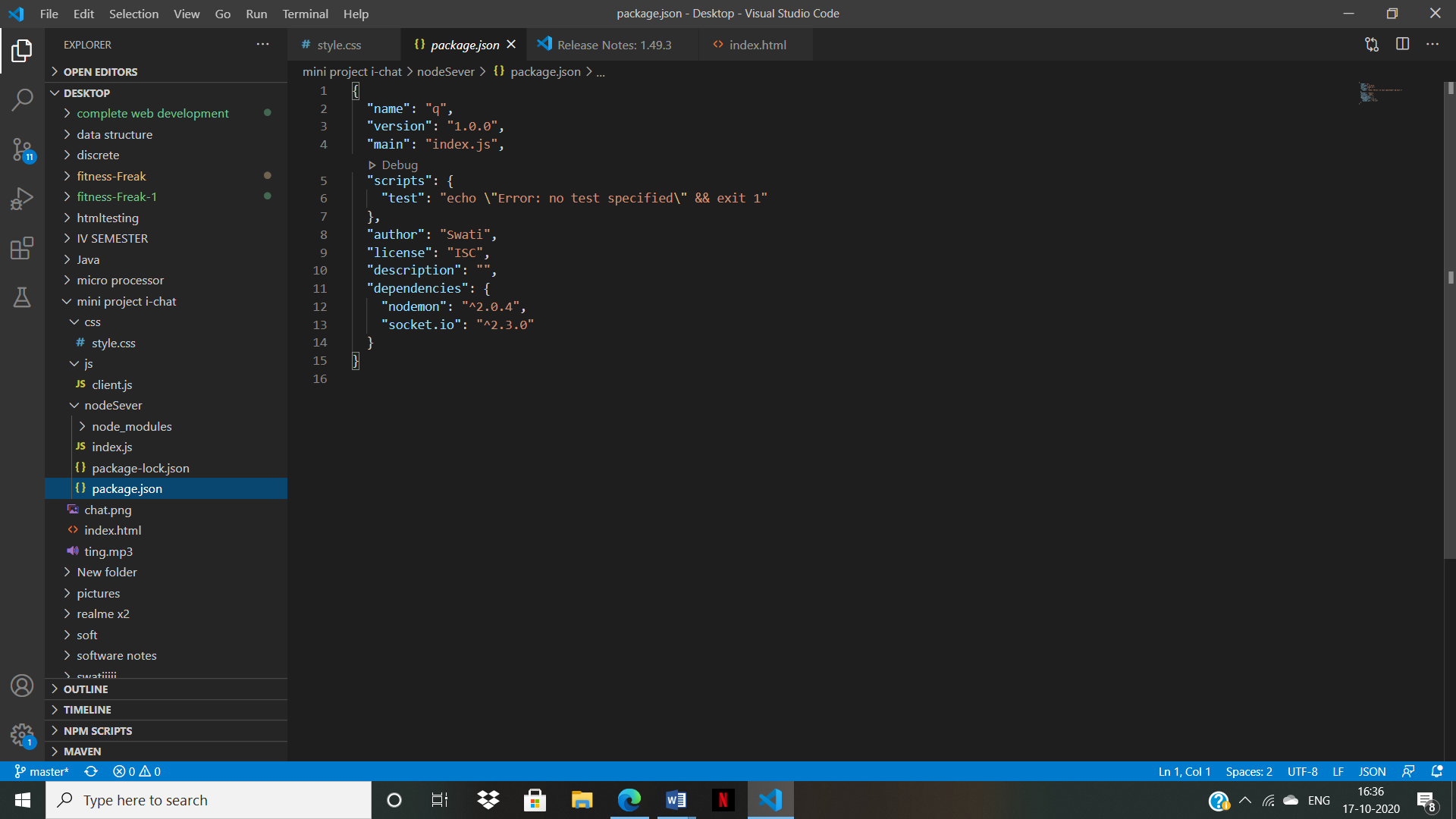


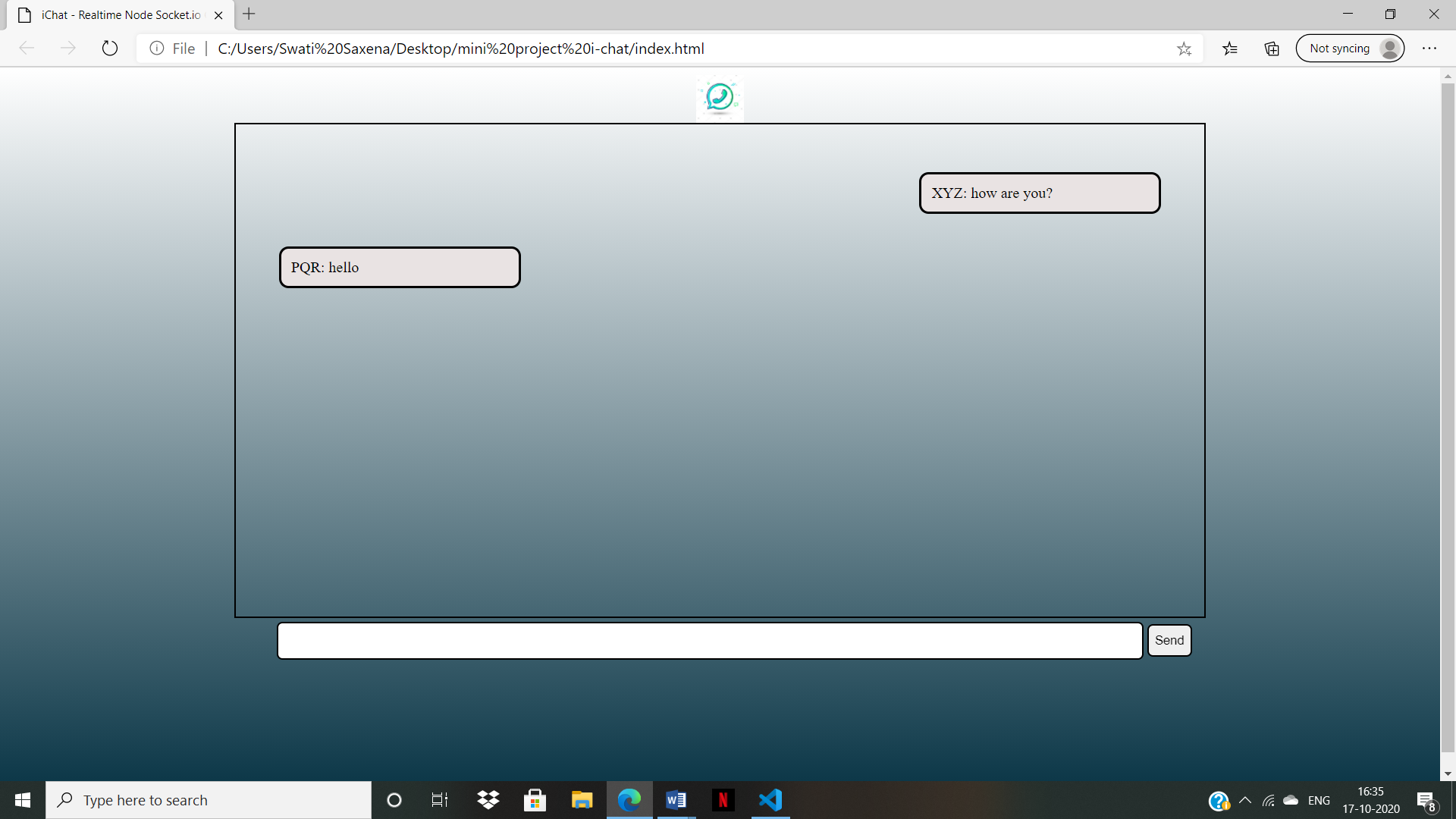


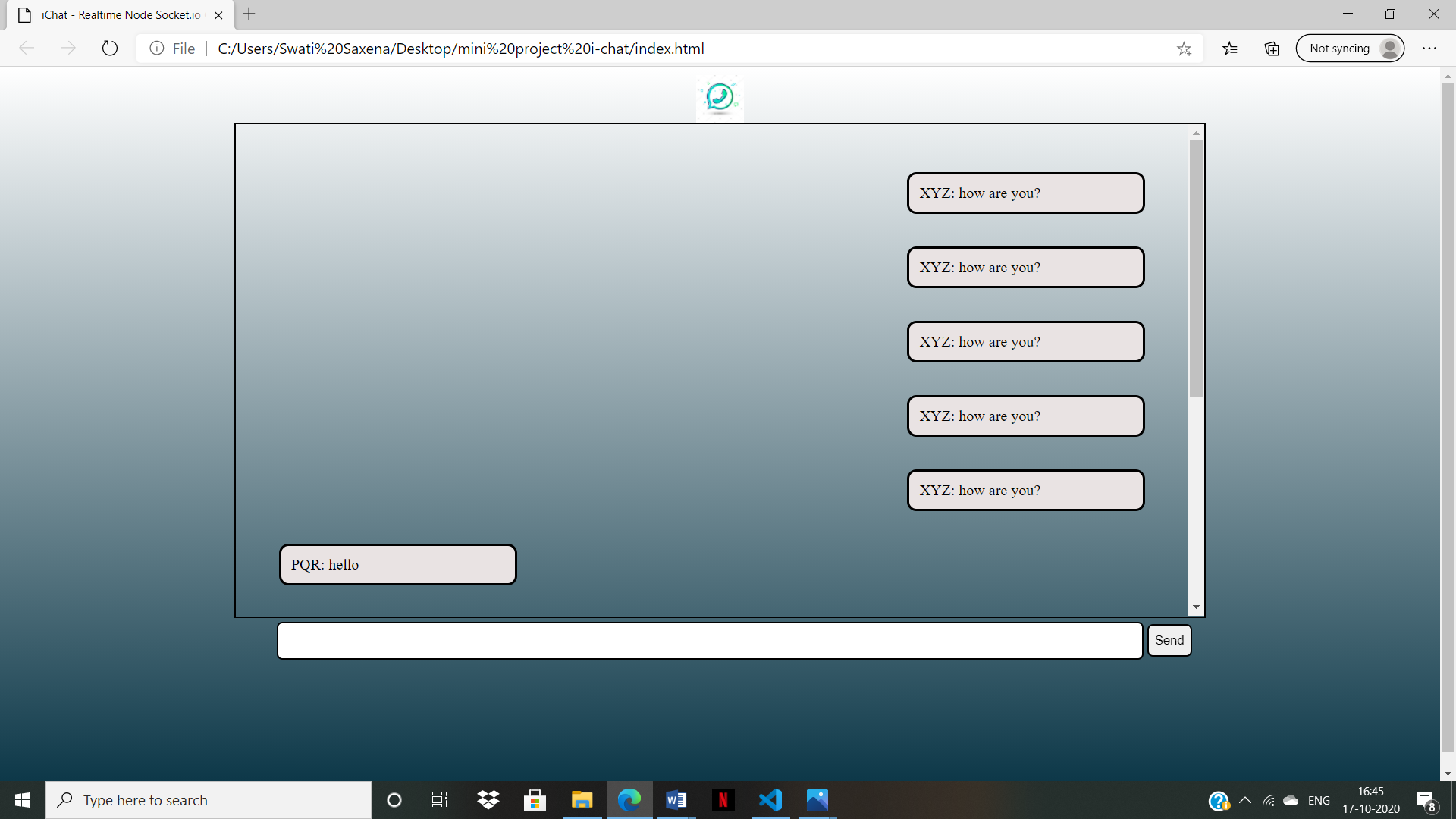


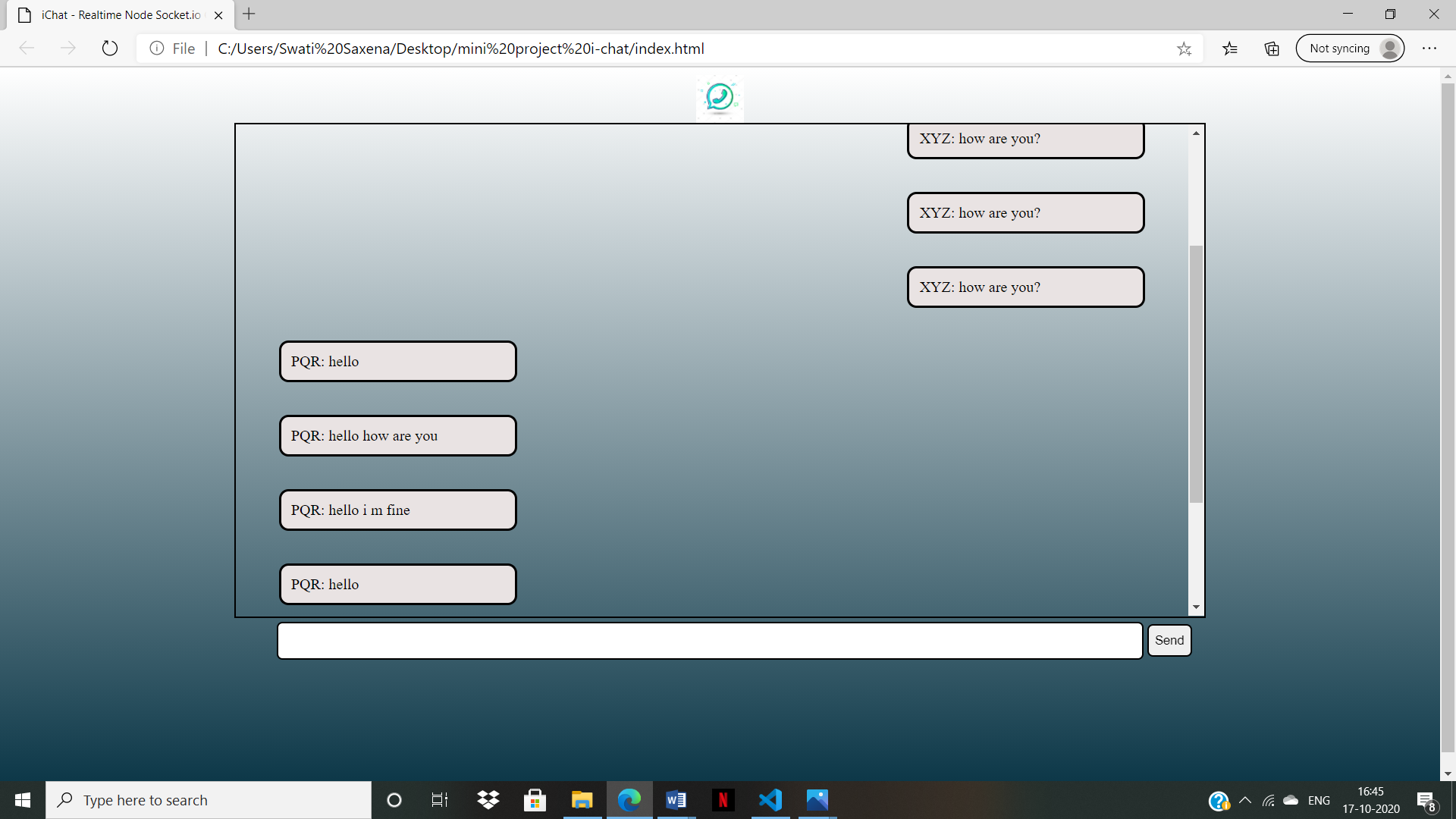












**References**

* <https://www.w3schools.com/js/DEFAULT.asp>
* <https://nodejs.dev/learn>
* <https://developer.mozilla.org/en-US/docs/Learn/CSS>
* <https://www.javatpoint.com/html-tutorial>