Full Stack Project

(SESSION-2021-2022)

Real-Time Chat System Report



**Institute of Engineering & Technology Team Members**

**Ishika Dubey (181500283)**

**Ashita Vijay Seth (181500145)**

**Aryan Garg (181500135)**

**Aryan Saxena (181500136)**

**Khush Sharma(181500343)**

***Supervised By***

Mr. Pankaj Kapoor (Assistant Professor)

**Department of Computer Engineering &Websites**

## Department of Computer Engineering and Applications

**GLA University, Mathura**

**17 km. Stone NH-2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**

# DECLARATION

I hereby declare that the work which is being presented in the Full Stack Project **“Real-Time Chat System”,** in partial fulfillment of the requirements for Full Stack project Lab is an authentic record of my own work carried under the supervision of **Mr. Pankaj Kapoor, Assistant Professor.**

## Aryan Saxena

## Ashita Vijay Seth

## Ishika Dubey

## Aryan Garg

## Khush Sharma

**Department of Computer Engineering and Applications**

**GLA University, Mathura**

**17 km. Stone NH-2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**

# CERTIFICATE

This is to certify that the project entitled “**Real-Time Chat System**” carried out in Full Stack Project – II Lab is a legitimate work done by Ashita Vijay Seth (181500145), Ishika Dubey (181500283), Aryan Garg(181500135), Aryan Saxena (181500136) ,Khush Sharma(181500343) and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).

### Signature of Supervisor:

**Name of Supervisor: Mr. Pankaj Kapoor Date: 06/05/202**

# ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Full Stack Project undertaken during B. Tech. Third Year. This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.

Our heartiest thanks to Dr. (Prof).Anand Singh Jalal, Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.

We owe special debt of gratitude to Mr.Pankaj Kapoor, Technical Trainer, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Aryan Garg

Aryan Saxena

Ashita Vijay Seth

Ishika Dubey

Khush Sharma

# Abstract

Messaging apps are surging in popularity nowadays, there are literally thousands of chat based apps and they are also very popular among the users as they provide a personal touch while connecting the world together.

Chatting is a method of using technology to bring people and ideas together despite of the geographical barriers. The technology has been available for years but the acceptance was quite recent. Our project is an example of a chat server.

In this project we will try to solve this lack of communication problem by providing a real time chat system which will allow the users to chat with each other. For Simplicity we have removed the authentication part, but the user can only access the chat after writing his name.

**Table of Contents**

[Declaration 2](#_TOC_250004)

[Certificate 3](#_TOC_250003)

[Acknowledgement 5](#_TOC_250002)

[Abstract 6](#_TOC_250001)

[Table Of Contents 7](#_TOC_250000)

1. Introduction 8
   1. Motivation and Overview 8
   2. Objective 9
2. Software Requirement Analysis 12
   1. Define the Problem 12
   2. Define the modules and their functionalities (SRS) 12
3. Software Design 18
   1. Types of Software Design 18
   2. Process 19
4. Software Testing 19
5. Requirements 20
   1. Hardware 20
   2. Software 20
6. Implementation and user interface 22
7. References/Bibliography 25

# Introduction:

Real-time chat system using JavaScript, Node Js and socket.io. For the frontend, we have used HTML, CSS, Bootstrap, and JavaScript. And for the backend, we have used Node Js, ExpressJs and Mongo DB for Database. We have used socket.io web socket library for real time data.

Our chat system will work in real-time. So we don't need to refresh the page to get new data.

We have used Socket.io for our Web Sockets connection.

# Motivation

Messaging apps are surging in popularity nowadays, there are literally thousands of chat based apps and they are also very popular among the users as they provide a personal touch while connecting the world together. This is the most basic chat app we can create. It is small and light weight but effective. You can add your functionalities in the app with a better UI and you are good to go.

# Overview

The main purpose of this project is to provide a real time service to the user to chat with each other.

# Objective:

The Objective of this project is to develop a front-end part of a chat website. It will allow the user to chat with each others in real time. Our main objective is developing a simple chat application; therefore for simplicity we have avoided the authentication part. The messages must be updated without refreshing the page.

It is a real time web application which means user does not have to refresh the page to see their chat. As soon as user presses the send button the message will be visible to everyone.

The chats are also stored in the mongo dB database with data, time and the name of the user who sent that message. These data can be used in the future by the admin to improve his/her content as per user requirements.

Our main focus:

* Communication: To develop an instant messaging solution to enable users to seamlessly communicate with each other.
* User friendliness: The project should be very easy to use enabling even a novice person to use it.

# Hypothesis:

If these would be successfully done then the user can chat with the other users. It will be extremely helpful as at this time when people are scrolling through different social media sites, and feeling lonely they can use it as a source to share their opinions, make friends and keep themselves entertained. In between all of this negativity, this app can help people be happier and keep themselves busy. To keep it simply we have skipped the authentication part.

Our main aim is to:

* This project is to create a chat application with a server and users to enable the users to chat with each other
* To develop an instant messaging solution to enable users to seamlessly communicate with each other.
* The project should be very easy to use enabling even a novice person to use it.
* Helps the user to express their feeling.
* Helps the admin to improve the content
* Real time application means no need to refresh the page again and again
* Can store messages with date, time and user name.

# Operational Definitions:

### Real-Time Chat System:

Our main objective through this project is to provide a simple user friendly we application through which users can interact.

# How it works:

* + In this website, while accessing for the first time, customers’ needs to give their name in the prompt box in order to access the website.
  + Once the user entered his/her name successfully, user can join the chat and send message.
  + Below the main chat area, there is a text box where the user can type their message and press the send button to send the message. User can type the message and have to press the send button in order to post his/her comment.
  + Date, time and name of the user is also shown on the posted comment.
  + It is real time application so user does not have to refresh the page in order to see his/her comment.
  + All the messages are stored in the database with date, time and name of the sender.

## About Real-Time Chat System

* + Let users express their feelings/opinion about any topic.
  + Simple, fast and convenient for sending messages.
  + Can store all the sent messages.
  + Can chat with other users without refreshing the page.

# Advantages:

* + User can show express their view/opinions about any topic while chatting..
  + Provides a medium of interaction between users.
  + Real time application so, no needs to refresh the page.
  + Can save all the messages in the database with the date, time and name of the sender.

# Disadvantages:

* + It requires a reliable internet connection.
  + System may provide inaccurate results if data not entered correctly.

# Software Requirement Analysis

**PROBLEM STATEMENT**

The functionality of the chat application is to give the ability to chat with whoever is online on the application. The users and stakeholders will be a small group for now, the use cases will be what is available to the user, and the functional/nonfunctional requirements will be covered, as well as the milestones of the chat application. It is a real time application so no needs to refresh the page. All the chats also get stored in the database.

# DEFINITIONS

ABOUT FRONT-END TECHNOLOGY**:**

The front-end stack is made up of many different languages and libraries. While these vary from application to application, there are only a few generic languages understood by all web browsers. These three main front-end coding languages are HTML, CSS and JavaScript.

Together, they create the underlying scaffolding that web browsers use to render the web pages that we interact with every day. All other libraries and front-end engineering are built upon these three main languages, which makes them must-have skills for any front-end developer.

In fact, you can think of a webpage like a house. The initial UX design is the blueprint. HTML is the basic structure of the house. The CSS is the paint, fixtures, and other aesthetic decisions that make the house look attractive. And finally, JavaScript is the inner workings of the house (lights, heating, and water) that we, the owner or renter, use and enjoy.

# VISUAL STUDIO:

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including [Java,](https://en.wikipedia.org/wiki/Java_(programming_language)#_blank) [JavaScript](https://en.wikipedia.org/wiki/JavaScript#_blank), [Go](https://en.wikipedia.org/wiki/Go_(programming_language)#_blank), [Node.js](https://en.wikipedia.org/wiki/Node.js#_blank) and [C++.](https://en.wikipedia.org/wiki/C%2B%2B#_blank) It is based on the [Electron](https://en.wikipedia.org/wiki/Electron_(software_framework)#_blank) framework, which is used to develop [Node.js](https://en.wikipedia.org/wiki/Node.js#_blank) [Web applications](https://en.wikipedia.org/wiki/Web_application#_blank) that run on the [Blink](https://en.wikipedia.org/wiki/Blink_layout_engine#_blank) [layout engine.](https://en.wikipedia.org/wiki/Blink_layout_engine#_blank) Visual Studio Code employs the same editor component (codenamed "Monaco") used in [Azure DevOps](https://en.wikipedia.org/wiki/Azure_DevOps_Server#_blank) (formerly called Visual Studio Online and Visual Studio Team Services).

Instead of a project system, it allows users to open one or more directories, which can then be saved in workspaces for future reuse. This allows it to operate as a [language-agnostic](https://en.wikipedia.org/wiki/Language-agnostic#_blank) code editor for any language. It supports a number of programming languages and a set of features that differs per language. Unwanted files and folders can be excluded from the project tree via the settings. Many Visual Studio Code features are not exposed through menus or the user interface, but can be accessed via the command palette.

Visual Studio Code can be extended via [extensions,](https://en.wikipedia.org/wiki/Plug-in_(computing)#_blank) available through a central repository. This includes additions to the editor and language support. A notable feature is the ability to create extensions that add support for new [languages,](https://en.wikipedia.org/wiki/Programming_language#_blank) [themes,](https://en.wikipedia.org/wiki/Theme_(computing)#_blank) and [debuggers,](https://en.wikipedia.org/wiki/Debugger#_blank) perform [static code](https://en.wikipedia.org/wiki/Static_code_analysis#_blank) [analysis,](https://en.wikipedia.org/wiki/Static_code_analysis#_blank) and add [code linters](https://en.wikipedia.org/wiki/Lint_(software)#_blank) using the [Language Server Protocol.](https://en.wikipedia.org/wiki/Language_Server_Protocol#_blank)

Visual Studio Code includes multiple extensions for FTP, allowing the software to be used as a free alternative for web development. Code can be synced between the editor and the server, without downloading any extra software.

Visual Studio Code allows users to set the [code page](https://en.wikipedia.org/wiki/Code_page#_blank) in which the active document is saved, the [newline](https://en.wikipedia.org/wiki/Newline#_blank) character, and the programming language of the active document. This allows it to be used on any platform, in any locale, and for any given programming language.

.

# WEB BROWSER:

A web browser (commonly referred to as a browser) is a [software application](https://en.wikipedia.org/wiki/Software_application#_blank) for accessing information on the [World Wide Web.](https://en.wikipedia.org/wiki/World_Wide_Web#_blank) Each individual [web page,](https://en.wikipedia.org/wiki/Web_page#_blank) image, and video is identified by a distinct [Uniform Resource Locator](https://en.wikipedia.org/wiki/URL#_blank) (URL), enabling browsers to retrieve these resources from a [web server](https://en.wikipedia.org/wiki/Web_server#_blank) and display them on the [user](https://en.wikipedia.org/wiki/User_(computing)#_blank)'s device.

A web browser is not the same thing as a [search engine,](https://en.wikipedia.org/wiki/Web_search_engine#_blank) though the two are often confused. For a user, a search engine is just a [website,](https://en.wikipedia.org/wiki/Website#_blank) such as [google.com](https://en.wikipedia.org/wiki/Google_Search#_blank), that stores searchable data about other websites. But to connect to a website's server and display its web pages, a user needs to have a web browser installed on their device.

The most popular browsers are [Chrome,](https://en.wikipedia.org/wiki/Google_Chrome#_blank) [Firefox,](https://en.wikipedia.org/wiki/Firefox#_blank) [Safari,](https://en.wikipedia.org/wiki/Safari_(web_browser)#_blank) [Internet Explorer](https://en.wikipedia.org/wiki/Internet_Explorer#_blank), and [Edge](https://en.wikipedia.org/wiki/Microsoft_Edge#_blank).

# Technical Feasibility:

The proposed system is developed using HTML, CSS and bootstrap as front-end tool and PHP and JS node as the back end. The proposed system needs a Personal Web Server to serve the requests submitted by the users. The Web browser is used to view the web page that is available within the Windows operating system itself. The proposed system will run under Win9x, NT, and win2000 environment. As Windows is very user friendly and GUI OS it is very easy to use. All the required hardware and software are readily available in the market. Hence the system is technically feasible.

# Operational Feasibility:

The proposed system is operationally feasible because of the following reasons.

* The customer is benefited more as most of his time is saved. The customer is serviced at his place of work.
* The purpose of this website serves the good and needy people.

# Economic Feasibility:

As the necessary hardware and software are available in the market at a low cost, the initial investment is the only cost incurred and does not need any further enhancements. Hence it is economically feasible. The system is feasible in all respects and hence it encourages taking up the system design. We have used different languages and technologies for preparing the project.

## LANGUAGES USED:

1. **HTML5**:

**Introduction**

HTML is a mark-up language whose role is to prepare written documents using formatting tags. HTML forms the skeleton of a web page. HTML stands for Hypertext Mark-Up Language.

The WWW (World Wide Web) is a worldwide network which is formed by multiple web pages which are connected to each other via hyperlinks.

Web pages are usually organized around a main page, mostly named as index.html, the other web pages are connected through links(hyperlinks) to the main page.

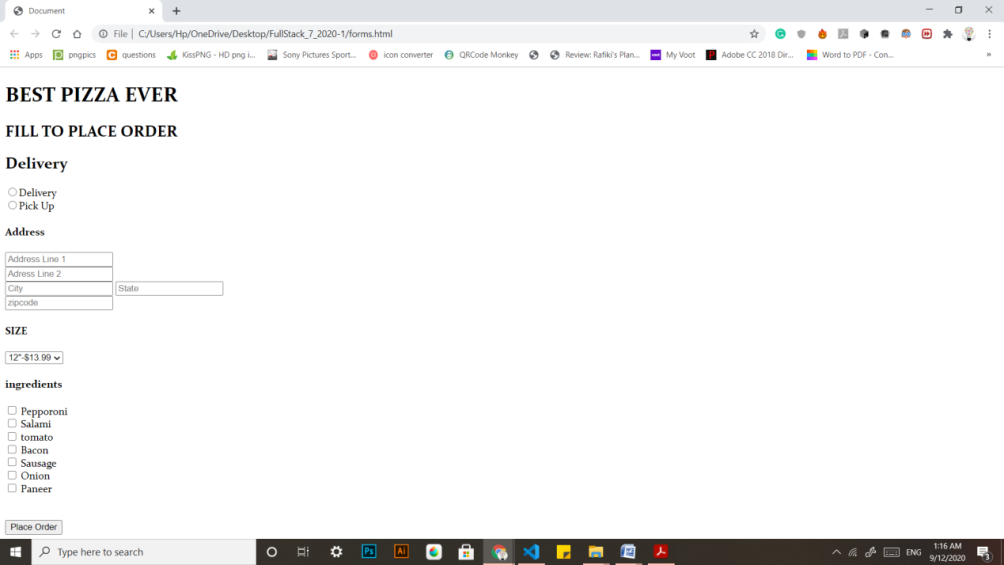


Fig 2 shows the skeleton programmed using HTML only

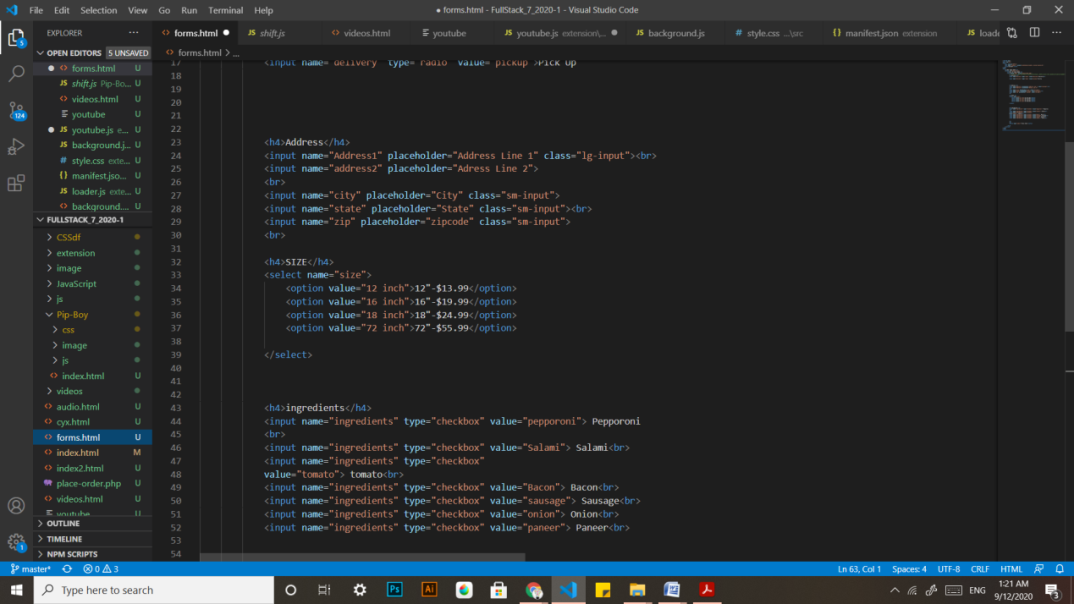


Fig 3 shows the HTML code for the above webpage

**Versions and Structure of HTML:**

HTML was designed by Tim Berners- Lee in 1989. The latest version of HTML is HTML 5.0.

Basic Structure of HTML file:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<!--HTML COMMENT-->

</body>

</html>

## CSS3:

**What is CSS?**

CSS stands for Cascading Style Sheets. While HTML gives a basic Structure of the web page , CSS gives the style to the web page i.e. it can change the whole look or the appearance of a web page.

CSS saves a lot of work. It can control the layout of multiple web pages all at once . CSS can be either external or internal.

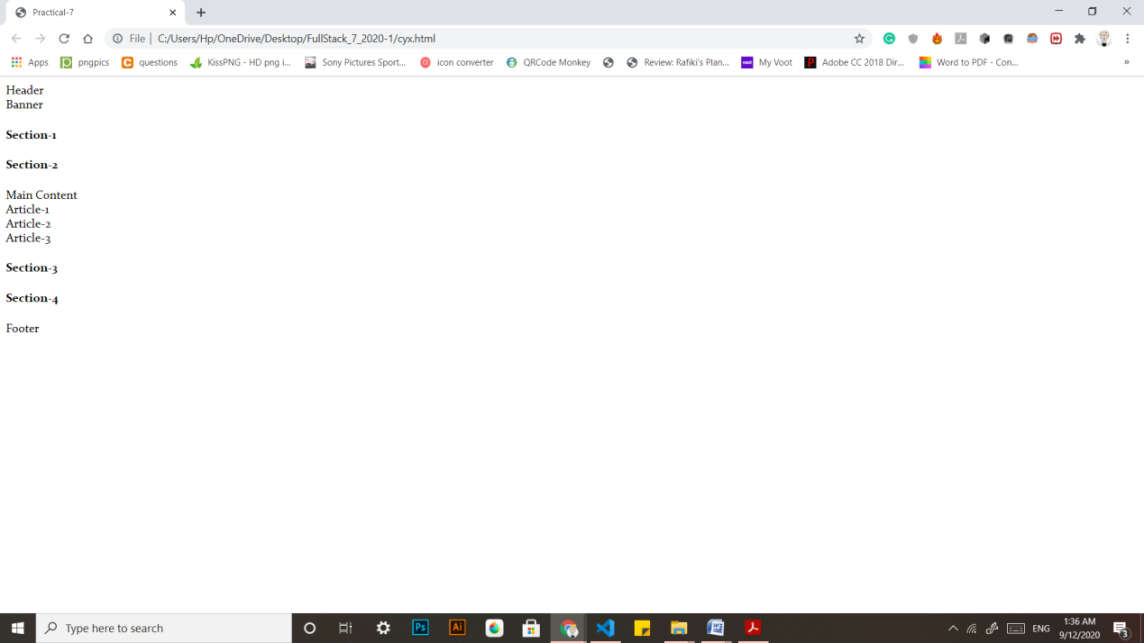


Fig. 4 shows the HTML structure of a web page

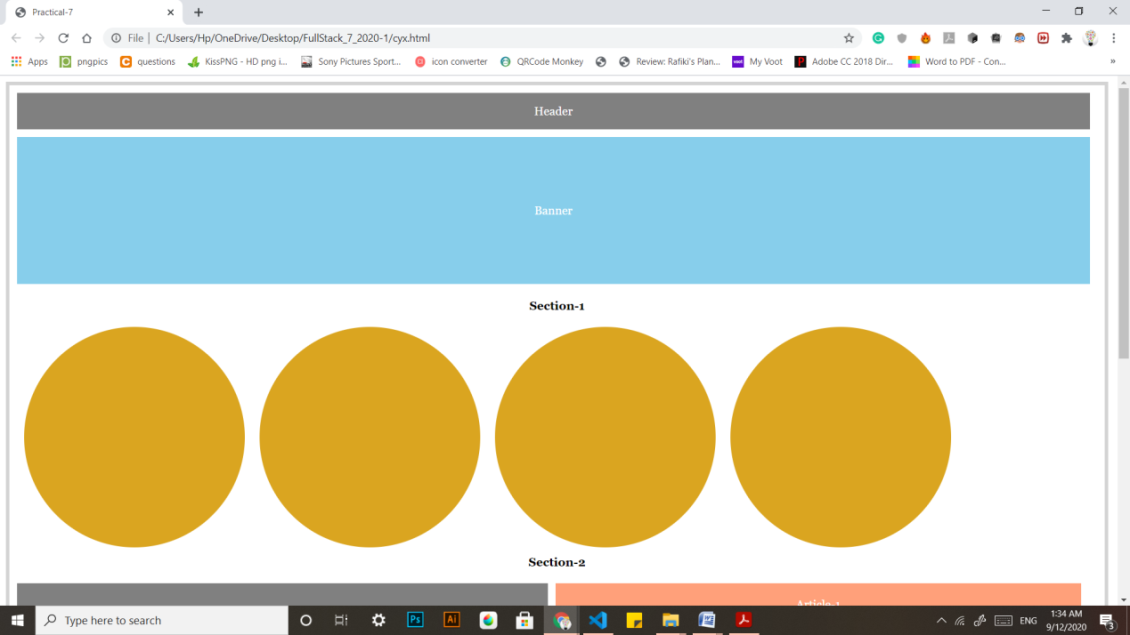


Fig. 5 shows the webpage after styling the above web page using CSS

**5.2 CSS Syntax**:

* External CSS

The external CSS can be declared in the required HTML pages as:

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="./style.css">

</head>

The external CSS file is saved by using the “.css” extension.

* Internal CSS

The internal CSS is saved in corresponding HTML file using the <style> tag in the <head> section

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Practical-7</title>

    <style>

        body{

            background-color: teal;

            font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;

        }

</style>

</head>

**Which CSS is better?**

Using External CSS is better than using internal CSS because:

* + It is easy to maintain external CSS
  + It reduces the file size
  + Can be connected to multiple HTML files at the same time
  + Improves flexibility

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

## Bootstrap4:

Bootstrap 4 is the latest version of bootstrap, which is the most popular HTML,CSS, JavaScript framework for developing responsive websites.

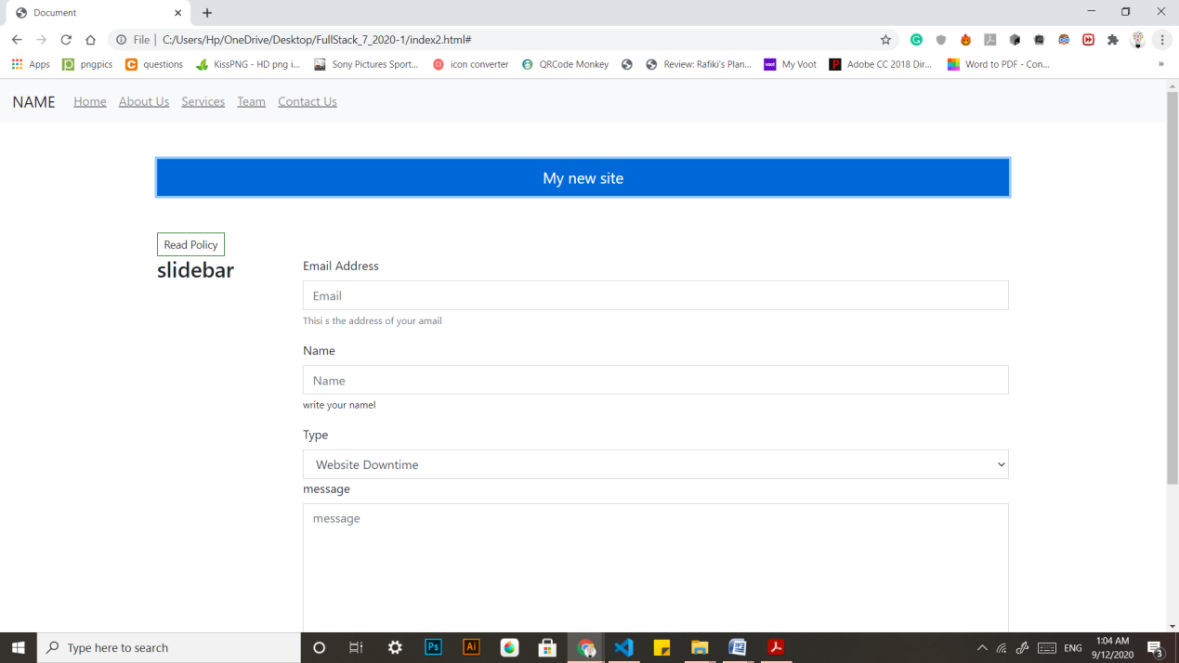


Fig.7 shows a form build using Bootstrap

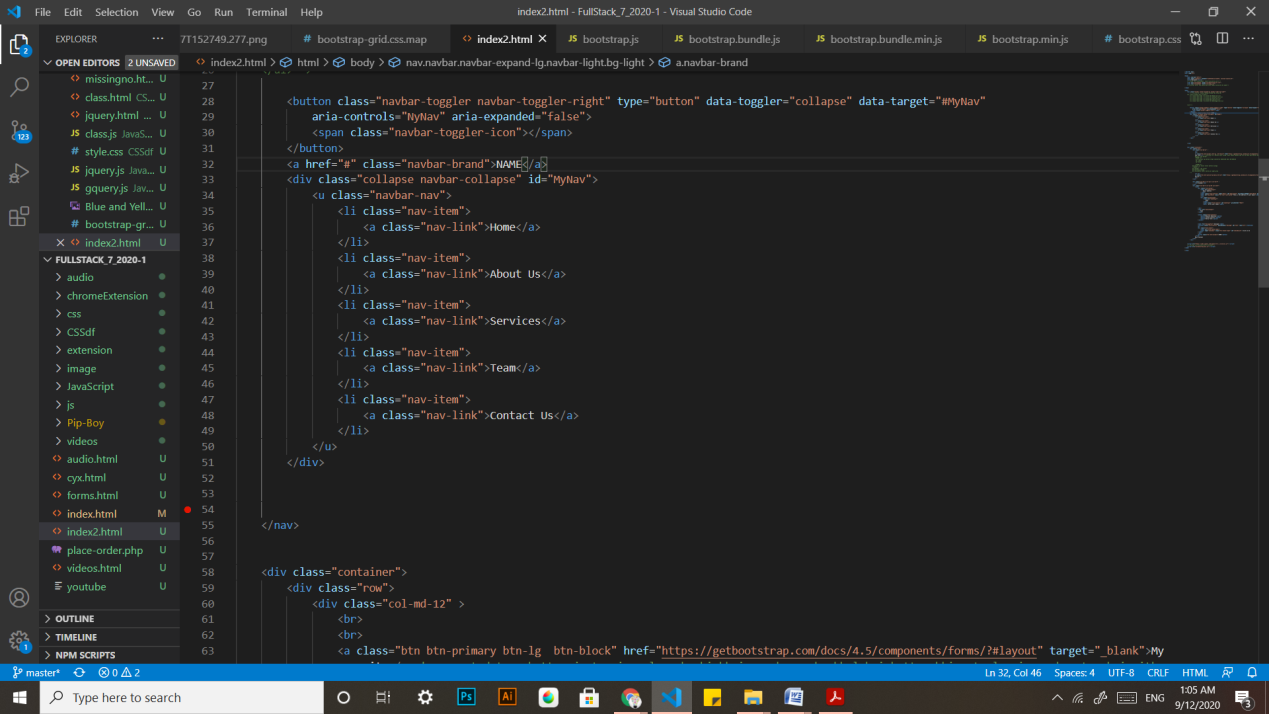
****

Fig.8 shows HTML and Bootstrap implementation for Fig 7 web page

Bootstrap is a free front-end framework for faster and easier web development. It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plug ins

Linking Bootstrap to the html page:

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="css/bootstrap.min.css">

    <link rel="stylesheet" href="css/swag.css">

    <!--written after the bootstrap so that the changes can be overwrittten-->

</head>

### Getting Started with Bootstrap Basics

Bootstrap is available in two forms; as a precompiled version, and as a source code version. The source code version uses the [Less](http://lesscss.org/#_blank) CSS preprocessor, but if you are more into Sass, there is an official Sass port of Bootstrap also available. To make it easier to make use of CSS vendor prefixes, Bootstrap uses [Autoprefixer.](https://github.com/postcss/autoprefixer#_blank)

The source code version comes styles source code written in Less (or Sass), all the JavaScript, and accompanying documentation. This allows more ambitious designers and developers to change and customize, at their will, all the provided styles, and to build their own version of Bootstrap.

### JavaScript:

**What is JavaScript?**

JavaScript is an object-based scripting language that is lightweight and cross-platform. JavaScript is not compiled but translated. The JavaScript Translator (embedded in browser) is responsible to translate the JavaScript code**. JavaScript** is used to program the behavior of web pages

It is mainly used for:

* Dynamic Drop-down menus
* Displaying Date and time
* Displaying pop-ups and alerts
* Displaying clocks etc.

**Syntax of JavaScript:**

* External linking:

<script src=”myScript.js”></script>

* Internal linking:

<script>

    function myfunc(){

        document.getElementById(“demo”).innerHTML=”PARA CHANGED”;

    }

    </script>

Like CSS, JavaScript also can be placed in:

1. between the body tag of html

2. In .js file (external JavaScript)

3. between the head tag of html

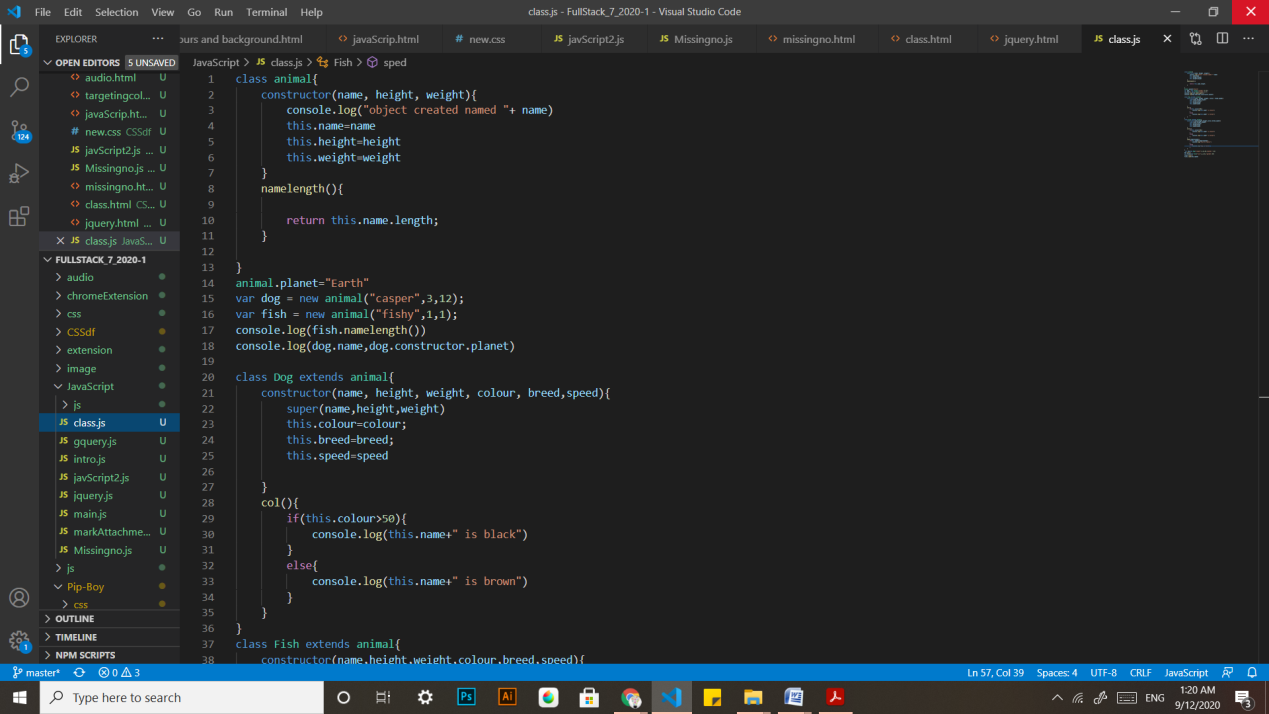


Fig .6 shows a basic JavaScript code

# ABOUT BACK-END TECHNOLOGY:

### Node JS:

Node.js isan [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform,](https://en.wikipedia.org/wiki/Cross-platform) [back-end](https://en.wikipedia.org/wiki/Front_end_and_back_end) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [runtime environment](https://en.wikipedia.org/wiki/Runtime_environment) that runs on the [V8 engine](https://en.wikipedia.org/wiki/V8_(JavaScript_engine)) and executes JavaScript code outside a [web browser](https://en.wikipedia.org/wiki/Web_browser). Node.js lets developers use JavaScript to write command line tools and for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting)—running scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying [web-](https://en.wikipedia.org/wiki/Web_application) [application](https://en.wikipedia.org/wiki/Web_application) development around a single programming language, rather than different languages for server-side and client-side scripts.

Node.js brings [event-driven programming](https://en.wikipedia.org/wiki/Event-driven_programming) to [web servers,](https://en.wikipedia.org/wiki/Web_server) enabling development of fast web servers in JavaScript. Developers can create scalable servers without using [threading,](https://en.wikipedia.org/wiki/Thread_(computing)) by using a simplified model of [event-driven programming](https://en.wikipedia.org/wiki/Event-driven_programming) that uses callbacks to signal the completion of a task. Node.js connects the ease of a scripting language (JavaScript) with the power of UNIX network programming.

### MONGODB:

**Mongo DB** is a [source-available](https://en.wikipedia.org/wiki/Source-available) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [document-oriented database](https://en.wikipedia.org/wiki/Document-oriented_database) program. Classified as a [NoSQL](https://en.wikipedia.org/wiki/NoSQL) database program, Mongo DB uses [JSON](https://en.wikipedia.org/wiki/JSON)-like documents with optional [schemas.](https://en.wikipedia.org/wiki/Database_schema) Mongo DB is developed by [Mongo DB Inc.](https://en.wikipedia.org/wiki/MongoDB_Inc)

Mongo DB supports field, [range query,](https://en.wikipedia.org/wiki/Range_query_(database)) and [regular-expression](https://en.wikipedia.org/wiki/Regular_expression) searches. Queries can return specific fields of documents and also include user-defined [JavaScript](https://en.wikipedia.org/wiki/JavaScript) functions. Queries can also be configured to return a random sample of results of a given size.

Mongo DB provides high availability with replica sets. A replica set consists of two or more copies of the data. Each replica-set member may act in the role of primary or secondary replica at any time.

### ExpressJS:

Expressis a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. It is an open source framework developed and maintained by the Node.js foundation.

Express provides a minimal interface to build our applications. It provides us the tools that are required to build our app. It is flexible as there are numerous modules available on npm, which can be directly plugged into Express.

With ExpressJS, you need not worry about low level protocols, processes, etc.

# SOFTWARE DESIGN

**Preliminary Design:** Preliminary design is basically concerned with deriving an overall picture of the system. Deriving entire system into modules and sub-modules while keeping Cohesion and Coupling factors in mind. Tools, which assist in preliminary design process, are Data Flow Diagrams.

**Code design:** The purpose of code is to facilitate the identification and retrieval for items of information. A code is an ordered collection of symbols designed to provide unique identification of an entity or attribute. To achieve unique identification there must be only one place where the identified entity or the attribute can be entered in the code; conversely there must be a place in the code for everything that is to be identified. This mutually exclusive feature must be built into any coding system. The codes for this system are designed with two features in mind. Optimum human oriented use and machine efficiency. Length of the code range from length of one to length of five characteristics:

* The code structure is unique; ensuring that only one value of the code with a single meaning may be correctly applied to a given entity or attributes.
* The code structure is expansible allowing for growth of its set of entities and attributes.
* The code is concise and brief for recording, communication, transmission and storage efficiencies.
* They have a uniform size and format.
* The codes are simple so that the user can easily understand it.
* The codes are also versatile i.e., it is easy to modify to reflect necessary changes in condition, chart eristic and relationships of the encode entities.
* The codes are also easily storable for producing reports in a predetermined order of format.
* The codes are also stable and do not require being frequently updated thereby promoting user efficiency.
* The codes are also meaningful.
* They are also operable i.e., they are adequate for present and anticipate data processing both for machine and human use.

# SOFTWARE TESTING

Testing is a process of executing a program with the intent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. A good test case is one that has a high probability of finding an as undiscovered error.

The test approach is divided into three main phases:

Module testing, integration tests and system testing.

In addition, the system testing includes two sub-phases: functional and usability testing. These planned tests are explained briefly below.

1. **Module testing** will perform during coding by using debug messages to check that the written code produces wanted results. An important requirement is that the code will compile with zero bugs.
2. **Integration testing** will perform after finish module testing in order to validate if each module can work fine with each other. Integration Test proves that system works as integrated unit when all the fixes are complete.
3. **System testing** includes two phases: functional testing and usability testing. These will perform after the product reaches its final version. During functional test phase, the tester will test if the product meets the game requirements. The tester tests the requirements using the use cases listed below in Test Cases section. The usability test will perform to understand how easy it is to use this restaurant website. Any person out of the team members will perform this test by selecting the category and placing the order via call.

# REQUIREMENTS:

Following are the hardware and the software requirements for our project:

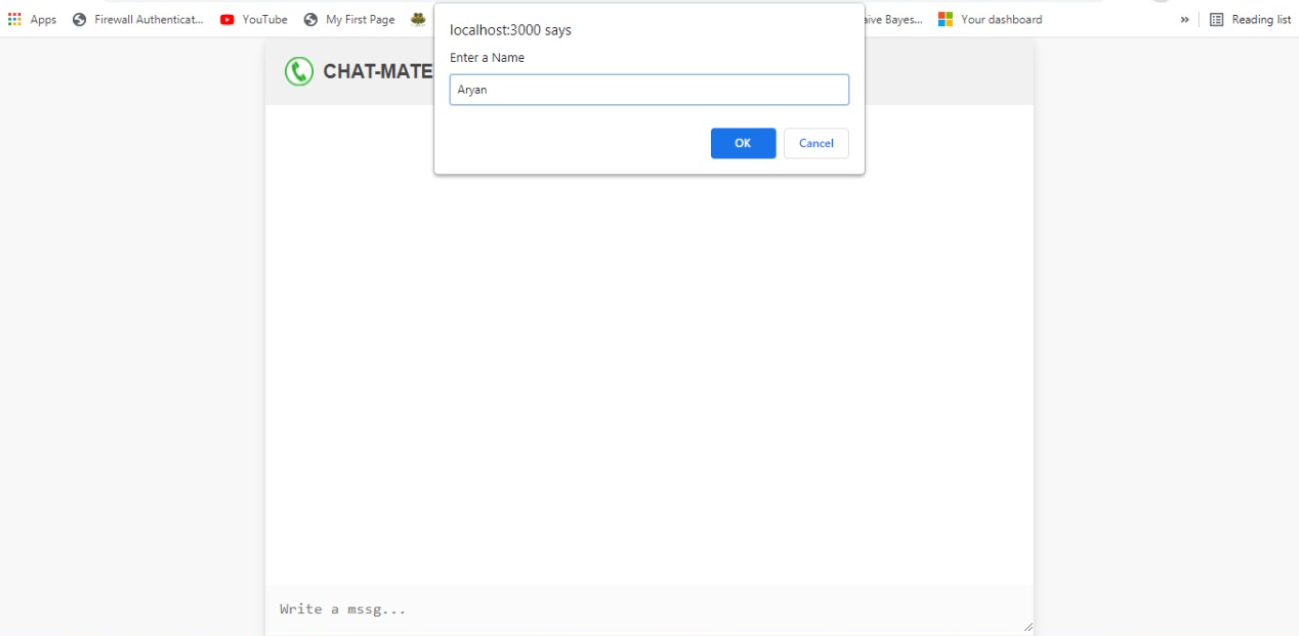
### Hardware:

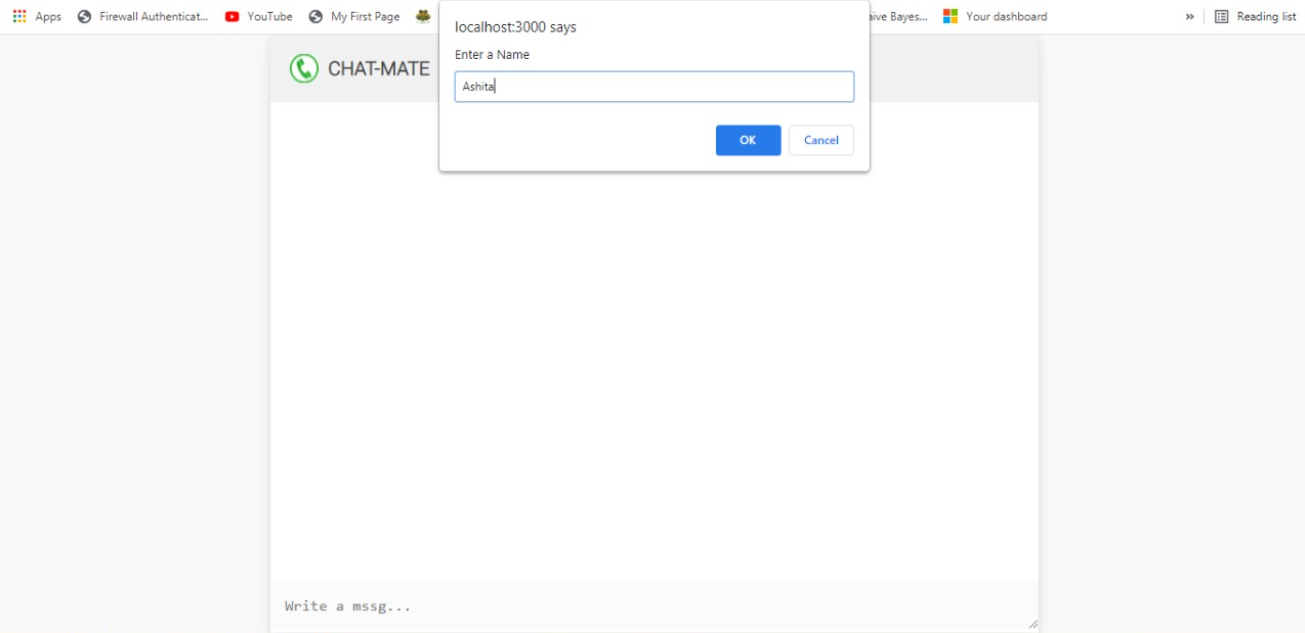
* Laptop/Desktop
* 1.8 GHz or faster processor. Quad-core or better recommended
* 4 GB of RAM
* Hard disk space: Minimum of 800MB up to 210GB of available space
* Video card that supports a minimum display resolution of 720p (1280 by 720)

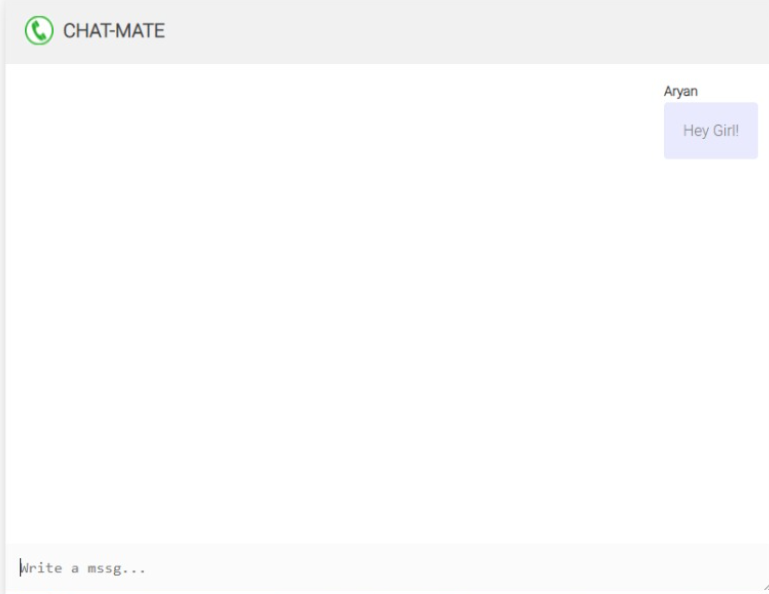
### Software:

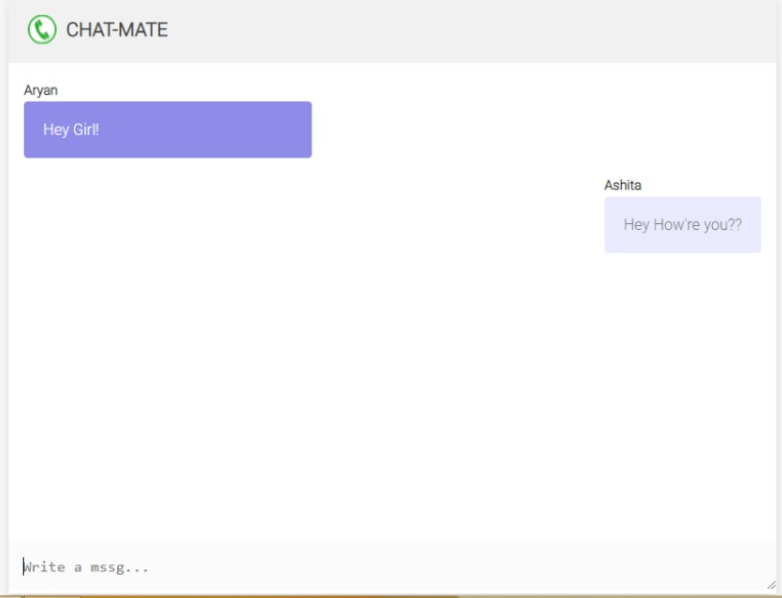
* Windows 8.1 and above
* Visual Studio
* Web Browser

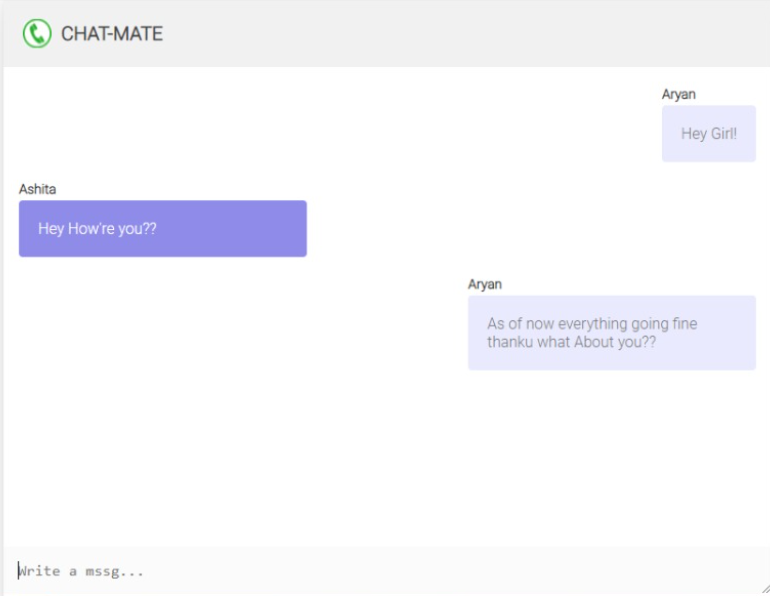
**User Interface:**

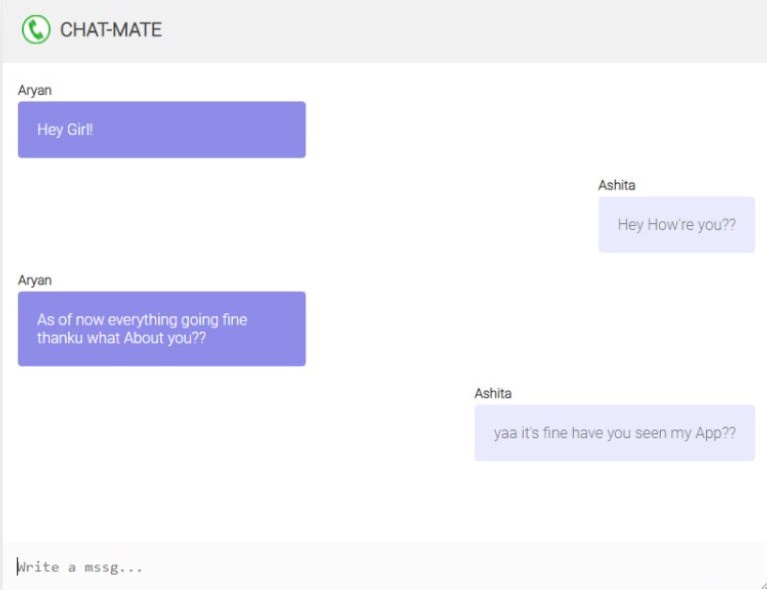


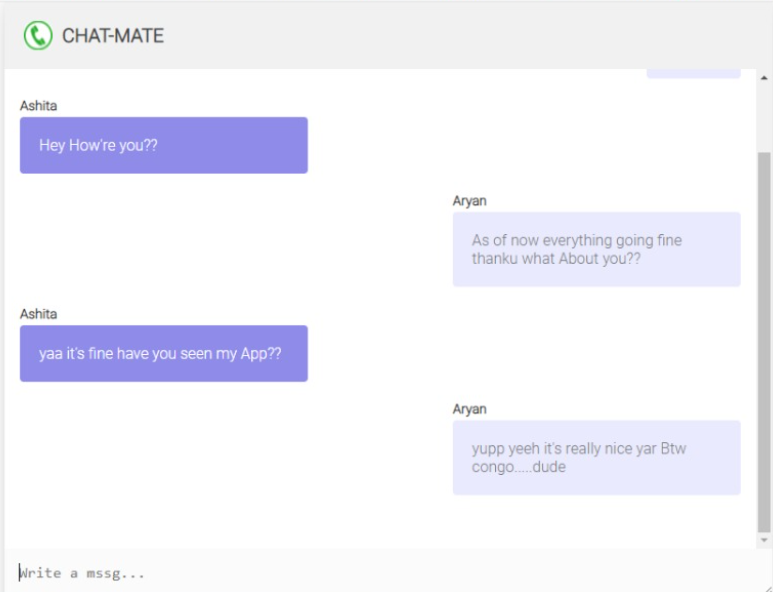












# Conclusion

So, in this web application we have built a real time chat system in which user has enter his/her name first in order to access the webpage. One can chat with anyone who is online by typing a messaging and pressing sent. It is a real time web application so user does not have to bother about refreshing the page again and again. The messages will be stored in the database with sender’s name, time and date.

# References:

* <https://www.w3schools.com/>
* <https://stackoverflow.com/>
* [www.bootstrap.com](http://www.bootstrap.com/)
* [www.youtube.com](http://www.youtube.com/)
* <https://github.com/>