
Table of Contents

Acknowledgement	i
Abstract	ii
List of Figures	v
INTRODUCTION.....	1
REQUIREMENT ANALYSIS.....	3
2.1 One-on-one interviews	3
2.2 Group Interviews.....	3
2.3 Questionnaires/Surveys.....	3
2.4 User Observation.....	4
2.5 Requirements Pertaining to the Project.....	4
SYSTEM REQUIREMENT SPECIFICATION.....	5
3.1 Hardware Requirements	5
3.2 Software Requirements	5
ANALYSIS AND DESIGN	6
IMPLEMENTATION	9
5.1 File Structure	9
5.2 Technologies Used	10
5.2.1 HTML (Hyper Text Markup Language).....	10
5.2.2 CSS (Cascading Style Sheets)	10
5.2.3 JavaScript.....	10
5.2.4 PHP	10
5.3 Code used at client and server sides.....	11
5.3.1 Code at client side:.....	11
5.3.2 Functions at server side:.....	12
TESTING.....	13
6.1 Testing Techniques	13

6.2	Software Testing Fundamentals	13
6.3	Testing Objective	13
6.4	Test Cases.....	14
SNAPSHOTS.....		15
CONCLUSION		17
FUTURE ENHANCEMENTS.....		18
REFERENCES.....		19

List of Figures

Fig 1.1: Web Application Diagram	2
Fig 4.1: Sitemap of the Website	7
Fig 4.2: Web System Architecture	8
Fig 5.1: File Structure of the Project	9
Fig 7.1: Sign in Section	15
Fig 7.2: Register Section	15
Fig 7.3: The Homepage	16
Fig 7.4: Sign in with validation	16

CHAPTER 1

INTRODUCTION

Web Technology

- **Web programming**, also known as **web development**, is the creation of dynamic web applications. There are two broad divisions of web development – front-end development (also called client-side development) and back-end development (also called server-side development).
- Front-end development refers to constructing what a user sees when they load a web application – the content, design and how you interact with it. This is done with three codes – HTML, CSS and JavaScript. HTML, short for Hyper Text Markup Language, is a special code for ‘marking up’ text in order to turn it into a web page. Every web page on the net is written in HTML, and it will form the backbone of any web application. CSS, short for Cascading Style Sheets, is a code for setting style rules for the appearance of web pages. CSS handles the cosmetic side of the web. Finally, JavaScript is a scripting language that’s widely used to add functionality and interactivity to web pages.
- Back-end development controls what goes on behind the scenes of a web application. A back-end often uses a database to generate the front-end.

Web applications

- In computing, a **web application** or **web app** is a client-server program which the client (including the user interface and client-side logic) runs in a web server.
 - Common web applications include webmail, online retail sales, and online auction.
 - Chatting applications are some such web apps that serve user needs by providing a platform for them to discuss about various topics of interest.
-



Fig 1.1: Web Application Diagram

- Web applications use a combination of server-side scripts (like PHP) to handle the storage and retrieval of the information, and client-side scripts (using JavaScript) to present information to users.
- This allows users to interact with the company using online forms, content management systems, shopping carts and more. In addition, the applications allow employees to create documents, share information, collaborate on projects, and work on common documents regardless of location or device.

1.1 Problem Statement

The purpose of this project is to enable and quantify the service satisfaction levels of users who use chatting applications to discuss about topics of interest, topics about work, topics about entertainment, etc.

1.2 Objective

The main objective of this mini project is to increase the user satisfaction and to determine the relationship with user satisfaction by developing an application using web technologies.

1.2 Scope

This project enables people to use the chatting application to discuss on various topics. It can be done by creating an account and chatting on the discussion channel.

CHAPTER 2

REQUIREMENT ANALYSIS

The purpose of this paper is to examine the different methods in gathering requirements. Requirements are one of the most vital pieces to ensuring the success of a system or project. To ensure the optimal requirements are received, the methods in which those requirements are obtained are equally important.

Types of requirements analysis:

2.1 One-on-one interviews

One-on-one interviews are the most common technique for gathering requirements, as well as one of the primary sources of requirements. To help get the most out of an interview, they should be well thought out and prepared before sitting with the interviewee. The analyst should identify stakeholders to be interviewed. These can be users who interact with the current or new system, management, project financiers or anyone else that would be involved in the system.

2.2 Group Interviews

Group interviews are like one-on-one interviews, except there is more than one person being interviewed. Group interviews work well when the interviewees are at the same level or position. A group interview also has an advantage when there is a time constraint.

2.3 Questionnaires/Surveys

Questionnaires, or surveys, allow an analyst to collect information from many people in a relatively short amount of time. This is especially helpful when stakeholders are spread out geographically, or there are dozens to hundreds of respondents whose input will be needed to help establish system requirements.

2.4 User Observation

The direct approaches of interviewing and questionnaires provide valuable user feedback based on the questions asked of them; however, there are times when direct observation may be better suited in requirement gathering. To get a better understanding of a user in their in current work environment, the analyst may observe the user themselves. User observation is helpful in assisting the analyst by getting a full grasp of how the user interacts with the system, first hand.

2.5 Requirements Pertaining to the Project

- To reduce clutter of information all dumped into a single discussion group.
- Having parallel discussions on different topics all under same group but with different topics grouped together individually as channels.
- Increase in response time and make way for an effective discussion/conversation on all the topics due to the organised content.
- Enhanced user experience of only providing to relative information to the topic out of all the messages in a channel.
- Making each message as thread structure will give finer grain of organization.
- Limiting notifications by only notifying the user of the thread that he/she has been a part of and their marked important channles.

CHAPTER 3

SYSTEM REQUIREMENT SPECIFICATION

- Requirements analysis is critical for project development. Requirements must be documented, actionable, measurable, testable and defined to a level of detail sufficient for system design.
- Requirements can be architectural, structural, behavioural, functional, and non-functional. A software requirements specification (SRS) is a comprehensive description of the intended purpose and the environment for software under development.

3.1 Hardware Requirements

- Minimum of 1 GB of main memory
- Minimum of 3 GB of storage
- Keyboard
- Mouse
- Display Unit
- Dual-Core CPU with a minimum speed of 1.5 GHz

3.2 Software Requirements

- Windows – XP and above or Linux based OS
 - A text editor
 - XAMPP/WAMP
 - A web browser
-

CHAPTER 4

ANALYSIS AND DESIGN

The core part of website development and design is not necessary the coding process. Indeed, such technologies as HTML, CSS, and JavaScript give the web we know its shape and define the way we interact with the information. But what usually stay behind the scenes and, at the same time, remain the crucial part of website development life cycle are the stages of preliminary information gathering, detailed planning, and post-launch maintenance.

We have followed some of the steps of Website Development Life Cycle to do our mini project.

4.1 Website Development Life Cycle

Step 1. Gathering Information:

The most important task at this point is to get the clear understanding of your future website purposes, the main goals you wish to get, and the target audience you want to attract to your site. Such kind of a website development questionnaire helps to develop the best strategy for further project management.

Step 2. Planning:

At this stage of website development cycle, the developer creates the data that can give to a customer an opportunity to judge how the entire site will look like.

Based on the information that was gathered together in the previous phase, the sitemap is created as shown in the following figure. The site map is designed so that all the similar files are kept together in single directories so that navigating the site structure for developers becomes easier for

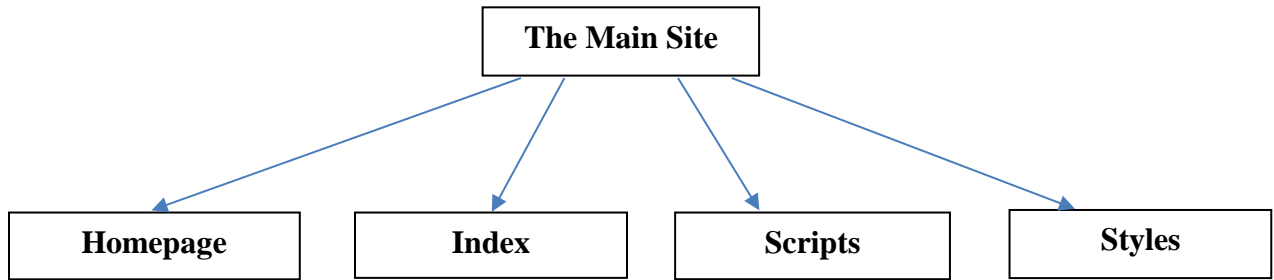


Fig 4.1: Sitemap of the Website

Step 3. Design:

Website layout is the result of designer's work. It can be a graphic sketch or an actual graphic design. The primary function of the layout is to represent the information structure, visualize the content, and demonstrate the basic functional. Layouts contain colors, logos, images and can give a general understanding of the future product.

Step 4. Content Writing and Assembly:

Content writing and compiling usually overlaps with other stages of website creation, and its role can't be underestimated. At this step it is necessary to put in writing the very essence you'd like to communicate to the audience of your website and add calls-to-action. Content writing involves also creation of catching headlines, text editing, writing new text, compiling the existing text, etc., which takes time and effort. As a rule, the client undertakes to provide website content ready to migrate to the site. It is better when all website content is provided before or during website coding.

Step 5. Coding:

At this step, you can finally start creating the website itself. Graphic elements that have been designed during the previous stages should be used to create an actual website. Usually, the home page is created first, and then all sub-pages are added, according to the website hierarchy.

Step 6. Testing:

Testing is probably the most routine and important part of a process. Every single link should be tested to make sure that there are no broken, forms and scripts that are used must be tested before deploying the project, validation of user input must be done.

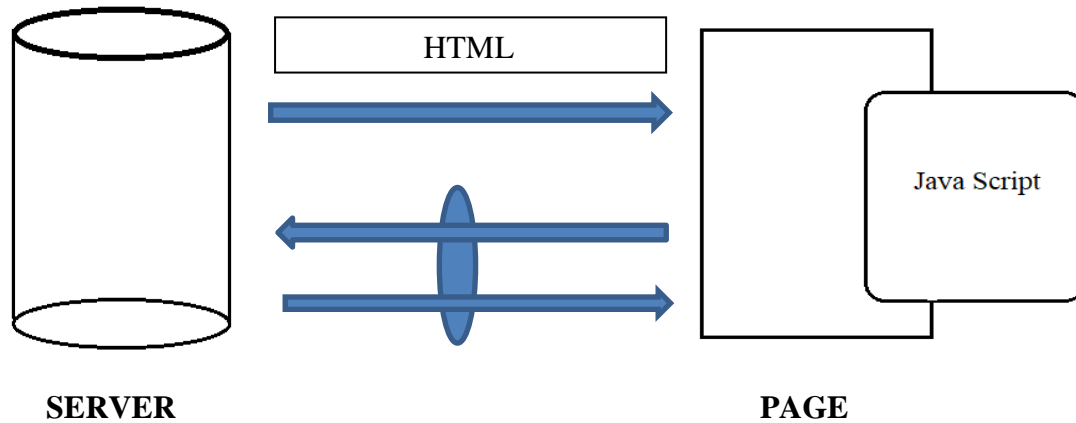


Fig 4.2: Web System Architecture

CHAPTER 5

IMPLEMENTATION

5.1 File Structure

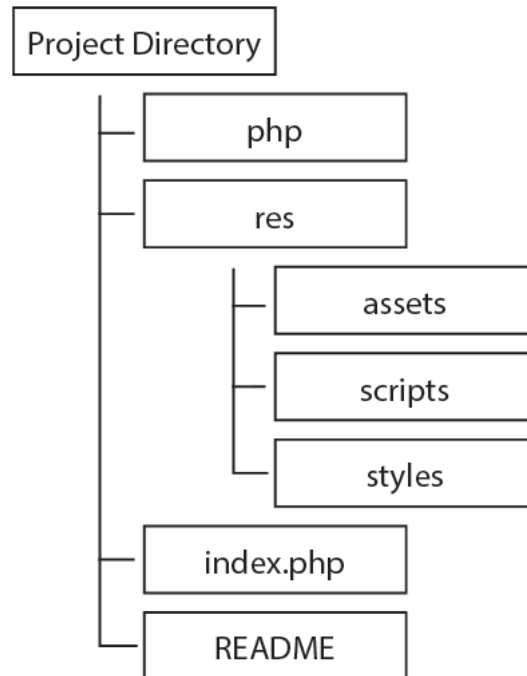


Fig 5.1: File Structure of the Project

The above figure depicts the file structure of the project that has been implemented. The root node or the directory in which all the essential files are stored is the slackchat directory (or the main directory). It contains many other important subdirectories and the index.php file.

- File index.php contains the initial html and PHP code. It contains several important tags like <div>, <body>, <html> etc.
 - File README.md contains the information about the entire project.
 - File index.css under the 'styles' directory is the main stylesheet file.
 - 'Assets' directory is the main directory which contains all the resources that are used.
-

5.2 Technologies Used

5.2.1 HTML (Hyper Text Markup Language)

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. HTML tags are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, links, quotes and other items. Tags such as `` and `<input/>` directly introduce content into the page. Other tags such as `<p>` surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

5.2.2 CSS (Cascading Style Sheets)

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts.

5.2.3 JavaScript

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.

5.2.4 PHP

PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for web development, and also used as a general-purpose programming language. PHP code

may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable.

5.3 Code used at client and server sides

5.3.1 Code at client side:

1. Login form

```
<form id="login-form" method="POST" action="/php/login.php">
  <label for="inputEmail" class="sr-only">Username</label>
  <input required type="text" id="username" name="username" class="form-
    control" placeholder="Username">
  <label for="inputPassword" class="sr-only">Password</label>
  <input required type="password" id="password" name="password"
    class="form-control" placeholder="Password">
  <input class="btn btn-lg btn-primary btn-block" type="submit" value="Sign
In"></form>
```

This is the HTML5 login form code that sends data to the login.php server-side code, with HTML5 form validation using the 'required' keyword.

2. To import the Bootstrap framework:

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap.min.js"
crossorigin="anonymous"></script>
```

This code is used to import the Bootstrap framework, jQuery, and Popper.js.

5.3.2 Functions at server side:

1. Database connection

```
$conn = new mysqli ($dbhost, $dbuser, $dbpass, $db);  
if($conn->connect_error)  
die ("connection failed". $conn->connect_error);
```

The above code is used to connect to the database and obtain connection object.

2. Inserting message records into the database

```
<?php  
include 'connect.php';  
$db = dbconnect ();  
$username = $_POST['username'];  
$message = $_POST['message'];  
$channel = 'messages_general';  
$sql = "INSERT INTO messages_general (username, message_text) VALUES  
('$username','$message')";  
$res = mysqli_query ($db, $sql);  
include 'homepage.php';  
?>
```

The above code is used to insert a new message chat record into the database. On success it redirects to the homepage of the chat application.

CHAPTER 6

TESTING

6.1 Testing Techniques

Software testing techniques help you design better cases. Since exhaustive testing is not possible, testing techniques help reduce the number of test cases to be executed while increasing test coverage. They help identify test conditions that are otherwise difficult to recognize.

6.2 Software Testing Fundamentals

During earlier definition and development of software quality assurance and represents the ultimate review of specification, design and coding. The increasing visibility of software as a system element and attendant “costs” associated with a software failure is a motivating force for well planned, thorough testing.

6.3 Testing Objective

Several rules that can serve well as testing objectives are:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probabilities of finding an undiscovered error. A successful test is one that uncovers an undiscovered error.
- The objective is to design a test systematically to uncover different classes of errors and do so with minimum amount of time and effort.
- Once testing is done, we need to validate the web application with an actual user base so that we can be sure that the application works properly in real environments.

Testing cannot show the absence of defects, it can only show the software defects that are present.

6.4 Test Cases

SI # Test Case:	1
Name of the Page	Index
Expected Output	Name-ABC
Actual Output	Field Name Missing
Status	Fail
Remarks	User Has Not Filled the Information

Table 6.1: Test Case 1

SI # Test Case:	2
Name of the Page	Index
Expected Output	User Has Been Created Successfully
Actual Output	New User Has Not Been Created
Status	Fail
Remarks	User Has Given Wrong Data

Table 6.2: Test Case 2

SI # Test Case:	3
Name of the Page	Index
Expected Output	Name-ABC
Actual Output	Name-ABC
Status	Pass
Remarks	User Has Filled the Information

Table 6.3: Test Case 3

CHAPTER 7

SNAPSHOTS

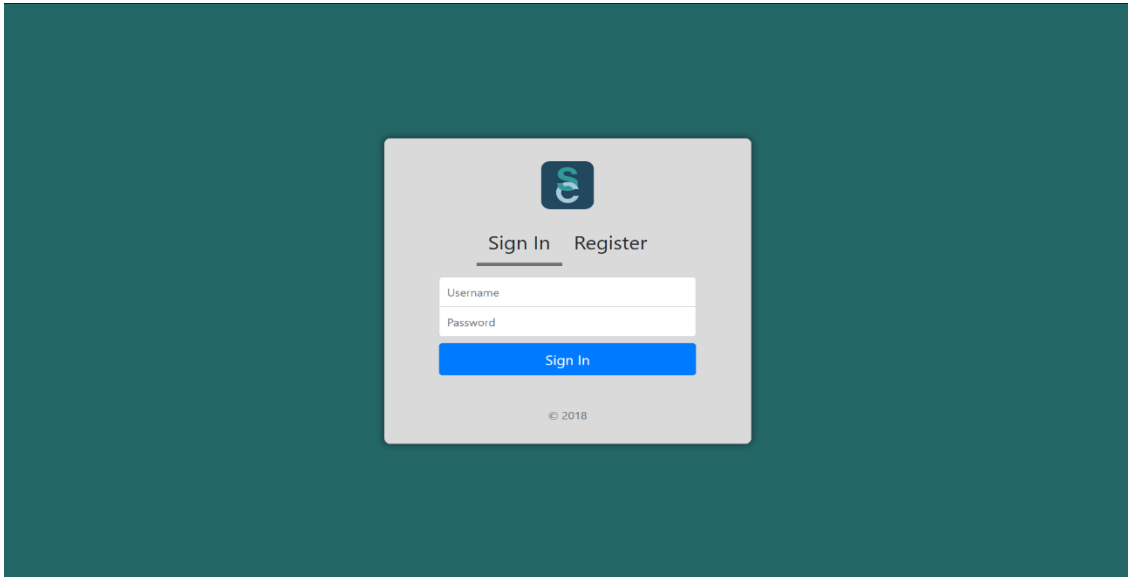


Fig 7.1: Sign in Section

This page is the landing page of the web page which prompts the user to sign in.

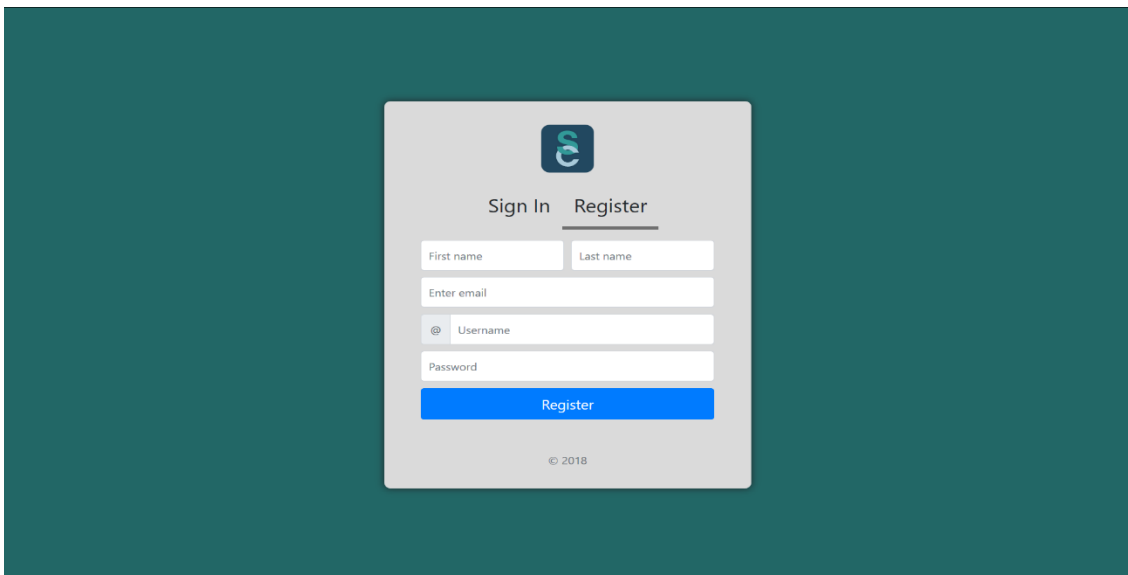


Fig 7.2: Register Section

This section lets a user register an account on the chat web application.

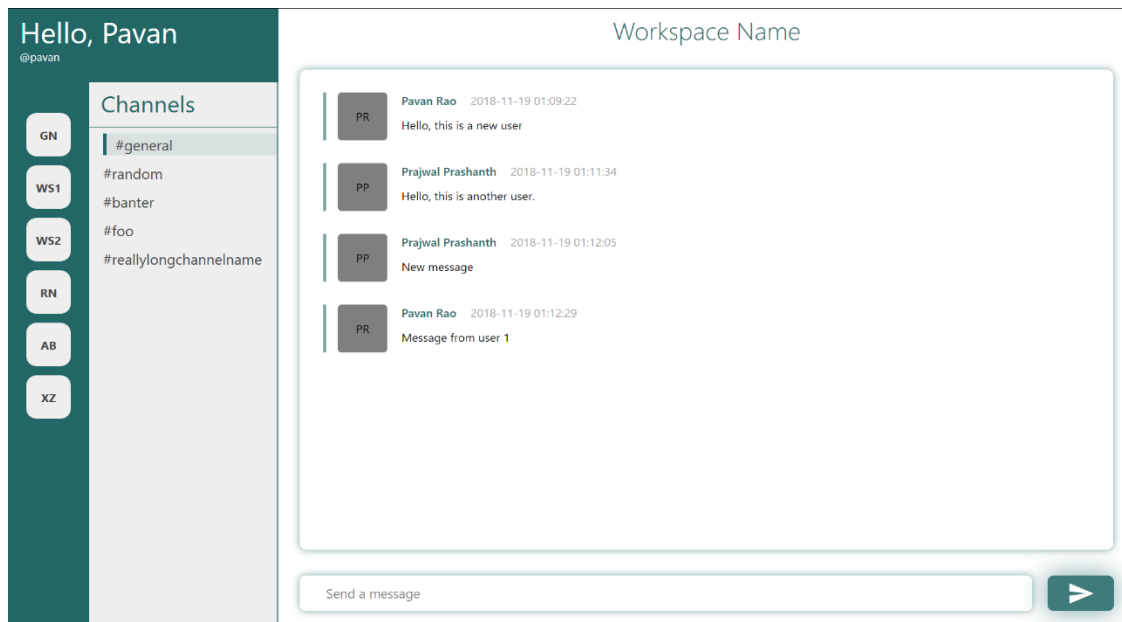


Fig 7.3: The Homepage

This page is the main homepage of the web application where the users can chat.

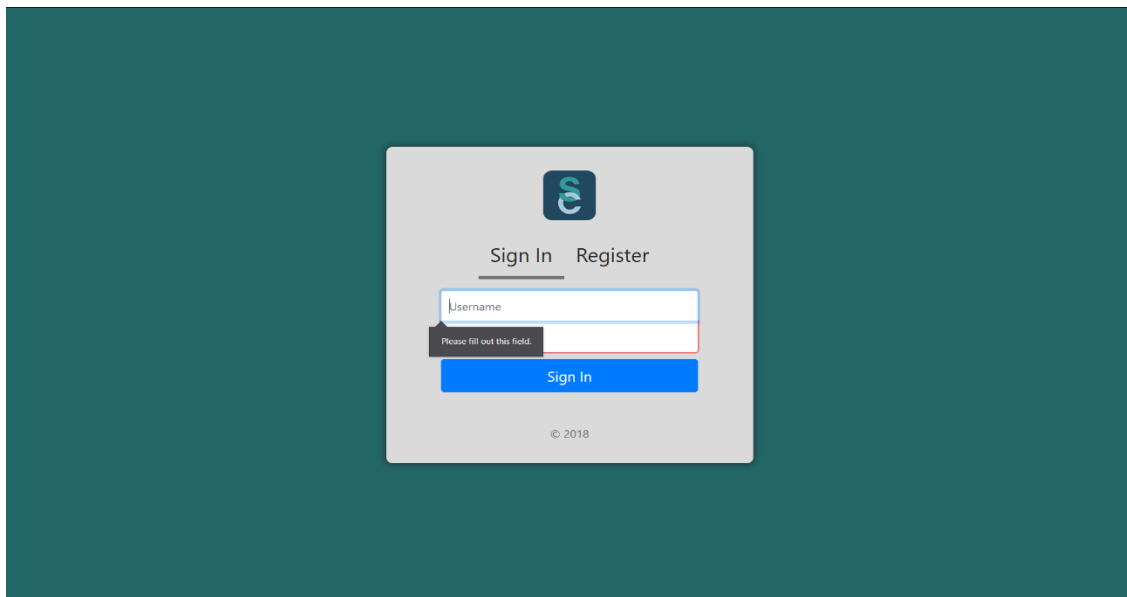


Fig 7.4: Sign in with validation

This page shows the form validation when a user tries to sign in without providing the required details to sign in to the website.

CONCLUSION

This project is to develop a simple chat application with the help of web technologies like HTML, CSS, JavaScript and PHP.

Web technology plays a major role in today's world where visualization and interaction take the upper hand when compared to textual interaction. This is largely true as we can see user interfaces becoming more and more attractive, all thanks to major leaps in the field of web development and design. The project is implemented using the XAMPP, a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. It is clear to see how web technology reduces costs and makes a company more efficient, raising business potential and increasing customer satisfaction.

FUTURE ENHANCEMENTS

- The look and feel of the website can be enhanced with media queries and optimized for viewing on a mobile device.
 - Advanced properties of HTML and CSS can be used to make the website to work much more smoothly with regards to the logging in and sending of messages.
 - Asynchronous requests to the back end will enable the messages to appear as soon as they are sent, which makes for a more optimized user experience along with faster response times.
 - Addition of multiple workspaces will enable users to create their own organizations' group.
 - Making each message as a thread so the any reply to that message specifically can be in a separate enclosure.
 - The ability to send media such as images and documents will greatly enhance the functionality of this web application.
-

REFERENCES

We have referred the following resources in order to complete our mini project:

- Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 1st Edition, Pearson Education India. (ISBN:978-9332575271)
 - <https://en.wikipedia.org>
 - <https://www.w3schools.com>
 - <https://www.stackoverflow.com>
-