

Heriot Watt University

School of Mathematical & Computer Science

Network Applications

F21NA

Coursework – Stage 2

6th April 2016

**Submitted By**

Anton Tolken - H00232218

Beni Iyaka – H00181266

Ritu John – H00232500

Sabiha Nuzhat – H00232316

Contents

[Introduction 2](#_Toc448160124)

[Application 2](#_Toc448160125)

[Front-end 2](#_Toc448160126)

[Server 6](#_Toc448160127)

[Server.php 6](#_Toc448160128)

[User.php 7](#_Toc448160129)

[Database 7](#_Toc448160130)

[Security 8](#_Toc448160131)

[Test Cases 8](#_Toc448160132)

[Front-end Validations 8](#_Toc448160133)

[Validating user input for registration 8](#_Toc448160134)

[Login validation 9](#_Toc448160135)

[Other Test Cases 10](#_Toc448160136)

[Technology Used 10](#_Toc448160137)

[Web Sockets 10](#_Toc448160138)

[Custom Chat Server 10](#_Toc448160139)

[Database Choice 11](#_Toc448160140)

[Security Issues 11](#_Toc448160141)

[Code Listing 11](#_Toc448160142)

# Introduction

The objective of this coursework is to create a synchronous chat service which is a three tier web based application. We have named it as ‘Wise Web Chat’.The client side is realised using Extended Hyper Text Transfer Protocol (XHTML) with Cascading Style Sheets (CSS) and the server side is a custom service built using PHP. The chat service is implemented using sockets on the web. This is achieved using Javascript. Using this chat, a user can:

* Authenticate themselves and log in to the application
* Browse a list of available channels
* Join a channel or create a new one
* Chat on the channel in real time

The course work has been divided between four members equally. Web sockets, server in PHP, and the javascript files have been done by Ben and Anton. The front end of the application, the database and the connection between the database and the chat application for login and registration along with validation has been jointly done by Sabiha and Ritu. The CSS and HTML were coordinated among all the group members.

The strength of the application, i.e., the web sockets in particular, have been further elaborated in Technology Used section.

The application can be furthered improved to monitor the use of foul language in the chat and to host the application on the internet.

# Application

## Front-end

The front end is coded using XHTML and CSS. The application has the following screens:

1. Overview Screen – This page gives the overview of the chat application and the details of the developers. CSS3 features were used in the application to give new features like onmouseover and onmouseout which shows a highlighted text as shown below and marquees show a rolling text below the title of the page.

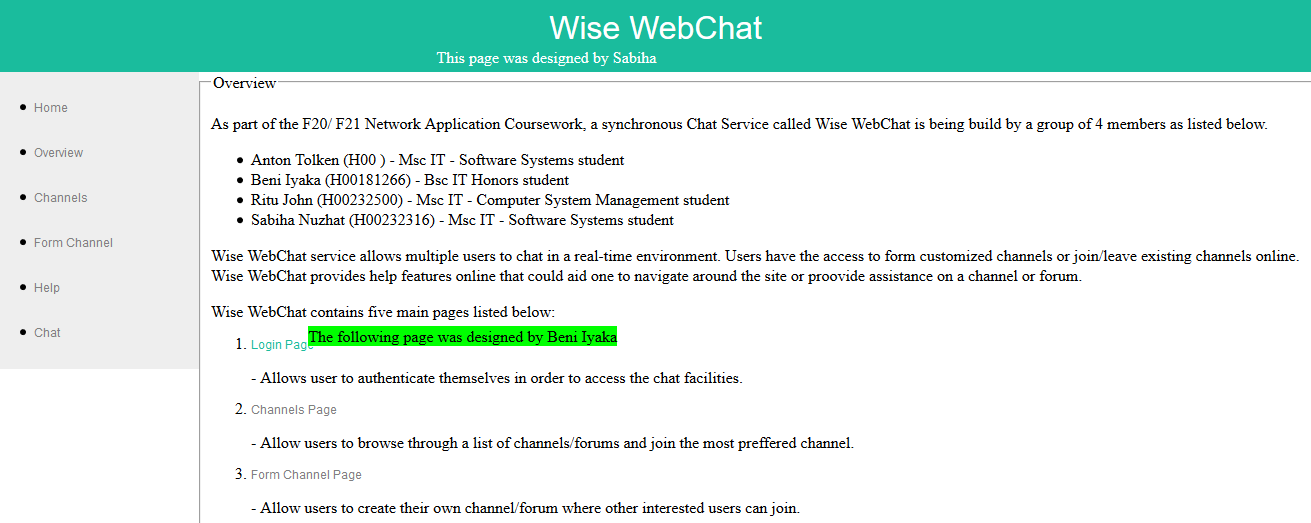


Figure : Wise WebChat overview page

1. Home page – This interface is what the user will see first when visiting this chat application. An existing member of the application can login with his username and password, whereas a new user can register. Figure 3 shows the interface for registration.

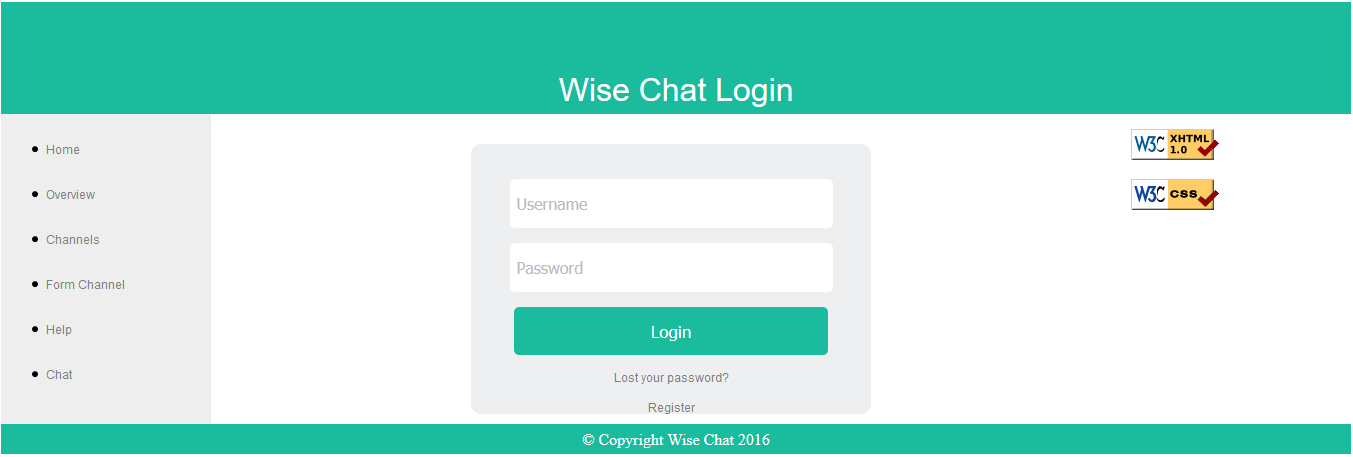


Figure : Wise WebChat Login Page

The user enters his details on this registration screen. It also has a security check to ensure it is genuine. The register button is enabled only when the answer is verified.

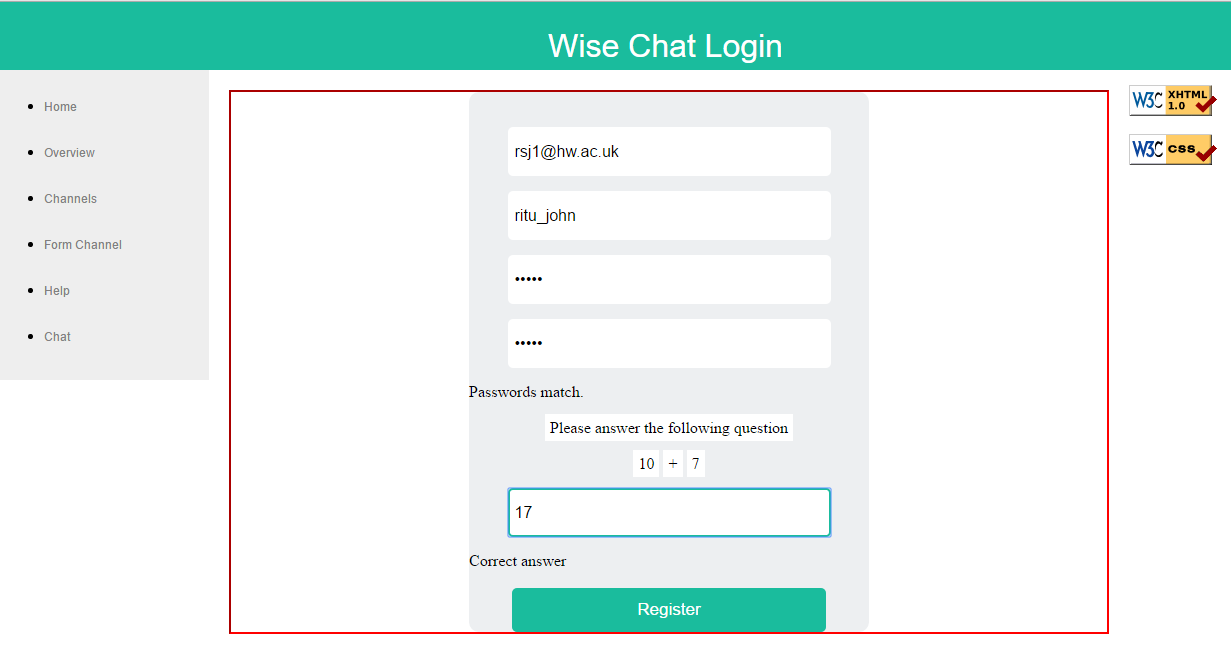


Figure : Wise WebChat Login Page

1. Channels – This screen shows the available channels to which a user can join. A button to create a new channel is also provided.

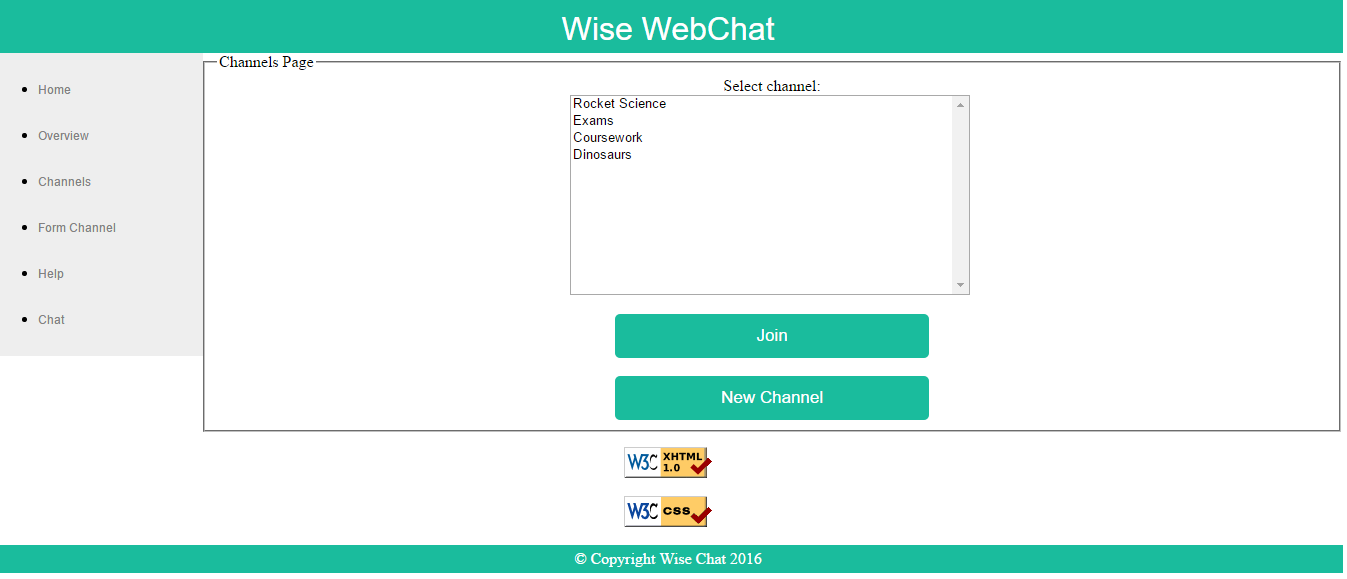


Figure : Wise WebChat Channels Page

1. Form Channel – The user can create a new channel using form channel screen.

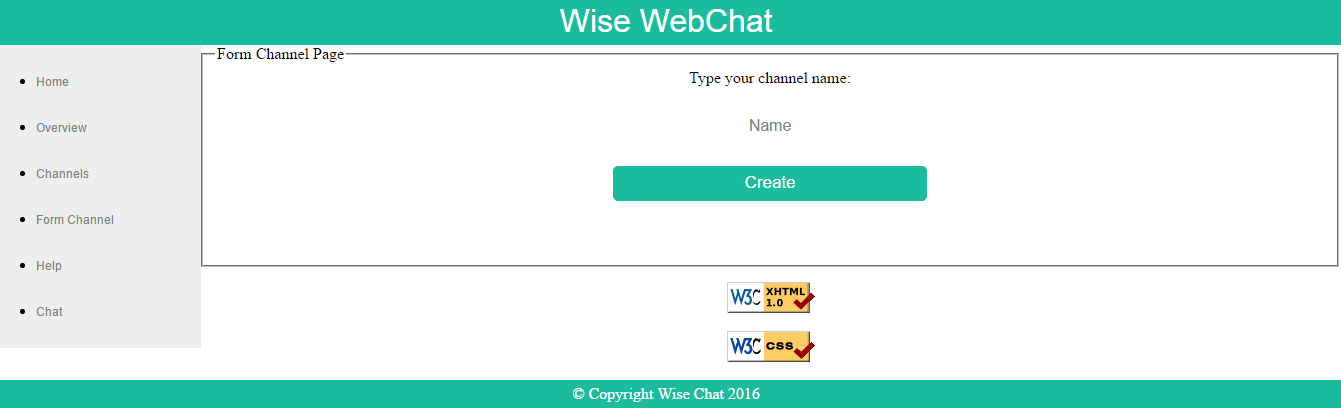


Figure : Wise WebChat Form Channels Page

1. Chat – The chat allows the user to see the other online users in the channel and chat with them in real time. It also verifies the connection to the server and displays whether the user is connected or not. It also has a app navigation and logout button.

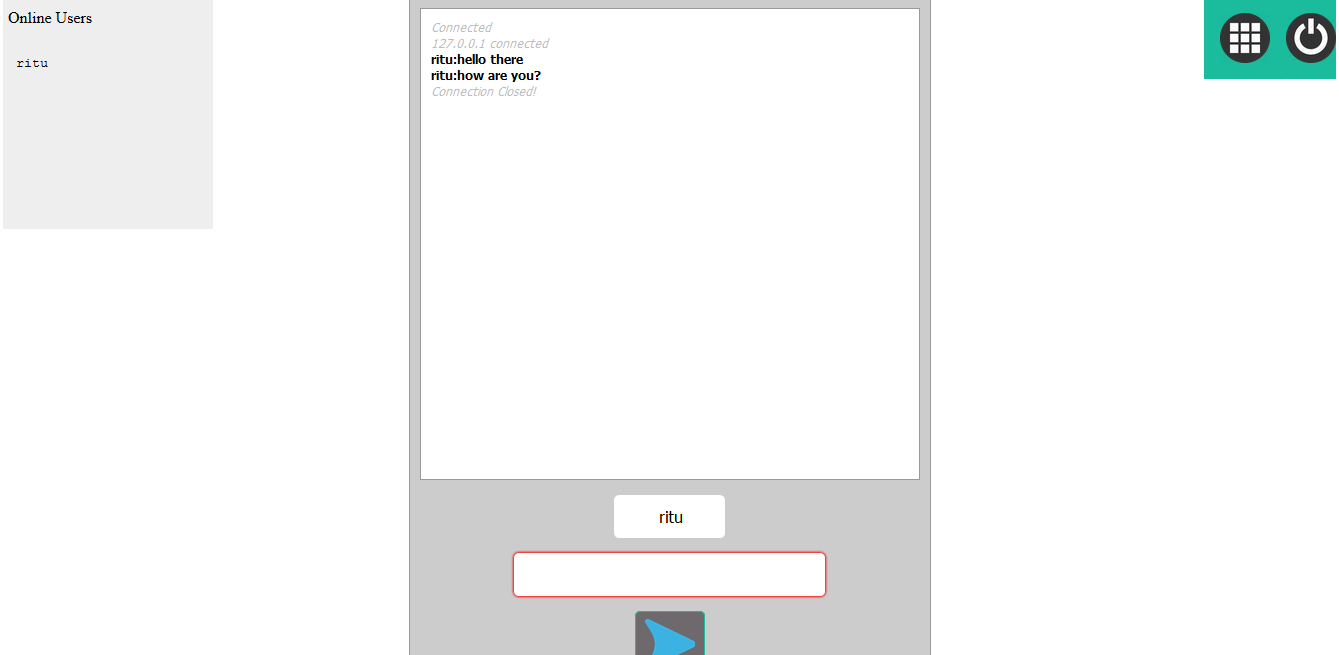


Figure : Chat page

1. Help – This page gives tips to the user to get started and provides answers to frequently asked questions.

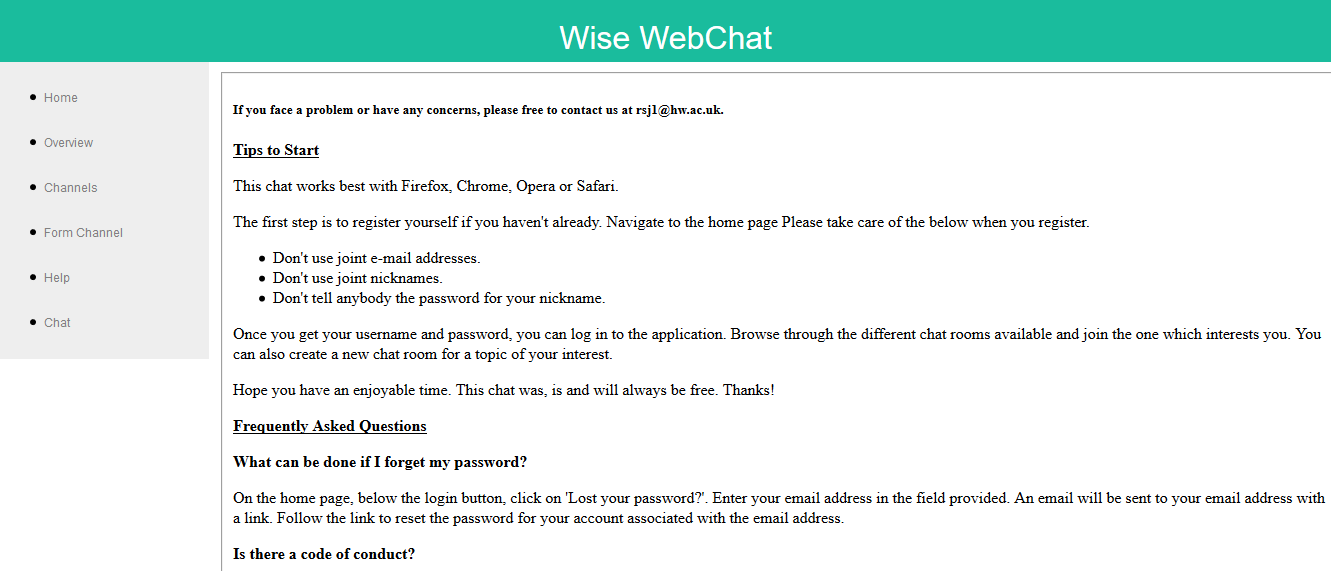
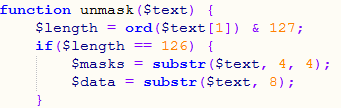


Figure : Wise WebChat Help Page

## Server

### Server.php

The back-end server is custom made in PHP. It uses web sockets to enable real-time two way communication over TCP/IP. Unlike a TCP connection, the messages are transmitted as text using web sockets and not bytes.

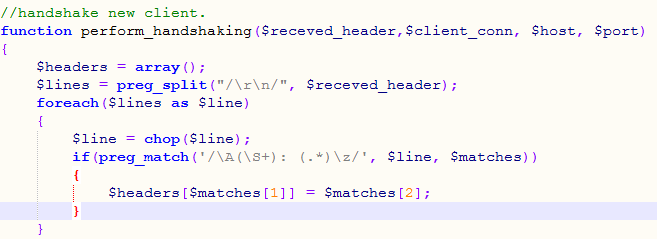


Once the socket connection is made, the user is notified by the server.





The connection between the clients is established with the handshake protocol as shown below.



### User.php

This file contains the backend procedures for tasks such as user registration, login, logout and form channels.

The details of the user entered on the registration screen are captured and entered into the MySQL database.

During login, it verifies whether the username and password exists in the database. Once logged in, it enters the username and channel to the current online users’ entity.

On logout, the username is deleted from the current online users’ table.

Moreover, it also adds the new channels created to the database.

Sessions are created to remember the username of the user.

## Database

The database is used is MySQL. It contains three tables – chat, online and channels. The table chat contains details of the registered users on the Wise WebChat with fields such as email, username and password.

The online table includes the username and channel of the logged in users.

It also contains a table named channels with the list of all channels created.

# Test Cases

## Validating user input for registration

If the user enters an invalid combination of passwords during registration, an error message is showed saying ‘Passwords do not match!’ In case, the form is submitted without entering all the mandatory details, the action won’t take place as part of validation. The security check will also prevent the user from registering in case the automated question hasn’t been answered correctly. The screenshot below shows the combination of these checks:

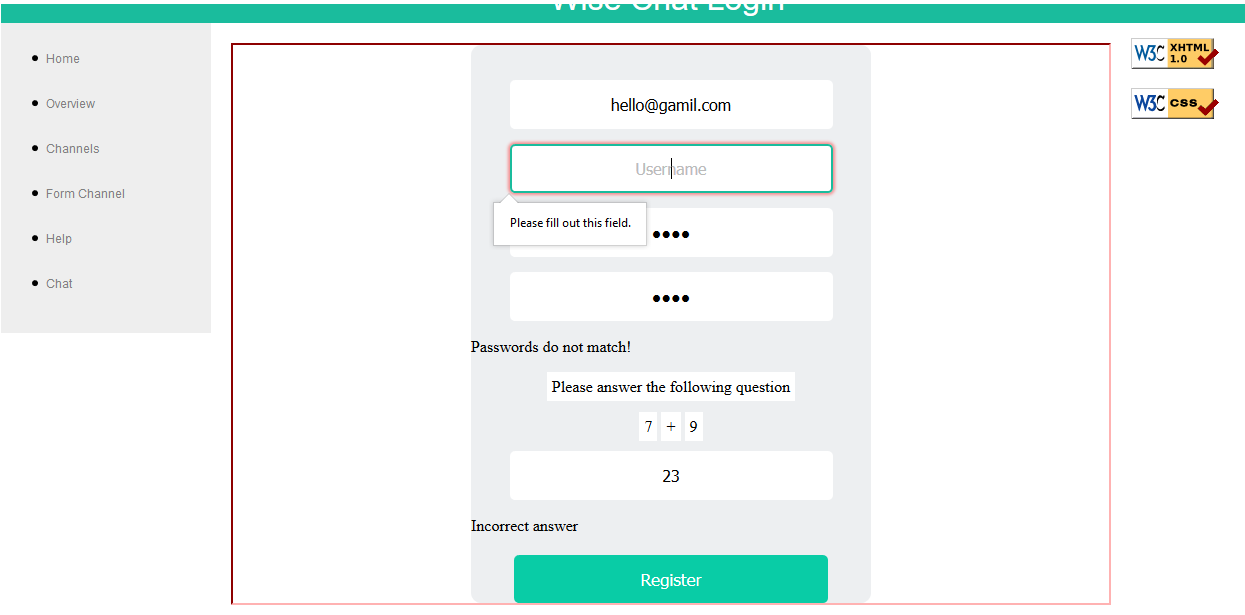


Figure : User Registration validation test case

## Login validation

If the user tries logging in with incorrect username or password that doesn’t match with the details saved on the database, an error messages pops up as shown below:

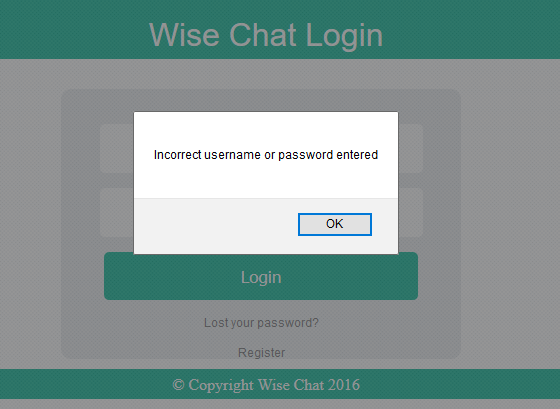


Figure : User Login Test Case

# Technology Used

## Web Sockets

Web sockets is an internet application protocol. Use of these can be considered as a hybrid approach of web and non-web technology. With Hyper Text Transfer Protocol (HTTP), it is not possible to have a real time chatting service due to the client server model. Thus, by using sockets, the use of HTTP is bypassed and uses peer to peer (P2P) mode of communication. It enables continuous two way communication over a TCP connection. However, it is neither a TCP nor a HTTP connection. The interface between client and server is realised using Javascript and that between server and socket connection using a server program.

Implementing socket in Javascript:

* Create socket using constructor
* Access socket through readyState attribute
* Control with event handlers like onopen, onmessage, onerror, onclose

Among other technologies like short and long polling, HTTP streaming and applets for web chat, web sockets is the chosen technology to implement ‘Wise Web Chat’ since this is supported by all browsers and is the latest in technology. Short polling is not apt for real time communication and there are implementation challenges for long polling. HTTP Streaming is no longer supported by Chrome or Internet Explorer and Applets have considerable security issue. However, web sockets also pose security issues but in spite of these issues, it is most apt for our application.

## Custom Chat Server

The server is developed using PHP rather than other programming languages like Javascript, C#, etc. for the following reasons:

* Easy deployment
* Optimised for making web application quickly
* Can support several databases such as MySQL, Oracle, Postgre SQL, etc.
* PHP can be easily embedded into HTML code.

## Database Choice

The initial idea was to use PHP with MongoDB. A few different attempts were to made to configure PHP to use MongoDB, including Docker and manual configuration. However none of these worked well enough, and thus we switched to MySQL. An online MySQL was chosen due to its popularity and ease-of-use with PHP. It is accessible from anywhere from a web browser. Moreover, it is multi-tenant in nature, i.e., multiple customers can have a shred model to use it.

## Short Polling

Short polling is to have a constant update of people that are currently joining the chat room. A request gets sent to the server every 2 second for updates of the online users and in this way, it shows the usernames of the current users in the chat rooms.

# Security

The application contains security mechanisms to support a moderate level of security.

* It contains user authentication and only registered user are allowed to login to the application.
* The passwords in the database are stored after encryption.
* The registration process contains an extra check to ensure that the users registering are genuine and not scripts used by spammers.

## Security Issues

Ideally we would have preferred a single PHP application, with a local MySQL database, but due to difficulties with configuring security related settings like the Windows firewall, we chose to use PHP on both laptops, and an online MySQL database.

# Code Listing

## PHP Files:

### Online.php



### Server.php

<?php

$host = 'localhost'; //host

$port = '443'; //port

$null = NULL; //null var

//Create TCP/IP sream socket

$socket = socket\_create(AF\_INET, SOCK\_STREAM, SOL\_TCP);

//reuseable port

socket\_set\_option($socket, SOL\_SOCKET, SO\_REUSEADDR, 1);

//bind socket to specified host

socket\_bind($socket, 0, $port);

//listen to port

socket\_listen($socket);

//create & add listning socket to the list

$clients = array($socket);

//start endless loop, so that our script doesn't stop

while (true) {

//manage multipal connections

$changed = $clients;

//returns the socket resources in $changed array

socket\_select($changed, $null, $null, 0, 10);

//check for new socket

if (in\_array($socket, $changed)) {

$socket\_new = socket\_accept($socket); //accpet new socket

$clients[] = $socket\_new; //add socket to client array

$header = socket\_read($socket\_new, 1024); //read data sent by the socket

perform\_handshaking($header, $socket\_new, $host, $port); //perform websocket handshake

socket\_getpeername($socket\_new, $ip); //get ip address of connected socket

$response = mask(json\_encode(array('type'=>'system', 'message'=>$ip.' connected'))); //prepare json data

send\_message($response); //notify all users about new connection

//make room for new socket

$found\_socket = array\_search($socket, $changed);

unset($changed[$found\_socket]);

}

//loop through all connected sockets

foreach ($changed as $changed\_socket) {

//check for any incomming data

while(socket\_recv($changed\_socket, $buf, 1024, 0) >= 1)

{

$received\_text = unmask($buf); //unmask data

$tst\_msg = json\_decode($received\_text); //json decode

$user\_name = $tst\_msg->name; //sender name

$user\_message = $tst\_msg->message; //message text

$user\_color = $tst\_msg->color; //color

//prepare data to be sent to client

$response\_text = mask(json\_encode(array('type'=>'usermsg', 'name'=>$user\_name, 'message'=>$user\_message, 'color'=>$user\_color)));

send\_message($response\_text); //send data

break 2; //exist this loop

}

$buf = @socket\_read($changed\_socket, 1024, PHP\_NORMAL\_READ);

if ($buf === false) { // check disconnected client

// remove client for $clients array

$found\_socket = array\_search($changed\_socket, $clients);

socket\_getpeername($changed\_socket, $ip);

unset($clients[$found\_socket]);

//notify all users about disconnected connection

$response = mask(json\_encode(array('type'=>'system', 'message'=>$ip.' disconnected')));

send\_message($response);

}

}

}

// close the listening socket

socket\_close($socket);

function send\_message($msg)

{

global $clients;

foreach($clients as $changed\_socket)

{

@socket\_write($changed\_socket,$msg,strlen($msg));

}

return true;

}

//Unmask incoming framed message

function unmask($text) {

$length = ord($text[1]) & 127;

if($length == 126) {

$masks = substr($text, 4, 4);

$data = substr($text, 8);

}

elseif($length == 127) {

$masks = substr($text, 10, 4);

$data = substr($text, 14);

}

else {

$masks = substr($text, 2, 4);

$data = substr($text, 6);

}

$text = "";

for ($i = 0; $i < strlen($data); ++$i) {

$text .= $data[$i] ^ $masks[$i%4];

}

return $text;

}

//Encode message for transfer to client.

function mask($text)

{

$b1 = 0x80 | (0x1 & 0x0f);

$length = strlen($text);

if($length <= 125)

$header = pack('CC', $b1, $length);

elseif($length > 125 && $length < 65536)

$header = pack('CCn', $b1, 126, $length);

elseif($length >= 65536)

$header = pack('CCNN', $b1, 127, $length);

return $header.$text;

}

//handshake new client.

function perform\_handshaking($receved\_header,$client\_conn, $host, $port)

{

$headers = array();

$lines = preg\_split("/\r\n/", $receved\_header);

foreach($lines as $line)

{

$line = chop($line);

if(preg\_match('/\A(\S+): (.\*)\z/', $line, $matches))

{

$headers[$matches[1]] = $matches[2];

}

}

$secKey = $headers['Sec-WebSocket-Key'];

$secAccept = base64\_encode(pack('H\*', sha1($secKey . '258EAFA5-E914-47DA-95CA-C5AB0DC85B11')));

//hand shaking header

$upgrade = "HTTP/1.1 101 Web Socket Protocol Handshake\r\n" .

"Upgrade: websocket\r\n" .

"Connection: Upgrade\r\n" .

"WebSocket-Origin: $host\r\n" .

"WebSocket-Location: ws://$host:$port/demo/shout.php\r\n".

"Sec-WebSocket-Accept:$secAccept\r\n\r\n";

socket\_write($client\_conn,$upgrade,strlen($upgrade));

}

### User.php

<?php

session\_start();

$db\_host = 'db4free.net';

$db\_user = 'wonderben';

$db\_pass = 'Wonder12';

$db\_database = 'mylab1';

$link = mysqli\_connect($db\_host,$db\_user,$db\_pass,$db\_database) or die('Unable to establish a DB connection');

mysqli\_query($link, "SET names UTF8");

if(isset($\_POST['register'])){

$user=$\_POST['username'];

$password = md5($\_POST['pass']);

$email = $\_POST['email'];

if( $link -> query("INSERT INTO chat(username,pswrd,email) VALUES('$user', '$password','$email')")){

echo("<script>alert('Registration successful. Thank you for using Wise Chat')</script>");

}

}

if (isset($\_POST['submit'])){

$user=$\_POST['userid'];

$password = md5($\_POST['pswrd']);

if ($link->connect\_errno) {

echo "Failed to connect to MySQL: " . $mysqli->connect\_error;

}

$res = $link->query("SELECT \* FROM chat where username='$user' and pswrd='$password'");

$row = $res->fetch\_assoc();

if(isset($row['username']))

{

$\_SESSION['mysesi']=$row['username'];

echo "<script>window.location.assign('./channels.xhtml?username=$user')</script>";

}

else{

echo'<script>alert("This username or password not same with database."); location.assign("./index.xhtml");</script>';

}

$userexist=in\_array($user,$onlineusers\_file);

if ($userexist)$userindex=array\_search($user,$onlineusers\_file);

if($userexist==false){

$onlineusers\_file[]=$user;

saveUsers($onlineusers\_file);

echo "<script>window.location.assign('./chat.xhtml?username=$user')</script>";

exit();

}

if($userexist==true){

echo "userexist";

exit();

}

if($userexist==true){

array\_splice($onlineusers\_file,$userindex,1);

saveUsers($onlineusers\_file);

echo "signout";

exit();

}

if($userexist==false){

echo "usernotfound";

exit();

}

}

if (isset($\_POST['signout'])){

echo "<script>alert('You have Successfully logout')</script>";

echo "<script>window.location.assign('./index.xhtml')</script>";

}

if (isset($\_POST['join'])){

$user = $\_SESSION['mysesi'];

if( $link -> query("INSERT INTO online(user) VALUES('$user')")){

echo "<script>window.location.assign('./chat.xhtml?username=".$user."')</script>";

}

}

if (isset($\_POST['create'])){

$name = $\_POST['textinput'];

if( $link -> query("INSERT INTO channel(name) VALUES('$name')")){

echo("<script>alert('Channel created. Thank you for using Wise Chat')</script>");

echo "<script>window.location.assign('./channels.xhtml')</script>";

}

}

?>

### userC.php



### Class.PHPWebSocket.php

<!DOCTYPE html>

<html>

<head>

<meta charset='UTF-8' />

<style type="text/css">

.chat\_wrapper {

width: 500px;

margin-right: auto;

margin-left: auto;

background: #CCCCCC;

border: 1px solid #999999;

padding: 10px;

font: 12px 'lucida grande',tahoma,verdana,arial,sans-serif;

}

.chat\_wrapper .message\_box {

background: #FFFFFF;

height: 150px;

overflow: auto;

padding: 10px;

border: 1px solid #999999;

}

.chat\_wrapper .panel input{

padding: 2px 2px 2px 5px;

}

.system\_msg{color: #BDBDBD;font-style: italic;}

.user\_name{font-weight:bold;}

.user\_message{color: #88B6E0;}

</style>

</head>

<body>

<?php

$colours = array('007AFF','FF7000','FF7000','15E25F','CFC700','CFC700','CF1100','CF00BE','F00');

$user\_colour = array\_rand($colours);

?>

<script src="//ajax.googleapis.com/ajax/libs/jquery/2.0.0/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function(){

//create a new WebSocket object.

var wsUri = "ws://localhost:9000/server.php";

websocket = new WebSocket(wsUri);

websocket.onopen = function(ev) { // connection is open

$('#message\_box').append("<div class=\"system\_msg\">Connected!</div>"); //notify user

}

$('#send-btn').click(function(){ //use clicks message send button

var mymessage = $('#message').val(); //get message text

var myname = $('#name').val(); //get user name

if(myname == ""){ //empty name?

alert("Enter your Name please!");

return;

}

if(mymessage == ""){ //emtpy message?

alert("Enter Some message Please!");

return;

}

//prepare json data

var msg = {

message: mymessage,

name: myname,

color : '<?php echo $colours[$user\_colour]; ?>'

};

//convert and send data to server

websocket.send(JSON.stringify(msg));

});

//#### Message received from server?

websocket.onmessage = function(ev) {

var msg = JSON.parse(ev.data); //PHP sends Json data

var type = msg.type; //message type

var umsg = msg.message; //message text

var uname = msg.name; //user name

var ucolor = msg.color; //color

if(type == 'usermsg')

{

$('#message\_box').append("<div><span class=\"user\_name\" style=\"color:#"+ucolor+"\">"+uname+"</span> : <span class=\"user\_message\">"+umsg+"</span></div>");

}

if(type == 'system')

{

$('#message\_box').append("<div class=\"system\_msg\">"+umsg+"</div>");

}

$('#message').val(''); //reset text

};

websocket.onerror = function(ev){$('#message\_box').append("<div class=\"system\_error\">Error Occurred - "+ev.data+"</div>");};

websocket.onclose = function(ev){$('#message\_box').append("<div class=\"system\_msg\">Connection Closed</div>");};

});

</script>

<div class="chat\_wrapper">

<div class="message\_box" id="message\_box"></div>

<div class="panel">

<input type="text" name="name" id="name" placeholder="Your Name" maxlength="10" style="width:20%" />

<input type="text" name="message" id="message" placeholder="Message" maxlength="80" style="width:60%" />

<button id="send-btn">Send</button>

</div>

</div>

</body>

</html>

## JavaScript Files:

### Fancywebsocket.js

var FancyWebSocket = function(url)

{

var callbacks = {};

var ws\_url = url;

var conn;

this.bind = function(event\_name, callback){

callbacks[event\_name] = callbacks[event\_name] || [];

callbacks[event\_name].push(callback);

return this;// chainable

};

this.send = function(event\_name, event\_data){

this.conn.send( event\_data );

return this;

};

this.connect = function() {

if ( typeof(MozWebSocket) == 'function' )

this.conn = new MozWebSocket(url);

else

this.conn = new WebSocket(url);

// dispatch to the right handlers

this.conn.onmessage = function(evt){

dispatch('message', evt.data);

};

this.conn.onclose = function(){dispatch('close',null)}

this.conn.onopen = function(){dispatch('open',null)}

};

this.disconnect = function() {

this.conn.close();

};

var dispatch = function(event\_name, message){

var chain = callbacks[event\_name];

if(typeof chain == 'undefined') return; // no callbacks for this event

for(var i = 0; i < chain.length; i++){

chain[i]( message )

}

}

};

### Me.js

function reveal() {

document.getElementById("items").style.visibility = "visible";

}

function hide() {

document.getElementById("items").style.visibility = "hidden";

}

function reveal1() {

document.getElementById("items1").style.visibility = "visible";

}

function hide1() {

document.getElementById("items1").style.visibility = "hidden";

}

function reveal2() {

document.getElementById("items2").style.visibility = "visible";

}

function hide2() {

document.getElementById("items2").style.visibility = "hidden";

}

function reveal3() {

document.getElementById("items3").style.visibility = "visible";

}

function hide3() {

document.getElementById("items3").style.visibility = "hidden";

}

function reveal4() {

document.getElementById("items4").style.visibility = "visible";

}

function hide4() {

document.getElementById("items4").style.visibility = "hidden";

}

function be(){

$( "#push" ).css({

borderStyle: "inset",

cursor: "wait"

});

$( "#test" ).slideDown( 1000, function() {

$( this )

.css( "border", "2px red inset" )

.filter( ".middle" )

.css( "background", "yellow" )

.focus();

$( "#tes" ).css( "visibility", "hidden" );

});

}

var num1 = document.getElementById("num1");

var num2 = document.getElementById("num2");

num1.innerHTML = Math.floor(Math.random()\*10+1);

num2.innerHTML = Math.floor(Math.random()\*10+1);

function checkMath() {

var num1 = parseInt(document.getElementById("num1").innerHTML, 10);

var num2 = parseInt(document.getElementById("num2").innerHTML, 10);

var answer = parseInt(document.getElementById("answer").value, 10);

if (answer === num1 + num2) {

$("#check1").html("Correct answer");

} else {

$("#check1").html("Incorrect answer");

}

randomNum();

}

function be(){

$( "#push" ).css({

borderStyle: "inset",

cursor: "wait"

});

$( "#test" ).slideDown( 1000, function() {

$( this )

.css( "border", "2px red inset" )

.filter( ".middle" )

.css( "background", "yellow" )

.focus();

$( "#tes" ).css( "visibility", "hidden" );

});

$( "#tes" ).hide();

}

var num1 = document.getElementById("num1");

var num2 = document.getElementById("num2");

num1.innerHTML = Math.floor(Math.random()\*10+1);

num2.innerHTML = Math.floor(Math.random()\*10+1);

function checkMath() {

var num1 = parseInt(document.getElementById("num1").innerHTML, 10);

var num2 = parseInt(document.getElementById("num2").innerHTML, 10);

var answer = parseInt(document.getElementById("answer").value, 10);

if (answer === num1 + num2) {

$("#check1").html("Correct answer");

} else {

$("#check1").html("Incorrect answer");

}

randomNum();

}

function checkPasswordMatch() {

var password = $("#pass").val();

var confirmPassword = $("#conf").val();

if (password != confirmPassword)

$("#check").html("Passwords do not match!");

else

$("#check").html("Passwords match.");

}

$(document).ready(function () {

$("#conf").keyup(checkPasswordMatch);

});

function check(form){

var me = $("#userid").val();

/\*the following code checkes whether the entered userid and password are matching\*/

if( form.pswrd.value == "1234"){

window.open('server.php?username='+ me);

window.location.assign('chat.xhtml?username='+ me);

} else

{

alert("Error Password")/\*displays error message\*/

}

}

function checkPasswordMatch() {

var password = $("#pass").val();

var confirmPassword = $("#conf").val();

if (password != confirmPassword)

$("#check").html("Passwords do not match!");

else

$("#check").html("Passwords match.");

}

$(document).ready(function () {

$("#conf").keyup(checkPasswordMatch);

});

function check(form){

var me = $("#userid").val();

/\*the following code checkes whether the entered userid and password are matching\*/

if( form.pswrd.value == "1234"){

window.open('server.php?username='+ me);

window.location.assign('chat.xhtml?username='+ me);

} else

{

alert("Error Password")/\*displays error message\*/

}

}

## CSS Files:

### Style.css

.chat\_wrapper {

width: 500px;

margin-right: auto;

margin-left: auto;

background: #CCCCCC;

border: 1px solid #999999;

padding: 10px;

font: 12px 'lucida grande',tahoma,verdana,arial,sans-serif;

}

ul

{

list-style-position:outside;

}

h1{

color: #fff;

text-align: center;

font-family: Arial;

font-weight: normal;

margin: 2em auto 0px;

}

.outer-screen{

background: #13202c;

width: 900px;

height: 540px;

margin: 50px auto;

border-radius: 20px;

-moz-border-radius: 20px;

-webkit-border-radius: 20px;

position: relative;

padding-top: 35px;

}

.outer-screen:before{

content: "";

background: #3e4a53;

border-radius: 50px;

position: absolute;

bottom: 20px;

left: 0px;

right: 0px;

margin: auto;

z-index: 9999;

width: 50px;

height: 50px;

}

.outer-screen:after{

content: "";

background: #ecf0f1;

width: 900px;

height: 88px;

position: absolute;

bottom: 0px;

border-radius: 0px 0px 20px 20px;

-moz-border-radius: 0px 0px 20px 20px;

-webkit-border-radius: 0px 0px 20px 20px;

}

.stand{

position: relative;

}

.stand:before{

content: "";

position: absolute;

bottom: -150px;

border-bottom: 150px solid #bdc3c7;

border-left: 30px solid transparent;

border-right: 30px solid transparent;

width: 200px;

left: 0px;

right: 0px;

margin: auto;

}

.stand:after{

content: "";

position: absolute;

width: 260px;

left: 0px;

right: 0px;

margin: auto;

border-bottom: 30px solid #bdc3c7;

border-left: 30px solid transparent;

border-right: 30px solid transparent;

bottom: -180px;

box-shadow: 0px 4px 0px #7e7e7e

}

.inner-screen{

width: 800px;

height: 340px;

background: #1abc9d;

margin: 0px auto;

padding-top: 80px;

}

.form1{

width: 400px;

height: auto;

background: #edeff1;

margin: 0px auto;

padding-top: 20px;

border-radius: 10px;

-moz-border-radius: 10px;

-webkit-border-radius: 10px;

}

input[type="password"]:focus{

border: 2px solid #1abc9d

}

input[type="password"]{

display: block;

width: 309px;

height: 35px;

margin: 15px auto;

background: #fff;

border: 0px;

padding: 5px;

font-size: 16px;

border: 2px solid #fff;

transition: all 0.3s ease;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

}

span{

background: #fff;

padding: 5px;

font-size: 16px;

}

input[type="text"]{

display: block;

width: 309px;

height: 35px;

margin: 15px auto;

background: #fff;

border: 0px;

padding: 5px;

font-size: 16px;

border: 2px solid #fff;

transition: all 0.3s ease;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

}

input[type="text"]:focus{

border: 2px solid #1abc9d

}

input[type="email"]{

display: block;

width: 309px;

height: 35px;

margin: 15px auto;

background: #fff;

border: 0px;

padding: 5px;

font-size: 16px;

border: 2px solid #fff;

transition: all 0.3s ease;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

}

input[type="email"]:focus{

border: 2px solid #1abc9d

}

#send-btn1{

height: 50px;

width: 50px;

}

#send-btn{

height: 70px;

width: 70px;

background-color: #6d696c;

border: solid 1px #1abc9d;

border-radius: 5px

}

input[type="button"]{

display: block;

background: #1abc9d;

width: 314px;

padding: 12px;

cursor: pointer;

color: #fff;

border: 0px;

margin: auto;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

font-size: 17px;

transition: all 0.3s ease;

}

input[type="button"]:hover{

background: #09cca6

}

a, button{

font-family: Arial;

color: gray;

display: block;

margin: 15px auto;

text-decoration: none;

transition: all 0.3s ease;

font-size: 12px;

}

a:hover{

color: #1abc9d;

}

::-webkit-input-placeholder {

color: gray;

}

:-moz-placeholder { /\* Firefox 18- \*/

color: gray;

}

::-moz-placeholder { /\* Firefox 19+ \*/

color: gray;

}

:-ms-input-placeholder {

color: gray;

}

#list{

height: 400;

width: 95%;

float: right;

}

#name{

text-align: center;

}

#menu { position: relative }

#items { position: absolute; left: 25%; top: 63%;

background-color: lime; visibility: hidden; }

:link, :visited { text-decoration: none }

#items1 { position: absolute; left: 25%; top: 74%;

background-color: lime; visibility: hidden; }

:link, :visited { text-decoration: none }

#items2 { position: absolute; left: 25%; top: 84%;

background-color: lime; visibility: hidden; }

:link, :visited { text-decoration: none }

#items3 { position: absolute; left: 25%; top: 100%;

background-color: lime; visibility: hidden; }

:link, :visited { text-decoration: none }

#items4 { position: absolute; left: 25%; top: 110%;

background-color: lime; visibility: hidden; }

:link, :visited { text-decoration: none }

#marquee { animation: going 25s linear infinite; }

#marquee:hover { animation-play-state: paused; }

@keyframes going { from { margin-left: 20em; }

to { margin-left: -50em; } }

#user {

border: 1px dotted red;

padding: 10px;

}

#block {

font-weight:bold;

color: #88B6E0;

}

#header {

background-color:#1abc9d;

color:white;

text-align:center;

padding:5px;

}

#t{

border: none;

}

.dropbtn{

height: 50px;

background-color: #1abc9d;

color: white;

padding: 16px;

font-size: 16px;

border: none;

cursor: pointer;

float: right;

}

div#form input{

height: 82px;

width: 50px;

background-image: url("./logout.png");

background-size: 50px;

background-repeat: no-repeat;

background-position: center;

color: white;

padding: 16px;

font-size: 16px;

border-radius: 1%;

cursor: pointer;

float: right;

}

.dropbtn:hover, .dropbtn:focus {

background-color: #1abc9d;

}

.dropdown{

position: relative;

display: inline-block;

float: right;

}

.dropdown-content {

display: none;

top: 80%;

position: absolute;

background-color: #f9f9f9;

min-width: 160px;

overflow: auto;

box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);

}

.dropdown-content a {

color: black;

padding: 12px 16px;

text-decoration: none;

display: block;

}

.dropdown a:hover {background-color: #f1f1f1}

.show {display:block;}

#nav {

line-height:30px;

background-color:#eeeeee;

height:auto;

width:200px;

float:left;

padding:5px;

}

#section {

width:250px;

float:left;

padding:10px;

margin-left:auto;

margin-right:200px;

}

#footer {

background-color:#1abc9d;

color:white;

clear:both;

text-align:center;

padding:5px;

}

.chat\_wrapper .message\_box {

background: #FFFFFF;

height: 450px;

overflow: auto;

padding: 10px;

border: 1px solid #999999;

}

.chat\_wrapper .panel input{

padding: 2px 2px 2px 5px;

}

.system\_msg{color: #BDBDBD;font-style: italic;}

.user\_name{font-weight:bold;}

.user\_message{color: #88B6E0;}#test {

display: none;

margin: 10px;

}input, textarea {border:1px solid #CCC;margin:0px;padding:0px}

#body {max-width:800px;margin:auto}

#log {width:100%;height:400px}

#message {width:100%;line-height:20px}

h1{

color: #fff;

text-align: center;

font-family: Arial;

font-weight: normal;

margin: 2em auto 0px;

}

.outer-screen{

background: #13202c;

width: 900px;

height: 540px;

margin: 50px auto;

border-radius: 20px;

-moz-border-radius: 20px;

-webkit-border-radius: 20px;

position: relative;

padding-top: 35px;

}

.outer-screen:before{

content: "";

background: #3e4a53;

border-radius: 50px;

position: absolute;

bottom: 20px;

left: 0px;

right: 0px;

margin: auto;

z-index: 9999;

width: 50px;

height: 50px;

}

.outer-screen:after{

content: "";

background: #ecf0f1;

width: 900px;

height: 88px;

position: absolute;

bottom: 0px;

border-radius: 0px 0px 20px 20px;

-moz-border-radius: 0px 0px 20px 20px;

-webkit-border-radius: 0px 0px 20px 20px;

}

.stand{

position: relative;

}

.stand:before{

content: "";

position: absolute;

bottom: -150px;

border-bottom: 150px solid #bdc3c7;

border-left: 30px solid transparent;

border-right: 30px solid transparent;

width: 200px;

left: 0px;

right: 0px;

margin: auto;

}

.stand:after{

content: "";

position: absolute;

width: 260px;

left: 0px;

right: 0px;

margin: auto;

border-bottom: 30px solid #bdc3c7;

border-left: 30px solid transparent;

border-right: 30px solid transparent;

bottom: -180px;

box-shadow: 0px 4px 0px #7e7e7e

}

.inner-screen{

width: 800px;

height: 340px;

background: #1abc9d;

margin: 0px auto;

padding-top: 80px;

}

.form{

width: 400px;

height: 250px;

background: #edeff1;

margin: 0px auto;

padding-top: 20px;

border-radius: 10px;

-moz-border-radius: 10px;

-webkit-border-radius: 10px;

}

input[type="text"]{

display: block;

width: 309px;

height: 35px;

margin: 15px auto;

background: #fff;

border: 0px;

padding: 5px;

font-size: 16px;

border: 2px solid #fff;

transition: all 0.3s ease;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

}

input[type="text"]:focus{

border: 2px solid #1abc9d

}

input[type="submit"]{

display: block;

background: #1abc9d;

width: 314px;

padding: 12px;

cursor: pointer;

color: #fff;

border: 0px;

margin: auto;

border-radius: 5px;

-moz-border-radius: 5px;

-webkit-border-radius: 5px;

font-size: 17px;

transition: all 0.3s ease;

}

input[type="submit"]:hover{

background: #09cca6

}

a{

font-family: Arial;

color: gray;

display: block;

margin: 15px auto;

text-decoration: none;

transition: all 0.3s ease;

font-size: 12px;

}

a:hover{

color: #1abc9d;

}

::-webkit-input-placeholder {

color: gray;

}

:-moz-placeholder { /\* Firefox 18- \*/

color: gray;

}

::-moz-placeholder { /\* Firefox 19+ \*/

color: gray;

}

:-ms-input-placeholder {

color: gray;

}

## HTML Files:

### Index.xhtml

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" lang="en">

<head>

<meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript"/>

<title>Wise Live chat</title>

<script src="https://code.jquery.com/jquery-1.10.2.js" type="text/javascript"></script>

<script src="/JS/me.js" type="text/javascript"></script>

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

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

<style type="text/css">

input{

text-align: center;

}

body{

background-color: white;

}

#lost, #eis{

text-align: center;

}

#header {

background-color:#1abc9d;

color:white;

text-align:center;

padding:5px;

}

#nav {

line-height:30px;

background-color:#eeeeee;

height:300px;

width:200px;

float:left;

padding:5px;

}

#section {

width:900px;

margin-left:auto;

margin-right:auto;

}

#footer {

background-color:#1abc9d;

color:white;

clear:both;

text-align:center;

padding:5px;

}

#push{

text-align: center;

}

</style>

</head>

<body>

<div id="header">

<h1>Wise Chat Login</h1>

</div>

<div id="nav">

<ul>

<li><a href="index.xhtml" class="btn">Home</a></li>

<li><a href="overview.xhtml" class="btn">Overview</a></li>

<li><a href="channels.xhtml" class="btn">Channels</a></li>

<li><a href="form\_channel.xhtml" class="btn">Form Channel</a></li>

<li><a href="help.xhtml" class="btn">Help</a></li>

<li><a href="chat.xhtml" class="btn">Chat</a></li>

</ul>

</div>

<div id="section">

<div id="test">

<form method="post" action="./user.php">

<div class="form1">

<input type="email" id="email" name="email" class="zocial-dribbble" placeholder="Email" required="required"/>

<input type="text" id="username" name="username" placeholder="Username" required="required"/>

<input type="password" id="pass" name="pass" placeholder="Password" required="required" />

<input type="password" id="conf" placeholder="Confirm Password" required="required"/>

<div id="check"></div><br/>

<div id="eis">

<span>Please answer the following question</span><br /><br />

<span id="num1"></span>

<span class="plus">+</span>

<span id="num2"></span>

</div>

<input type="text" id="answer" placeholder="Answer" required="required"/>

<div id="check1"></div><br/>

<input type="submit" value="Register" id="register" name="register"/>

</div>

</form>

</div>

<div id="tes">

<br/>

<form id="login" method="POST" action="user.php">

<div class="form">

<input name="userid" id="userid" type="text" class="zocial-dribbble" required="required" placeholder="Username"/>

<input name="pswrd" id="pswrd" type="password" required="required" placeholder="Password"/>

<input type="submit" name="submit" value="Login" id="submit" onclick="check(this.form)"/>

<a href="" id="lost">Lost your password?</a>

<a id="push" onclick=" be();">Register</a>

</div>

</form>

</div>

</div>

<div>

<a href="http://validator.w3.org/check?uri=referer"><img

src="http://www.w3.org/Icons/valid-xhtml10"

alt="Valid XHTML 1.0!" height="31" width="88" />

</a><a href="http://jigsaw.w3.org/css-validator/check/referer">

<img style="border:0;width:88px;height:31px"

src="http://jigsaw.w3.org/css-validator/images/vcss"

alt="Valid CSS!" />

</a>

</div>

<script type="text/javascript">

$('#conf').bind('input', function() {

var password = $("#pass").val();

var confirmPassword = $("#conf").val();

if (password != confirmPassword)

$("#check").html("Passwords do not match!");

else

$("#check").html("Passwords match.");

});

$('#answer').bind('input', function() {

var num1 = parseInt(document.getElementById("num1").innerHTML, 10);

var num2 = parseInt(document.getElementById("num2").innerHTML, 10);

var answer = parseInt(document.getElementById("answer").value, 10);

if (answer === num1 + num2) {

$("#check1").html("Correct answer");

} else {

$("#check1").html("Incorrect answer");

}

randomNum();

});

function be(){

$( "#push" ).css({

borderStyle: "inset",

cursor: "wait"

});

$( "#test" ).slideDown( 1000, function() {

$( this )

.css( "border", "2px red inset" )

.filter( ".middle" )

.css( "background", "yellow" )

.focus();

$( "#tes" ).css( "visibility", "hidden" );

});

$( "#tes" ).hide();

}

var num1 = document.getElementById("num1");

var num2 = document.getElementById("num2");

num1.innerHTML = Math.floor(Math.random()\*10+1);

num2.innerHTML = Math.floor(Math.random()\*10+1);

$(document).ready(function () {

$("#conf").keyup(checkPasswordMatch);

});

</script>

<div id="footer">

&#169; Copyright Wise Chat 2016

</div>

</body>

</html>

### Overview.xhtml

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml" lang="en">

<head>

<title>Overview Page</title>

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

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

<script type="text/javascript" src="js/me.js"/>

<style>

#nav{

float: left;

}

#section{

width: 100%;

}

</style>

</head>

<body>

<div id="section">

<div id="header">

<h1>Wise WebChat</h1>

<div style="margin-right: 30px; overflow: hidden">

<div id="marquee">This page was designed by Sabiha</div>

</div>

</div>

<div id="nav">

<ul>

<li><a href="index.xhtml" class="btn">Home</a></li>

<li><a href="overview.xhtml" class="btn">Overview</a></li>

<li><a href="channels.xhtml" class="btn">Channels</a></li>

<li><a href="form\_channel.xhtml" class="btn">Form Channel</a></li>

<li><a href="help.xhtml" class="btn">Help</a></li>

<li><a href="chat.xhtml" class="btn">Chat</a></li>

</ul>

</div>

<fieldset>

<legend>Overview</legend>

<p> As part of the F20/ F21 Network Application Coursework, a synchronous Chat Service called Wise WebChat is being build by

a group of 4 members as listed below.

</p>

<ul>

<li>Anton Tolken (H00 ) - Msc IT - Software Systems student</li>

<li>Beni Iyaka (H00181266) - Bsc IT Honors student</li>

<li>Ritu John (H00232500) - Msc IT - Computer System Management student</li>

<li>Sabiha Nuzhat (H00232316) - Msc IT - Software Systems student</li>

</ul>

<p>Wise WebChat service allows multiple users to chat in a real-time environment. Users have the access to

form customized channels or join/leave existing channels online. Wise WebChat provides help features online that could aid one to navigate

around the site or proovide assistance on a channel or forum.

</p>

<p>Wise WebChat contains five main pages listed below:</p>

<ol>

<li><a href="index.xhtml" class="btn" onmouseover="reveal()" onmouseout="hide()">Login Page</a> - Allows user to authenticate themselves in order to access the chat facilities.</li>

<li><a href="channels.xhtml" class="btn" onmouseover="reveal1()" onmouseout="hide1()">Channels Page</a> - Allow users to browse through a list of channels/forums and join the most preffered channel.</li>

<li><a href="form\_channel.xhtml" class="btn" onmouseover="reveal2()" onmouseout="hide2()">Form Channel Page</a> - Allow users to create their own channel/forum where other interested users can join.</li>

<li><a href="help.xhtml" class="btn" onmouseover="reveal3()" onmouseout="hide3()">Help Page</a> - Contains instructions on using and navigating through the website. Also contains a list of frequently asked questions.</li>

<li><a href="chat.xhtml" class="btn" onmouseover="reveal4()" onmouseout="hide4()">Chat Page</a> - Once the user selects their preffered channel to join, the Chat page is visible for the users to chat in real time.</li>

</ol>

<h3>HTML5 and CSS3 Features Used:</h3>

<ol>

<li>HTML5 brings in new features that were implemented on this website, such as </li>

<li>CSS3 brings new featiures that were implemeted on the website such as onmouseover, marquees and onmouseout </li>

<li> </li>

<li> </li>

</ol>

<h3>External Files Used</h3>

<ul>

<li><a href="http://validator.w3.org/check?uri=referer">

<img src="http://www.w3.org/Icons/valid-xhtml10" alt="Valid XHTML 1.0!" height="31" width="88" /></a></li>

<li><a href="http://jigsaw.w3.org/css-validator/check/referer">

<img src="http://jigsaw.w3.org/css-validator/images/vcss" style="border:0;width:88px;height:31px" alt="Valid CSS!" />

</a></li>

</ul>

</fieldset>

<div id="items">The following page was designed by Beni Iyaka</div>

<div id="items1">The following page was designed by Anthon</div>

<div id="items2">The following page was designed by Anthon</div>

<div id="items3">The following page was designed by Ritu John</div>

<div id="items4">The following page was designed by Beni Iyaka</div>

<div id="footer">

&#169; Copyright Wise Chat 2016

</div></div>

</body>

</html>

### Form\_channel.xhtml

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge"/>

<meta name="viewport" content="width=device-width, initial-scale=1"/>

<title>Form Channel Page</title>

<meta name="description" content="Source code generated using layoutit.com"/>

<meta name="author" content="LayoutIt!"/>

<style type="text/css">

input{

text-align: center;

display: block;

width: 309px;

height: 35px;

margin: 15px auto;

background: #fff;

border: 0px;

padding: 5px;

font-size: 16px;

border: 2px solid #fff;

}

</style>

<script type="text/javascript">

</script>

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

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

</head>

<body>

<div id="header">

<h1>Wise WebChat</h1>

</div>

<div id="nav">

<ul>

<li><a href="index.xhtml" class="btn">Home</a></li>

<li><a href="overview.xhtml" class="btn">Overview</a></li>

<li><a href="channels.xhtml" class="btn">Channels</a></li>

<li><a href="form\_channel.xhtml" class="btn">Form Channel</a></li>

<li><a href="help.xhtml" class="btn">Help</a></li>

<li><a href="chat.xhtml" class="btn">Chat</a></li>

</ul>

</div>

<form method="post" class="form-horizontal" action="./user.php">

<fieldset>

<legend>Form Channel Page</legend>

<div style="text-align:center;">

Type your channel name:<br/>

<input type="text" id="textinput" name="textinput" required="required" />

<input type="submit" id="create" name="create" value="Create" style="padding: 5px"/>

<br/><br/><br/>

</div>

</fieldset>

</form>

<div style="text-align:center;">

<a href="http://validator.w3.org/check?uri=referer"><img

src="http://www.w3.org/Icons/valid-xhtml10"

alt="Valid XHTML 1.0!" height="31" width="88" />

</a><a href="http://jigsaw.w3.org/css-validator/check/referer">

<img style="border:0;width:88px;height:31px"

src="http://jigsaw.w3.org/css-validator/images/vcss"

alt="Valid CSS!" />

</a>

</div>

<script type="text/javascript">

document.getElementById("textinput").required = true;

document.getElementById("textinput").placeholder = "Name";

</script>

<div id="footer">

&copy; Copyright Wise Chat 2016

</div>

</body>

</html>

### Chat.xhtml

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">

<head>

<title>Wise chat</title>

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

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

</head>

<body>

<div id="header"></div>

<div id="nav">

<div ><div>Online Users</div>

<p><iframe src="./online.php" id="t"></iframe></p>

</div></div>

<div id="form">

<form method="POST" action="user.php" id="mes">

<input class="dropbtn" id="signout" name="signout" type="submit" value=""/>

<input type="hidden" name="name1" id="name1"/>

</form></div>

<div class="dropdown">

<img class="dropbtn" src="./css/menu.png" onclick="myFunction()" alt=""/>

<div id="myDropdown" class="dropdown-content">

<a href="./index.xhtml">Home</a>

<a href="./channels.xhtml">Choose Channels</a>

<a href="./form\_channel.xhtml">Create Channel</a>

<a href="/help.xhtml">Help</a>

</div></div>

<script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/2.0.0/jquery.min.js"></script>

<script type="text/javascript">

$(document).ready(function(){

//create a new WebSocket object.

var wsUri = "ws://10.11.4.38:443";

websocket = new WebSocket(wsUri);

var colors = ['red', 'green', 'blue', 'orange', 'black'];

var result = colors[Math.floor(Math.random() \* colors.length)];

var query = window.location.search.substring(1);

var vars = location.search.split('username=')[1];

document.getElementById("name").value = vars;

document.getElementById("name1").value = vars;

websocket.onopen = function(ev) { // connection is open

var div2 = document.createElement('div');

div2.textContent = "Connected";

div2.setAttribute('class', 'system\_msg');

$('#message\_box').append(div2);

};

$('#send-btn').click(function(){ //use clicks message send button

var mymessage = $('#message').val(); //get message text

var myname = $('#name').val(); //get user name

if(myname == ""){ //empty name?

alert("Enter your Name please!");

return;

}

if(mymessage == ""){ //emtpy message?

alert("Enter Some message Please!");

return;

}

//prepare json data

var msg = {

message: mymessage,

name: myname,

color : result

};

//convert and send data to server

websocket.send(JSON.stringify(msg));

});

//#### Message received from server?

websocket.onmessage = function(ev) {

var msg = JSON.parse(ev.data); //PHP sends Json data

var type = msg.type; //message type

var umsg = msg.message; //message text

var uname = msg.name; //user name

var ucolor = msg.color; //color

if(type == 'usermsg')

{

var divt = document.createElement('div');

var di = document.createElement('span');

var divs = document.createElement('span');

var k = di.innerHTML = uname;

var q = divs.innerHTML = umsg;

di.className = 'user\_name';

divs.className = 'user\_name';

divt = k + ":" + q;

var iDiv = document.createElement('div');

iDiv.style.color = ucolor;

iDiv.id = 'block';

document.getElementById('message\_box').appendChild(iDiv);

iDiv.innerHTML = divt;

// Now create and append to iDiv

// $('#message\_box').append(divt);

}

if(type == 'system')

{

var div3 = document.createElement('div');

div3.textContent = umsg;

div3.setAttribute('class', 'system\_msg');

$('#message\_box').append(div3);

}

$('#message').val(''); //reset text

};

websocket.onerror = function(ev){

var div4 = document.createElement('div');

div4.textContent = "Error Occurred - "+ev.data+"";

div4.setAttribute('class', 'system\_error');

$('#message\_box').append(div4);

};

websocket.onclose = function(ev){

var divs5 = document.createElement('div');

divs5.textContent = "Connection Closed!";

divs5.setAttribute('class', 'system\_msg');

$('#message\_box').append(divs5);

};

});

var auto\_refresh = setInterval(

function()

{

$('#t').fadeOut('slow').load('./online.php').fadeIn("slow");

}, 10000);

/\* When the user clicks on the button,

toggle between hiding and showing the dropdown content \*/

function myFunction() {

document.getElementById("myDropdown").classList.toggle("show");

}

// Close the dropdown if the user clicks outside of it

window.onclick = function(event) {

if (!event.target.matches('.dropbtn')) {

var dropdowns = document.getElementsByClassName("dropdown-content");

var i;

for (i = 0; i = dropdowns.length; i++) {

var openDropdown = dropdowns[i];

if (openDropdown.classList.contains('show')) {

openDropdown.classList.remove('show');

}

}

}

}

</script>

<div class="chat\_wrapper" >

<div class="message\_box" id="message\_box"></div>

<div class="panel">

<input type="text" name="name" id="name" maxlength="10" style="width:20%" readonly="readonly"/>

<input type="text" name="message" id="message" maxlength="80" style="width:60%" required="required" />

<button id="send-btn"><img id="send-btn1" src="css/send.png" alt=""/></button>

</div>

</div>

</body>

</html>

### Channels.xhtml

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml" lang="en">

<head>

<meta http-equiv="X-UA-Compatible" content="IE=edge"/>

<meta name="viewport" content="width=device-width, initial-scale=1"/>

<title>Channels Page</title>

<meta name="description" content="Source code generated using layoutit.com"/>

<meta name="author" content="LayoutIt!"/>

<style type="text/css">

select{

width: 400px;

height: 200px;

}

button{

font-family: Arial;

color: gray;

display: block;

margin: 15px auto;

text-decoration: none;

transition: all 0.3s ease;

font-size: 12px;

}

#lab{

left: 50%;

right: 50%;

}

</style>

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

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

</head>

<body>

<div id="header">

<h1>Wise WebChat</h1>

</div>

<script src="https://code.jquery.com/jquery-1.9.1.min.js"></script>

<script>

function ben(){

$.get("./userC.php", function(data){

$("#selectmultiple").html(data);

setTimeout(function(){ben()},20000);

});

}

$( document ).ready(function() {

ben();

});

</script>

<div id="nav">

<ul>

<li><a href="index.xhtml" class="btn">Home</a></li>

<li><a href="overview.xhtml" class="btn">Overview</a></li>

<li><a href="channels.xhtml" class="btn">Channels</a></li>

<li><a href="form\_channel.xhtml" class="btn">Form Channel</a></li>

<li><a href="help.xhtml" class="btn">Help</a></li>

<li><a href="chat.xhtml" class="btn">Chat</a></li>

</ul>

</div>

<form method="post" class="form-horizontal" action="./user.php">

<fieldset>

<legend>Channels Page</legend>

<div style="text-align:center;">

Select channel:<br/>

<select id="selectmultiple" name="selectmultiple" class="form-control" multiple="multiple">

</select>

<br/><br/>

</div>

<input type="submit" id="join" name="join" value="Join"/>

</fieldset>

</form>

<a href="./form\_channel.xhtml"> <input type="button" name="channel" value="New Channel"/></a>

<div style="text-align:center;">

<a href="http://validator.w3.org/check?uri=referer"><img

src="http://www.w3.org/Icons/valid-xhtml10"

alt="Valid XHTML 1.0!" height="31" width="88" />

</a><a href="http://jigsaw.w3.org/css-validator/check/referer">

<img style="border:0;width:88px;height:31px"

src="http://jigsaw.w3.org/css-validator/images/vcss"

alt="Valid CSS!" />

</a>

</div>

<div id="footer">

&#169; Copyright Wise Chat 2016

</div>

</body>

</html>

### Help.xhtml

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml" lang="en">

<head>

<meta http-equiv="X-UA-Compatible" content="IE=edge"/>

<meta name="viewport" content="width=device-width, initial-scale=1"/>

<title>Help Page</title>

<meta name="description" content="Source code generated using layoutit.com"/>

<meta name="author" content="LayoutIt!"/>

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

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

<style type="text/css">

.heading {

text-decoration: underline;

}

</style>

</head>

<body>

<div id="header">

<h1>Wise WebChat</h1>

</div>

<div id="nav">

<ul>

<li><a href="index.xhtml" class="btn">Home</a></li>

<li><a href="overview.xhtml" class="btn">Overview</a></li>

<li><a href="channels.xhtml" class="btn">Channels</a></li>

<li><a href="form\_channel.xhtml" class="btn">Form Channel</a></li>

<li><a href="help.xhtml" class="btn">Help</a></li>

<li><a href="chat.xhtml" class="btn">Chat</a></li>

</ul>

</div>

<div id="section">

<fieldset style="width:1100px">

<div class="form-group">

<h5>

<b>If you face a problem or have any concerns, please free to contact us at rsj1@hw.ac.uk.</b>

</h5>

<p class="heading">

<b>Tips to Start</b>

</p>

<p> This chat works best with Firefox, Chrome, Opera or Safari. </p>

<p class="clr">

The first step is to register yourself if you haven't already.

Navigate to the home page

Please take care of the below when you register.</p>

<ul> <li>Don't use joint e-mail addresses.</li>

<li>Don't use joint nicknames.</li>

<li>Don't tell anybody the password for your nickname.</li></ul>

<p class="clr">

Once you get your username and password, you can log in to the application.

Browse through the different chat rooms available and join the one which interests you.

You can also create a new chat room for a topic of your interest.</p>

<p class="clr">

Hope you have an enjoyable time. This chat was, is and will always be free. Thanks!

</p>

<p class="heading">

<b>Frequently Asked Questions</b>

</p>

<p class="clr">

<b>What can be done if I forget my password?</b>

</p>

<p>On the home page, below the login button, click on 'Lost your password?'.

Enter your email address in the field provided.

An email will be sent to your email address with a link.

Follow the link to reset the password for your account associated with the email address.</p>

<b>Is there a code of conduct?</b>

<p>While we try not to put too many restrictions, we do want you to be aware that you are chatting

with a live person. We appreciate a positive and polite tone. Any abusive language will not be tolerated.</p>

</div>

<div style="text-align:center;">

<a href="http://validator.w3.org/check?uri=referer"><img

src="http://www.w3.org/Icons/valid-xhtml10"

alt="Valid XHTML 1.0!" height="31" width="88" />

</a><a href="http://jigsaw.w3.org/css-validator/check/referer">

<img style="border:0;width:88px;height:31px"

src="http://jigsaw.w3.org/css-validator/images/vcss"

alt="Valid CSS!" />

</a>

</div>

</fieldset>

</div>

<div id="footer">

&#169; Copyright Wise Chat 2016

</div>

</body>

</html>