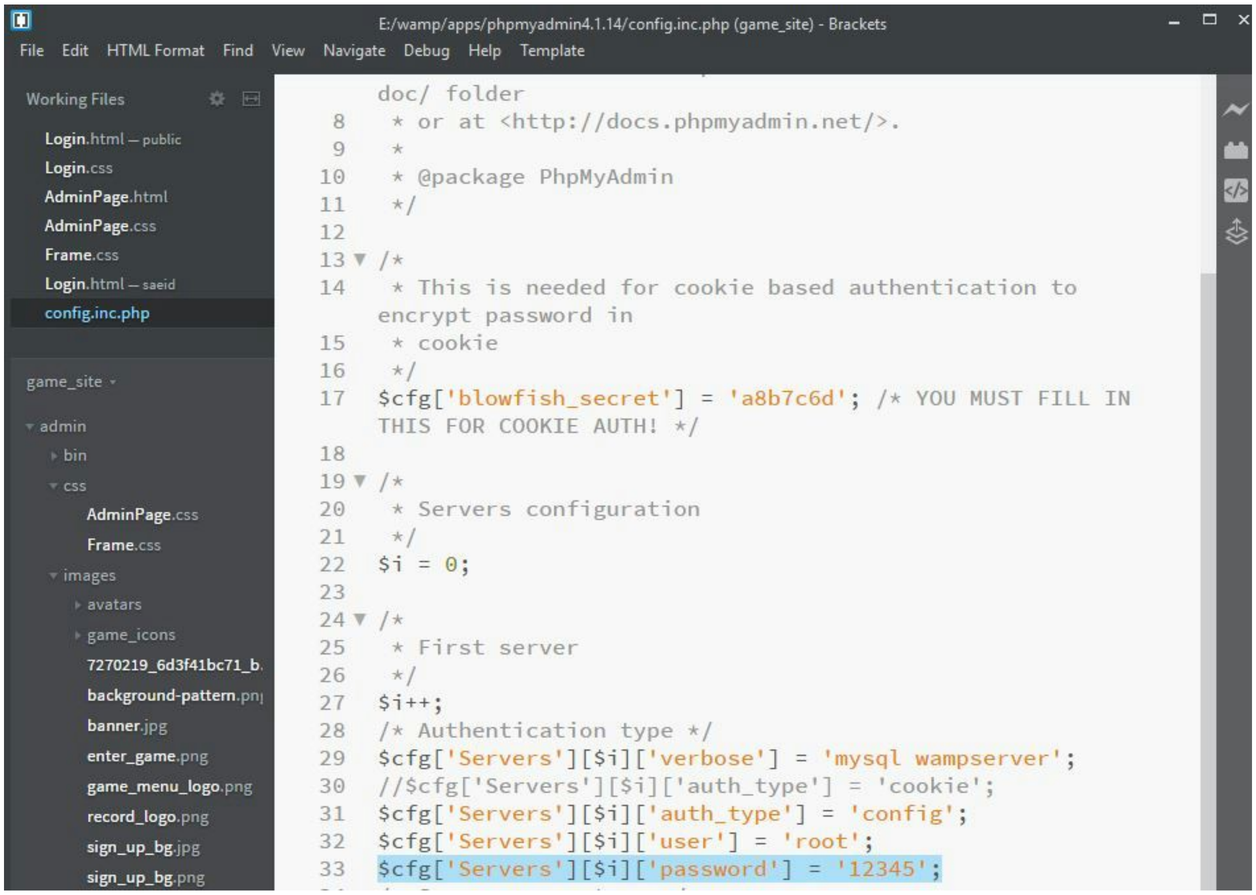
<u>Instruction manual for Group 5 game project</u>

Server Installation Instruction:

In order get the website functional, our work need some prerequisite applications to be installed.

Database Preparation:

Our database uses the Mysql server which is in the Wamp or Mamp server.
We have provided an SQL file which contains all required tables. At the first step a new database should be created with the name to be “***game_board***”. after this part our SQL file should be imported using the import procedures in the Mysql. The SQL compatibility mode: should be set to the Mysql at the time of import. Another step is to set the root password for Mysql server as “12345”. This process can be done using the simple text editor and editing the “config.inc.php” as the following figure.



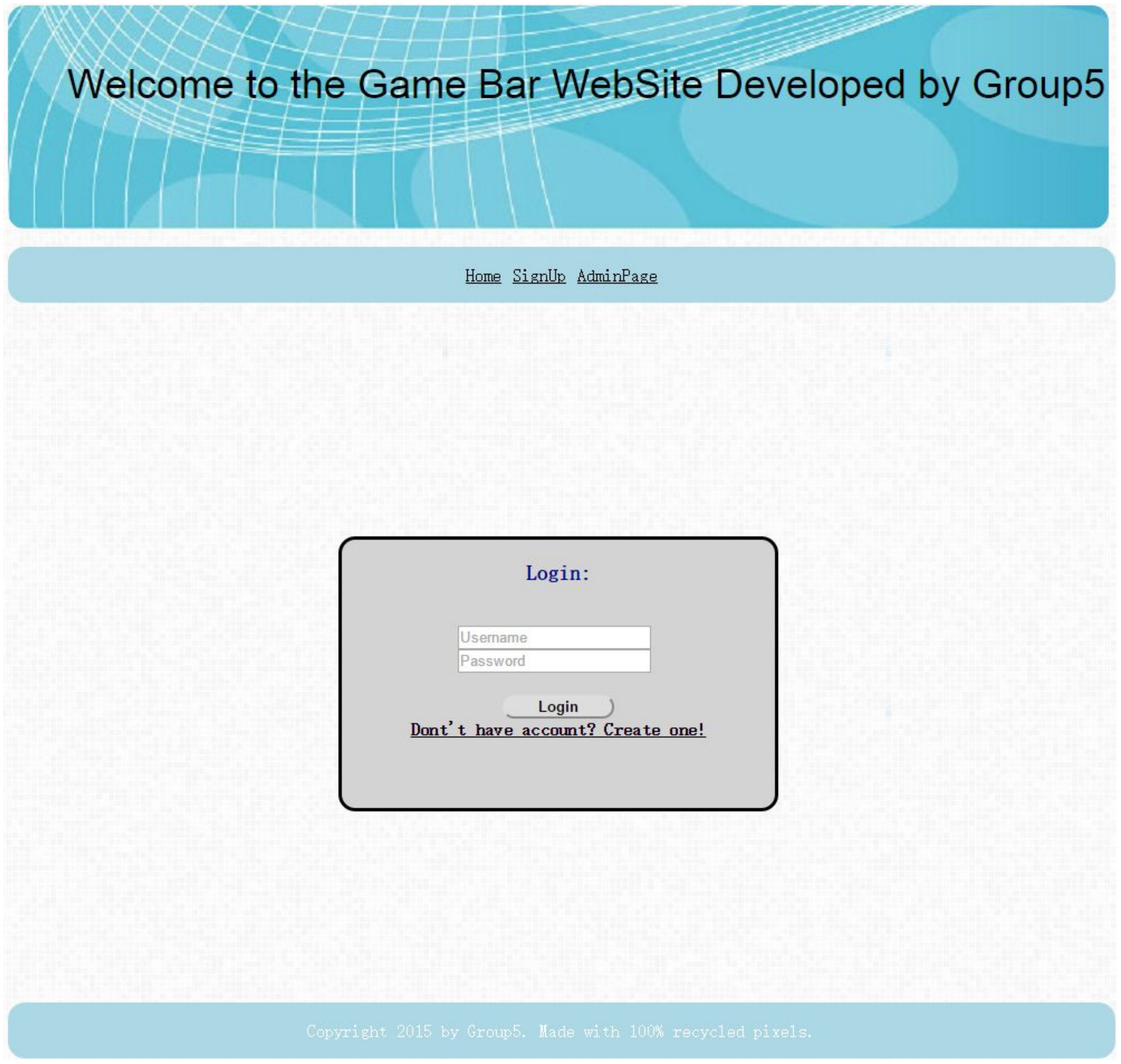
```
doc/ folder
 * or at <http://docs.phpmyadmin.net/>.
 *
 * @package PhpMyAdmin
 */

/*
 * This is needed for cookie based authentication to encrypt password in
 * cookie
 */
$cfg['blowfish_secret'] = 'a8b7c6d'; /* YOU MUST FILL IN THIS FOR COOKIE AUTH! */

/*
 * Servers configuration
 */
$i = 0;

/*
 * First server
 */
$i++;
/* Authentication type */
$cfg['Servers'][$i]['verbose'] = 'mysql wampserver';
//$cfg['Servers'][$i]['auth_type'] = 'cookie';
$cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '12345';
```

Ratchet and Composer Installation:

We have implemented the "Tic Tac Toe" game server using the Ratchet library which is a toolset that allows creating bi-directional applications between our clients over WebSockets. From the technical aspect, Websockets are full-duplex and establish background connections between our registered clients and the Tic Tac Toe game server.

Instruction for installing Ratchet:

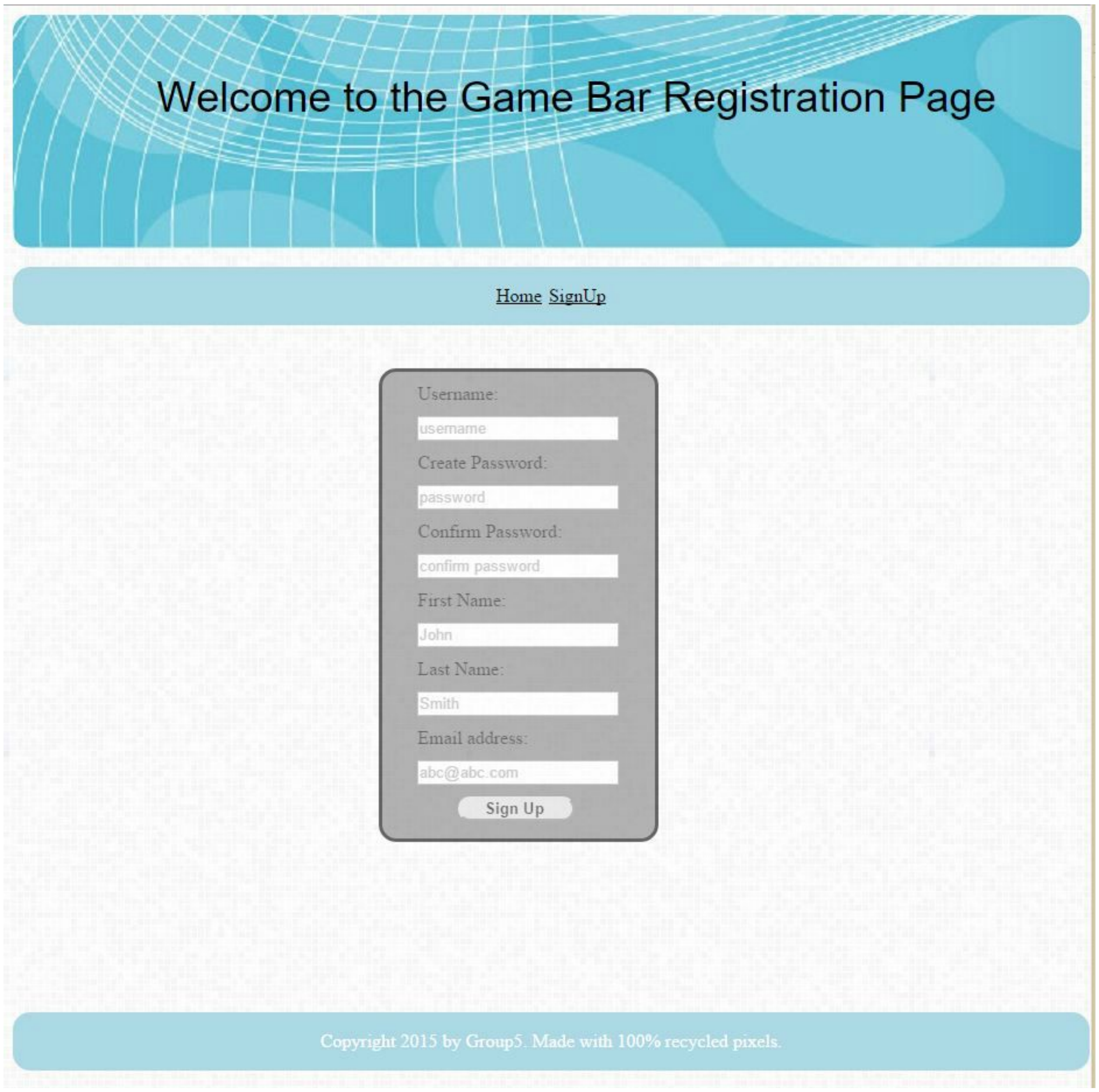
1- Copy the project folder into your WAMP or MAMP folder.
2- Install Composer: https://getcomposer.org/
There are windows, linux, and ubuntu versions for composers. You can find the installation guide here:https://getcomposer.org/doc/00-intro.md or
 http://www.dev-metal.com/install-update-composer-windows-7-ubuntu-debian-centos/.
3- Install Ratchet: php ~/composer.phar require cboden/ratchet
4- run php bin/server.php
5- Open the index.html file located in the public folder two times
6- Then start playing the game!


Welcome to the Game Bar WebSite Developed by Group5
Home SignUp AdminPage
Login:
Username
Password
Login
Dont't have account? Create one!
Copyright 2015 by Group5. Made with 100% recycled pixels.

Login Page:

The website starts from the Login page. In this page user has two choices, either “Login” or “Create Account”. The user can create an account using the navigation bar or from the provided link in the “Login” form.

The administrator can click on “AdminPage” to switch to administrator login page.



Welcome to the Game Bar Registration Page

Home SignUp

Username:

username

Create Password:

password

Confirm Password:

confirm password

First Name:

John

Last Name:

Smith

Email address:

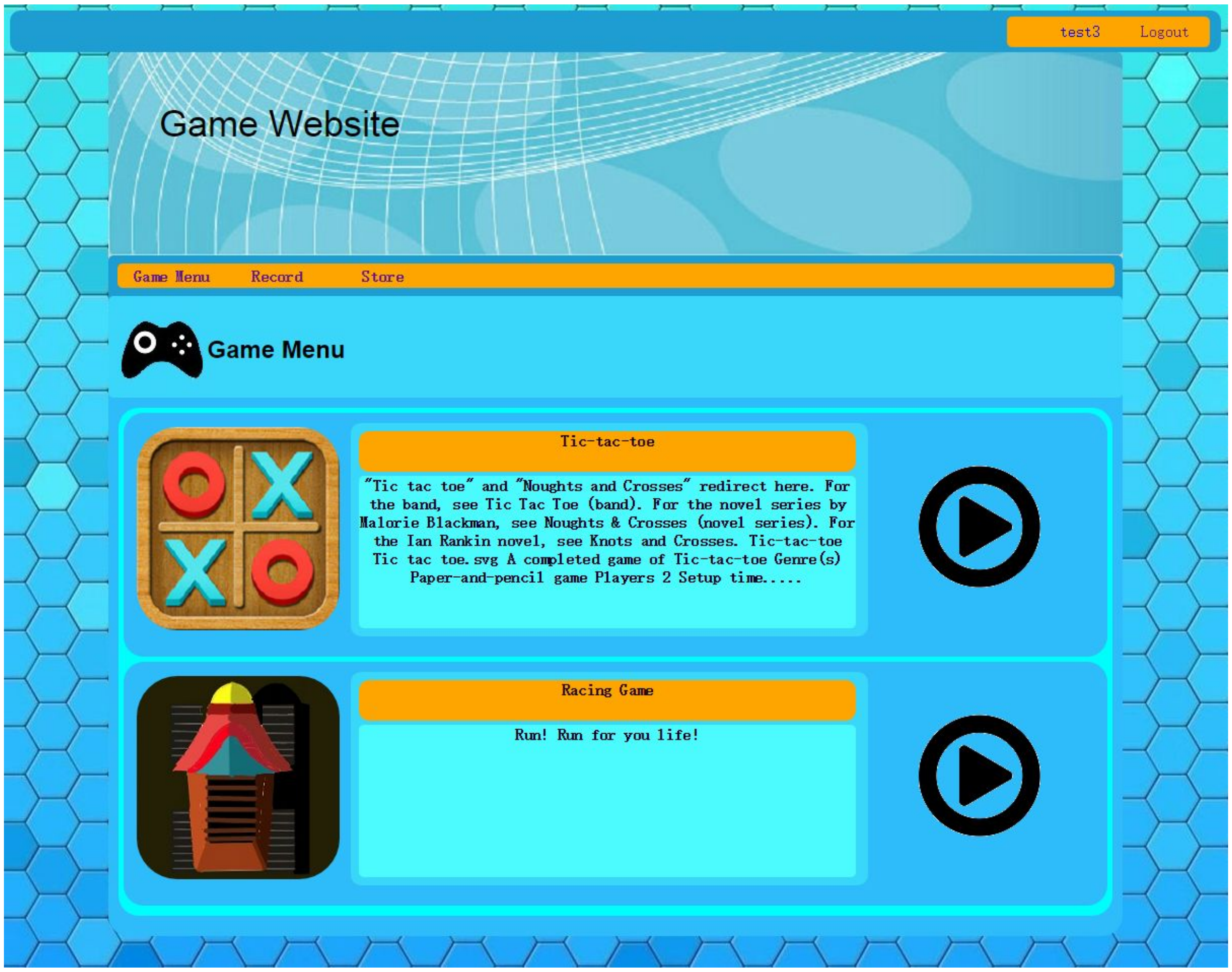
abc@abc.com

Sign Up

Copyright 2015 by Group5. Made with 100% recycled pixels.

Registration Page:

From the registration page, a new user can create an account on our server by using the form. This page can be reached using the “SignUp” section or from the provided link on the “Login” page.
After this procedure user should use “Home” section of the navigation bar and try to sign in the website.

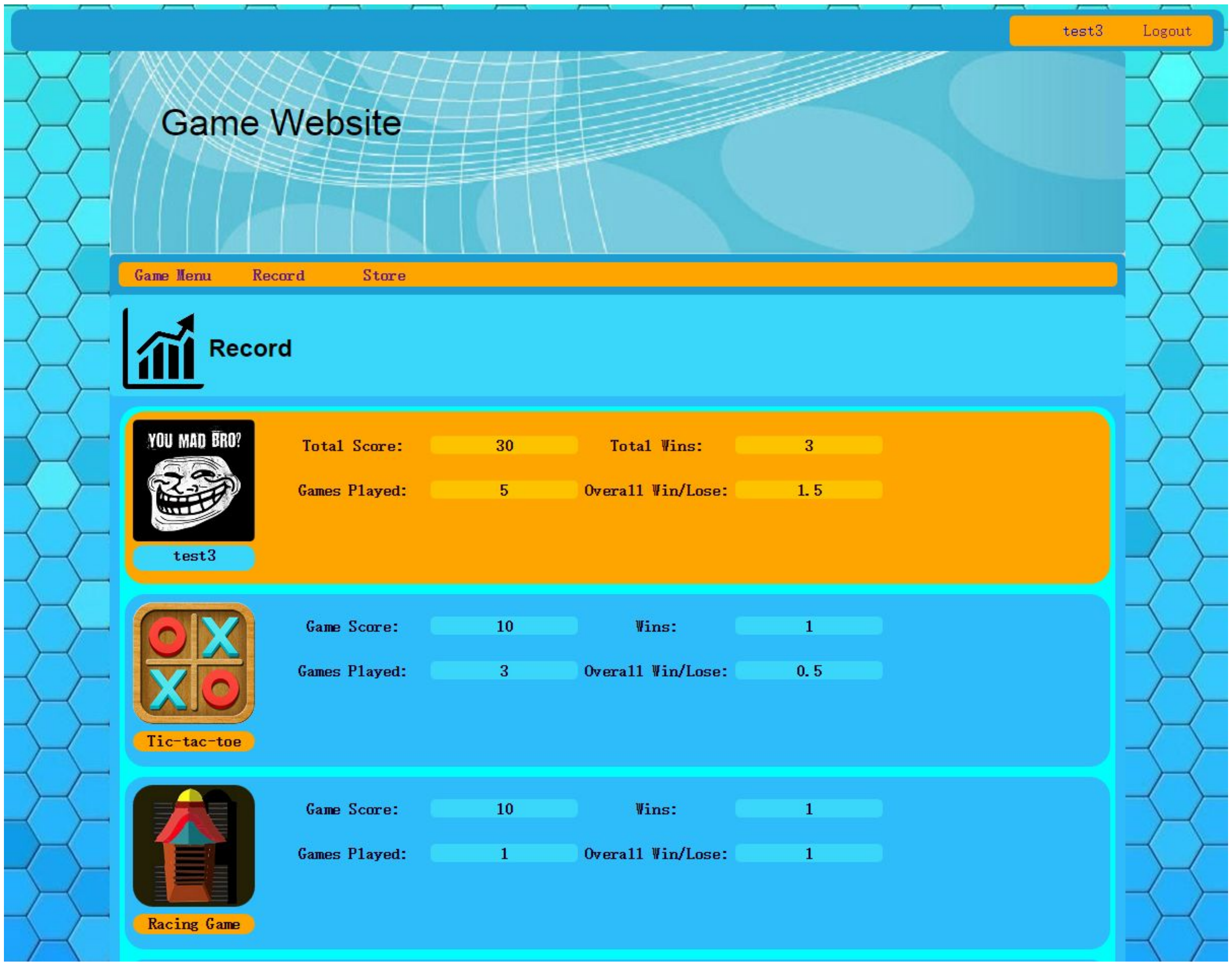


"Tic tac toe" and "Noughts and Crosses" redirect here. For the band, see Tic Tac Toe (band). For the novel series by Malorie Blackman, see Noughts & Crosses (novel series). For the Ian Rankin novel, see Knots and Crosses. Tic-tac-toe Tic tac toe.svg A completed game of Tic-tac-toe Genre(s) Paper-and-pencil game Players 2 Setup time.....

Run! Run for you life!

Game Menu:

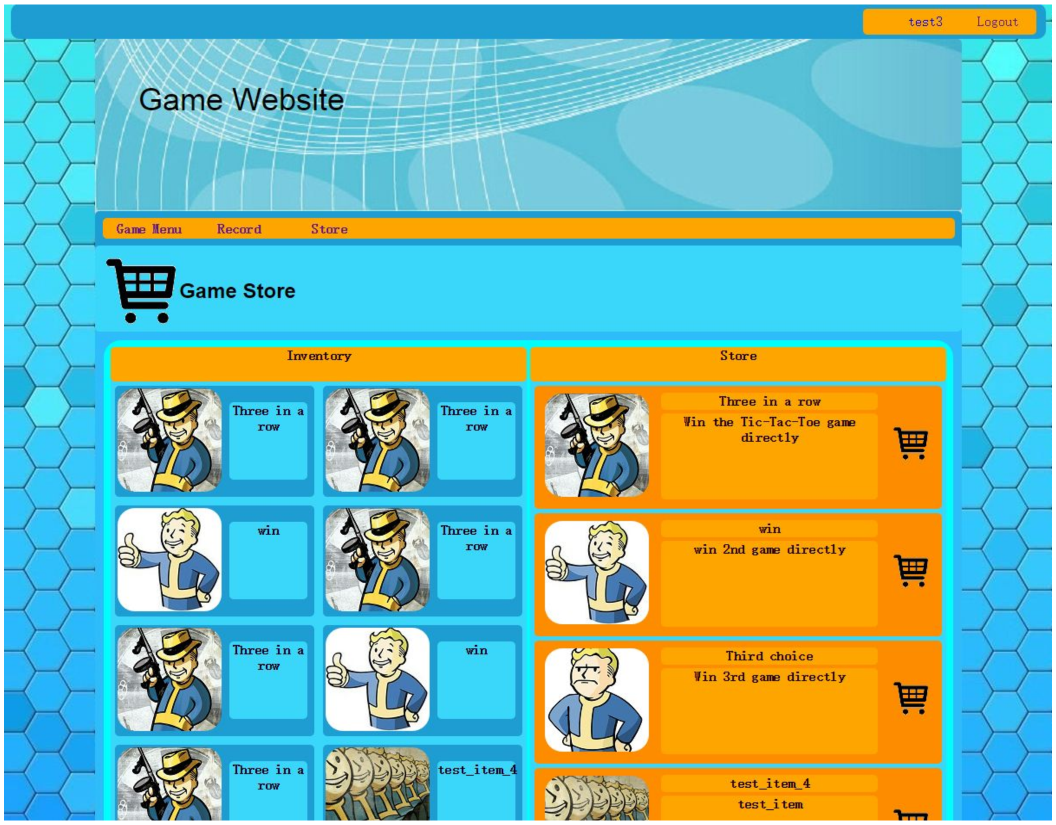
After a successful login, a user will reach the “Game Menu” page.
In this page, there are two different games. The first game is "tic tac toe" which is our multiplayer game. The second game is called, "Tank Race" which is our single player game. Click the “play” icon to play the corresponding game.
On top of that, the user can access the "Record" or "Store" page using the game navigation bar. Also, the user can see her username or logout on top of the page.

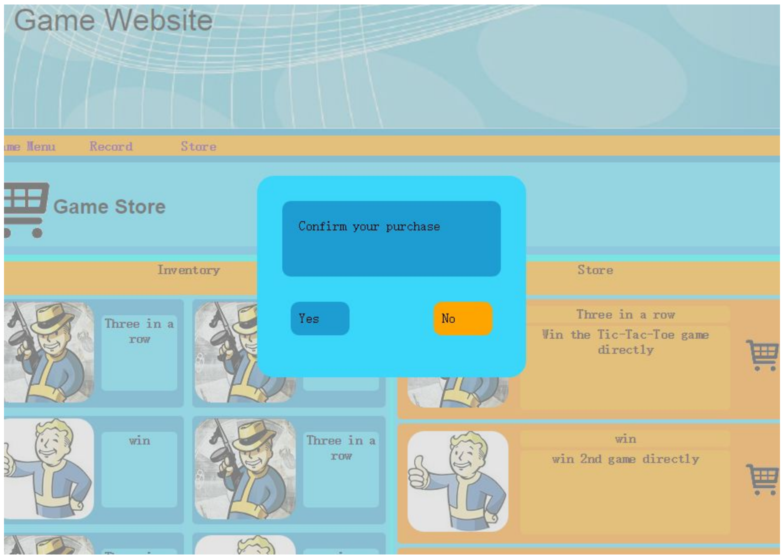


test3	Total Score:	30	Total Wins:	3
	Games Played:	5	Overall Win/Lose:	1.5
Tic-tac-toe	Game Score:	10	Wins:	1
	Games Played:	3	Overall Win/Lose:	0.5
Racing Game	Game Score:	10	Wins:	1
	Games Played:	1	Overall Win/Lose:	1

Game Record:

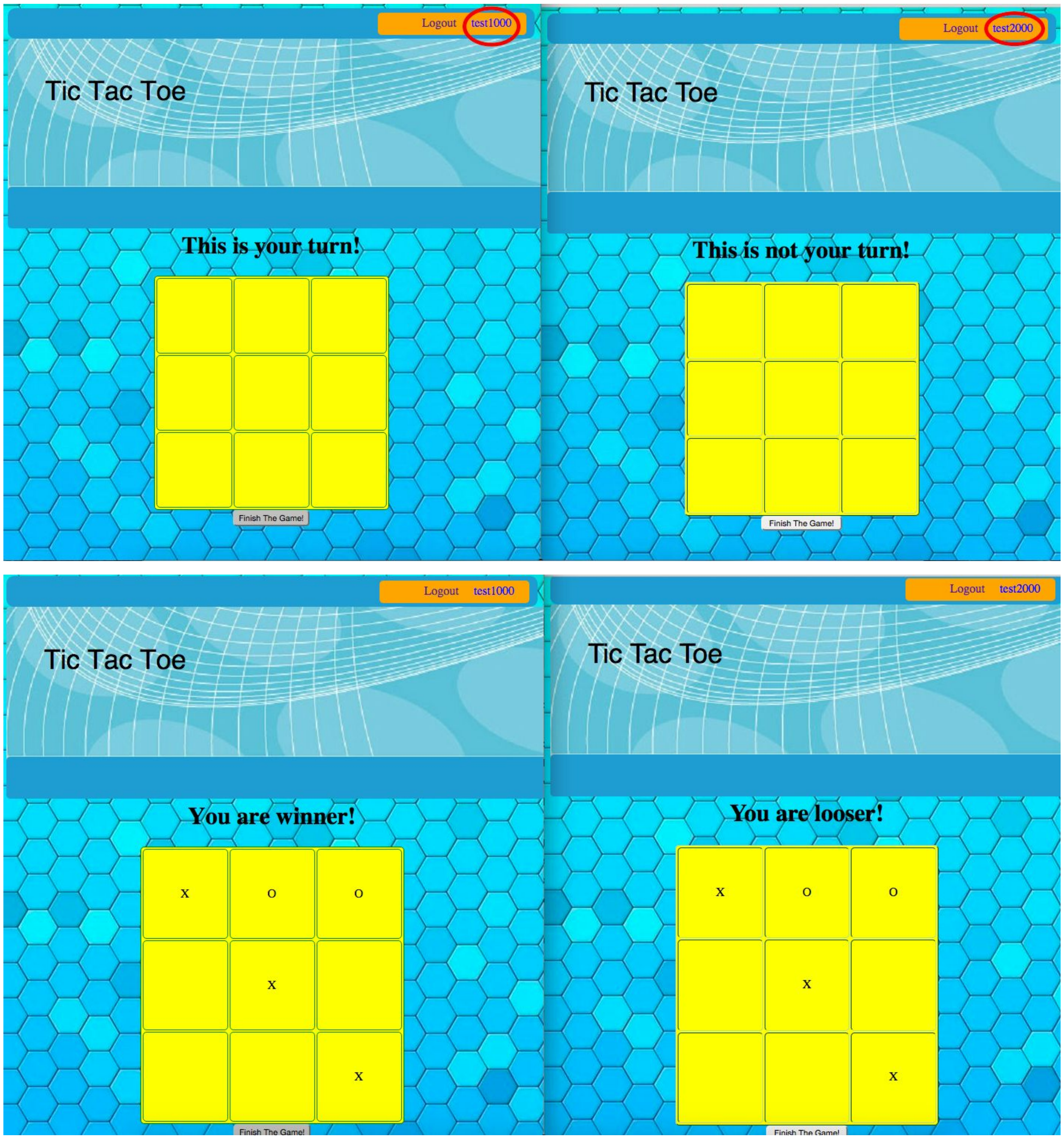
In this page, the user can see her total score and a history of all games she played.
Also, the user can navigate between other pages or logout using the corresponding buttons.





Game Store:

In this page, the user can spend her credits and buy extras or items for the games. Click the “shopping cart” icon and confirm the purchase to buy item.
Also, the user can navigate between other pages or logout using the corresponding buttons.



1: Tic Tac Toe (Realtime Multiplayer Game):

We have implemented the Tic Tac Toe game server using the Ratchet library which is a toolset that allows creating bi-directional applications between our clients over Websockets. From the technical aspect, Websockets are full-duplex and background connections between our registered clients and the Tic Tac Toe game server. We have implemented the game using the following structure (located in the game_tic_tac folder).

- There is "*bin*" folder that stores the websocket server file, which runs in the background and listens to new game connections and handles events.
- There is folder called "*public*" that contains the client part of the game and event handling implementations.
- There is a folder called "*src*" holds our custom server side code
- The last folder is "*vendor*" that holds the third-party PHP libraries.

Server-side Websocket and Logic

Our Tic Tac Toe game server is aware of all background or websocket connections, while having three responsibilities

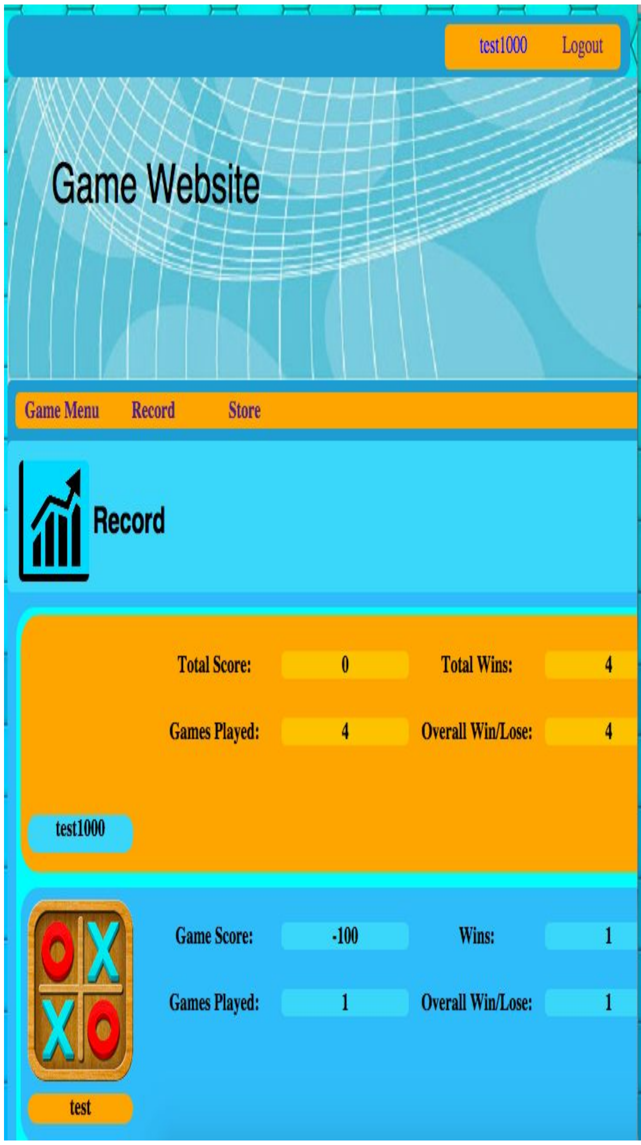
- It randomly pairs players into separate games.
- It keeps the actions of each player in the game across different rounds.
- It makes sure a player cannot play more than one time in each round.
- It determines the winner and loser in each game and then sends back the winner and looser information through invisible forms to the record page for persisting into the database.
- It uses JSON to encode and decode messages exchanged with players for the ease of implementation.

Details: Our game server keeps track of all players by storing their connections and actions into multiple PHP storage and dictionary objects (SplObjectStorage). To implement the game server using Ratchet: First, we have used Composer to manage the PHP library dependencies and handle autoloading. Second, we have constructed namespaces for the Ratchet library and then created the Chat class (located in the game_tic_tac/src/MyAPP) that implements the game logic. Finally, we have created our websocket server (game_tic_tac/bin/server.php), give it our Chat handler class, and tell it to listen on port 8080. In the websocket server, Ratchet exposes some components that we can use. For the tic tac toe, we only need to use the WsServer (WebSocket server) that provides "MessageComponentInterface". When we start up our game socket server, it creates a new instance of Chat class that implements the game logic. This class handles "onOpen" event that is issued when a player selects to play the tic tac toe game by connecting to the websocket server. In this function, we stores the new connection information, bootstrap our dictionary and list data structures, and finally send the necessary welcome message using JSON message encoding. When our game logic receives a message from "OnMessage" event, it decides to updates the data structures and then issues updates to the other player.

Client-side JQuery and Logic

In the player side, we use use jQuery. Our app.js runs in the client browser and allows a player to connect to the socket server, and send and receive messages between players. Similar to the server side, our websocket API exposes the same events so we define a callback function to handle each one. The “onopen” event happens when our client connects to the socket server. The “onmessage” event happens when the client receives a message from the socket server. Each of these callbacks receives an event object as an argument. The actual payload of the message is stored in the event object’s “data” property.

As can be seen in the above figure, after the game is finished, each user gets her winner and loser status, by pressing the “Finish the game” button each user will be sent to the record page. The winner in this game will earn +100 point, and the loser will punish -100 to her total score. The following describes the detail of the Tic Tac Toe game. If a user selects to play the “tic tac toe” then, she will be send to the “tic tac toe” page but the game never starts unless another player selects to play “ tic tac toe”. This game also supports more than two parallel game.

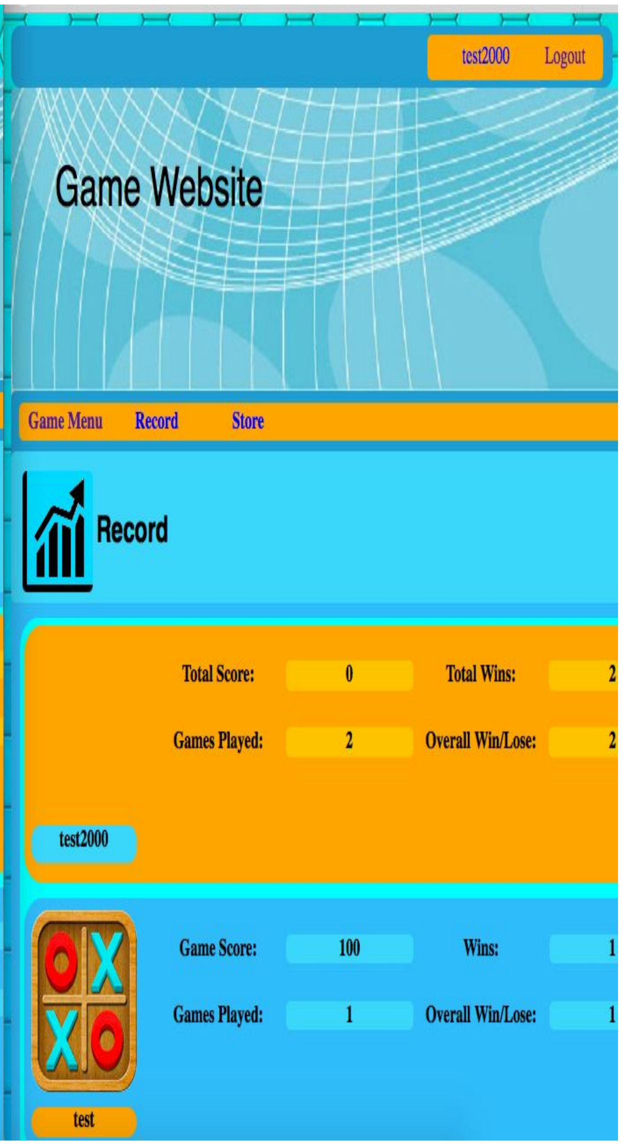


Total Score:	0	Total Wins:	4
Games Played:	4	Overall Win/Lose:	4

test1000

Game Score:	-100	Wins:	1
Games Played:	1	Overall Win/Lose:	1

test

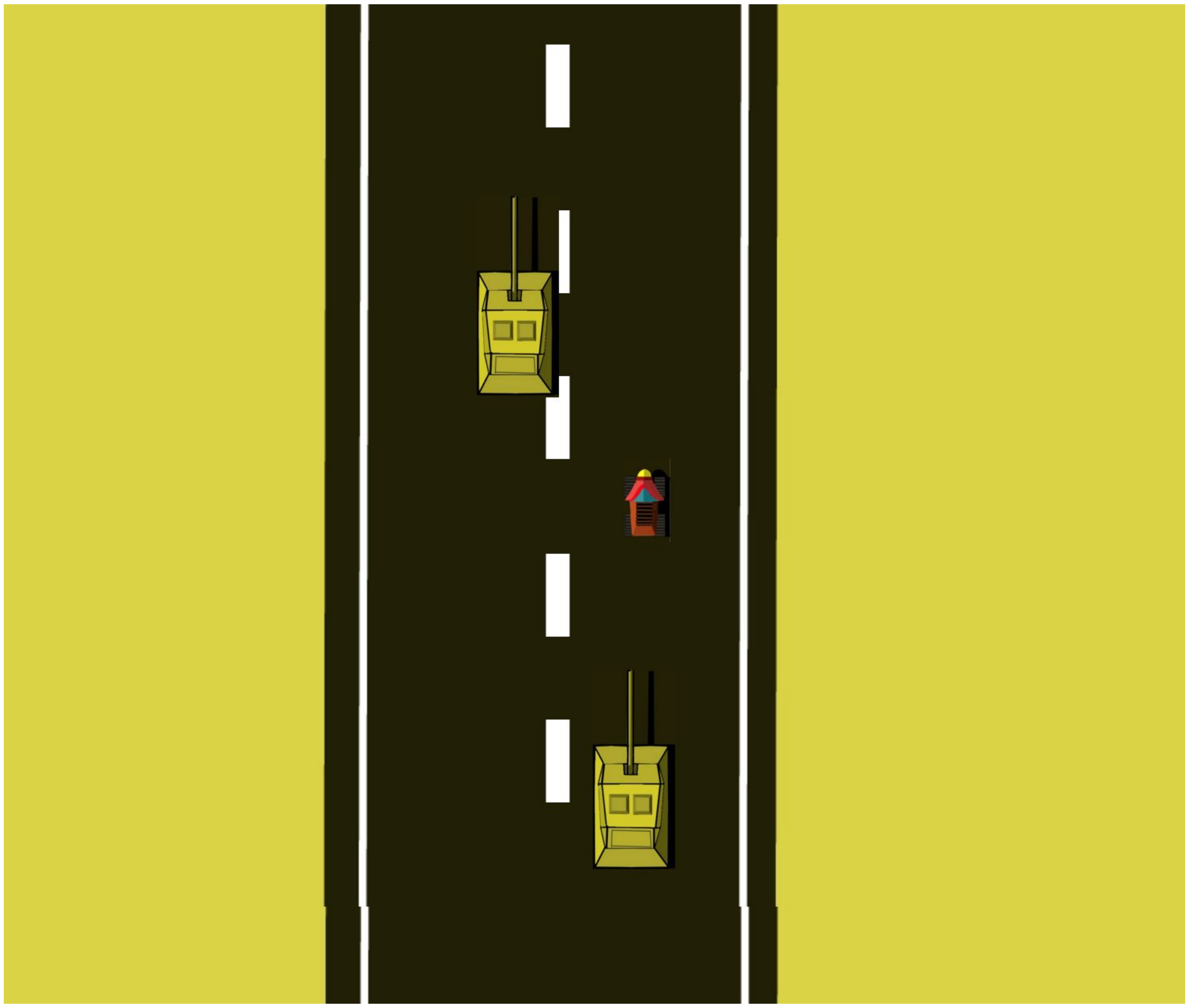


Total Score:	0	Total Wins:	2
Games Played:	2	Overall Win/Lose:	2

test2000

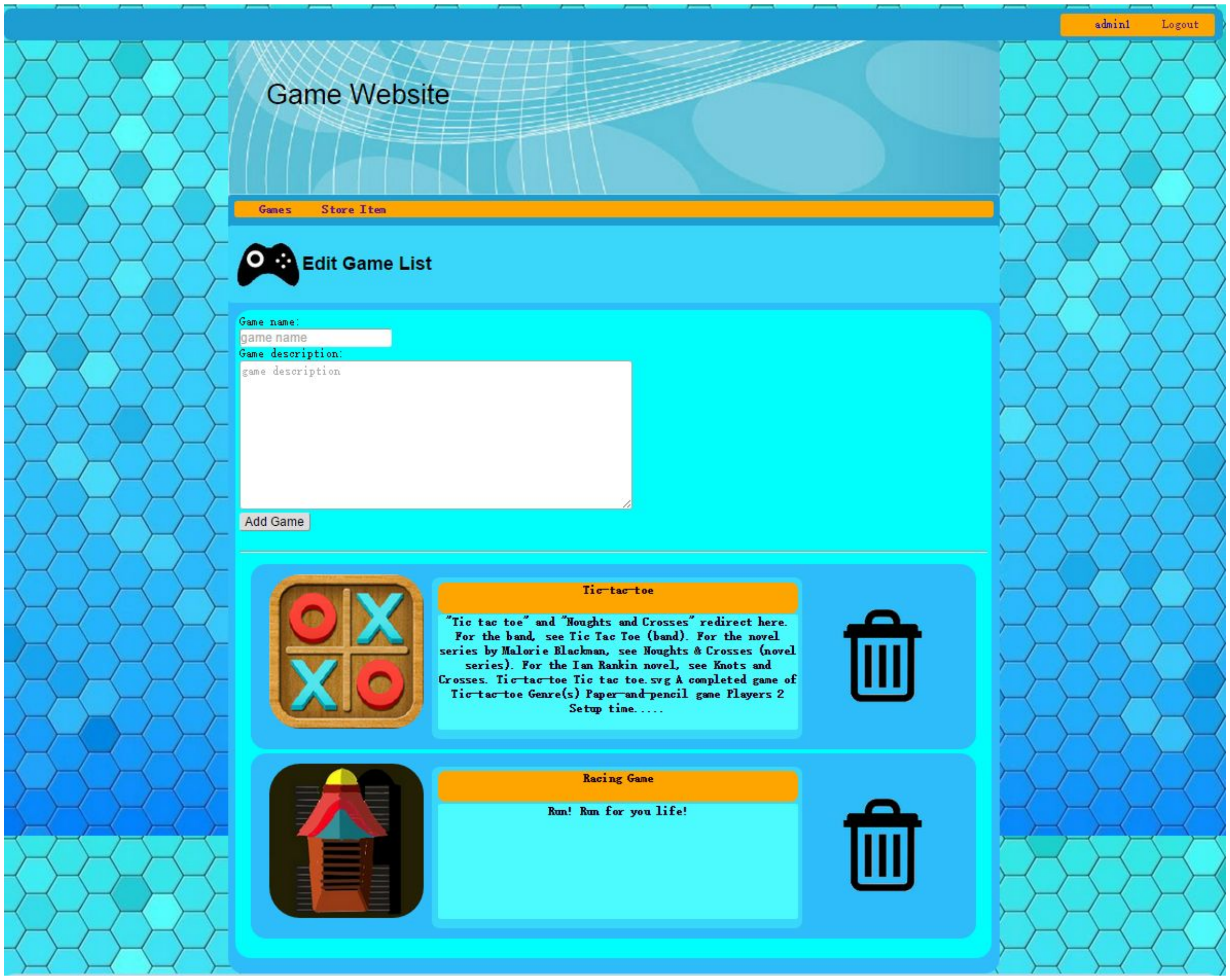
Game Score:	100	Wins:	1
Games Played:	1	Overall Win/Lose:	1

test



2: Tank Race game (Single Player game):

In this game, the player should avoid hitting the barriers, and she will earn 10 points every 2 seconds. After the game is finished, the user will see her total score will be sent to the game menu page.



Game name:
game name
Game description:
game description

Tic-tac-toe

"Tic tac toe" and "Noughts and Crosses" redirect here. For the band, see Tic Tac Toe (band). For the novel series by Malorie Blackman, see Noughts & Crosses (novel series). For the Ian Rankin novel, see Knots and Crosses. Tic-tac-toe Tic tac toe.svg A completed game of Tic-tac-toe Genre(s) Paper-and-pencil game Players 2 Setup time.....

Racing Game

Run! Run for you life!

Administration page:

The Administration pages of the website consist of “Edit game” page, “Edit items” page, “Manage Accounts” page and “Manage item list” page. An administrator needs to login from the Admin login page using administrator’s account to access administration pages. Unfortunately, we don’t have time to complete them all. But we have “edit game” page major function works. The “edit game” page has two part. The form above is used to create a new game in the database. Filling the form and click ‘add game’, then the new game will be inserted into ‘games’ table. If we had more time, we would allow the administrator to select a game source file and icons and add them to the website directory automatically. For the moment, the administrator needs manually add game icon and game folder into the website directory.
The list below shows the games that already exist. Clicking the “trash” icon on the right side will delete the corresponding game in the database.