Tic Tac Toe using PHP

Survey of Programming Languages
ITCS-4102/5102
Final Project Report

10th August 2018

Dhiksha Ramkumar

Table of Contents

1. Introduction

The motivation of this project is the fast-growing fascination towards game technology using advanced technologies. This project aims at creating a game using PHP, a language so basic and versatile, so that people who are a novice in programming can attempt to create a game too. Game design requires high-level visuals and complex algorithms to produce an interactive game for the user, but rudimentary games like sudoku, tic tac toe, etc. can be built using easier languages like PHP.

PHP is an open source general-purpose scripting language suited for web development. PHP can also be embedded in HTML to create diverse web pages. The difference between PHP and JavaScript is that PHP code is executed on the server side and then generates a HTML code which is then sent to the client, but the client would not be aware of the underlying code whereas in the case of JavaScript it is a client-side program, meaning it is executed on the client side itself. PHP can be advantageous to people who are new to coding but want to perform advanced tasks.

1.1 Purpose

Game design is an emerging field that has gained huge popularity among all age groups. They involve high-end technologies for visuals and complex algorithms to achieve the end product. I propose to take the problem of Tic Tac Toe, a rudimentary yet popular game, and try to use a simple language such as PHP and create the game. This would help widen the extent to which PHP can be used and can open opportunities for beginners to create simple games using simple projects.

1.2 Scope

This is a Tic Tac Toe game product that can be played by a single player. The product has the ability to place an 'o' in a randomly generated place for every 'x' that the player places. The game, however, does not learn the technique to efficiently win the game. This project can help widen the scope of PHP for using it not only as a web development tool but also for game development. This mainly aims for simplicity in creating simple games rather than using complex algorithms to achieve it. The goal is to create a user interactive game where a single player plays against the system and tries their hand at winning the game.

1.3 References

1. PHP tutorials: https://www.w3schools.com/pHP/default.asp
2. Concept of creating tic tac toe game in python: https://www.youtube.com/watch?v=9jw4ndKDpBE
3. Tic Tac Toe in java script: https://github.com/kennycason/ultimate_tictactoe
4. Apache setup: https://websitebeaver.com/set-up-localhost-on-macos-high-sierra-apache-mysql-and-php-7-with-sslhttps

1.4 Overview

This project focuses on creating a simple single player game that can be created using just PHP. The game chosen to be recreated is a Tic Tac Toe game, where the player plays against the computer and tries to win against it. Tic Tac Toe is game with grids containing 3 rows and 3 columns, with a total of 9 blocks. The goal of the game is to get vertical, horizontal or diagonal alignment of the same characters. PHP, being a language used for web development, this game will be a web passed game where the player plays against a computer on the page and attempts to win against it.

2. General Description

2.1 Specifications

- Required specifications: Apache, Operating system (MacOS was used during the creating of this project)
- Specified Time Span: 1 week
- Goal: Creation of Tic Tac Toe game using PHP

2.2 Project Components

This project has only two components. One is the php file itself containing the code that must run on the local host server and the next one is the image file that acts a banner for the homepage.

2.3 Steps to Execute the project

[Note: These steps are listed based on the specifications of MacOS, it will vary depending on the system used]

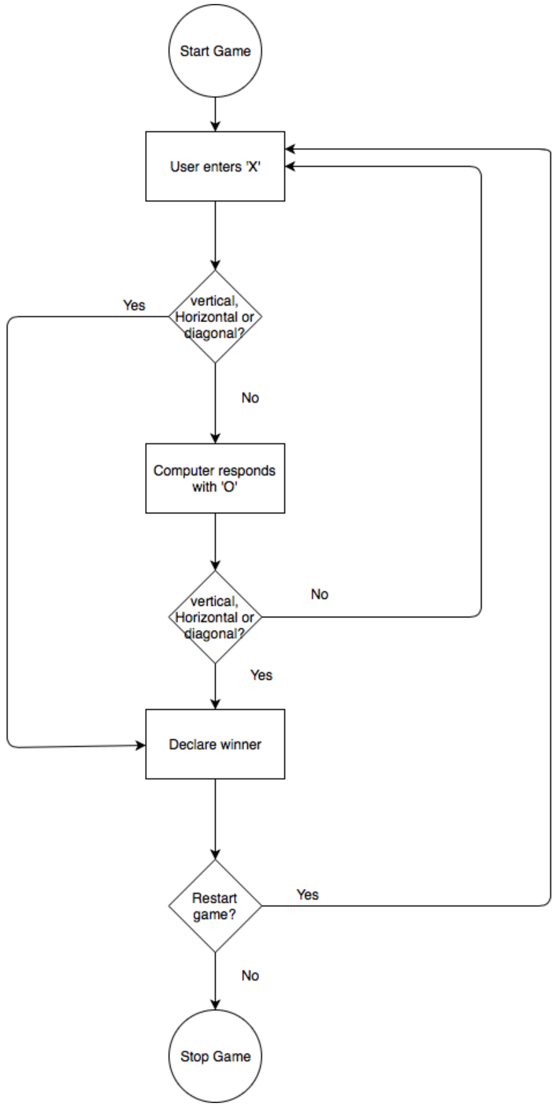
To begin with, the Apache server must be start in order to execute the PHP file. Perform the following steps to get it started:

- Open Terminal
- Enter the command:
  ```
  sudo apachectl start
  ```
- Open safari ,enter localhost in the URL bar. This should display a screen with “it works!” on it.
- Go to the httpd.conf file accessed using the command:
  ```
  sudo nano /etc/apache2/httpd.conf
  ```
- In that remove the # from the LoadModule php7_module libexec/apache2/libphp7.so
- Exit the page after saving it
- Restart apache using the command:
  ```
  sudo apachectl restart
  ```
- Next, click Go on the top bar and go to Home and create a folder “Sites” and place a index.php in that folder.
- Again go to the httpd.conf file and replace the occurrence of library from ‘/Library/WebServer/Documents” to “/Users/”computer_name”/Sites”.
- Save file and restart apache again
- Open safari again and refresh the localhost webpage and you’ll see the contents of the index.php file on the webpage

The index.php file consists the code mentioned in the next section and also place the image file you want as banner in the same sites folder to easily access it without having to specify the directory.

3. Program Specifications

3.1 Flow Chart


Start Game
User enters 'X'
Yes
vertical, Horizontal or diagonal?
No
Computer responds with 'O'
No
vertical, Horizontal or diagonal?
Yes
Declare winner
Restart game?
Yes
No
Stop Game

3.2 Program Logic and Code Explanation

```
<?php
$winner = 'n';
$box = array('','','','','','','','','');
if (isset($_POST["submitbtn"]))
   {
     $box[0] = $_POST["box0"];
     $box[1] = $_POST["box1"];
     $box[2] = $_POST["box2"];
     $box[3] = $_POST["box3"];
```

```
    $box[4] = $_POST["box4"];
    $box[5] = $_POST["box5"];
    $box[6] = $_POST["box6"];
    $box[7] = $_POST["box7"];
    $box[8] = $_POST["box8"];

    if(($box[0] == 'x' && $box[1] == 'x' && $box[2] == 'x')  || ($box[3] == 'x' && $box[4] ==
'x' && $box[5] == 'x') || ($box[6] == 'x' && $box[7] == 'x' && $box[8] == 'x') || ($box[0] == 'x'
&& $box[3] == 'x' && $box[6] == 'x')  || ($box[1] == 'x' && $box[4] == 'x' && $box[7] == 'x')
|| ($box[2] == 'x' && $box[5] == 'x' && $box[8] == 'x') || ($box[0] == 'x' && $box[4] == 'x' &&
$box[8] == 'x') || ($box[2] == 'x' && $box[4] == 'x' && $box[6] == 'x') )
    {
        $winner = 'x';
        Print "<h1>X Wins!</h1>";
    }

    $blank = 0;
    for ($i = 0; $i <= 8 ; $i++)
    {
        if($box[$i] == '')
        {
                $blank = 1;
        }
    }
    if($blank == 1)
    {
        $i = rand() % 8;
        while($box[$i] != '')
        {
                $i = rand() % 8;
        }
        $box[$i] = 'o';
        if(($box[0] == 'o' && $box[1] == 'o' && $box[2] == 'o')  || ($box[3] == 'o' && $box[4]
== 'o' && $box[5] == 'o') || ($box[6] == 'o' && $box[7] == 'o' && $box[8] == 'o') || ($box[0] ==
'o' && $box[3] == 'o' && $box[6] == 'o')  || ($box[1] == 'o' && $box[4] == 'o' && $box[7] ==
'o') || ($box[2] == 'o' && $box[5] == 'o' && $box[8] == 'o') || ($box[0] == 'o' && $box[4] == 'o'
&& $box[8] == 'o') || ($box[2] == 'o' && $box[4] == 'o' && $box[6] == 'o') )
        {
                $winner = 'o';
                Print "<h1>O Wins!</h1>";
        }

    }
    else if ($winner == 'n')
    {
        $winner = 't';
        Print "<h1>Game Tied!</h1>";
    }
  }
?>
<html>
<head>
  <title>Tic Tac Tow</title>
  <style>
  body {
    background-color: brown;
    text-align: center;
  }
  #ip{
    border-radius: 50px;
    border: 2px solid black;
    padding: 50px;
    width: 200px;
    height: 15px;
    margin-bottom: 20px;
    margin-top: 20px;
    margin-right: 20px;
    font-size: 30px;
  }
  #go{

    width: 200px;
    height: 15px;
    margin-top: 20px;
    padding: 50px;
    border-radius: 50px;
  }
  </style>
</head>
<body>
  <img src="Tic-Tac-Toe-Pro.jpg" alt="banner" style="width:1000px;height:300px;">
  <div style="margin:0 auto;width:75%;text-align:center;">
  <form name = "ticktactoe" method = "post" action = "index.php">
    <?php
        for($i = 0; $i <=8; $i++)
        {
                printf('<input type = "text" id = "ip" name = "box%s" value = "%s">', $i, $box[$i]);
                if ($i == 2 || $i == 5 || $i == 8){
                print("<br>");
                }
        }
        if($winner == 'n')
        {
                print('<input type = "submit" name = "submitbtn" value = "Next Move" id =
"go">');
        }
        else
        {
                print('<input type = "button" name = "newgamebtn" value = "Play Again" id =
"go" onclick = "window.location.href=\'index.php\'">');
        }

    ?>
  </form>
  </div>
</body>
</html>
```

This project has a simple code logic where it consists of just one "index.php" file with a PHP section to begin with, followed by HTML code and the CSS code for the presentation which is internally present in the html code.

PHP Section:

The winner is set to "n" in the beginning and an array "box" is created for storing all the places with 'x' and 'o' throughout the game. On the event of clicking the go button, the values entered are stored in the respective places in the box array. Then the condition for 'x' winning is checked by checking if the places (0,1,2),(3,4,5),(6,7,8) [horizontal case] or (0,3,6),(1,4,7),(2,5,8) [vertical case] or (0,4,8), (2,4,6)[diagonal case] is filled with 'x'. If yes, then the winner is declared to be 'x' else 'o' is placed on a random place with the help of the" rand ()" function and later the condition for winning is checked for 'o' as well and if it is satisfied, 'o' is declared as winner. If at the end the winner variable is not changed and still remains "n" then the game is declared to be tied.

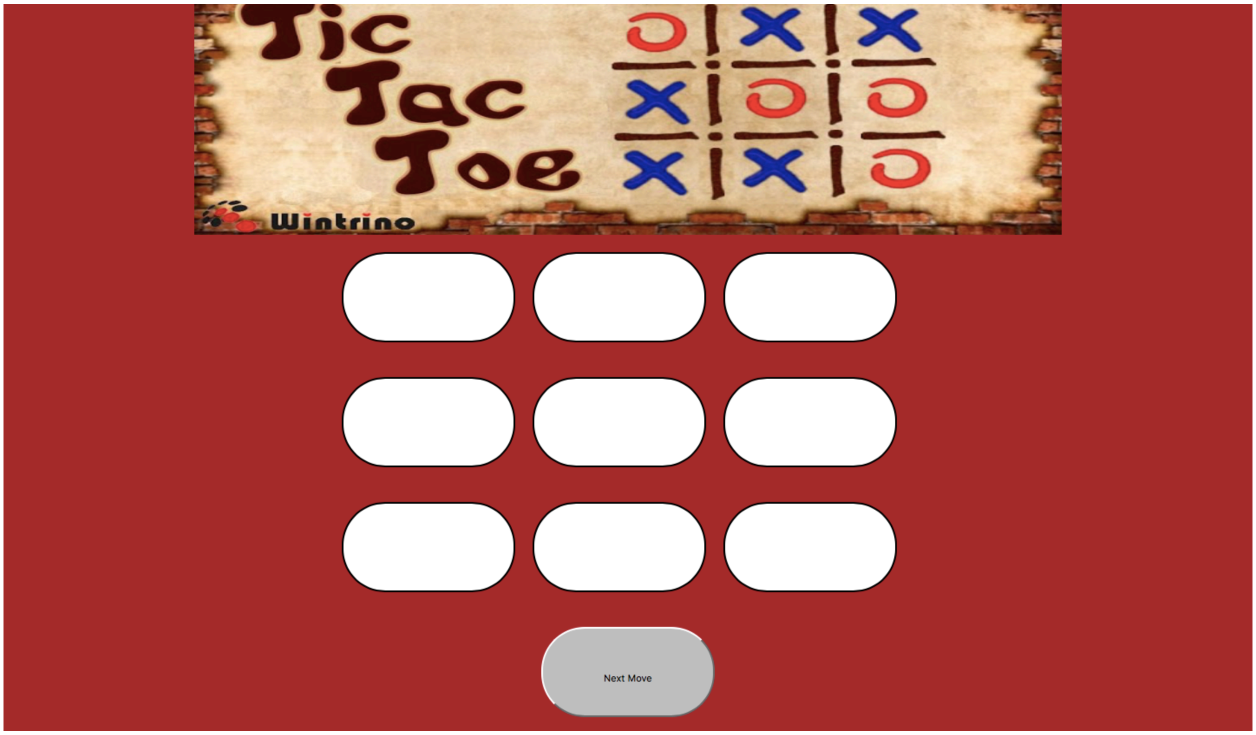
HTML Section:

In the head section of HTML, the title is declared to be "tic tac toe" and a style tag is defined for the body of HTML and the input texts where 'x' and 'o' is present and for the buttons that must be clicked for next move and for restarting the game. The body is given a background color of "brown" and aligned to be "center". The input boxes are curved, and sizes are defined, also the same is done for the buttons.

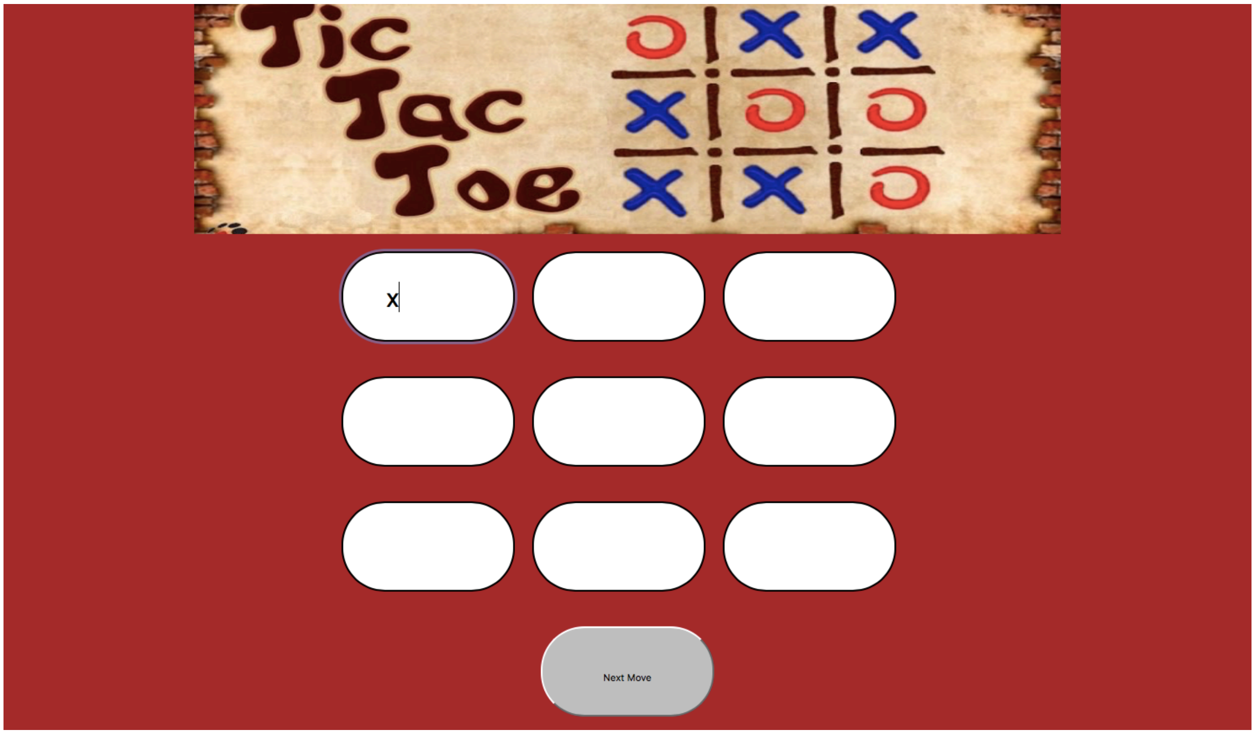
In the body section, to enhance the presentation of the game, a banner of a tic tac toe is given as an image source and added using the <img> tag. The form is placed in a <div> container and is called tic tac toe with the method to be post and action as itself. Next the grids are formed and in order to get the format of a 3x3 matrix, after every third box, i.e., at 2,5 and 8 there is a break. Next, if the winner remains to be "n" then the button remains as the next move button, on clicking which the computer makes its next move otherwise, if the game is over after the declaration of a winner or if the game is tied, then the button changes to a Play Again button that refreshes the page.

3.3 Screenshots

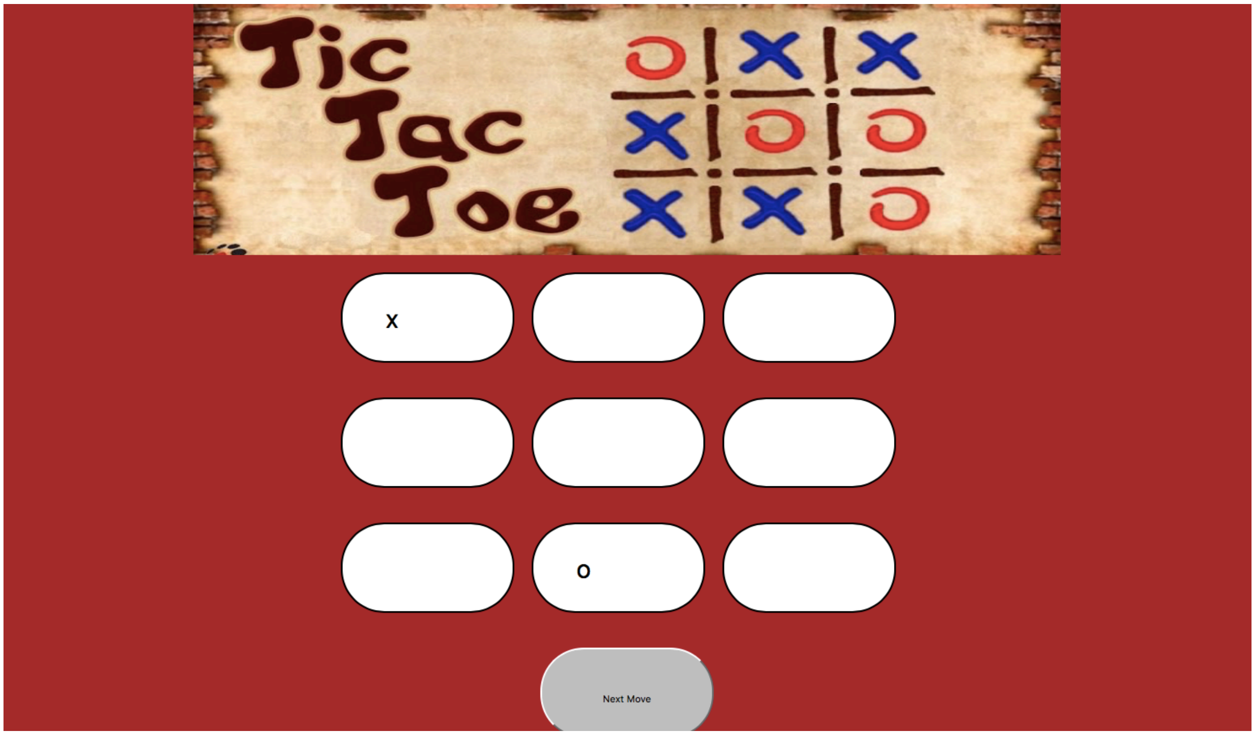
Once the index page is opened, the page will look like the screenshot below



The user then enter 'x' in any position he wants. For example, in the first position below

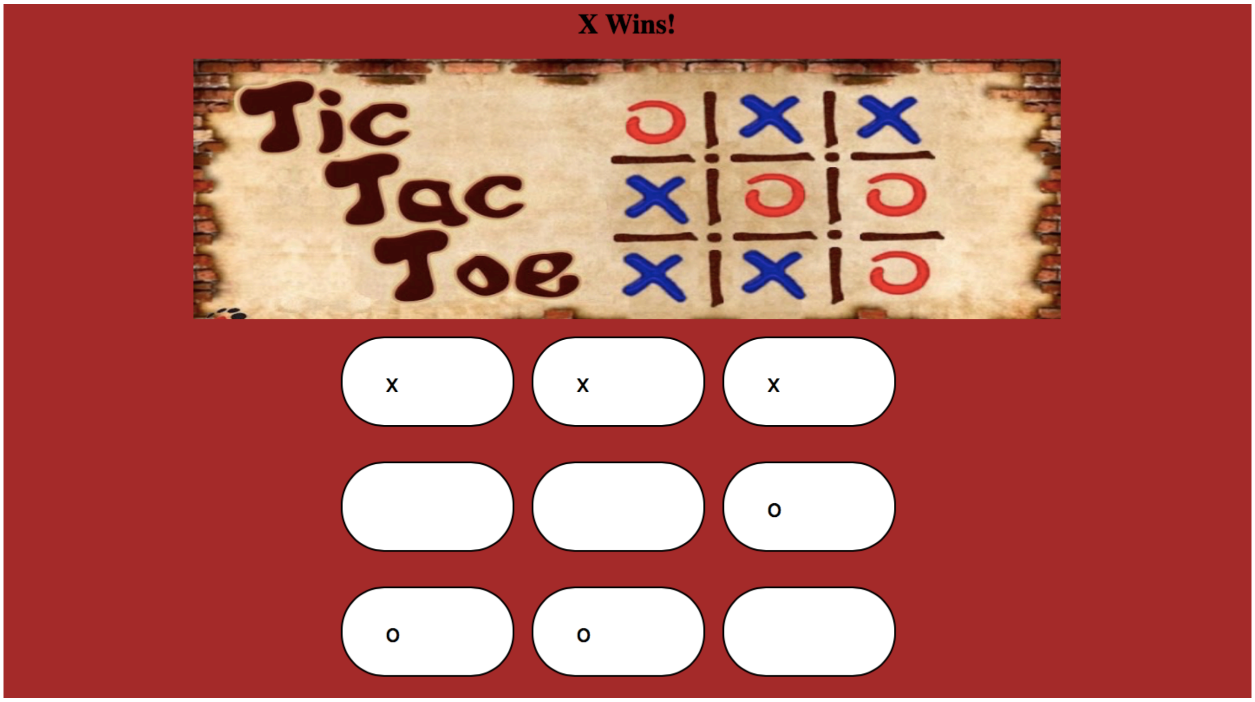


Then the computer enters ‘o’ in any position other than position 1 as shown below



x		
	o	

In the Event the X wins the game after playing, the page looks like so


X Wins!

x	x	x
		o
o	o	

4. Future Work

This project can further be extended by implementing machine learning on the backend of the system, so that the program automatically learns from different possibility how to win the game. Reinforcement learning in the machine learning field can be used to do it with minimum training data set.

5. Conclusion

To conclude, in this project, creating a game using a diverse and simple language such as PHP was attempted. This helps widen out the used for PHP, away from just web development. The game picked to be implemented was Tic Tac Toe as it is a game that is immensely popular all over the world.