Course:

CIS-17B

Title:

Online Connect 4

Assignment:

Project Reports

Due Date

June 9th, 2024

Authors:

Aleksandar Videv, Amare Terrell, Albert Ruiz, Anthony Nguyen, Cristian Magana, Francisco Sanchez, Janaye Jackson,

Kelby Knight, Kyle Riebeling, Patrick Pascual,

Rafaan Hyder, Ryan Westfall.

Table of contents:

[**File Layout 2**](#_1lexl9rm698e)

[**Introduction 3**](#_wauin64jrg7j)

[What is Connect 4? 3](#_1tksr6ycw5xf)

[Rules: 3](#_302670r0d3w3)

[**GANTT Chart 4**](#_ixxkclap84vh)

[**Pseudo-code 5**](#_ffqkzzrgrno)

[**UML 8**](#_5jpztxhiejxn)

[**Screen Shots 9**](#_o4vm6y49oj7v)

[**How were these features used? 15**](#_u5lze46l6h1e)

[1. MVC - How you delineated your objects 15](#_6p80rdmeu9vk)

[2. Objects - Javascript/PHP - Serialization, Passing info with Javascript 15](#_4u0qavyc7xgo)

[3. Objects to PHP Objects with Cookies 15](#_2bcsjjmr0ai6)

[4. Reading/Writing Files/Local Storage JSON - Javascript 15](#_3avtcw1iykw)

[5. Databases SQL - Identify Entities, Xref, and Enum Tables 15](#_z0ffbalwxw)

[6. Form Validation - Regular Expressions 15](#_m1k5vhhrjvzl)

[7. User-Admin-Login 15](#_e1q1kzq87s61)

[8. Cookies - Sessions - Securing Pages 16](#_ws0es9knx37r)

[**Group Info & Github: 17**](#_httc5a6627qn)

[● Time and Location of Meetings 17](#_dx5q7bro7vkv)

[● List of Members, Responsibilities, and Assignments 17](#_srd3ftslbhsu)

# File Layout

1. **ConnectFour\_V6\_1**
   1. Connect4.js
   2. GameOver.html
   3. allusers.php
   4. dbconnect.php
   5. deluser.php
   6. displaystats.php
   7. edituser.php
   8. index.php
   9. loginhandler.php
   10. playgame.php
   11. registerform.php
   12. registerhandler.php
   13. savestats.php
   14. **SQL Dump V6\_1**
       1. connect\_4\_entity\_user.sql
   15. **includes**
       1. board.png
       2. circle.png
       3. displayfunctions.js
       4. footer.html
       5. gameboard.html
       6. header.php
       7. main.css
       8. red.png
       9. user.php
       10. yellow.png
   16. **nbproject**
       1. project.properties
       2. project.xml
       3. **private**
          1. config.properties
          2. private.properties
          3. private.xml

# Introduction

The game of Connect 4 has been a popular pastime for many years, often played with physical tokens and a vertical board. Our group has developed a digital version of Connect 4 with Web Development including HTML, JavaScript, PHP, SQL, and databases. This version allows players to register an account, log in to that account, and compete against each other and includes an option to play against a computer-controlled opponent utilizing artificial intelligence. The game saves user data and game statistics in local storage and an SQL database, allowing players to keep track of their progress over time. This implementation aims to enhance user engagement through strategic gameplay and persistent data management online.

## What is Connect 4?

Connect 4 is a two-player connection game in which the players take turns dropping colored discs into a vertically suspended grid. The grid is typically 7 columns wide and 6 rows high. The objective of the game is for a player to connect four of their own discs of the same color next to each other vertically, horizontally, or diagonally before their opponent does.

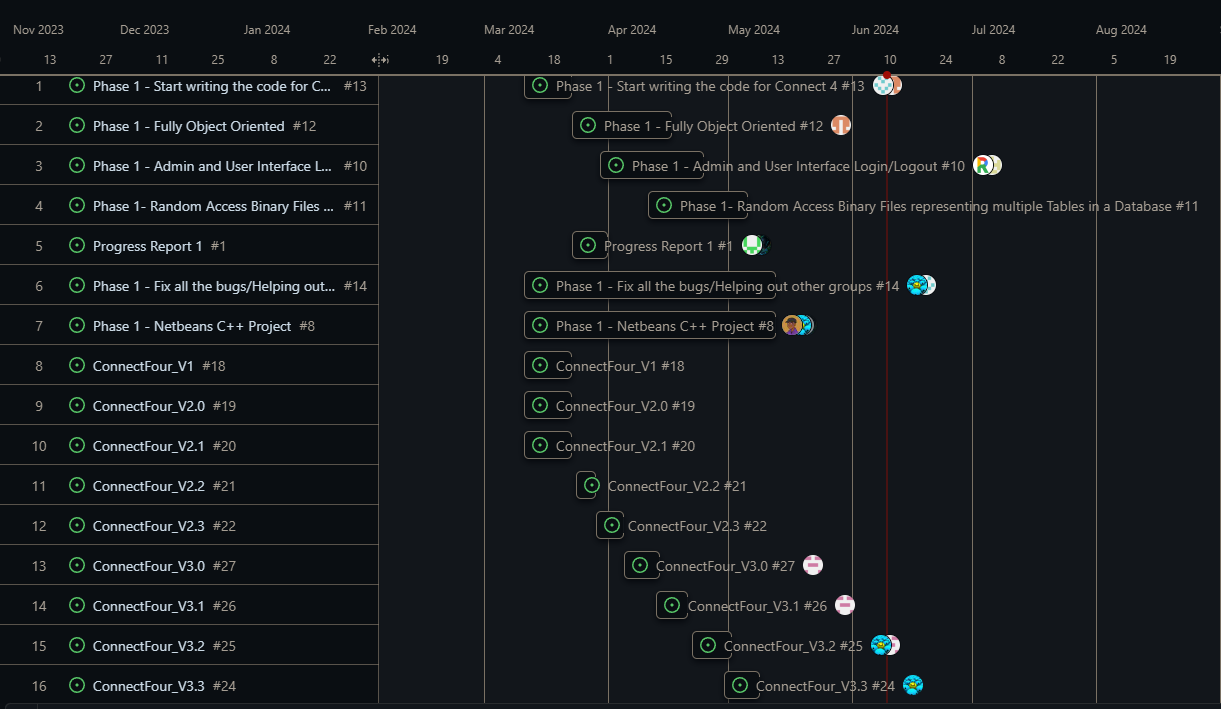
## Rules:

Each player chooses a color for their discs and decides who will go first. Players then take turns dropping one of their colored discs into the grid. The objective is to be the first player to connect four of their own discs in a row, which can be horizontally, vertically, or diagonally.

## 

# GANTT Chart

The full GANTT Chart can be found on [GitHub](https://github.com/users/4mxr3/projects/2/views/5).



# 

# Pseudo-code

// Main flow of the application

START

// User visits the index.php

IF user is not logged in THEN

REDIRECT to login page

ELSE

DISPLAY game options

// User selects to play a game

IF playgame.php is requested THEN

LOAD Connect4.js

INITIALIZE game board

WHILE game is not over

WAIT for user move

VALIDATE move

UPDATE game board

CHECK for win condition

END WHILE

IF game is over THEN

DISPLAY GameOver.html

END IF

END IF

// User can register

IF registerform.php is requested THEN

DISPLAY registration form

IF form is submitted THEN

VALIDATE input with regular expressions

SAVE user data to database

SET user session

REDIRECT to game options

END IF

END IF

// User can log in

IF loginhandler.php is requested THEN

VALIDATE login credentials

IF valid THEN

SET user session

REDIRECT to game options

ELSE

DISPLAY error

END IF

END IF

// User can edit their profile

IF edituser.php is requested THEN

DISPLAY user data

IF form is submitted THEN

VALIDATE input with regular expressions

UPDATE user data in database

REDIRECT to game options

END IF

END IF

// Admin can manage users

IF allusers.php or deluser.php is requested THEN

IF user is admin THEN

DISPLAY user management options

HANDLE user addition, deletion, and updates

ELSE

REDIRECT to game options

END IF

END IF

// Save game statistics

IF savestats.php is requested THEN

READ game data

SAVE game statistics to database

END IF

// Display statistics

IF displaystats.php is requested THEN

QUERY database for statistics

DISPLAY statistics

END IF

// Connect to the database

INCLUDE dbconnect.php

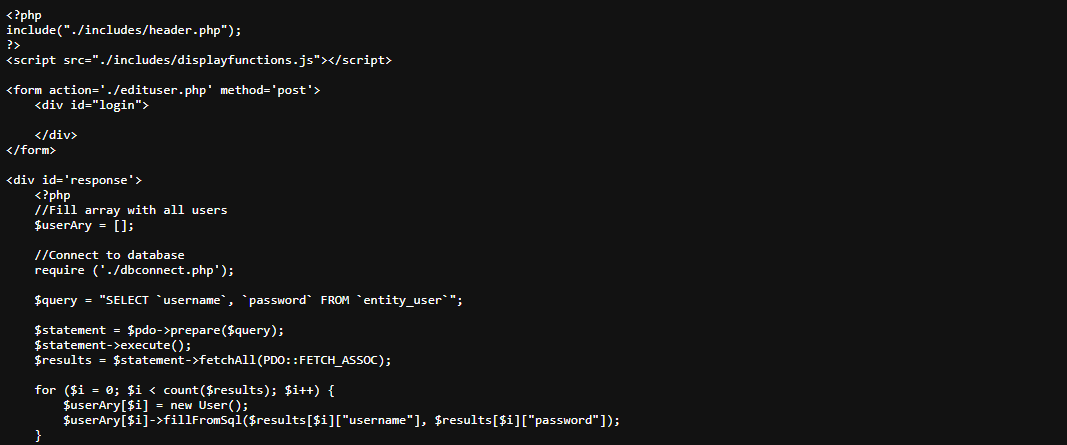
END

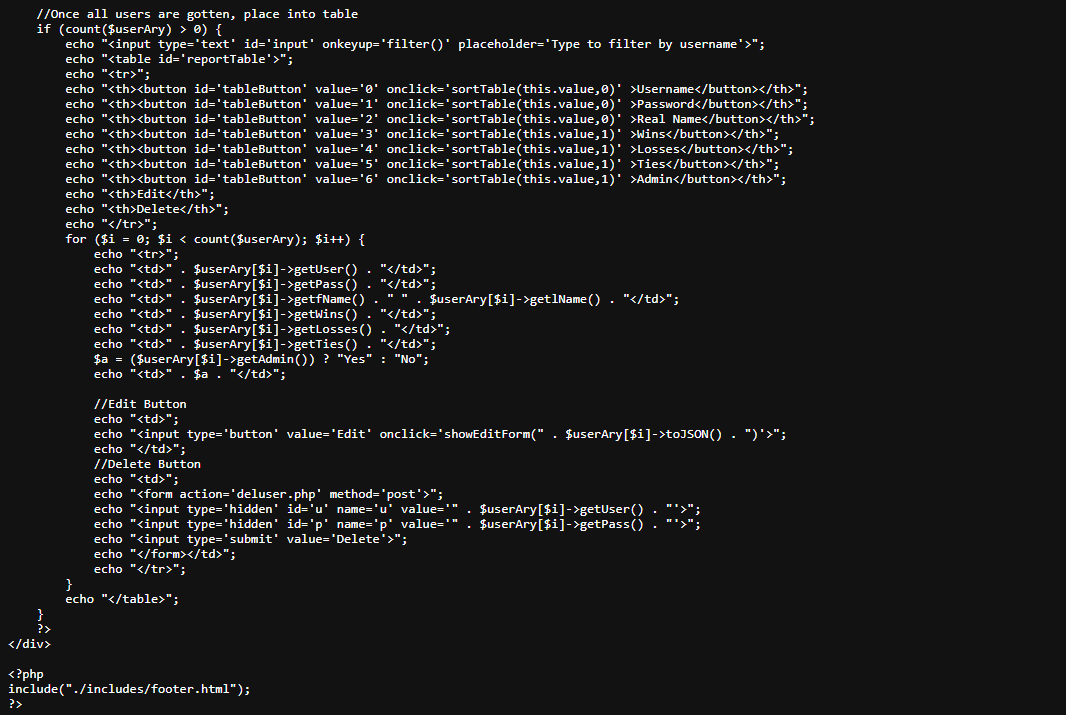
# UML

# Screen Shots

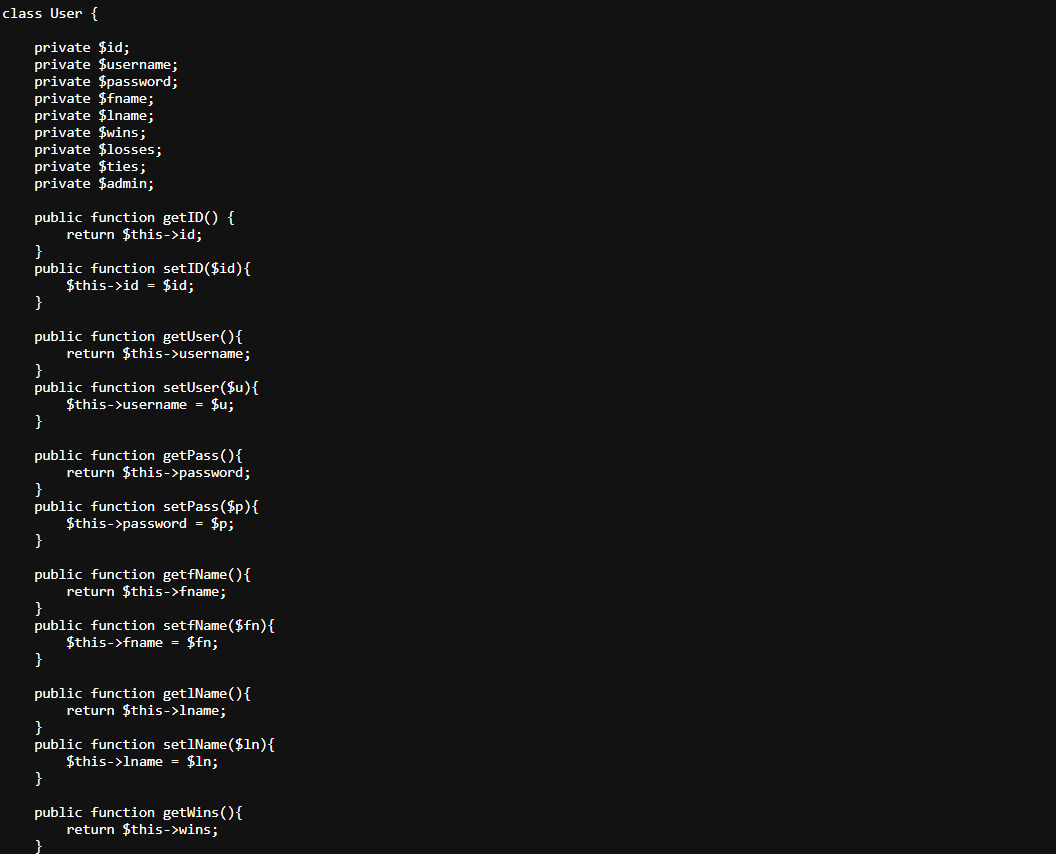
#### Index.php

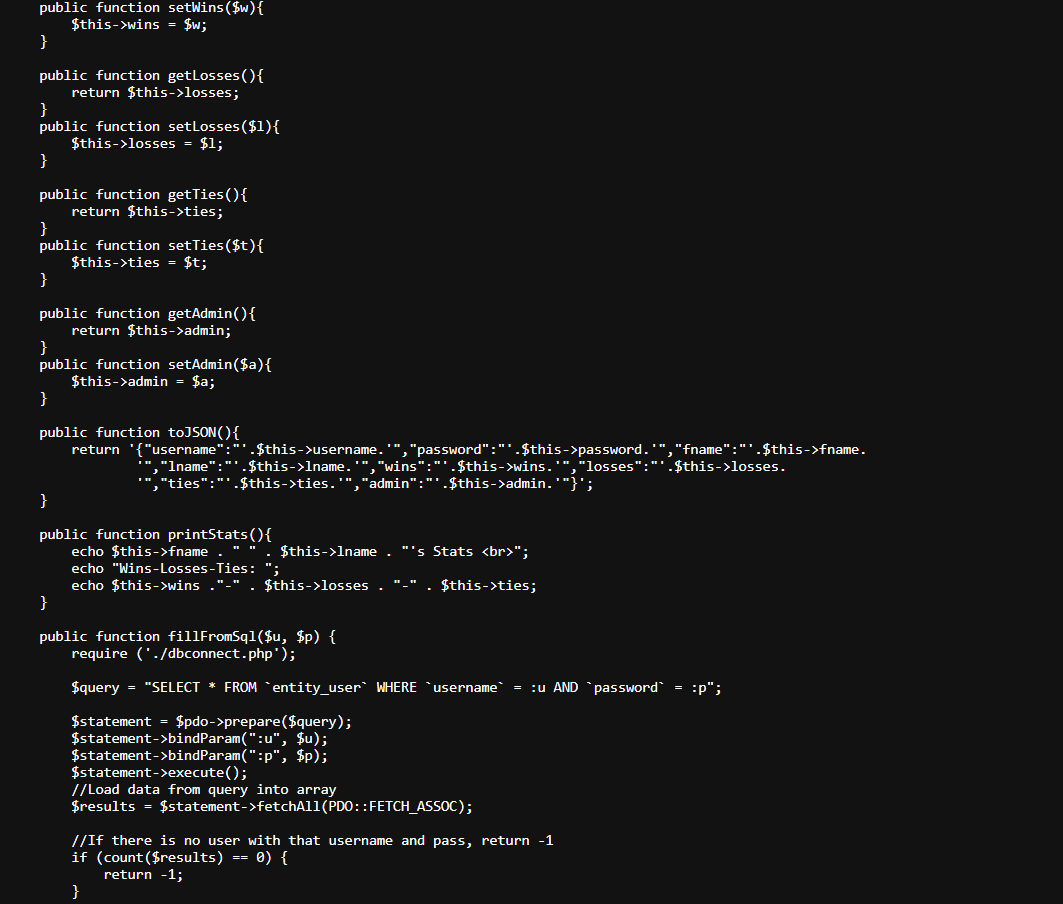
#### Allusers.php

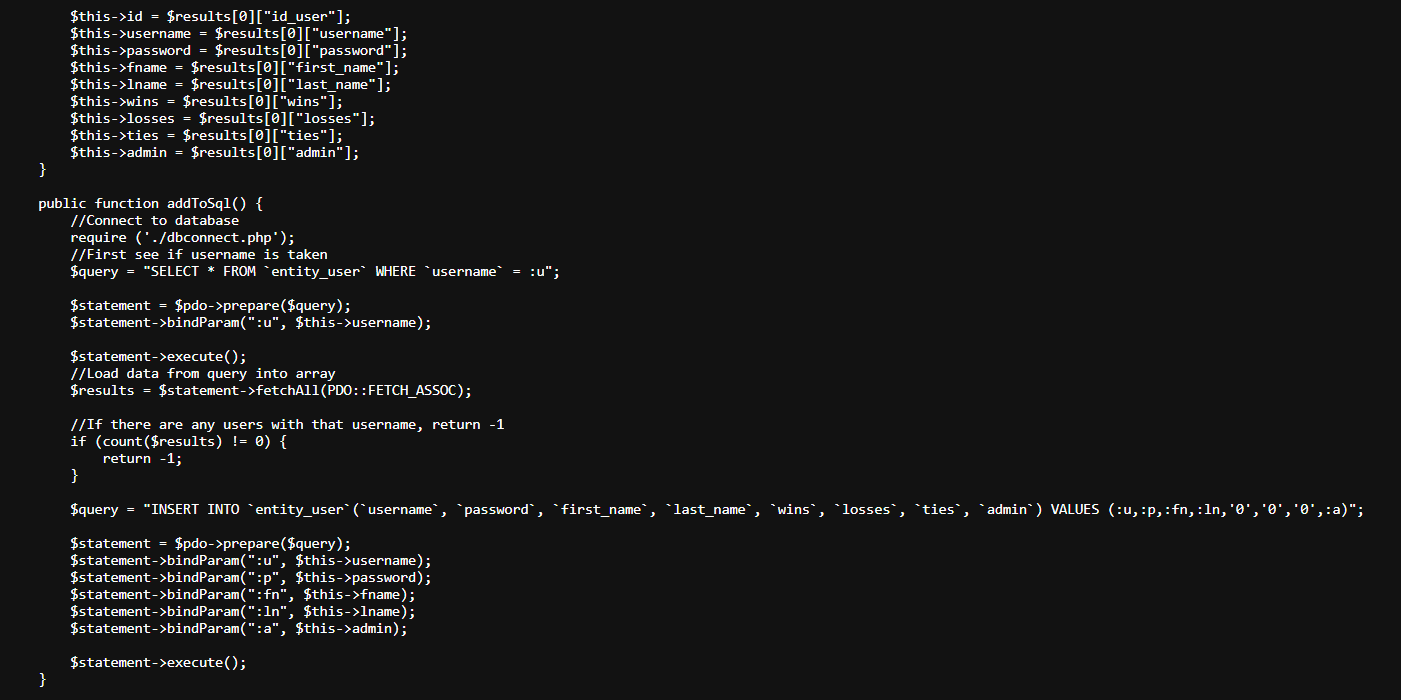


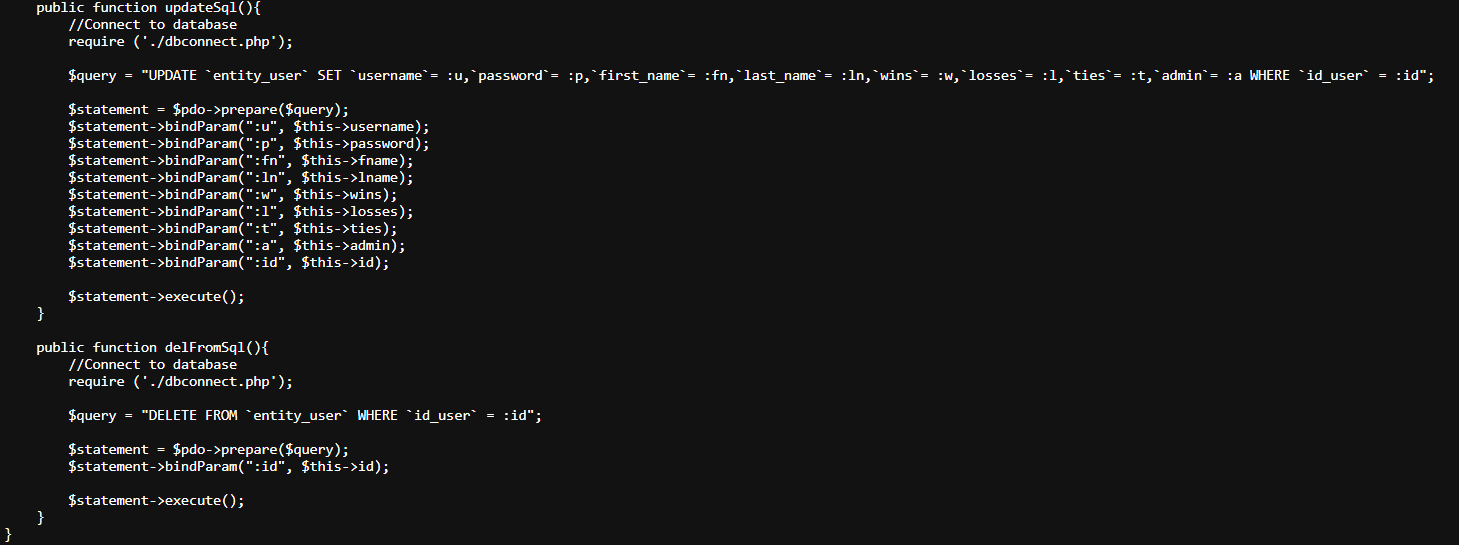


#### User.php

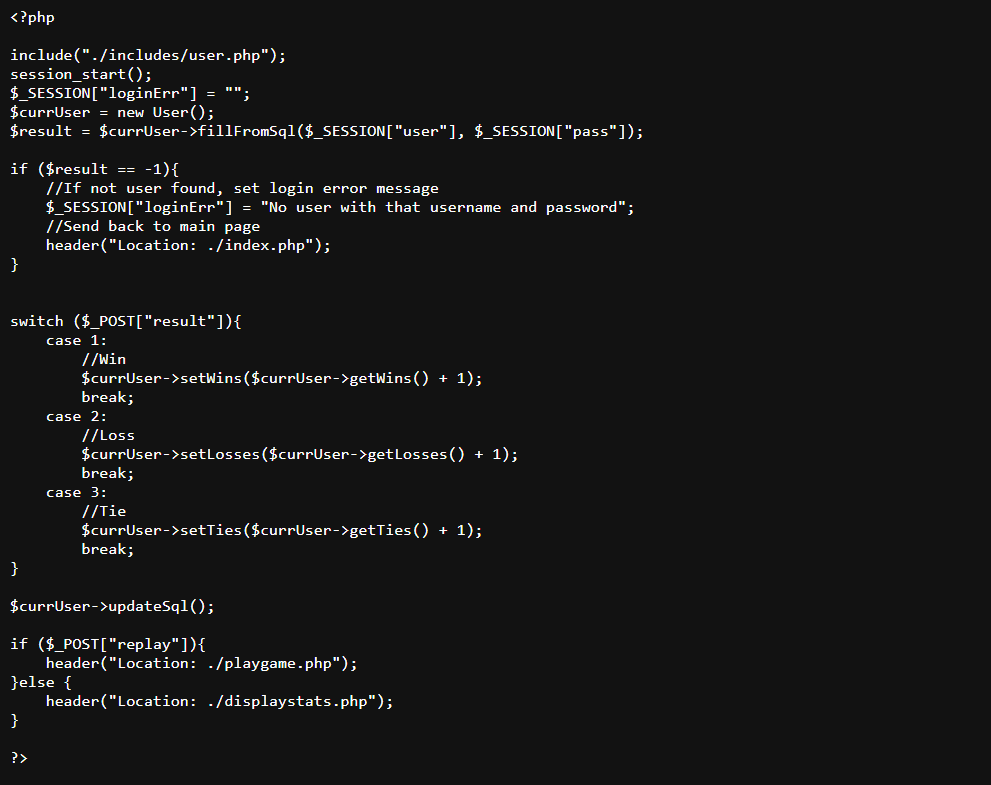






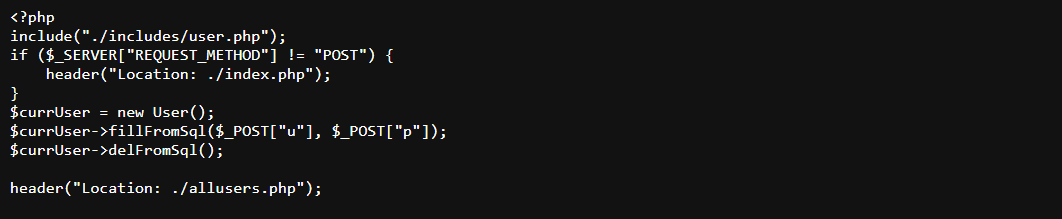


#### Savestats.php

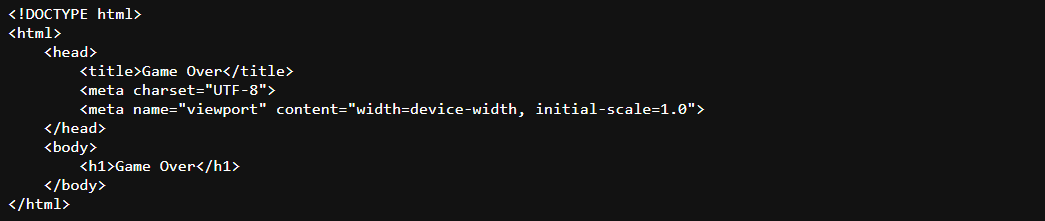


#### Displaystats.php

#### Deluser.php



#### GameOver.php



# How were these features used?

## MVC - How you delineated your objects

* 1. The project's objects are delineated by placing each object into its own .php file. In addition, the javascript functions are kept in their own file folder with easy access to their needed files.

## Objects - Javascript/PHP - Serialization, Passing info with Javascript

* 1. Information is passed from PHP to the SQL Database using the “$\_SESSION” variable, which fills data from the SQL Database when; for example, printing the user’s stats.

## Objects to PHP Objects with Cookies

* 1. The PHP Objects that save data write data to the SQL Database and redirect the user back to the main page.

## Reading/Writing Files/Local Storage JSON - Javascript

* 1. Each PHP file handling users and game data will read and write to the SQL Database progressively, or as the data is produced. Unfortunately, game data is not able to be saved in the Database because it does not accept arrays, which is where LocalStorage is used.

## Databases SQL - Identify Entities, Xref, and Enum Tables

* 1. The SQL Database is handled by the “connect\_4\_entity\_user.sql” in the SQL Dump Folder. This essentially passes the user-entered and validated information to the SQL Database. Default values of the length of each variable are outlined here. Sample users are placed here for reference.

## Form Validation - Regular Expressions

* 1. Form Validation is handled by “registerform.php” which goes through the process of validating if the information from the user entered into the form is correct.

## User-Admin-Login

* 1. The User-Admin Login is handled by “registerhandler.php”, “loginhandler.php”, and “allusers.php” header files. These files are separated into handling the registration of an account, logging in to an account, passing the received information to the SQL Database, and creating the sample users that have already been registered in the system.

## Cookies - Sessions - Securing Pages

* 1. Cookies are used to store the login information, username, and password, of the user to the device they are playing from.

# Group Info & Github:

### **Time and Location of Meetings**

* To coordinate our schedules efficiently, we set up a Google Calendar. This helped us identify the best times for all members to meet. The calendar is accessible [here](https://calendar.google.com/calendar/u/0/r?cid=NmY3MGFhOWM3NjJhYjgwMTdhMzg1Yjg1MjkyYzU5MDIwOWEyZjMxZjhhYjVhMDllNDljMTg4ODkwNTUwNDkxNkBncm91cC5jYWxlbmRhci5nb29nbGUuY29t). Based on our availability, we decided to schedule meetings twice a week, with flexibility depending on our workload.
* **Scheduled Meeting Times:**Tuesdays and Sundays from 5 PM to 12 AM

### **List of Members, Responsibilities, and Assignments**

* **Aleksandar Videv:** Managed the project using a Gantt chart and maintained GitHub. Developed version 3.3, addressing input and menu errors, and version 3.4, improving AI with minimax. Contributed to the documentation.  
  I worked on V7 - V10 by improving the design, creating JavaScript validation, and fixing bugs.
* **Amare Terrell:** Update the Github Project to stay organized (GANTT Chart, To-Do List), Assistance with Documentation, and later work on bug checks for the Javascript and PHP portions of code.
* **Anthony Nguyen:** Documentation and creation of UML/FlowChart
* **Cristian Magana:** Helped out with the project
* **Albert Ruiz:** Assisted with AI and user interface
* **Francisco Sanchez:** Added the menu in v2.2 and implemented the player vs player mode and moved some logic to the Game class to keep main.cpp cleaner. Fixed the output bug where the board was printing twice for AI and player,Helped with some documentation.
* **Janaye Jackson:** Reconfigured the already written code in V1 into object-oriented programming, created the AI, and added more conditions to make it more complicated. Debugged some logic I wrote that was inconsistent. Also converted the project into basic javascript, have not implemented database or user/admin.
* **Kelby Knight:** Organizing group meetings, Documentation, comments,php
* **Kyle Riebeling:** V6 and V6\_1: Created User.php class, User/admin login and registration, site navigation with dynamic header using php session variables, admin utilities, table sort and filter functions, accessing, deleting and updating from an SQL database
* **Patrick Pascual:** Set up initial base codeline, or first verson as well as helped with the documentation of the code.
* **Rafaan Hyder:** Helped out with the project
* **Ryan Westfall:** Documentation.

Github Link: <https://github.com/4mxr3/Connect4ON>