

---

# FINAL REPORT

---

## 1 Introduction

### 1.1 Problem Description

This is a web site where users can play small web games and developers can deploy their small games. To better illustrate how the website runs, I wrote a sample game named connected 4.

The online game is based on a tradition game called connect 4. The user(player) must place the game pieces to a typically 6-high-7-wide game board. The winner is whoever place 4 consecutive pieces in the same row or column or triangle. This game is already deployed on the website and can be run in a browser.

### 1.2 Objectives for Project

- Design a web site where users can play small web games and developers can deploy their small games.
- Design a connect-4 game to deploy on the website

### 1.3 Overview of Approaches Taken

- JQuery to select elements in HTML and make http request to the server
- JavaScript animation
- HTML 5 features
- CSS
- Bootstrap
- Responsive Design

### 1.4 Organization of the rest of the report

#### Contents

1	Introduction .....	1
1.1	Problem Description .....	1
1.2	Objectives for Project.....	1
1.3	Overview of Approaches Taken .....	1
1.4	Organization of the rest of the report .....	1
2	Detailed features of the product design.....	2
2.1	Responsive Design .....	2
2.2	User Authentication .....	2
2.3	Game .....	2

3	Detailed features developed.....	3
3.1	Responsive Design .....	3
3.2	User Authentication .....	6
3.3	Game .....	6
3.4	Memo.....	8
4	Assessment of how well approach works for problem.....	9
5	Appendix: Any details about the GUI, codes and other information about the product. ....	9
6	References .....	9

## 2 Detailed features of the product design

### 2.1 Responsive Design

With JavaScript, the web site should be super responsive to user. To keep page to clean and easy to use, we should hide unnecessary information and provide several ways for users to guide themselves.

We should also provide ways to let user get where we want them to go. For example, the index page is show how easy it can be for developers to deploy their game on our website. Therefore, after each bander, we should provide a button or link so that developers can easily give us the feedback, which is the memo area.

### 2.2 User Authentication

The original plan is to use a local passport strategy used to do the authentication, which has changed to reuse the User group in the native Django SQLite to improve the code reuse.

### 2.3 Game

For the game part, the previous version is shown as Figure 1 .

The game will also provide a scoreboard to show the top winners of this game. The way to display is not ceratain. It can be another page or a side table.

The game should also provide a way for players to submit their score.

By the way, due to the time schedule, I probably won't do a lot of security issues, which specifically, I will pass the entire game script to the client rather than holding a running session from the server end. However, it breaks the security, which shouldn't really happen in a real world.

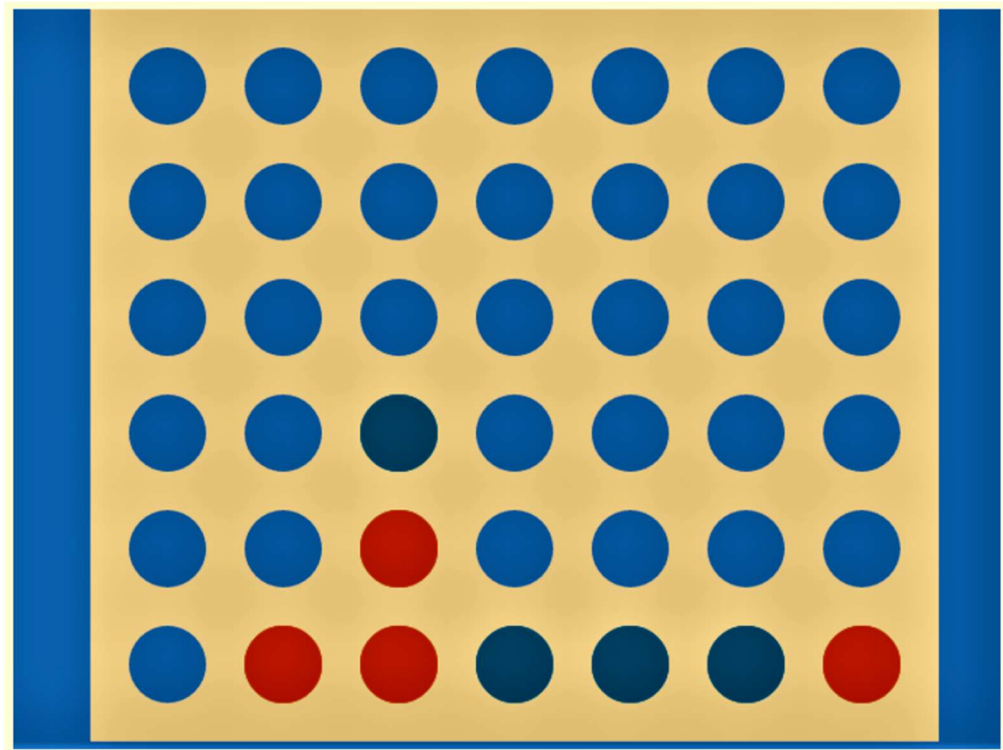


Figure 1 Previous Version of Connect 4 Snapshot

### 3 Detailed features developed

#### 3.1 Responsive Design

The index page contains four sections, namely “Home”, “Services”, “Portfolio”, “Contact”.

The idea is that if a user wants to contact us for his game product, we should let him contact as soon as possible. Therefore, we provide a button “Contact us” at the end of every section. And a button “to-bottom” is also added.

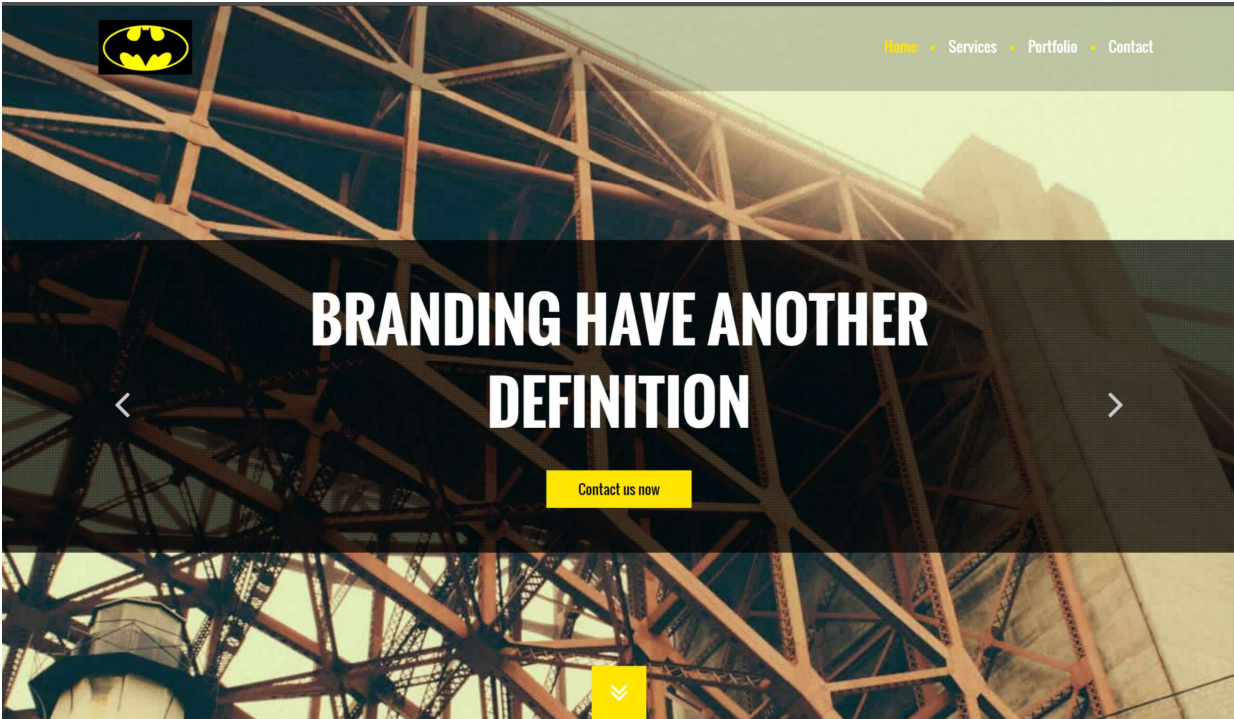


Figure 2 index-home

As shown in Figure 3, the index page has fixed-to-top navigate bar that contains for bookmarks to help users browsing the web page.

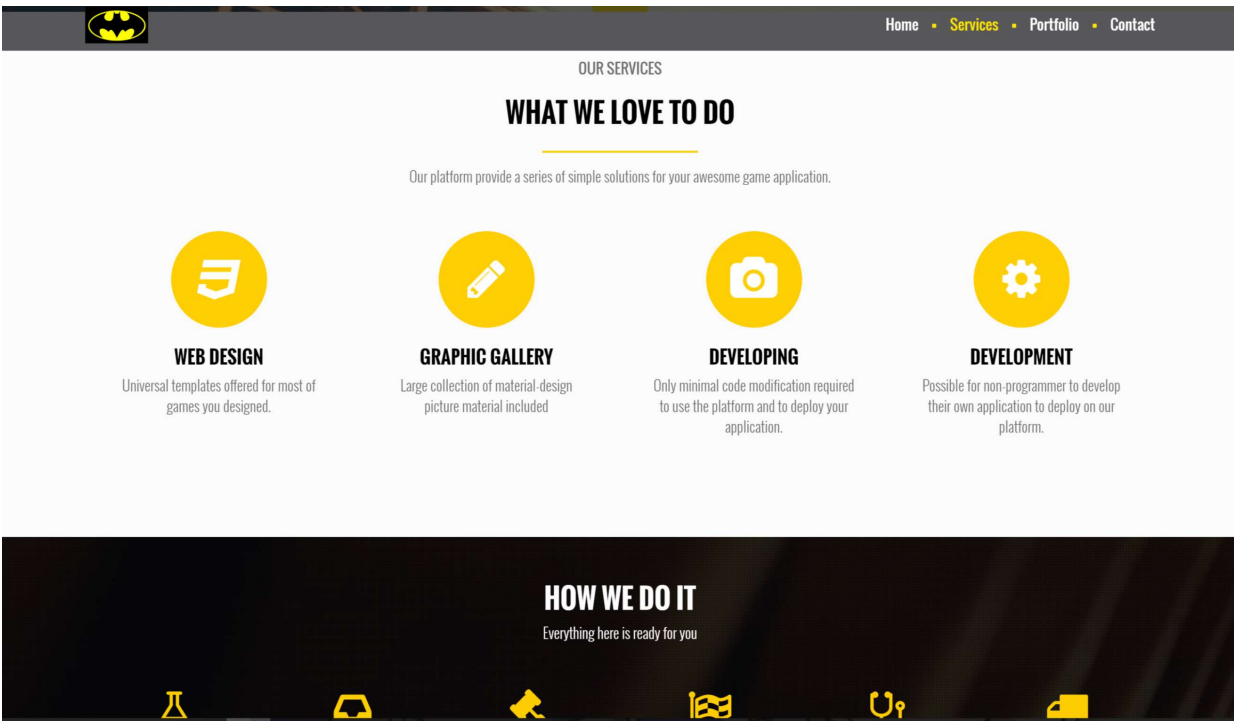


Figure 3 index-Services

The Portfolio section offer several samples. Due to limited time, I only developed one and duplicate it 12 times to fill the void. It's a bit awkward here now, but works.

One good design is that once a user click on a game, it will show the detain information as Figure 5 does.

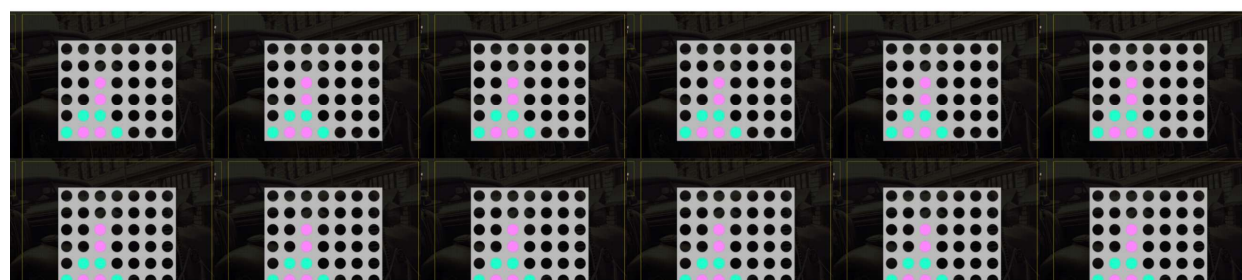
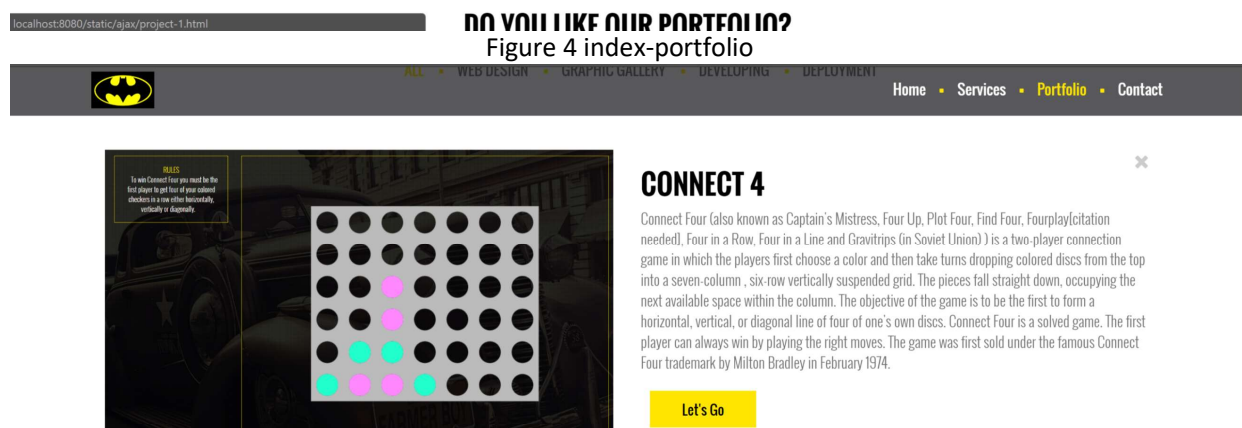
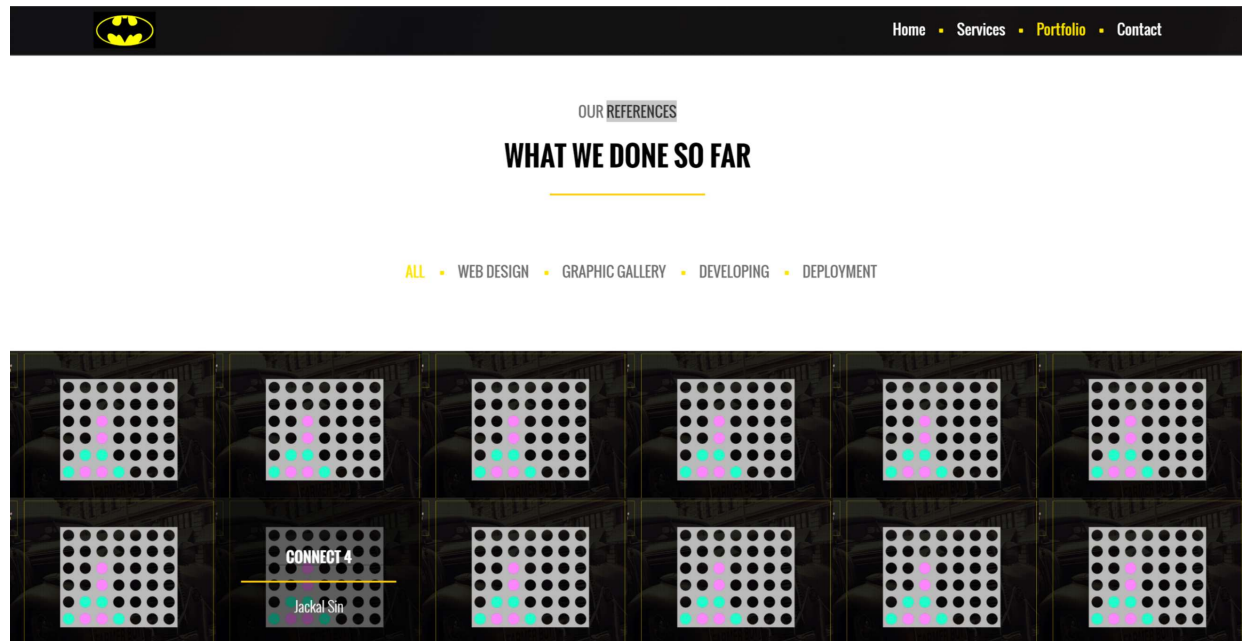


Figure 5 index-portfolio-



Contact section is simply a memo. Users can submit their feedback and communicate with administrator directly. The website contains validator in every field, only a valid format field value can be passed to the server and we have server end validation to prevent some user directly sending request to do injection attack to our server.

5000 Forbes Ave, Pittsburgh, PA 15213

zxin@andrew.cmu.edu

+1 (412) 551-2612

SEND US A MESSAGE

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer et viverra ipsum.

Name

Email

Phone 1234567890 or 123-456-7890

Your comment..

Send message

Figure 6 index-contact

## 3.2 User Authentication

The first version of user authentication is discarded to improve the code reuse. Currently, I use the native User table in Django. The main reason to do this, I developed another system to let user to submit their current score, so there is no need to let user login to submit again. Besides, due to there are some private information in the memo, only allow localhost administrator will be a more reasonable to view all the memo. The style of the feedback also changed from index-like to the administrative-page-like.

## 3.3 Game

When you click button “Let’s go” in Figure 5, it will lead the user to the game connect4 page. It has the same fix-to-top navigate bar as the index uses.

There are mainly three buttons below, namely rules, wiki links and score board. Clicking Rules will give the user rules to win at the left column; clicking wikilink will lead the user to Wikipedia; clicking scoreboard will display the top winner of the game.

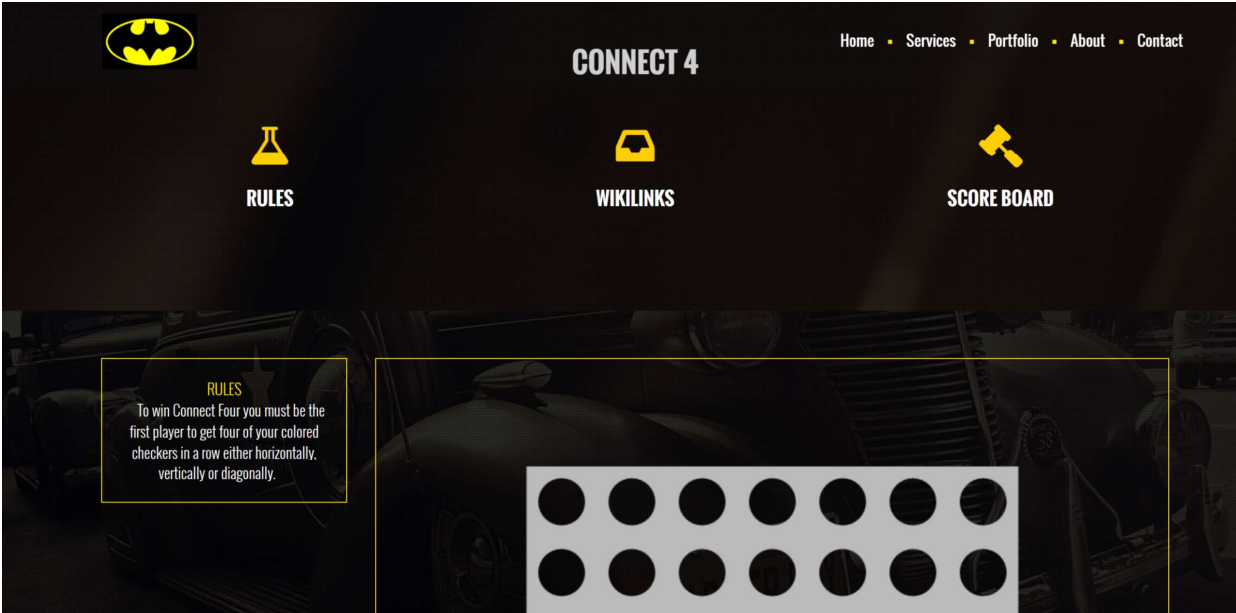


Figure 7 game-rules

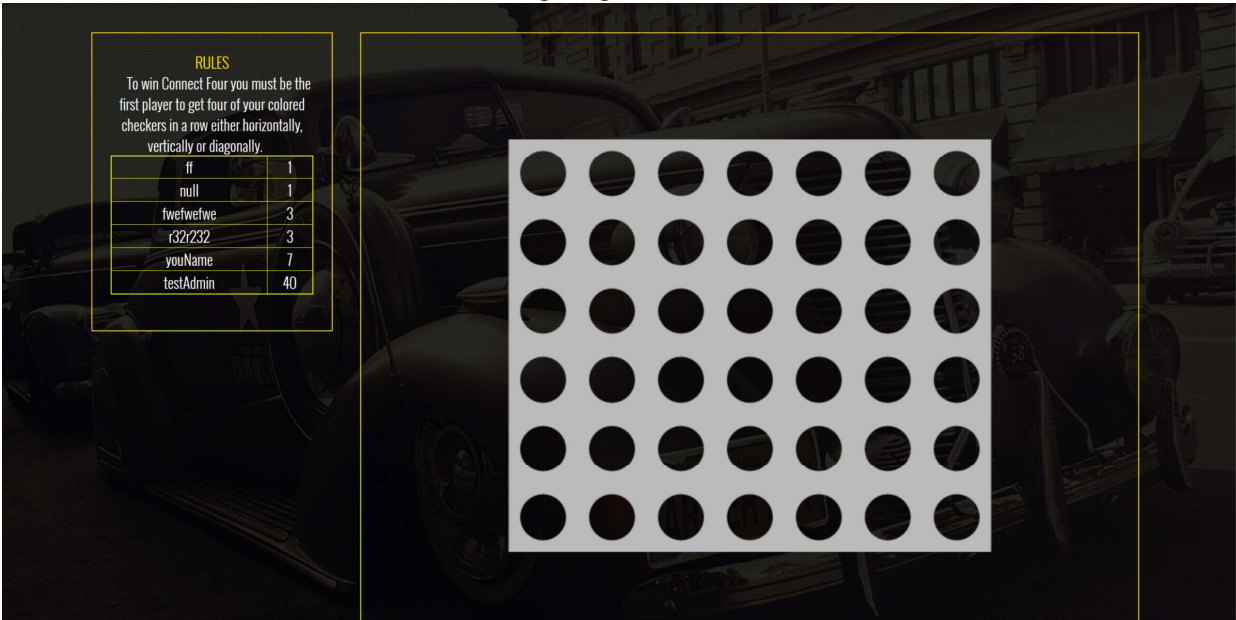


Figure 8 scoreboard

To play the game, the user can simply click any column on the board. After the user’s clicking, the computer will automatically place another one. When the user loses, he can click to restart; if he wins, he can send his score and name to the server to record.

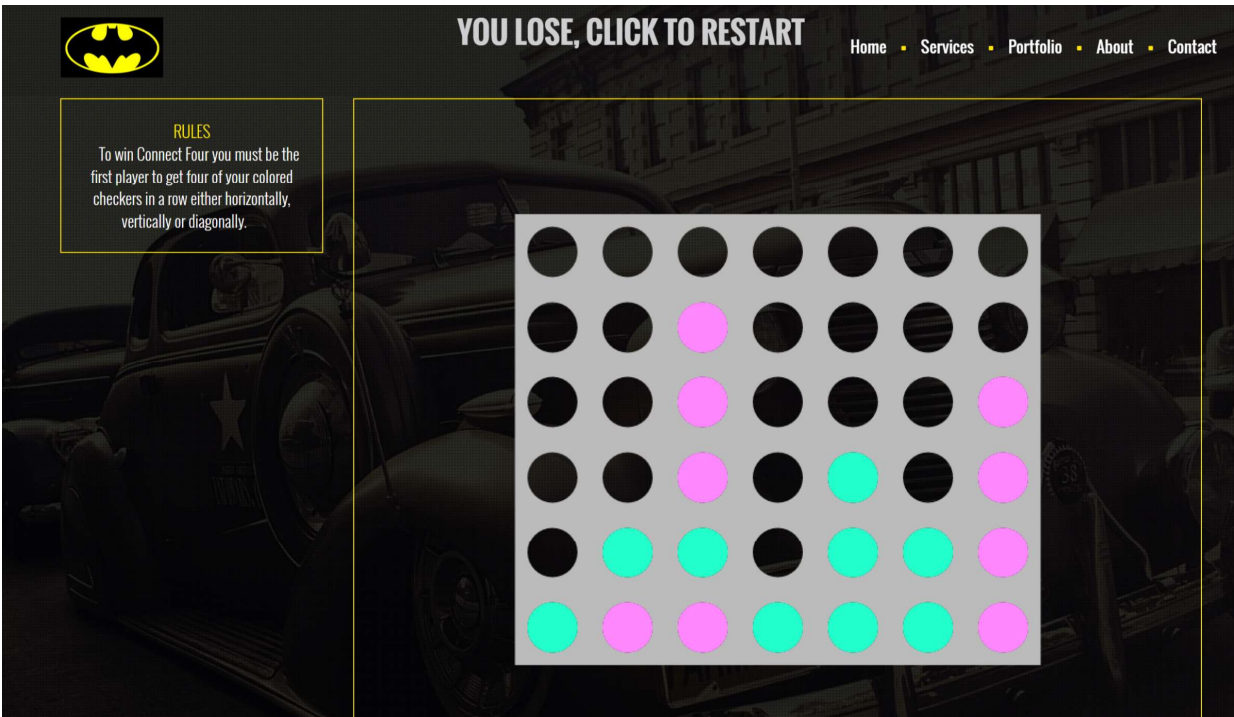


Figure 9 You Lose

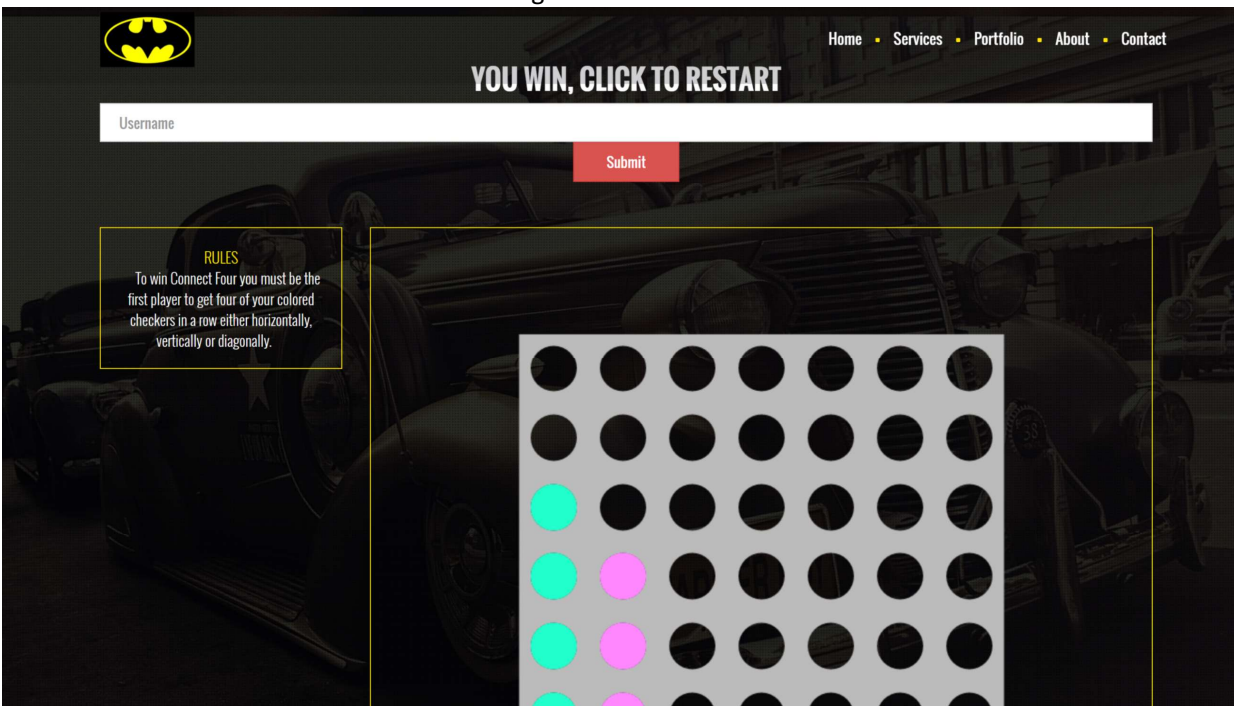


Figure 10 You win

### 3.4 Memo

If a develop find the website is interesting and want to give us feedback and deploy their own app on the website, he can send us a memo in Section Contact.



The administrator can view these memos after logged in. The memo page is shown as below.

The screenshot shows the Django administration interface for viewing private memos. The header is blue with 'Django administration' in yellow, and 'Admin' and 'Memo' tabs. A 'Welcome JACKA' message is on the right. The main content area is titled 'Private Memos'. It lists three memos:

- Name**  
email@email.com  
1234567890  
your comments
- Jackal**  
jackal.sin@hotmail.com  
412-551-2612  
无可奉告
- JackalSin**  
jackal.sin@hotmail.com  
412-551-2612  
无可奉告

Figure 11 Administrative Memo

## 4 Assessment of how well approach works for problem

I think it works but only in can-do level. There are several reasons.

Firstly, the game record is not securely checked, which means users can send the record himself. This is a big drawback. However, due to the requirement of this project, I must use http protocols instead of socket, which I think is more proper under this circumstance.

Secondly, the memory usage is beyond my control. At first, I use different bookmarks to separate different sections where it should be a different page. The way I implemented it was popular when Bootstrap was just released. Since I never do it in this way, I just do it. Besides, I used a lot of animation combined from 2 templates from Internet. Way to slow on a 4GB machine when using Chrome.

## 5 Appendix: Any details about the GUI, codes and other information about the product.

## 6 References

Some of the animations and css files are from graphberry and TemplateMonsters.

<http://www.graphberry.com/item/montreal-creative-portfolio-theme>

[https://en.wikipedia.org/wiki/Connect\\_Four](https://en.wikipedia.org/wiki/Connect_Four)