

# Coursework Report

Bradley Jones

4016745@napier.ac.uk

Edinburgh Napier University - Advanced Web Technologies(SET09103)

**Keywords** – CSGO, CSS, HTML, Napier, SET09103, 40167459, Advanced Web Tech, static, templates

## 1 Introduction

**Overview** The purpose of my web app was to document all the current official Counter-Strike: Global Offensive game types and give a brief description of each game type.

**Homepage** The homepage is the first page the user sees. It has a light blue background, since it is often used to indicate calmness in design. This gives the user links to the other pages of the web app, each game type is linked under the appropriate heading.

### Counter Strike Global Offensive Game Type List

This is a list of all Counter Strike: Global Offensive game types, as of the most recent patch (10/18/2017).

#### List of Game types

Arsenal: Arms Race (ar)

Arsenal: Demolition (de)

Hostage Rescue (cs)

Bomb Defusal

Deathmatch

Figure 1: **Homepage** - A screenshot of the homepage

**Game Type Page** The game type page is the other style of page used in the development of the web app. Its design is replicated by the other game types, a good example of this type of page is the "Armsrace" page gives the user a description of the game mode, including a brief description of its origins and the rules of the game type. It carries a similar design to the homepage, with the light blue background and the use of a sans-serif type face. However unlike the homepage, this type of page has a screenshot from Counter-Strike: Global Offensive (CSGO) Showing the game type in question.

## 2 Design

### 2.1 Homepage

The homepage was the first page designed, and as such, it's design influenced that of all the other pages in the Web App. The design of the homepage is rather minimalist, with it only having a header which links back to itself and the links to the other game types. As previously mentioned light blue was chosen as it is regarded as a calming colour, however it is also regarded as a depressing, sad colour. During the design process Arial was chosen as the font of choice. A sans-serif font



Figure 2: **Armsrace** - A screenshot of the armsrace page

was chosen because the strokes are a near uniform width, keeping the font readable when reduced in resolution or reduced in font size. Since Counter-Strike: Global Offensive is a frequently changing game, a sub-heading mentioning how the number of game types is accurate as of the most recent patch, which came out on the 18/10/2017 was added underneath the main header.

### 2.2 Game Type Page

The second page designed was the game type page, all the pages of this type are the same, so "Armsrace" can be considered a sample page. As mentioned before, all pages designed after the homepage took several design cues from it, for example the use of the light blue background colour and Arial as the font of choice. This was done to maintain uniformity among my pages. After the screenshot, there is the page heading followed by a brief description of the game mode, the rules and the game winning scenarios for both teams.

### 2.3 Python

The .py files are where the routing and error handling is carried out. The templates are also loaded within the home.py file. The template files are named very similarly as the actual routes themselves. The IP collection is also carried out within the home.py file.

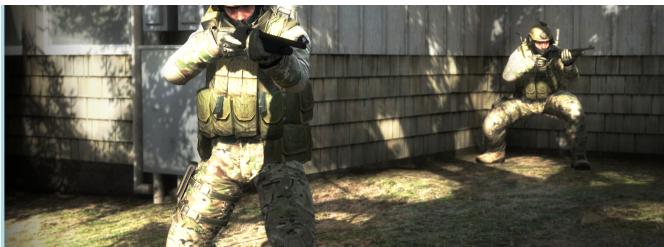
### 2.4 HTML Templates

The reason why I opted to use HTML templates is that it can be rather difficult and annoying to write HTML directly into python. During the development, I opted to create a template for every page. This made it easy to add specific things to certain pages, for example appropriate images.



**Hostage Rescue**  
Hostage Rescue or Hostage Scenario is a game mode available in all of the Counter-Strike Games.

Figure 3: **Hostage Rescue (cs)** - A screenshot of the Hostage Rescue game mode page, highlighting the design uniformity



**Bomb Defusal**  
Bomb Defusal mode, also known as "Bomb Scenario" is considered by the majority of the CSGO playerbase to be the most popular and balanced game mode. For those reasons it's usually the only game mode used for tournaments or other competitive matches.

The round winning conditions for the two teams are:

Figure 4: **Bomb Scenario** - Another page example showing the uniformity of the game type pages

## 2.5 CSS

CSS was used in the design of my website as made it easier to modify the aesthetics of the website, getting it to a state that was more pleasing to the eye. Some examples of how CSS was used were changing the background colour to light blue and changing the font to Arial. A sample of the CSS used is documented below.

## 2.6 Code Listing

A segment of code taken from the style.css file.

Listing 1: style in css

```

1 body{
2   background-color: lightblue;
3 }
4 h1{
5   font-family: Arial, Helvetica, sans-serif;
6 }
7 h2{
8   font-family: Arial, Helvetica, sans-serif;
9   font-width: 50%
10 }
```

## 3 Enhancements

### 3.1 Python

If I had more time I would have liked to implement more varied python. For example a system that logs the amount of times a specific page is visited. Another potential feature I would have liked to implement is a forum, a place where CSGO players could discuss the game types in detail.

### 3.2 HTML Templates

I would have liked to implement if I had more time, would have been to use a generic "Game Type" template for all the game type pages and then have python input all the specific text and images, based on whatever the routing was. This would make it easier to roll out website wide changes as I would only need to change one page. Another feature I would like to have implemented was more web pages, documenting more aspects of CSGO, for example the maps and the weapons.

### 3.3 CSS

There would be a much greater use of CSS to help improve the overall appearance of the web pages. Appropriate colour schemes would have been used and better layout and overall design would have been chosen. Different sizes of display and devices would have also been taken into consideration.

## 4 Critical Evaluation

### 4.1 HTML Templates

During the development of the coursework, the use of templates was very effective as it allowed me to quickly update the web pages. However, there is a definite possibility that they could have been implemented more effectively, for example the use of a "master" HTML Template could have been better than making templates for each page.

### 4.2 Python

The use of python is also rather effective as all the routing and error handling that was implemented works as intended. However I could have implemented a few more features or used more varied python code.

### 4.3 CSS

The use of CSS is definitely the weakest aspect, and despite being functional and working as intended, it definitely could use improvement. The CSS in the current web app is rather bare bones and could be to consider different a layout or a more well defined colour scheme.

## 5 Personal Evaluation

During the course of this coursework, my grasp of python, HTML and CSS have improved. I faced a few challenges

during my efforts to create a functioning web app, like learning how to adapt the information I learned from the work-book to the coursework, but the biggest challenge I faced was when I suffered virtual hardware failure on Levinux. This was due to having to abandon my work in the JKCC due to a fire alarm going off and when I returned I had attempted to reopen my repository containing my coursework and it did not work. Thankfully, I had recently backed up my work to Github meaning that after re-installing levinux and pulling my repository down again, I had only lost about an hour's work. Overall, I feel like I have performed well but should have spent more time on the coursework and more time should have been spent earlier on it before the last couple of weeks.