F28WP Coursework - Edinburgh Campus

Mark: 25% of the overall mark of the course

Due Date: Week xx

A develop an online multiplayer game that that lets players interact and play other people in real-time. Store and view information (e.g., high-scores, number of online active players). Play other people or join other games. On interaction should be communicated to a web server and recorded in server side storage (e.g. MongoDB, MySQL or files), as well as being displayed back to the user side (e.g., leaderboards, other player interaction). The front-end should also let a user list all or some game information that is available (e.g., player’s past scores and other where they sit in the global high scores).

The assignment should be in a group pairs (3-5). Students on the Edinburgh campus who can't find a partner may contact

Benjamin Kenwright by e-mail (**b.kenwright@hw.ac.uk**) for help in doing so.

The following Web technologies should be used:

* HTML 5 (XHTML syntax), CSS 3, DOM 1+ and JavaScript
* Server side technologies and Database

The project will examine web technologies and algorithms to allow you to develop a complex web-based system (emphasis on web development best practices).

Objectives

* Design and implementation web-based `multiplayer' game
* Theory and practice of server-side scripting and SQL database for web games
* Code modulation in Javascript
* Techniques to implement best practices in web development such as Test-Driven -Development, Version Control and Javascript documentation API
* Minimum Viable Product (MVP) and testing

This coursework requires you to spend your time programming and doing practical development work. The coursework aims to develop a web-based skills through a complex web-based solution (interactive online multiplayer game) combined with the techniques covered in lectures and labs.

* Design and develop a sufficiently complex web-based game using advance web techniques such as design pattern and modularity
* Efficiently manage and document the code using appropriate software tools and techniques
* Incorporate back-end services into the game for asset management and game-state management
* Use appropriate web technologies to comply with best practices

# Development environment

Access to a web server for online publication, access to a software version control system (GIT), access to a limited range of web services (e.g. a high-score service, game-state management services)

# Date Due

The coursework for Edinburgh students is due by:

1. Version control (regular checking, show team work, ..)
2. How you manage security (e.g., cheating)?
3. Reliable connection? (e.g., delays, intermittent spikes, ..)
4. Scalability (5 or 5000 connections/players)
5. Store and display the game statistics to players (e.g., number of live players, )
6. Complexity of the software
7. Testing (bugs, user interaction, software reliability, …)
8. Browser compatibility (Chrome, IE, Mobile, Desktop, …)

Criteria

* 2D Game (i.e., NOT 3D)
* Interactive and animated (move around/explore, can’t be a static/board game)
* Store separate player profiles/histories/details (securely)
* Communicate data/information without reloading the browser

Submit:

2 Page Report, include:

* Name, Student No
* List all other team members (% of work each member did)
* Youtube URL (2 Min Trailer):
* GitHub (Repo) URL :
* Live Pages URL:

**Marking Critera**

Demonstrate Understanding of HTML, CSS and Javascript

HTML validates fully

CSS does not contain syntax errors

Javascript contains no errors

Creative use of CSS styling

Advanced CSS3 layout (e.g., flexbox)

Creative use of Javascript

HTML5 semantic elements used appropriately

Game/site is responsive

Site is responsive in mobile

Hand written code (clean, structured, commented, ..)

Database/Security

Data is managed for the website effectively and efficiently (Server/Database)

Code Structure

Clear separate of code

Tidy code organisation, file and folder names

Appropriate code nesting and indentation

Informative code comments

Usability of Website/Game

Easy to navigate and use

Clear awareness of accessibility principles

Effective navigation at all sizes, content easy to locate

Development Progress

Shows even work distribution (group work)

Evidence of testing

Live site (Github and webpage)