COMP125 – Client-Side Web Development

Final Exam - Practical
The Dice Roller

Due class #14 (Wednesday August 17, 2016) by end of class.

Value 15%

The Dice Roller **Maximum Mark: 30**

Overview: Use your accumulated knowledge of JavaScript and Web technologies to build a dice roller. Your web app will display a random result of two dies set side by side within the webpage. Each time the player presses the Roll button the dice result displayed will change.

Instructions :

(5 Marks: GUI, 13 Marks: Functionality, 4 Marks: Internal Documentation, 4 Marks: Revision Control, 4 Marks: Cloud Services)

1. Add a **Roll Button** control to your GUI that allows the user to generate a random die result each time he or she clicks the button (2 Mark GUI).
2. Add javascript to respond to the user's clicking of the Roll Button. This code will generate a random number between 1 and 6 for each die. You will create code so that two dice are rolled (6 Marks: Functionality).
3. Display the dice result by using the images (bitmaps) provided for the die faces and displaying the result on your web page. Use HTML5 and CSS to display the die images next to each other. (4 Marks: GUI, 4 Marks: Functionality).
4. Add an appropriate text label control that displays the dice result underneath the corresponding graphic representation (1 Marks: GUI, 1 Marks: Functionality).
5. Include **Internal Documentation** for your program **(4 Marks: Internal Documentation):**
    a. Ensure you include a program header that indicates: The Source file name, Author's name, Student Number, Date, Program description (2 Marks: Internal Documentation).
    b. Ensure you include a header for all of your functions (1 Marks: Internal Documentation
    c. Ensure your program uses contextual variable names that help make the program human-readable (1 Marks: Internal Documentation).
6. Share your files on **GitHub** and deploy to a **Cloud Service** (Microsoft Azure, Heroku, etc.) to demonstrate Version Control Best Practices **(4 Marks: Version Control, 4 Marks: Cloud Deployment).**

a. Your repository must include **your code** and be well structured (2 Marks: Version Control).
b. Your repository must include **commits** that demonstrates the project being updated at different stages of development – each time a major change is implemented (2 Marks: Version Control).
c. Ensure your game is live and online. Deploy to a Cloud Service of your choice (4 Marks: Cloud Deployment).

Feature	Description	Marks
GUI / Interface Design	Display elements meet requirements. Appropriate spacing, graphics, colour, and typography used.	5
Functionality	Site deliverables are me and site functions are met. No errors, including submission of user inputs.	13
Internal Documentation	File header present, including site & student name & description. Functions and classes include headers describing functionality & scope. Inline comments and descriptive variable names included.	4
Version Control	GitHub commit history demonstrating regular updates.	4
Cloud Deployment	Deploy site to Cloud Service	4
Total		**30**

SUBMITTING YOUR WORK

Your submission should include:

1. A zipped archive of your project files
2. A link to your project files on GitHub.
3. A link to your live site on a Cloud Service of your choice.

This assignment is weighted **15%** of your total mark for this course.

External code will **not be allowed** for this test.