**COMP229 – Advanced Web Applications**

**Final Team Project**

Due: Monday, April 28th, 2014 @ 11:59 PM

Value 35%

Final Team Project **Maximum Mark:** **76**

**Overview**: Working with your group and utilizing your accumulated knowledge of web technologies including HTML5, CSS3, PHP, MySQL, JavaScript and jQuery create a web app.

**Instructions :**

**(5 Marks: Site Structure, 5 Marks: Internal Documentation, 10 Marks: External Documentation, 5 Marks: Version Control)**

1. Your Web App must be designed using well-structured semantic HTML, CSS, JavaScript and jQuery (front end technologies) along with a dynamic back-end (using PHP and MySQL) **(4 Marks: Site Structure):**
   1. Your **CSS** rules reside in separate file(s) in their own folder and adhere to best practices (1 Mark: Site Structure).
   2. Your **JavaScript** files (and other external code) are contained in their own folder and are appropriately linked to your site. (1 Mark: Site Structure).
   3. Your **images and multimedia assets** are contained in their own folder and appropriately linked to your site. Your multimedia assets may need to be modified for mobile (1 Mark: Site Structure).
   4. All Your Code (HTML, CSS, JavaScript, jQuery, jQuery Mobile etc.) is error free (1 Mark: Site Structure).
2. Include **Internal Documentation** for your site **(5 Marks: Internal Documentation):**
   1. Ensure you include a **comment** **header** for your **html**, **CSS, JavaScript and PHP files** that indicate: the **File name**, **Author's name**, **web site name**, **file description** (2 Marks: Internal Documentation).
   2. Ensure you include a **section headers** for all of your **html structure, CSS style sections,** and any **functions** (1 Marks: Internal Documentation)
   3. Ensure all your code uses **contextual variable names** that help make the files human-readable (1 Marks: Internal Documentation).
   4. Ensure you include **inline comments** that describe your GUI Design and Functionality. (1 Marks: Internal Documentation)
3. Create an **External Document** for your Web App that includes **(10 Marks: External Documentation)**:
   1. A **company Logo** (1 Marks: External Documentation).
   2. **Table of contents** (1 Marks: External Documentation).
   3. A **Detailed description** of your Web App including its core functionality (3 Marks: External Documentation).
   4. A **Wireframes Section** that include a wireframe image and appropriate arrows and labels for each page template of your Web App (2 Marks: External Documentation)
   5. **Screen Capture Section** that includes Screen Shots (samples) of each of your site’s templates. (2 Marks: External Documentation).
   6. **Potential Future Functionality –** a section that describes features that could be added to your app but do not as yet exist given the time constraints. (1 Mark: External Documentation).
4. Share your files on **Github** for this assignment to demonstrate Version Control Best Practices **(5 Marks: Version Control).**
   1. Your repository must include **your code** and be well structured (2 Marks: Version Control).
   2. Your repository must include **commits** from both you and your partner that demonstrates the project being updated at different stages of development – each time a major change is implemented (3 Marks: Version Control).

***APP SPECIFIC REQUIREMENTS (30 MARKS)***

SURVEY SITE

**(15 Marks: GUI, 15 Marks: Functionality)**

1. User Management and site security **(5 Marks: GUI, 3 Marks: Functionality):**
   1. **User Registration** must be included. A form will allow the user to enter profile information (**username**, **password, email address, etc)** which will be stored in a database table (2 Marks: GUI, 2 Marks: Functionality).
   2. The user will be able to **Login**, **Logout** and **modify** his or her profile (1 Mark: GUI, 1 Mark: Functionality).
   3. **Site security** will prevent non-registered users from **creating** a survey or entering secure areas of the site (2 Marks: Functionality).
2. Users can **Create** a Survey **(6 Marks: GUI, 6 Marks: Functionality):**
   1. After a user is **registered** and **logged** **in**, he or she can create a survey based on 1 or 2 possible **survey templates** (e.g. Multiple Choice, Agree/Disagree, Short Answer, etc.) (3 Marks: GUI, 3 Marks: Functionality).
   2. The user should be able **customize** **survey** **questions**. This includes the question text and response options (2 Marks: GUI, 2 Marks: Functionality).
   3. The user should be able to create a **lifetime** for the survey (i.e. when the survey becomes **active** and when it **expires**) (1 Mark: GUI, 1 Mark: Functionality).
3. Anonymous users can **Respond** to any active survey **(2 Marks: GUI, 3 Marks: Functionality):**
   1. Anonymous users should be able to **select** an active survey and respond to survey questions. (2 Marks: GUI, 2 Marks Functionality).
   2. Survey responses will be stored in the database for later use (1 Mark: Functionality).
4. Secure Reporting Section **(2 Marks: GUI, 3 Marks: Functionality):**
   1. A **registered user** will be able to get simple analysis for any survey that he or she **owns** including **number of respondents** and survey answer **statistics** (1 Mark: GUI, 2 Marks: Functionality).
   2. The statistics from each survey can be **exported** in some manner (e.g. **emailed**, **printed**, exported to **excel**, etc.) (1 Mark: GUI, 1 Mark: Functionality).

**SUBMITTING YOUR WORK**

Your submission should include:

1. An external document (MS Word or PDF).
2. A zip archive of your website’s Project files or a link to GitHub (preferable).

Please zip all files in to a single project archive.

This assignment is weighted **35%** of your total mark for this course.

All Assignments are due at the beginning of class.

Late submissions:

* 10% deducted for each day late.

External code (e.g. from the internet or other sources) can be used for student submissions within the following parameters:

1. The code source (i.e. where you got the code and who wrote it) must be cited in your internal documentation.
2. It encompasses a maximum of 10% of your code (any more will be considered cheating).
3. You must understand any code you use and include documentation (comments) around the code that explains its function.
4. You must get written approval from me via email.