COB290 Team Projects – Part 2

Dr Sara Saravi, Dr Magda Zajaczkowsa, Dr George Vogiatzis, and Dr Shirin Dora

1 Introduction

There are several aspects of this assessment. One part of this coursework is an extension of the previous task, still focusing on the productivity and knowledge management system and seeking ways in which the task of the client company's staff might be made easier using the Web App. You will use the MySQL database management system to store and manage information and PHP and/or JavaScript with HTML, CSS to build your dynamic websites.

2 The Problem

The task is to implement a working system based on your prototype which you have shown and discussed with the users in the demonstration at the end of Part 1. Of course, if the users made any requests for changes in the demonstration or you made any promises, then these too will need to be implemented.

3 The Programming

You will all be expected to work in your GCP (Google Cloud Platform) accounts with a Linux system and GitHub repository to coordinate the development. The company users will be using a range of web browsers (e.g., Firefox, Safari, Chrome, and Edge) and at least one of the browsers will be minimal. All the functionality of the websites must be available through that browser (functions that cannot be accessed from that browser will be treated as if they had not been provided).

In Semester 2, help with the programming will be available in the lab at the times shown on your Semester 2 timetable.

4 Deliverables

There are four deliverables for this part, detailed as following:

4.1 Demonstration

(D) During week 3 of semester 2 each group will be expected to give a demonstration (15 minutes), explaining their proposed solution, showing us the dynamic website they have produced. During the demonstration you may be asked to describe and justify the methods chosen for creating your solution.
Attendance of each member of the team for demos is compulsory.

4.2 Source code

(S) Listings of all of your code scripts on your Team's GitHub repository. Each script should be written so as to make it as easy as possible for it to be understood and altered by someone not familiar with the code, e.g. choose meaningful variable names, include appropriate comments, etc. All team members are expected to generate some Git commits to your repository. You will also need to demonstrate that Git/GitHub is effectively used for supporting the system development.

4.3 Reports

(R1) A report discussing the system design (e.g. system architecture) and implementation strategy (e.g. planing and prioritising tasks) of your productivity and knowledge management system to satisfy the requirements. You should apply the knowledge you have learned from the '*Software Engineering 1*' module. The report has a page limit of 6 pages maximum (this includes everything except an additional cover page, if any) with minimum 12 point text, with Arial font.

(R2) Assess the way the group carried out the task and what contribution each member of the group made to each aspect of the work – this will also include the contribution made to the organisation, coordination and communication of the team. The peer assessment submission point and instructions are on Learn. You will be asked to allocate a score for each member of your team based on their contribution to different aspects of the Part 2 deliverables.

Report (R1), and Source code (S) must be submitted by **11am on Monday of week 3** in semester 2. Report (R1) must be submitted on Learn and Source code (S) must be checked in to your team's GitHub repository. Peer assessment (R2) must be completed by **11am on Friday of week 3** in semester 2 (instructions are on the module Learn page).

5 Assessment

The split of marks awarded to the group will be:

- Website demonstration (D) 60%
- Source code (S) 20%
- Report (R1) 20%

Report (R2) will not be directly assessed as part of the group mark, but will be used to enable individual marks to be awarded. The individual marks will be based on, but will not necessarily exactly follow, the peer assessment scores for each member's contribution to the project.

6 Communication

As for the first part, you should use the same Learn forums you used in Part 1 to ask questions. Make sure you ask your question on the right forum or you won't get a good answer. Don't forget the three rules for asking questions:

1. Always use the correct forum, never ask questions by email to any of the staff team.
2. Always put your team number in the subject line: Eg. "Team 21 – Question about delivery dates".
3. Always give possible alternative answers to any question you pose to give the receiver a better idea of your understanding and enable them to give more helpful answers.

7 Plagiarism

Any copying is plagiarism unless the source is specifically referenced.

See the marking criteria on the next page

8 Marking Schemes

8.1 Demonstration

Criterion	Weight	Excellent (A)	Good (B)	Adequate (C,D)	Unsatisfactory (E,F)
Engagement	20%	Presentation/demo is carried out in a professional manner. The whole process is smooth and well organised. All team members have engaged.	Presentation/demo is carried out in a mostly professional manner. The whole process is mostly smooth and well organised.	Some attention to the presentation/demo is given. The whole process is somewhat smooth and well organised.	The presentation/demo is disorganised and the team seems to lack of interest.
User-friendly and responsive design	30%	User-friendly and responsive design is considered to an excellent level. All operations are intuitive and streamlined.	User-friendly and responsive design is considered to a good level. Most operations are intuitive and streamlined.	User-friendly and responsive design is considered to an adequate level. Some operations are intuitive and streamlined.	User-friendly and responsive design is considered to an unsatisfactory level. Most operations are confusing and unintuitive.
Meeting requirements	50%	The system is excellent. All requirements are met and clearly explained.	The system is good. Most requirements are met and clearly explained.	The system is adequate. Some requirements are met and clearly explained.	The system is unsatisfactory. Few requirements are met and the explanation was ambiguous.

8.2 Source code

Criterion	Weight	Excellent (A)	Good (B)	Adequate (C,D)	Unsatisfactory (E,F)
Code readability	40%	Code is clean, self-explanatory and appropriately commented and well organised.	Minor issues with consistent indentation, use of whitespace, variable naming, or general organisation. Some places could benefit from more detailed comments or the code is overly commented.	At least one major issue with indentation, whitespace, variable names, or organisation. The whole implementation would benefit from more adequate comments.	Code is not readable and/or messy.
Use of Git/GitHub	60%	Every team member has demonstrated excellent use of Git/GitHub which includes frequent regular commits, other features such as branches and merges, issues tracking, etc. Commits are modularised.	Every team member has demonstrated good use of Git/GitHub which includes regular commits, and some other features. Occasionally, commits are not modularised.	All or some team members has demonstrated adequate regular commits, and simple attempts occasionally, commits are not modularised.	Git/GitHub is not used or inadequately used.

8.3 Report

Criterion	Weight	Excellent (A)	Good (B)	Adequate (C,D)	Unsatisfactory (E,F)
Organisation and presentation	20%	The document is presented in a professional-looking document, using informative headings and figures/tables where appropriate. Language is appropriate and professional.	The document is presented in a mostly professional-looking document with reasonable use of headings, figures/tables where appropriate. Language is mostly appropriate and professional.	Attention to the organisation and presentation is given, but may not be well-executed. Insufficient or excessive use of headings and figures/tables.	The document is disorganised. Overall presentation of the document is unprofessional.
Implementation strategy:	70%	System design and implementation strategy is consistent, thoroughly presented and justified to an excellent level.	System design and implementation strategy is fully presented with a good level of justification.	System design and implementation strategy is presented with adequate level of justification but some may be insufficient or excessive.	System design and implementation strategy is unclear and the justification is ambiguous.
Referencing	10%	References are correctly complied and cited in the document.	References contain minor errors but they are correctly cited in the document.	References contain major errors but they are correctly cited in the document.	No references or references contain major errors and they are not cited correctly in the document.