
| | | | |
|--|--|----------------------------------|--|
| COMP1640 (2019/20) | Enterprise Web Software Development | Faculty Header ID: 300711 | Contribution: 100% of course |
| Course Leader: Mr Matthew Prichard | Individual coursework | | Deadline Date: Tuesday 24/03/2020 |
| <p>This coursework should take an average student who is up-to-date with tutorial work approximately 50 hours</p> <p>Feedback and grades are normally made available within 15 working days of the coursework deadline</p> | | | |
| Learning Outcomes: All | | | |

Plagiarism is presenting somebody else's work as your own. It includes: copying information directly from the Web or books without referencing the material; submitting joint coursework as an individual effort; copying another student's coursework; stealing coursework from another student and submitting it as your own work. Suspected plagiarism will be investigated and if found to have occurred will be dealt with according to the procedures set down by the University. Please see your student handbook for further details of what is / isn't plagiarism.

All material copied or amended from any source (e.g. internet, books) must be referenced correctly according to the reference style you are using.

Your work will be submitted for plagiarism checking. Any attempt to bypass our plagiarism detection systems will be treated as a severe Assessment Offence.

Coursework Submission Requirements

- **An electronic copy of your work for this coursework must be fully uploaded on the Deadline Date of Tuesday 24/03/2020 using the link on the coursework Moodle page for COMP1640.**
- **For this coursework you must submit a single PDF document.** In general, any text in the document must not be an image (i.e. must not be scanned) and would normally be generated from other documents (e.g. MS Office using "Save As .. PDF"). An exception to this is hand written mathematical notation, but when scanning do ensure the file size is not excessive.
- **There are limits on the file size (see the relevant course Moodle page).**
- Make sure that any files you upload are virus-free and not protected by a password or corrupted otherwise they will be treated as null submissions.
- Your work will not be printed in colour. Please ensure that any pages with colour are acceptable when printed in Black and White.
- **You must NOT submit a paper copy of this coursework.**
- **All courseworks must be submitted as above. Under no circumstances can they be accepted by academic staff**

The University website has details of the current Coursework Regulations, including details of penalties for late submission, procedures for Extenuating Circumstances, and penalties for Assessment Offences. See <http://www2.gre.ac.uk/current-students/regs>

Scenario

This is a group coursework. You will be assigned to a group.

You need to adopt agile scrum working practices, and document your meetings appropriately. Ideally you need a database designer, an information architect, a programmer, a web designer, and a tester, as well as a scrum master and product owner, but more than one person can be in any technical role. No one is to take the role of project manager, but there could be a technical team leader.

You will get an individual grade (40%) based on your report. There is also a group grade (60%), weighted by your contribution to the team effort (from 0% to 100%).

Specification

You are required to build a secure web-enabled role-based system for collecting ideas for improvement from staff in a large University.

The system must meet the following criteria:

- The University has a Quality Assurance Manager to oversee the process.
- All Departments have a QA coordinator who is responsible for managing the process for their Department, and for encouraging staff to contribute.
- All staff (academic and support) have the opportunity to submit one or more ideas.
- All staff must agree to Terms and Conditions before they can submit.
- All staff can optionally upload documents to support their ideas.
- All ideas can be categorised (tagged) from a list of categories at the point when they are submitted.
- The QA Manager can add additional categories at any time, and can delete categories, but only if they have not been used.
- All staff can see all submitted ideas and can comment on any idea. They can also give the Thumbs Up or Thumbs Down for any idea, but only once for any idea.
- Ideas and comments can be posted anonymously, although the author's details will be stored in the database so any inappropriate ideas can be investigated.
- All new ideas are disabled after a closure date for new ideas, but comments can continue to be done until a final closure date.
- Once an idea is submitted the system emails a notification to the Department's QA Coordinator.
- The author of an idea receives an automatic email notification whenever a comment is submitted to any of their ideas.
- Lists of Most Popular Ideas (+1 for Thumbs Up, -1 for Thumbs Down), Most Viewed Ideas, Latest Ideas and Latest Comments must be made available to all users.

- Lists of Ideas need to be paginated (5 per page)
- The University QA Manager needs to be able to download all the data after the final closure date in a CSV file for transfer out of the system. Any uploaded documents need to be downloaded in a ZIP file.
- An administrator is needed to maintain any system data, e.g. closure dates for each academic year, staff details.
- Statistical analysis (e.g. number of ideas per Department) needs to be available.
- The interface must be suitable for all devices (eg mobile phones, tablets, desktops).

Assumptions

You must clearly state any assumptions you make.

Reports

A number of reports need to be made available. For example

- Statistics
 - Number of ideas made by each Department.
 - Percentage of ideas by each Department.
 - Number of contributors within each Department.
- Exception reports
 - Ideas without a comment.
 - Anonymous ideas and comments.

Tasks

1. Work as a team using agile scrum methods to develop and test a secure web-based system to meet the above specification.
2. Create a Panopto screencast recording (including screen and sound) demonstrating the key functionalities of the system.
3. Present the finished product to a non-technical audience to try to persuade them to purchase your system.
4. Document the system to an appropriate standard using a weighted scoring model with commentary, including an evaluation of the design process you followed and your reflection on the finished product, and on the contributions of your team members.

Deliverables

1. A **Group Report based on a Group Repository** containing all the artefacts produced by the team (eg ERD, minutes, test log, product backlog) with a menu allowing easy access to its content. The repository must be secure, but accessible by the Greenwich tutors. The Scrum Master is responsible to ensure this gets uploaded by the due date.
2. An **Individual PDF Report**
The report must give the **URL** of the Group Repository, the Screencast and the website and any **usernames or passwords** needed to access it. The **individual component of the marking will be based on your report**, so ensure this has evidence that your system meets the specified requirements. **The text in your individual report must be entirely your own words.**
3. A **Presentation and Screencast**
You must be present as part of the team that presents the finished product to your tutor, and should contribute to the screencast. The **presentation** should be pitched at a non-technical audience to try to persuade them to purchase the product; the **screencast** should demonstrate the functionality of the system

Assessment Breakdown

Group Component (60%)

This will be assessed based on a group report and a group repository created by the group on a secure shared area accessible to the Greenwich markers. Password and URL must be provided in individual reports. Must be suitably structured with a menu. Suggested location: GitHub, Google Docs, SharePoint 365, own website, DropBox or other repository.

Database 10%

Expect: Security, appropriate data types and validation, clear ERD, referential integrity implemented, enables roles to be implemented

Site design 10%

Expect: Responsive design, clear information architecture for both mobile and desktop, aesthetically pleasing, good usability, meets accessibility criteria

Functionality 10%

Expect: Role based security, submission of reports, email notification, summary and exception reports, UML diagrams, code snippets

Testing 10%

Expect: Test plan, test log, sufficient data to fully test, evidence of testing finding errors, test items linked to user stories in the product backlog

Agile methods followed 10%

Expect: Burn down chart, minutes of meetings, user stories, sprints, product backlogs

Screencast and Presentation 10%

Expect: Professional standard of presentation promoting the product, with contributions by all the team members, Screencast demonstrating all the main features of the product. Screencast can be done by one person.

Weighting factor for each student (scale 0 to 10)

| Commitment | Weight |
|---------------------------|--------|
| Fully committed | 10 |
| Committed | 8 |
| Contributed substantially | 6 |
| Contributed partially | 4 |
| Minimal contribution | 2 |
| No contribution | 0 |

Individual Component (40%)

N.B.: No shared content in the report, i.e. must be entirely in your own words, Must include title page with a list of team members and roles, URL and password of group repository, site and screencast.

Evaluation of product and process 10%

Expect: Appropriate screen shots and commentary, with cross references to group documents, evaluative comments on the product and on the agile process and design method used to build it.

Evaluation of team 10%

Expect: A weighted scoring model of the entire team (including yourself) with own choice of criteria and weighting, supported by commentary on each individual member. Model is expected to produce a range of scores for the individual members.

Self-evaluation 10%

Expect: Honest description of own contribution, and reflection on own performance and any lessons learnt

Quality of documentation 10%

Expect: NO SHARED CONTENT, professional standard, header page, page numbers, table of contents, headings, cropped images, figure captions, no spelling or grammatical errors.

Indicative Grading Criteria

>=70%

Well designed system to fully meet the requirements

Professional standard of report, with appropriate documentation

High level of individual commitment

High level of evaluative commentary

60-69%

Well designed system to meet most of the requirements

Professional standard of report

High level of individual commitment
Limited evaluative commentary

50-59%
Well designed system to meet most of the requirements
Acceptable standard of report
Good level of individual commitment
Limited evaluative commentary

40-49%
Acceptable system to meet most of the requirements
Acceptable standard of report
Acceptable level of individual commitment
Limited evaluative commentary

<40%
Poorly designed system
Few requirements met
Poor standard of report
Limited individual commitment
No evaluative commentary