****

**SESSION:** 2018 / 19 **DIET: 1st**

**Web Platform Development 2**

**(M3I322955)**

**Coursework Specification**

**LEVEL: 3**

**Module Leader: Katrin Hartmann**

**Submission:**

**On or before the 29th April 2019**

1. **Description of the Application**

Design and implement a milestone planner application.  A milestone planner contains a list of milestones that you are planning to carry out for a project, such as a coursework project. The application allows the user to define his or her own milestones, which are only visible to this user, i.e. require a login. In order to be useful the milestones exist for longer than one session.

The features of a milestone (object) can be designed by you but should at least consist of:

* A description of the milestone,
* An intended due date and
* The actual completion date.

The milestone planner should provide the following functionality for milestones:

* Milestones can be removed from the list.
* Milestones can be edited.
* A listing of all incomplete milestones.
* A milestone (list) can be shared with friends using a link.

Any application features not specified can be designed by you. Your application can extend the specification. In this case highlight the additional features in your report.

1. **Working Details**

You should work in teams of 3 or 4 students on this project. You can choose how to split the work between the team members but each team member should take designated responsibilities as half of the marks are allocated to individual team members for their work (see the marking scheme below).

Each team member should work on at least one separate branch. This branch should be regularly (or as deemed practical) merged with the main branch.

The project should be under GIT version control and regularly pushed to a Bitbucket remote.

Individual team members should pull as is sensible given the division of work within the team but the GIT functionality should be utilized, e.g. regular commits, useful comments, branches etc. Simply pushing a completed project to the remote before the submission date will not yield many marks. The Bitbucket repository is also part of the deliverables and should be well presented, e.g. have a readme.md file.

The project structure should adhere to the Maven project structure recommendations and the project should run from Maven (the command should be in the readme.md file).

1. **Submission Requirements**
2. Send confirmation of team membership no later than **one week after receip**t of this coursework specification by email to your supervisor using the subject line: “wpd2 coursework team membership”.
3. A team presentation in **week 8** addressing the following points (20%, cw1):
4. Team working. Here you should explain how the team is planning to work together. This may include (but is not limited to) a plan of the work ahead and division of work between team members as well as the communication between team members, including repo setup.
5. Initial design of the application. The design can be expressed in different ways, e.g. story boards or UML diagrams, such as use case diagrams, activity diagrams, sequence diagrams or class diagrams.
6. Initial implementation. This doesn’t have to include a lot of functionality as there is still time to develop this but you should have an application that can respond to requests and handle a landing page.
7. Share the Bitbucket repository with [k.hartmann@gcu.ac.uk](mailto:k.hartmann@gcu.ac.uk) and **22th April** and afterwards send an email to [k.hartmann@gcu.ac.uk](mailto:k.hartmann@gcu.ac.uk) with the subject line: “wpd2 coursework repo shared by team <id>” (replace <id> with the id of your team as allocated by week 8).

Push your final coursework to the remote repository on or before the **29th April**.

The remote repository contains the code and team documentation (see below).

The master branch contains the code of the implementation. All functionality of the application must be in the master branch.

Team members should work in individual branches. These should be visible.

There will be a separate area on GCULearn | Assignments set up for uploading individual reports.

1. **Marking Scheme**

50% of the final mark is team-based, i.e. each team member receives the same mark (see section 5). 50% of the mark is allocated for individual contributions.

Team marks are awarded for:

* The working application (30%) and
* an approximately 2000 – 3000 words team report outlining (20%):
* The link design within the application, i.e. a mapping of links to functionality.
* Describe the persistence mechanism including any database schema and the data access layer of the application.
* Documentation of the functionality of the application and test reports.
* An appraisal of the application security. NB: this should include a reflection on the actual security of the application as well as show your awareness of possible security threats. Highlight if you have implemented any security measures but also describe additional security measures that you haven’t implemented (as this isn’t a security module). Here you are mainly marked for awareness.
* The statement: “I declare that all work submitted for this coursework is the work of <insert names of the authors> alone unless stated otherwise.” on the title page.

Individual marks are awarded for:

* A report outlining the design and implementation of the work undertaken by this team member. Reflect on the design considerations and implementations of your individual contributions to this project. (20%)
* A reflection on your use of code reviews within the team (15%).
* A list of the branch(es) of this team member (15%). Marks are awarded for making continuous use of GIT and its features.

1. **Team Problems**

Part of the mark for this project is based on you working in a team setting. This is reflective of most software development working practices and undertaken with the intention to prepare you for working in a team after graduation. Working together is not always easy and there can be a point when you feel that a fruitful continuation is no longer possible as a team, e.g. in case one team member has ceased to contribute to this team.

As everybody in the team will benefit from the team-deliverable you should try and work with your team members, listen to their suggestions and be constructive in trying to resolve and work around any problems. The team presentation in week 9 is a checkpoint at which you will be asked whether you are working together as a team.

On the other hand, should you feel that a further fruitful continuation of the project is no longer possible as a team it is in your own best interest to raise the issue with your module leader verbally and in an email. In that case the following is likely to happen:

* A team meeting during which the problems are described and alternatives are discussed.
* Should the outcome of the meeting be such that it is in the best interest of everyone to dissolve the team this decision will be recorded and communicated to all team members.
* After that the team will split and each party continues on their own with the work they have already undertaken.
* Any deviations from this specification will have to be agreed with the module leader and recorded.