CSCM94 Software Engineering Principles
Assignment One
Object Oriented Software Design

February 1, 2022

Number of Credits: 20% of the 15 credit module.
Deadlines:
Week 1's Minutes and Contribution Breakdown:
Tuesday 8th February at 11am
Week 2's Minutes and Contribution Breakdown:
Tuesday 15th February at 11am
Week 3's Minutes and Contribution Breakdown:
Tuesday 22nd February at 11am
Week 4's Minutes and Contribution Breakdown:
Tuesday 1st March at 11am
Final Submission:
Tueday 1st March 11am.
Learning Outcome: To gain experience of designing larger software as part of a team.

1 Overview

In this assignment, each group will complete an object oriented design of an application detailed in the separate CSCM94 Functional Specification document. The system you design as part of this assignment shall adhere to the detailed list of the functionality and requirement specifications provided in that document. Please read the Functional Specification document before continuing.

2 A1 Tasks

Your team is asked to carry out a complete design for the entire system. However, you will not submit the entire design. Instead you will submit only certain pieces.

Each team member is required to contribute to at least one class in the design. The designer of the class does not necessarily need to implement it in A2.

2.1 Partial Design Document

A Design Document proposes an object-oriented design for the *entire* application. You should produce one of these. In other words, your team incorporates the full functional specification into the design. You should produce CRC Cards and UML Classes designs that cover all aspects of the design.

At this stage you can largely ignore the graphic side. Focus your design on capturing the data and behaviour of the application. If you have good foundations the graphics can be added on easily later on.

For the submission you will take aspects of your Design Document and produce a **Partial Design Document** that should be no more than 7 pages including text and diagrams (hat are readable when printed on A4 paper).

The Partial Design Document consists of the sections as detailed below.

2.1.1 Candidate Classes and Responsibilities

Choose between 3 and 5 candidate classes that are related and collaborate with each other (do not choose inheritance relationships for this). These classes should also have interesting and non-trivial behaviours that utilise the collaborations.

Note: If you choose poor classes here you will not be able to adequately present details in the following sections and will forfeit the marks. It is your responsibility to choose classes that exhibit the features required for the partial design document.

Present the chosen classes as CRC Cards. For each candidate class the following information (i.e., on the CRC card) must be provided:

1. **Class Name:** (in bold)
2. Author:
3. SuperClass:
4. SubClasses:
5. Responsibilities: a list of services this class provides.
6. Collaborations: a list of classes with which the class communicates.

Maximum 2 pages

2.1.2 Class Diagrams

After specifying the information in Section 2.1.1, draw the classes using UML Class Diagram notation. Your class diagrams must use appropriate arrows and UML syntax to represent relationships such as collaborations (i.e., associations, compositions, and aggregations).

You should carefully think about navigability and multiplicities when drawing the collaboration relationships. If you would like, you can use UML tools such as Papyrus[1] (or others).

When providing details about the attributes and operations, you must think carefully about type information and your design. The classes and methods should fit together and function to achieve the intended behaviour. Do not just add operations and collaborations without thinking about how the operations can actually carry out their jobs. You need to take time to think about what collaborations your classes need to have in order to provide the services they offer.

Describe the collaboration relationships and why you have chosen to model them as collaborations (i.e., associations, aggregations or compositions).

For each of your 5 most complex behaviours (i.e., actual operations/methods listed in your classes presented here) provide a short paragraph explaining what the operation should do overall, how the operation can be implemented, and how it has access to the data it needs (e.g., through collaborations). Justify that it can actually be realised when you code this system.

Maximum 2 pages.

[1] `1http://www.eclipse.org/papyrus/`

2.1.3 Hierarchy Descriptions

Choose 1 generalisation/specialisation (i.e., inheritance) relationship with at least 3 classes. Present 3 of the classes involved in a UML class diagram.

Describe and justify why you have chosen to place these classes in an inheritance hierarchy (e.g., what benefits to you gain in terms of attributes and behaviours? Why did you make this design choice?)

Maximum 1 page.

2.2 Deliverables

There are multiple deliverables during this assignment:

- **Weekly Contribution Breakdowns and Minutes**

  You must submit weekly Contribution Break- downs (see CSCM94 Assignment Overview document). These will include the Contribution Breakdown itself and the minutes of the weekly meeting where you create the Contribution Breakdown. These will help document the contribution of members and the progress over the duration of the assignment.

  The Contribution Breakdown and the minutes that accompany them will be submitted to Canvas.

- **Final Submission**

  The final submission will take place digitally via Canvas. The deadlines can be found at the start of this document. Please submit your Partial Design Document (Section 2.1) in a PDF format.

2.2.1 Weekly Minutes

15% of Coursework 1 Each week you will write up the minutes of the meetings where you discuss and create the Contribution Breakdown for that week. The minutes will be submitted **each week via Canvas as a PDF**, along with the Contribution Breakdowns, see CSCM94 Assignment Overview document). The deadlines can be found at the start of this document.

Your minutes files must be in PDF format and be named *"GroupGNMinutes-yyyy-mm-dd.pdf"* where *GN* is replaced by your group number and *yyyy / mm / dd* is replaced by the calendar date of the deadline for the minutes (with *yyyy* replaced by the year, *mm* with the month, and *dd* by the day), e.g., Group3minutes-2020-02-05.pdf.

The minutes format should follow the standard as set out in lectures.

2.2.2 Partial Design Document

85% of A1 (25% presentation, 60% content) Your Partial Design Document is to be included in your final submission to Canvas. The file must be named *"GroupGNDesignReport.pdf"* where *GN* is replaced by your group number. PDF format is required. Word document format (.doc or .docx) is not acceptable.

2.2.3 Overall Marks

The 2 components above (Weekly Minutes and Partial Design Document) will be used to produce a Group Mark. You will submit 4 Contribution Breakdowns over the duration of this assignment. The average of the weekly contribution scores will form each student's own Contribution Score. The final individual mark (for A1) for each student will be calculated by weighting the Group Mark with their Contribution Score.

Academic staff can overrule marks and change the process of peer assessment for a group in conjunction with the Year Head in atypical situations.

3 Issues with Contribution

Assignment One (A1) and Assignment Two (A2) assess the performance of the group when delivering a non-trivial piece of software. As mentioned in the course, real software is developed in teams and therefore group work is a necessary skill that needs to be developed. Individual contribution levels will be taken in to account via the Contribution Breakdown system. Under normal circumstances, the Contribution Breakdowns will be used to adjust marks and determine the level of contribution. However, we realise that abnormal circumstances during group work may occur.

In the event of any non-contribution, please refer to the CSCM94 Assignment Overview document and also follow this procedure:

- As early as possible, discuss the situation with your Module Coordinator.
- Work with your Module Coordinator to try and get the non-contributor participating.
- Keep your Module Coordinator posted on the situation.

Likewise, if any particular group member disruptively dominates group meetings and the assignment outputs in a way that is damaging to the team, the Module Coordinator should be notified immediately.

3.1 Academic Integrity and Academic Misconduct

By submitting coursework, electronically and/or hardcopy, you state that you fully understand and are complying with the university?s policy on Academic Integrity and Academic Misconduct. The policy can be found at https://myuni.swansea. ac.uk/academic-life/academic-misconduct/ .

In the event that you think you think you witness plagiarism you should get in touch with your module co-ordinator straight away and explain the issue to them.

4 Project Hints

- This coursework is very time consuming. Some teams usually lose marks because they don't manage to finish. Allow extra time for teamwork.
- All group members are expected to contribute to both design and implementation. Group members who are weak at programming may get help from other members who are better at it. However, all group members are responsible for understanding the full submission.
- You could use Discord (discordapp.com) or Slack (slack.com) for communication as a group. Many software development teams and companies use systems such as these a main form of communication.
- Why not use Skype or Slack for some of your group meetings?