

Cardiff School of Computer Science and Informatics

Coursework Assessment Pro-forma

Module Code: CM1202

Module Title: Developing Quality Software

Lecturer: Philipp Reinecke and Helen Phillips

Assessment Title: Design and Risk Management

Assessment Number: Coursework 2

Date Set: Monday 10th December 2018

Submission Date and Time: Wednesday 6th February 2019 at 9:30am.

Return Date: Feedback to Teams in Week 6 – Week starting Monday 4th March 2019.

Feedback on the draft will be in tutorial sessions.

This assignment is worth 15 % of the total marks available for this module. The penalty for late or non-submission is an award of zero marks.

Your submission must include the official Coursework Submission Cover sheet, which can be found here: <https://docs.cs.cf.ac.uk/downloads/coursework/Coversheet.pdf>

Submission Instructions

- **Team Submission** - A nominated TEAM member should submit your coursework, electronically, as a single pdf document, via Learning Central by 9:30am on Wednesday 6th February 2019. If you have any difficulties submitting via Learning Central you **MUST** e-mail the module leader Helen Phillips (PhillipsHR@cardiff.ac.uk) before the deadline. The team submission **must** include the team number on the first page of the document.
- **Individual Submission** – You must submit a self-review plus peer reviews for all other members of your team, **via Learning Central, Team File Exchange** by 9:30am on Wednesday 6th February 2019. These should be combined into a single pdf document. These will be used to monitor engagement within the team, and if necessary adjust marks. Please note these reviews will be shared. The Individual Submission **must** include your student ID on the first page of the document.

	Description		Type	Name
Team Submission	Coursework coversheet for each member of the team, documentation for Tasks 1a, 1b and Task 2.	Compulsory	One .pdf file	[team number]Design&Risk.pdf
Individual Submission Team File Exchange	Self-Review, and peer reviews for each team member	Compulsory	One PDF	Student_ID&CW2.pdf

Any deviation from the submission instructions above (including the number and types of files submitted) may result in a mark of zero for the assessment or question part.

Assignment

On the bottom of each page of the team portfolio type the name of the members of the team that have contributed to that particular section (this will be used to monitor individual student's engagement with the team).

INSTRUCTIONS:

Deliverables This coursework should consist of the following components:

- A class diagram for the example system.
- A class diagram for your use cases.
- Explanation of how this class diagram supports ALL the major system functional requirements or your use cases.
- Risk Analysis and Plan.
- A self-review and peer reviews for each of the other members of your team.

More detail on these deliverables is given below.

Non-participation of Team Members

If your team believes that someone is not contributing then you should email, me, the module leader **PhillipsHR@cardiff.ac.uk** as soon as possible so that I can investigate further. It is also important that anyone who is having difficulty contacting their team or has any other issues that are affecting their ability to work with the team inform me as soon as possible. The peer and self-reviews will be used by the module leader to monitor student engagement.

Task 1 Class Diagrams (70%)

a) Class Diagram for Example System

Model the following system in a UML diagram:

- Objects of class **DataStore** store 0 or more objects of type **StorageObject**. The **DataStore** class has an operation **iterate()** that can be used to iterate over all entries.
- A **StorageObject** has a private attribute **size** of type Integer and a public attribute **name** of type String. The class has the following public operations:
 - **create()**
 - **delete()**
 - **open()**
 - **close()**
 - **lock()**
 - **unlock()**
 - **read()**
 - **write()**

- **getSize()**
- A **File** class and a **DatabaseTable** class have been defined as subclasses of class **StorageObject**.
- The **File** class has a private attribute **location** (of type String) that can be read and written via the public operations **read()** and **write()**, respectively.
- The **DatabaseTable** class has the private attribute **accountInformation** (of type String) that can be accessed via the public operations **getAccountInfo()** and **setAccountInfo()**.
- Every **StorageObject** object has an associated **ContentObject**, with private attribute **data** and public operations **read()** and **write()**.
- The following **ContentObject** classes are defined: **String**, **Binary**, and **JSON**.

Your diagram should be as concise as possible, while still describing the whole system.

b) Class Diagram for Use Case and Explanation

Develop a UML Class diagram for your system. The diagram must include the classes and (possibly different types of) relationships between classes to capture the structure of the proposed system. For each class include representative attributes and operations

Explain how your class diagram will support ALL the major system functional requirements, as captured in the use cases diagram from your previous coursework. This explanation should be a maximum of two pages.

Include the use-cases diagram you submitted for your previous coursework as an Appendix for ease of reference (this will not count towards your page count).

Task 2 Risk Analysis / Plan (30%)

Identify and assess important specific legal, ethical, social and technical risks associated with the implementation, delivery and operation of your application and develop appropriate actions for mitigation. I am expecting one risk from each team member. Identify the most likely risks, comment on their seriousness, likelihood and the team's strategy/plan to minimize disruption. This information should be presented in a table along with a risk mapping. (see relevant lectures notes).

Weightings

	/100
1. Class Diagram for Example System	25
2. Class Diagram for Use Case and Explanation	45
3. Risk Analysis and Plan	30

Learning Outcomes Assessed

LEARNING OUTCOMES ADDRESSED:

- Understand the importance of basic Software Engineering concepts, principles and practices.
- Gain an appreciation of how main stages in the software development lifecycle contribute to the development of a high-quality software system by performing key technical tasks from each stage of the project
- Design a system using appropriate UML models to demonstrate how the main requirements can be delivered
- Show an understanding of how to plan and manage a project through the effective use of a variety of tools and technique

Criteria for assessment - Credit will be awarded against the following criteria.

Task 1a Class Diagram for Example System

HINTS

- Clear and concise class names, Appropriate attributes and methods, Correct syntax

☐ Class Diagram for Example System not done

Modelling conventions ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Correctness: ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Conciseness: ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Rating	Modelling Conventions	Correctness	Conciseness
Excellent	The model clearly follows UML modelling conventions and has no errors in modelling syntax.	The model clearly describes the example system given in the description and all details of the description have been captured.	The model concisely describes the system.
Good	The model is a good attempt at following UML modelling conventions and contains a few errors in modelling syntax.	The model is a good attempt at describing the example system and a few of the details are missing.	The model describes the system concisely, but some redundant information is present.
Pass	The model is a reasonable attempt at following UML modelling conventions and contains some errors in modelling syntax.	The model is a reasonable attempt at describing the example system and some of the details are missing.	The model describes the system, but a significant amount of redundant information is present.
Fail	The model does not follow UML modelling conventions and has many errors in modelling syntax.	The model does not correctly describe the example system and many of the details are missing.	The model contains a lot of redundancy or the model is not sufficiently complete to judge conciseness.

Task 1b Class Diagram for Use Case and Explanation

HINTS – Your explanation must demonstrate how the Class diagram sufficiently supports all the major system functional requirements. The explanation will be also be used in assessing completeness of the diagram.

Modelling conventions ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Explanation: ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Completeness: ☐ Excellent ☐ Good ☐ Pass ☐ Fail

Rating	Modelling Conventions	Explanation	Completeness
Excellent	The model clearly follows UML modelling conventions and has no errors in modelling syntax.	The explanation demonstrates clearly how the class diagram supports all the major system functional requirements	The UML model clearly meets the requirements of the project and no important information is missing.
Good	The model is a good attempt at following UML modelling conventions and contains a few errors in modelling syntax.	The explanation demonstrates how the class diagram supports all the major system functional requirements, but is limited and/or difficult to follow in some sections	The UML model is a good attempt at meeting the requirements of the project and little of the important information is missing.
Pass	The model is a reasonable attempt at following UML modelling conventions and contains some errors in modelling syntax.	The explanation only demonstrates how the class diagram supports some of the major system functional requirements and/or the explanation is limited/difficult to follow in some sections	The UML model is a reasonable attempt at meeting the requirements of the project and some of the important information is missing.
Fail	The model does not follow UML modelling conventions and has many errors in modelling syntax. Or: No UML model has been given.	No explanation has been given or the given explanation fails to demonstrate how the class diagram supports the system functional requirements.	The model meets few of the requirements of the project and much of the important information is missing. Or: No UML model has been given.

Risk Analysis

Excellent	Provides a comprehensive and insightful assessment of important legal, ethical and technical risks. Provides clear and highly appropriate actions for mitigating most of these risks.
Good	Provides an appropriate assessment of important legal, ethical and technical risks Provides appropriate actions for mitigating most of these risks.
Pass	Provides an adequate assessment of some important legal, ethical, business and/or technical risks. Provides appropriate actions for mitigating at least half of these risks.
Fail	A sparse or inappropriate assessment of risk with little/no indication of how most of these risks can be mitigated.

Feedback and suggestion for future learning

Feedback on your coursework will address the above criteria. Feedback and marks will be returned in Week 6 Week Beginning 4th March, via Learning Central as an attachment.

Feedback from this assignment will be useful for the Second year group project and CM2102 Database Systems.