

# Design Marking Guideline

## 1. Class Diagram (40%)

For this section's assessment, you are expected to assess the quality of the submitted UML class diagram. Things to check:

- **UML:**
  - Is the diagram drawn using UML class diagram elements? Arrows, classes and etc.
  - Is it following the correct syntax? Names, return types, input parameters and etc.
  - Are the ranges (if any) follow correct syntax?
- **Diagram:** You can check Section 3 for clarification of the design
  - Does the diagram follow domain model? If any differences, is it mentioned/justified in Section 3?
  - Are the associations (including aggregations) logical?
  - Are all the associations include multiplicities?
  - Are the multiplicities logical? Important to check if ranges are correct, if there is a 0 range is it logical? Is it missing any 0 range, for example in a design you might allow your customers not to have an account so in that case is the multiplication between customer and account allow this?
  - Are generalisations and specialisations logical and follow the domain model?
  - Are the methods and constructors logical?
- **Design patterns:**
  - Are any design patterns applied?
  - If applied is it designed correctly and in the right context.
- **Image:**
  - Is it clear and readable? All associations and input and return types.
  - All information is visible. No attributes or methods that are half missing.
- **Mark 5:** All the above is achieved without any mistakes.
- **Mark 4:** All the above is achieved with minor mistakes. No design patterns attempted (for a design pattern with very minor mistakes you might consider a 5).
- **Mark 3:** Most the above is achieved. Includes a high number of minor mistakes and might have few major mistakes. Image must be clear and readable (ignore very slightly blurry images if they are readable).
- **Mark 2:** Few of the above is achieved. Diagram might have a high number of major mistakes and/or diagram is blurry and hardly readable.
- **Mark 1:** Diagram is full of major mistakes (pretty much every element has a mistake) and/or Image quite blurry and barely readable.
- **Mark 0:** If the section is empty or missing.

A major mistake is having classes that doesn't belong on a class diagram such as a class representing a use case or a system function (a class called search or menu) or a class that needs to be an attribute. Missing multiplicities and associations also considered major mistakes. Having non-UML elements are also major mistake. Minor mistakes are incorrect associations, incorrect arrow types, incorrect multiplicities, arrow with an incorrect direction.

## 2. Traceability Matrix (10%)

For this section's assessment, you are expected to check if all the data RQs are fulfilled in the given design. Things to check:

- **Mark 5:** All data requirements are met, and the explanation strongly justifies the inclusion of attributes in the class. They must be logical.
- **Mark 4:** All data requirements are met, and the explanation adequately justifies the inclusion of attributes in the class. They must be logical.
- **Mark 3:** All or most of the data requirements are met, and the explanation weakly justifies the inclusion of attributes in the class. They must be logical.
- **Mark 2:** Most of the data requirements are met, and not all the included attributes is justified. The classes and attributes that fulfils the RQs are not logical.
- **Mark 1:** Few of the data requirements are met, and none of the included attributes is justified. The classes and attributes that fulfils the RQs are not logical.
- **Mark 0:** If the section is empty or missing.

Some of the data RQs might be fulfilled by a whole class while others with the inclusion of one or more attributes so be careful when judging. Suppose we have a requirement that says; "A user must login using a username and password". This RQ can be solved by multiple designs. A separate class for login details (username and password) is a common example as having password in the user object is considered a security weakness. However, a class design with username and password as attributes is a good as the fore mentioned design.

## 3. Design Discussion (20%)

For this section's assessment, you are expected to assess the quality of design justification. Things to check:

- Discussion on the differences between the entities in domain model and class diagram (if any differences):
  - Included and excluded entities and relations
  - Explanation for the changes (reasons for the changes feedback, RQ analysis and etc.)
- Design patterns applied (if any):
  - Justification for the use of the design pattern
  - How did you design it and why it is and effective/efficient solution?
- Justification for design decisions that **might not be clear to the reader**:
  - Justification for multiplicities
  - Justification for association and aggregations (including composition)
  - Justification for generalisations

You don't have to give explanation for very basic design decisions but don't assume others will understand your reasoning with one look.

- **Mark 5:** All above explained clearly in good detail.
- **Mark 4:** All above explained clearly but few decisions are not explained/justified clearly.
- **Mark 3:** Most of the above explained clearly but some decisions are not explained/justified clearly.
- **Mark 2:** Few of the above explained and majority of the decisions are not explained/justified clearly.
- **Mark 1:** None of above explained and the section doesn't justify any design decisions.
- **Mark 0:** If the section is empty or missing.

## 4. Sequence Diagrams (30% => 3x10%)

For this section's assessment, you are expected to assess the quality of the sequence diagrams. Things to check:

- **UML:**
  - Is the diagram drawn using UML class diagram elements? Lifelines, messages and etc.
  - Is it following the correct syntax? Message names, loops, alts and etc.
- **Diagram:**
  - Are the sequence diagrams for key use-cases? Is one of them for the use-case described in the previous report.
  - Does the diagram follow the class diagram? Do the messages call the correct methods and objects? Objects match classes in the diagram?
  - Is the sequence of events follow the use-case description (one of them) from the previous report? There might be slight differences.
  - Is the diagram logical? Are the sequence of event correct?
  - Are loops and alts used where necessary? Are they used logically?
  - Are the guards used where necessary? Are the correct guards used?
  - Correct create and destroy messages used (if any)?
  - Create messages (if any) call constructors?
  - Is the actor correct for the diagram?
- **Image:**
  - Is it clear and readable? All lifelines, messages and names.
  - All information is visible. No message or lifeline names that are half missing.

### Marking guideline

- **Mark 5:** All the above is achieved without any mistakes.
- **Mark 4:** All the above is achieved with minor mistakes. (No alts or loops attempted in any one of the diagrams, if one diagram includes any that's enough for both section 4 and 5).
- **Mark 3:** Most the above is achieved. Includes a high number of minor mistakes and might have few major mistakes. Image must be clear and readable (ignore very slightly blurry images if they are readable).
- **Mark 2:** Few of the above is achieved. Diagram might have a high number of major mistakes and/or diagram is blurry and hardly readable.
- **Mark 1:** Diagram is full of major mistakes (pretty much every element has a mistake) and/or Image quite blurry and barely readable.
- **Mark 0:** If the section is empty or missing.