THE INDEPENDENT INSTITUTE OF EDUCATION

MODULE NAME:	MODULE CODE:
SOFTWARE ENGINEERING	SOEN6222

ASSESSMENT TYPE: POE (PAPER & MARKING RUBRICS)

TOTAL MARK ALLOCATION: 300 MARKS

TOTAL HOURS: A minimum of 45 HOURS is suggested to complete this assessment

By submitting this assignment, you acknowledge that you have read and understood all the rules as per the terms in the registration contract, in particular the assignment and assessment rules in The IIE Assessment Strategy and Policy (IIE009), the intellectual integrity and plagiarism rules in the Intellectual Integrity and Property Rights Policy (IIE023), as well as any rules and regulations published in the student portal.

INSTRUCTIONS:

1. ***No material may be copied from original sources, even if referenced correctly, unless it is a direct quote indicated with quotation marks. No more than 10% of the assignment may consist of direct quotes.***
2. ***Please ensure that you submit your assignment through SafeAssign. Please make sure you attach a similarity report to your POE if you are required to submit a hard-copy of your PoE.***
3. ***Make a copy of your assignment before handing it in.***
4. *Assignments must be typed unless otherwise specified.*
5. *Begin each section on a new page.*
6. *Follow all instructions on the PoE cover sheet.*
7. *This is an individual assignment.*

Referencing Rubric

Providing evidence based on valid and referenced academic sources is a fundamental educational principle and the cornerstone of high-quality academic work. Hence, The IIE considers it essential to develop the referencing skills of our students in our commitment to achieve high academic standards. Part of achieving these high standards is referencing in a way that is consistent, technically correct and congruent. This is not plagiarism, which is handled differently.

Poor quality formatting in your referencing will result in a penalty **of a maximum of ten percent being deducted from the percentage awarded**, according to the following guidelines. Please note, however, that **evidence of plagiarism in the form of copied or uncited work (not referenced), absent reference lists, or exceptionally poor referencing, may result in action being taken in accordance with The IIE's Intellectual Integrity Policy (0023)**.

Markers are required to provide feedback to students by indicating **(circling/underlining) the information that best describes the student's work.**

Minor technical referencing errors: 5% deduction from the overall percentage – the student's work contains **five or more errors** listed in the minor errors column in the table below.

Major technical referencing errors: 10% deduction from the overall percentage – the student's work contains **five or more errors** listed in the major errors column in the table below**.**

If both minor and major errors are indicated, then 10% only (and not 5% or 15%) is deducted from the overall percentage. The examples provided below are not exhaustive but are provided to illustrate the error

Required: **Technically correct referencing style**	**Minor errors in technical correctness of referencing style** **Deduct 5% from percentage awarded**	**Major errors in technical correctness of referencing style** **Deduct 10% from percentage awarded**
Consistency • The same referencing format has been used for all in-text references and in the bibliography/reference list.	Minor inconsistencies. • The referencing style is generally consistent, but there are one or two changes in the format of in-text referencing and/or in the bibliography. • For example, page numbers for direct quotes (in-text) have been provided for one source, but not in another instance. Two book chapters (bibliography) have been referenced in the bibliography in two different formats.	Major inconsistencies. • Poor and inconsistent referencing style used in-text and/or in the bibliography/ reference list. • Multiple formats for the same type of referencing have been used. • For example, the format for direct quotes (in-text) and/or book chapters (bibliography/ reference list) is different across multiple instances.
Technical correctness • Referencing format is technically correct throughout the submission. • The correct referencing format for the **module's discipline** has been used, i.e., either **APA,** OR **Harvard** OR **Law.** • Position of the reference: a reference is directly associated with every concept or idea. • For example, quotation marks, page numbers, years, etc. are applied correctly, sources in the bibliography/reference list are correctly presented.	**Generally, technically correct with some minor errors.** • The correct referencing format has been consistently used, but there are one or two errors. • Concepts and ideas are typically referenced, but a reference is missing from one small section of the work. • Position of the references: references are only given at the beginning or end of every paragraph. • For example, the student has incorrectly presented direct quotes (in-text) and/or book chapters (bibliography/reference list).	**Technically incorrect.** • The referencing format is incorrect. • Concepts and ideas are typically referenced, but a reference is missing from small sections of the work. • Position of the references: references are only given at the beginning or end of large sections of work. • For example, incorrect author information is provided, no year of publication is provided, quotation marks and/or page numbers for direct quotes missing, page numbers are provided for paraphrased material, the incorrect punctuation is used (in-text); the bibliography/reference list is not in alphabetical order, the incorrect format for a book chapter/journal article is used, information is missing e.g. no place of publication had been provided (bibliography); repeated sources on the reference list.
Congruence between in-text referencing and bibliography/ reference list • All sources are accurately reflected and are all accurately included in the bibliography/ reference list.	**Generally, congruence between the in-text referencing and the bibliography/ reference list with one or two errors.** • There is largely a match between the sources presented in-text and the bibliography. • For example, a source appears in the text, but not in the bibliography/ reference list or vice versa.	**A lack of congruence between the in-text referencing and the bibliography.** • No relationship/several incongruencies between the in-text referencing and the bibliography/reference list. • For example, sources are included in-text, but not in the bibliography and vice versa, a link, rather than the actual reference is provided in the bibliography.
In summary: the recording of references is accurate and complete.	In summary, at least **80%** of the sources are correctly reflected and included in a reference list.	In summary, at least **60%** of the sources are incorrectly reflected and/or not included in reference list.

Overall Feedback about the consistency, technical correctness and congruence between in-text referencing and bibliography:

....

Background

Case Study 1: Helping Hands Software Solution

Helping Hands is a non-profit organization that provides support and resources to individuals and families in need. The organization has several programs and initiatives aimed at providing food, clothing, shelter, and educational resources to those who need it most.

Problem: Helping Hands was facing several challenges in managing their programs and initiatives effectively. The organization was relying on manual processes, spreadsheets, and paper-based systems, which were time-consuming, prone to errors, and difficult to track. The organization was facing difficulties in tracking the distribution of resources, monitoring program outcomes, and communicating with volunteers and beneficiaries.

Solution: To address these challenges, Helping Hands decided to implement a comprehensive software solution in the form of a website.

Case Study 2: Tree-Saving Non-Profit Software Solution

A non-profit organization dedicated to saving trees was facing challenges in managing its operations, communicating with supporters, and fundraising. The organization was committed to promoting the importance of trees for the environment and wildlife, but it needed a solution that would help it achieve its mission more effectively.

Problem: The organization faced several challenges, including:

- Difficulty in tracking and managing tree-planting efforts and monitoring their progress
- Ineffective communication with supporters and volunteers, resulting in low engagement and participation
- Lack of an efficient system for collecting and managing donations
- Inadequate information on the organization's activities and impact

Solution: Using an agile approach develop a custom to address these challenges and support the non-profit in its mission.

Important! Choose ONE project that you will stick with for the whole semester. If you change your mind later, it will cause a lot of unnecessary extra work!

Over the course of the semester, you will specify the requirements for the system, design the system, choose a technology for the website prototype and build a prototype showing what the user interface will look like.

Instructions

Part 1 — Requirements Document (LU1-LU4) (Marks:100)

A minimum of 10 hours is suggested to complete this task.

Please reference your work (both in text and a reference list at the end)

Create a software requirements specification document for your chosen project. The document must contain the following sections:

- Introduction: Cover the purpose of the software and the scope of the project in 150 to 200 words and you will implement the agile approach to the project
- Functional requirements: At least 10 different requirements.
- External interface requirements: Specify at least one external system that the website must communicate with.
- Non-functional requirements: At least five different requirements.

To get started, read this page for some guidance on writing a software requirements document: https://relevant.software/blog/software-requirements-specification-srs-document/ [Accessed 09 March 2023].

Summary of what to submit:

• Software requirements specification document.

Part 2 —Architectural Design (LU5-LU7) (Marks: 100)

A minimum of 20 hours is suggested to complete this task.

This task has two parts – choosing an architecture pattern and drawing a Unified Modelling Language (UML) class diagram showing the main classes that you will have in your application.

Question 1 – Architecture Pattern (30 marks)

Choose an architecture pattern to use for the website. In 300 to 500 words total.

- *Explain how the architecture pattern works; and*
- *Motivate why you chose the architecture pattern.*

Question 2 – UML Class Diagram (70 marks)

Draw a UML class diagram showing the design of the main classes (with their attributes and operations) that you will have in your application. Indicate on the diagram which requirements from your software requirements specification are satisfied by the diagram.

You may use any tool of your choice to draw the diagram. You need to be able to export the diagram from the tool somehow and embed it in your design document.

Summary of what to submit:

- **Design document containing the architecture decision and UML diagram with requirements mapping.**

Portfolio of Evidence (POE) — Document Updates and Prototype (Marks:100)

A minimum of 15 hours is suggested to complete this task.

This task has three parts – updating the existing documents, creating a user interface prototype and completing a self-evaluation.

Question 1 – Document Updates (20 marks)

For the POE, you must update your software requirements specification (task 1) and design document (task 2) to include the improvements suggested by your lecturer. Also include a brief write-up (100 to 200 words) as an appendix in each of the documents, explaining how the feedback was addressed.

Question 2 – User Interface Prototype (70 marks)

Develop a prototype of the website. You may do this using any technology of your choice.

The prototype should illustrate how users would use the website. You may choose to implement a full website using any web development framework (a fully developed backend is not a requirement here), or you may choose to use user interface prototyping software to draw the user interfaces.

The website has two types of users. Make sure that you implement the user interface for both types of users.

Submit at least six screenshots. Together with each screenshot, include a description (150 to 200 words) of which user uses the specific part of the system and what they would be doing there. Also include which requirement(s) each screenshot meets.

Question 3 – Self-Evaluation (10 marks)

You must also complete a **self-evaluation** to reflect on your learning experience. You can find the rubric for this at the end of this POE. You are also required to add a paragraph of 150-200 words on your self-reflection on the module.

Summary of what to submit:

- Updated software requirements document, including an appendix with a write-up of how the feedback was addressed.
- Updated design document, including an appendix with a write-up of how the feedback was addressed.
- User interface document containing screenshots of the prototype with descriptions.
- Completed self-evaluation rubric.

Appendix A - PoE Marking Rubrics

MODULE NAME:	MODULE CODE:
SOFTWARE ENGINEERING	SOEN6222

STUDENT NAME:	
STUDENT NUMBER:	

RUBRIC 1 — SKELETON OUTLINE	Levels of Achievement				Feedback
In order to be awarded full marks for these elements of Task 1, students need to have:	Excellent	Good	Developing	Poor	
	Score Ranges Per Level (½ marks possible)				
Introduction: The **purpose** of the software is well explained.	**9—10** Excellent description of the purpose of the software.	**7—8** Good description of the purpose of the software, providing sufficient details.	**4—6** Some details about the purpose of the software included.	**0—3** Purpose of the software not included, or very little details provided.	
Introduction: The **scope** of the project is well described and realistic with agile principles guiding the poject	**9—10** The scope of the project is excellently described.	**7—8** Good description of the project scope and the scope is realistic.	**4—6** Some details about the project scope provided but not realistic.	**0—3** Scope is not included or completely unrealistic.	

RUBRIC 1 — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of Task 1, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
Functional requirements: At least 10 unambiguous requirements are included.	**16—20** At least 10 excellently written requirements are included.	**10—15** At least eight good requirements are included, with minor improvements possible.	**5—9** At least five good requirements are included, or requirements are unrealistic.	**0—4** No functional requirements included, or requirements are unrealistic.	
Functional requirements: The included requirements are testable.	**9—10** All the requirements are written to be easily testable.	**7—8** Minor improvements can be made to make some of the requirements more testable.	**4—6** At least half of the requirements are testable, but the rest require changes.	**0—3** No functional requirements are included, or the requirements are not at all testable.	
Functional requirements: The included requirements are feasible.	**9—10** All the requirements are feasible.	**7—8** Most of the requirements are feasible.	**4—6** At least half of the requirements are feasible, but the rest need changes.	**0—3** No functional requirements are included, or the requirements are not feasible at all.	
Functional requirements: The included requirements are clear.	**9—10** All the requirements are clear.	**7—8** A few of the requirements could be clearer.	**4—6** Half of the requirements require clarification.	**0—3** No functional requirements included, or most of the requirements are unclear.	

RUBRIC 1 — SKELETON OUTLINE	Levels of Achievement				Feedback
In order to be awarded full marks for these elements of Task 1, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
External interface requirements: At least one external system interfaces to the website and the requirement for interfacing to that system is clearly described.	**9—10** Excellent description of an interface to a relevant external system.	**7—8** Good description of an external system with one or two points that need clarifying.	**4—6** Some details included about an external interface, but more information is required.	**0—3** No external interface included, or very little details provided.	
Non-functional requirements: At least five relevant non-functional requirements are included.	**16—20** At least two relevant non-functional requirements included.	**10—15** At least one relevant non-functional requirement included.	**5—9** Non-functional requirements included but only somewhat relevant.	**0—4** No non-functional requirements included or irrelevant requirements.	
TASK 1 SUBTOTAL					**/100**

RUBRIC 2 — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of Task 2, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
Architecture Pattern: Clear and detailed enough explanation of the chosen architecture pattern.	**9—10** Excellent explanation that clearly describes how the architecture works.	**7—8** A good description of the chosen architecture that could include some more details.	**4—6** The chosen architecture is presented but requires more details.	**0—3** No architecture pattern described, or almost no details provided.	
Architecture Pattern: Clear and detailed motivation for why the architecture pattern was chosen.	**16—20** An excellent motivation that clearly describes why the architecture is the right choice.	**10—15** A good motivation of the architecture choice that could have some more details.	**5—9** Some information is provided but the full argument for why the choice is not made.	**0—4** No motivation provided or the arguments made do not make sense.	
Class Diagram: Logical structure with enough detail to document the design.	**16—20** An excellent diagram that clearly details the design of the software.	**10—15** A good design with only a few additions that could be made.	**5—9** The basic design is in place but much more details are required.	**0—4** No class diagram included or very little details.	

RUBRIC 2 — SKELETON OUTLINE	Levels of Achievement				Feedback
In order to be awarded full marks for these elements of Task 2, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
Class Diagram: Attributes and operations are shown for the classes.	**16—20** All classes have excellent attributes and operation definitions.	**10—15** Attributes and operations are well defined with only a few things missing.	**5—9** Some attributes and operations are included but many are missing.	**0—4** No attributes and operations are included, or the provided information is irrelevant.	
Class Diagram: Correct UML notation used.	**9—10** Excellent use of UML notation.	**7—8** Good use of UML notation.	**4—6** Class diagram uses some elements of UML notation, but there are numerous mistakes.	**0—3** Diagram is not using UML notation at all, or many large mistakes.	
Class Diagram: Elements on the diagram are traceable to the requirements.	**16—20** All the elements are clearly traceable to the requirements.	**10—15** Most elements are traceable to the requirements, but some not.	**5—9** Lots of the elements cannot be traced to requirements.	**0—4** No traceability included or most elements cannot be traced.	
TASK 2 SUBTOTAL					**/100**

RUBRIC 3 (FOR POE) — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of the POE, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
Software requirements: Document updated to incorporate feedback.	**5** All the feedback was excellently incorporated.	**3—4** Most of the feedback was addressed, but not all.	**1—2** Only one or two comments were addressed.	**0** None of the suggested updates were incorporated.	
Software requirements: Description in an appendix of how the changes were handled.	**9—10** An excellent description included that clearly details all the changes.	**7—8** A good description included with a few things missing.	**4—6** Some details included but not everything is clearly described.	**0—3** No description included or the description is completely unclear.	
Software design: Document updated to incorporate feedback.	**5** All the feedback was excellently incorporated.	**3—4** Most of the feedback was addressed, but not all.	**1—2** Only one or two comments were addressed.	**0** None of the suggested updates were incorporated.	
Software design: Description in an appendix of how the changes were handled.	**9—10** An excellent description included that clearly details all the changes.	**7—8** A good description included with a few things missing.	**4—6** Some details included but not everything is clearly described.	**0—3** No description included or the description is completely unclear.	

RUBRIC 3 (FOR POE) — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of the POE, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
User interface: Screenshot 1 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	
User interface: Screenshot 2 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	

RUBRIC 3 (FOR POE) — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of the POE, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
User interface: Screenshot 3 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	
User interface: Screenshot 4 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	

RUBRIC 3 (FOR POE) — SKELETON OUTLINE	**Levels of Achievement**				**Feedback**
In order to be awarded full marks for these elements of the POE, students need to have:	**Excellent**	**Good**	**Developing**	**Poor**	
	Score Ranges Per Level (½ marks possible)				
User interface: Screenshot 5 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	
User interface: Screenshot 6 shows a relevant view of the prototype, with a comprehensive description of which user uses that screen and what they would be doing. The screenshot is traceable to requirements.	**9—10** Relevant view of the prototype, with a comprehensive description of how the screen is used. The screenshot is traceable to requirements.	**7—8** Relevant view of the prototype, with a good description and traceability. Some details are missing.	**4—6** A relevant screenshot is provided but the description and/or traceability need a lot more details.	**0—3** No screenshot included, or no description and traceability provided.	
Design: Overall design of the prototype according to UI and UX principles covered.	**9—10** Exceptional consideration and advanced understanding of the principles of good UX design.	**7—8** Good consideration and understanding of the principles of good UX and UI design.	**4—6** Superficial consideration and acceptable understanding of the principles of good UX and UI design.	**0—3** Insufficient consideration or underdeveloped understanding of the principles of UX and UI design.	
Self-evaluation: Record the mark awarded on the self-evaluation rubric here.	**0—10** (See self-evaluation rubric for details)				

POE SUBTOTAL				**/100**
RUBRIC POE — Self-Evaluation	**Levels of Achievement**			**Score**
	Excellent	**Good**	**Developing**	
	Score Ranges Per Level (½ marks possible)			
Criteria (3)	**3** • Bravo! You have done exceptionally well!	**2** You are on the right track, but you can do better!	**0 to 1** You have learned something – but you are not proving it!	
Reflection (3)	**3** • Reflection shows thorough thoughtfulness. • Reflection has several supporting details and examples. • All parts of the reflection are complete and done.	**2** Reflection shows little thoughtfulness. Reflection has few details or examples. Most parts of the reflection are incomplete.	**0 to 1** • Reflection shows no thoughtfulness. • Reflection has no details. • Reflection is incomplete.	

Demonstration of Learning (4)	**4** - Clearly explains what was learned. - Reflection is beyond simple description of event/experience to an analysis of how it contributed to learning and understanding.	**2 to 3** The reflection demonstrates student's attempt to analyse the event/experience but fails to demonstrate depth of analysis.	**0 to 1** - Reflection does not move beyond description of the event/experience.	
POE – Self Evaluation TOTAL				**/ 10**