1. Clearly define the problem.
    - Get familiar with the use-case.
    - Clearly state what the problem is and why it is important to be solved.
    - Break down the problem into subproblems.
2. Propose a solution.
    - **Do not** try to **design** a software, but **come up** with a solution to the problem on a very high level **(no code, no technology).**
    - Think about inputs and outputs.
3. Goals/objectives definition.
    - Use the hierarchy to define goals at different levels but **do not try** to finalize them. There is more that we need to see before deciding which goals (milestones) we need to achieve.
    - Typically these decisions are based on the dialogue between your team and the client (the entity that has come to you with the problem), in this case - the lecturer.
4. Prepare a maximum of 1 page A4 outlining the above.
    - Writing things down will allow you to see if something is wrong and if something else is needed.
    - Have three sections:
        - Problem definition
        - Objectives
        - Measurements / KPIs

Rubric

Task 1 [max 5pts]

Criteria	Ratings	Pts
This criterion is linked to a learning	**1.5 PtsExcellent**The problem has been clearly defined. The definition is complete (no ambiguity left) and justified. **1 PtsAverage**The problem definition is somehow correct and complete, but there is no justification. **0.5 PtsIncomplete / Unclear**The definition is incomplete or unclear. There are no or little supporting arguments. **0 PtsNo definition**No definition was provided, neither written, nor in conversation.	1.5 pts

outcome Problem definition		
This criterion is linked to a learning outcome Goals and objectives	**2 PtsExcellent**Goals and objectives follow a 4-step hierarchy. Goals / objectives are in line with the problem definition. **1.25 PtsAverage**Goals / objectives follow a hierarchy but do not completely reflect the problem statement. **0.75 PtsIncomplete / Unclear**Goals / objectives are rather unclear, sporadic and don't reflect the problem. **0 PtsNo definition**No goals / objectives presented.	2 pts
This criterion is linked to a learning outcome Measures / KPIs	**1.5 PtsGood**Measures / KPIs are considered in line with the goals and objectives. **1 PtsPartial (>50%)**Measures / KPIs considered are in line with more than half but not all goals/ objectives. **0.5 PtsIncomplete (<50%)**Measures / KPIs are in line with less than half but at least one goal / objective. **0 PtsNo measures / KPIs.**No measures are considered	1.5 pts

Total points: 5

Task 2 - From data to AI

Due 25 Sep by 21:00
Points 5
Submitting a file upload
File types doc, pdf, and docx
Available after 15 Sep at 0:00

1. Collect data:
    - Download data from a preferred source (clarifications in class or via email/canvas to the lecturer)
    - Data exploration:
        - Volume?
        - Velocity?
        - Variety?
        - Veracity?
2. Steps to train your (supervised) system:
    - Make sure data is aligned and you have a source (input) and a target (output, labels) side.
    - Preprocess the data
    - Split the data into train, test and dev sets
    - Clean the data by removing information you don't think is relevant, by removing or filling in missing data, etc.
    - Train the system
    - Evaluate the system
3. Form the intelligence - if necessary, combine trained models (each of which solves one part of the task) to cover the complete intelligence
4. Prepare a maximum 1 page A4 outlining the above.
    - Writing things down will allow you to see if something is wrong and if something else is needed.
    - Have three sections:
        - Data
        - Model (statistics about the mode, e.g. evaluation metrics, training time, inference time, number of epochs, etc.)

- Software used for the development, including software for data processing and preparation.

Rubric

Task 2 [max 5 points]

Criteria	Ratings	Pts
This criterion is linked to a learning outcome Data	**2 PtsExcellent**Data is collected and analysed. Statistics are collected. Data is preprocessed adequately. **1 PtsGood**Data is collected, and preprocessed. No statistics **0.5 PtsFair**Data is messy and not processed adequately. Requires external help befor training. **0 PtsBad**No data	2 pts
This criterion is linked to a learning outcome Model	**2 PtsExcellent**Model is trained and evaluated. Statistics are presented. **1.5 PtsGood**Model is trained and evaluated but no training statistics. **0.5 PtsFair**Model is trained but not evaluated. No training statistics are given. **0 PtsBad**No model has been trained.	2 pts

This criterion is linked to a learning outcome Model adequacy	**1 PtsExcellent**The selected model(s) is based on a deep learning or a machine learning approach that can adequately solve the task at hand. The model architecture is not too simple neither it is too complex. **0.5 PtsGood**The selected model(s) is based on a deep learning or a machine learning approach that can adequately solve the task at hand. The model architecture is either too simple or too complex. **0.25 PtsFair**The selected model(s) is not based on a deep learning or a machine learning approach although, in theory, it can adequately solve the task at hand. **0 PtsBad**No model has been trained.	1 pts

Total points: 5

Task 3 - Analysis

Due 9 Oct by 21:00
Points 6
Submitting a text entry box, a website url, or a file upload
Available after 29 Sep at 0:00

1. Requirements
    - Define user requirements
    - Define system requirements
2. Software requirement specification document.
    - Clearly outline user and system requirements.
    - Clearly outline functional and non-functional requirements.
    - Clearly indicate which requirements are related to the fact that an Intelligent system is being designed. That is, clearly state which are the AI-specific requirements.
3. No report is needed for this task. You need to submit the SRS document.
4. You can follow the attached template: SE4CSAI_SRS_Document_Template.docx
5. Download SE4CSAI_SRS_Document_Template.docx
6.
7. Do not write more than 4 pages (0.5-1 page of introduction, 0.5-1 for user requirements, 1-2 for system requirements and 0.5 for AI specific requirements).

Rubric

Task3 rubric [max 6 points]

Criteria	Ratings	Pts
This criterion is linked to a learning outcome User Requirements	**4 PtsExcellent**All requirements are verifiable, clear and concise, complete, consistent, traceable, viable, necessary and implementation free **3 PtsGood**4 to 7 criteria met for more than 50% of the requirements. **2.5 PtsAverage**1-3 criteria met for more than 50% of the requirements. **1 PtsBelow average**Still understandable but difficult to align with the 8 criteria. **0.5 PtsBad**Most of the requirements are incomplete, unclear. None of them meet the criteria. Some of them are not user requirements. **0 PtsNo marks**No requirements provided	4 pts
This criterion is linked to a learning outcome Non-Functional Requirements	**4 PtsExcellent**Clear, complete, not ambiguous, requirements clearly indicated **3 PtsGood**Complete but ambiguous **2.5 PtsAverage**Clear, but not complete **1.5 PtsBelow average**not complete and not clear. Still understandable and can be improved. **0.5 PtsBad**Incomplete, unclear **0 PtsNo marks**No requirements provided	4 pts

This criterion is linked to a learning outcome Functional vs Non-Functional	**3 PtsExcellent**Functional and non-functional requirements are well distinguished **2 PtsAverage**Minor overlaps between non-functional and functional requirements **1 PtsBad**Major overlaps between functional and non-functional requirements **0 PtsNo marks**No clear distinction at all	3 pts
This criterion is linked to a learning outcome AI-specific	**3 PtsExcellent**AI specific requirements are clearly indicated and make sense **2 PtsAverage**AI-specific requirements are clearly indicated, but are not completely relevant (not only AI-specific). **1 PtsBad**AI-specific requirements are clearly indicated but are not relevant (not AI-specific) at all. **0 PtsNo marks**No AI-specific requirements are indicated	3 pts
This criterion is linked to a learning outcome Functional	**4 PtsExcellent**Clear, complete, not ambiguous, requirements clearly indicated **3 PtsGood**Complete but ambiguous **2.5 PtsAverage**Clear, but not complete **1.5 PtsBelow average**not complete and not clear. Still understandable and can be improved. **0.5 PtsBad**Incomplete, unclear **0 PtsNo marks**No requirements provided	4 pts

Requirements		

Task 4 - Project management and implementation

Due 23 Oct by 21:00
Points 6
Submitting a file upload
Available after 6 Oct at 0:00

1. Project management:
    - Create a project in trello or jira (add me as a collaborator, but do not assign implementation tasks to me :))
        - Use your software requirements document to come up with implementation tasks
        - Assign priorities and who is responsible.
        - Identify expected time for implementation and implementation details (programming language, APIs, data, etc.)
    - Follow Agile scrum methodology:
        - Sprints should be of 1 week (simply due to time limitations)
        - Assign a scrum master; for each sprint it should be a different person; if you have exhausted the list of people of your team, start from the beginning
        - Try to do a 5-minutes scrum every 2 days
        - **Define tasks (todos)**
    - Redefine requirements when necessary (update your requirements document, but keep a backup of the original one so that at the end you can trace the differences)
        - User stories are not necessary (we have a limited time and the project should be defined almost at 100%).
        - Update your project board continuously (I will be monitoring it on irregular basis)
2. Create a github repository:
    - Write code and upload it to the repository
    - Create at least 2 branches

3. Prepare a half a page report:
    - Outline what you have done, who has been the scrum master in these two weeks and what is your experience.
    - Note the url of your github and what branches you created and for what purpose.
    - Notice 1: It is not intended that you finish your full implementation within this two weeks.
    - Notice 2: If you are following a test-first development approach beware that we will cover testing and future class.
    - Notice 3: After this task is finished, you will most probably continue working on your project. Continue following Agile.

Rubric

Some rubric (3)

Criteria	Ratings	Pts
This criterion is linked to a learning outcome Project Is a project management system (jira, trello, github projects) used?	**3 PtsYes**On regular basis. **2 PtsGood**Yes, but irregularly **1 PtsBelow average**A project is created but not used at all. **0 PtsBad**No	3 pts

This criterion is linked to a learning outcome Repository Is a repository management system (github, gogs, etc.) used?	**3 PtsYes**On regular basis. **2 PtsGood**Yes, but irregularly **1 PtsBelow average**A project is created but not used at all. **0 PtsBad**No	3 pts
This criterion is linked to a learning outcome Methodology Are you following AGILE?	**6 PtsYes**All development follows all steps of the methodology; sprints are used / or kanban boards; a sprint master drives the development and motivates the team. **4 PtsGood**All development follows all steps of the methodology; sprints are used / or kanban boards; no sprint master or a sprint master that does not drive the project forward. **2 PtsBelow average**The development tries to follow all steps of the methodology; but sprints are not well defined. **0 PtsBad**The development does not follow AGILE.	6 pts

This criterion is linked to a learning outcome Version control Is your code under version control?	**6 PtsYes**The code is updated regularly; new branches are created for the right purposes; after code is written (and tested), it is committed and merged into the master correctly. **4 PtsGood**The code is updated regularly; branches are not used properly (e.g., the work is done on the master). **2 PtsBelow average**The code is not updated regularly; no branches are created. **0 PtsBad**The development does not follow AGILE.	6 pts
This criterion is linked to a learning outcome Tasks Are the tasks created linked to user and system requirements?	**6 PtsYes**All tasks are associated with requirements. And all tasks cover all requirements. **3 PtsGood**The tasks cover partially the requirements. **1 PtsAcceptable**Tasks are logical but do not cover the requirements. There is a big disconnect between the two. **0 PtsBad**Either tasks are not created or the tasks are illogical and are not connected to the requirements.	6 pts

Task 5 - Implementation and testing

Due 6 Nov by 21:00
Points 6
Submitting a file upload
File types pdf
Available after 27 Oct at 14:30

1. Continue the implementation:
    - Select tasks from the ToDo list and implement them
    - Add new tasks to the ToDo list
2. Test the software:
    - Define at least 5 unit tests and at least 3 integration tests for your code
    - Define the test cases, the test data and the results
    - These tests should run automatically. Depending on the language you are writing your code in, use the appropriate testing tool.
3. Update the repo on regular basis.
4. Demonstrate the testing: present your tests in front of the class during the practical session after the deadline
5. Prepare a maximum of one page report to outline the test cases, test data, results and the why you decided for these tests

Rubric

Task 5 [max 6 points]

Criteria	Ratings	Pts
This criterion is linked to a learning outcome Unit Tests Are the unit tests correctly defined?	**2 PtsExcellent**All of them are perfectly defined, with the right input and expected output. The unit tests are on the level of a unit and not integration tests. **1.2 PtsGood**3 or 4 are ok. The rest are not well defined in terms of input / output. However, they are all at the level of a unit (and not component). **0.8 PtsAverage**3 or 4 are ok. The rest are not well defined in terms of input / output. At least 1 and at most 2 are not defined on the level of a unit, but on component level. **0 PtsBad**None of them are well defined/make sense.	2 pts
This criterion is linked to a learning outcome Component/Integration Tests Are the integration tests correctly defined?	**2 PtsExcellent**All of them are perfectly defined, with the right input and expected output. The component tests are on the component level and not on a unit level. **1.2 PtsGood**3 or 4 are ok. The rest are not well defined in terms of input / output. However, they are all at the level of a component. **0.8 PtsBear minimum**3 or 4 are ok. The rest are not well defined in terms of input / output. At least 1 and at most 2 are not defined on the component level. **0 PtsBad**None of them are well defined/make sense.	2 pts

This criterion is linked to a learning outcome Implementation Tracing implementation progress	**2 PtsExcellent**Implementation is clearly based on the tasks in todo and the software grew at steady paste. **1.5 PtsGood**The software is implemented with a steady progression; not all implementation follows the tasks defined in the todo list. **1 PtsAverage**The implementation was sporadic but the software evolves over time; all or major part of the implementation follows the tasks defined in the todo list. **0.5 PtsBelow average**The implementation was sporadic and the software did not progress extensively; not all implementation follows the tasks defined in the todo list. **0 PtsBad**No implemention	2 pts

Total points: 6

Task 6 - Scaling (up) your system

Due 13 Nov by 21:00
Points 3
Submitting a file upload
File types pdf
Available after 3 Nov at 0:00

Until now you should have implemented or are implementing parts of your software that, at the end, you should connect in a well-orchestrated system that fulfills the overall objective. The different components, models are perhaps on one computer. This task is about organising these building blocks in an effective distributed software, following the architecture we have seen.

To do so, you should develop (at least):

1. One server that provides the interface. It can deal with user requests (sending them to the orchestrator), check for errors, store data, etc. This server should not call directly the AI component.
2. One or more servers to contain the AI component

3. One or more servers to host the orchestrator. The orchestrator can be implemented using queue services, such as RabbitMQ or using a database, or other ways to provide an effective intermediator between the interface and the AI component.
    - It is not required to use multiple machines; one machine can host many different logical servers.
    - This distributed implementation can be done locally (using your machines) or on the cloud. If you wish to do that, please let me know.
4. Prepare 1 page preliminary report.

Rubric

Task 6 rubric [max 3 points]

Criteria	Ratings	Pts
This criterion is linked to a learning outcome Distribution Distribution of the developed system	**1.5 PtsExcellent**Each component is hosted on a different server. **0.75 PtsGood**Two components are hosted on different servers. **0 PtsBad**The code is not distributed.	1.5 pts

This criterion is linked to a learning outcome Interfacing Interfacing with other services.	**1.5 PtsFull marks**Communication between different components is facilitated by either a centralized point, a peer-to-peer fashion or via APIs and services. **0.75 PtsOK**The interface directly communicates with the AI component. **0 PtsBad**Everything is implemented as a monolithic script.	1.5 pts

Total points: 3

Task 7 - Wrapping up

Due 20 Nov by 21:00
Points 2
Submitting a file upload
File types pdf
Available after 10 Nov at 12:59

Finalize your code:

- Clean your code
- Document your code
- Ensure all code works (run tests)
- Make sure everything is committed to your repository and that it is in the master branch

Operation:

- Make sure your software runs as designed
- Check and update your requirements document to make sure they correspond to the software design

AI:

- Ensure that the AI model(s) work(s) and that the appropriate APIs are in place
- Check that your models are the latest version/update

Value Sensitive Design?

- **Would you have conducted the design, development and testing differently considering the VSD principles?**

Environmental aspects (Comment on the carbon footprint of your project):

- **What can you say about the AI component?**
- **What can you say about the non-AI part of your system?**

Teamwork:

- **What was/were the main role/s and tasks of each member of your team**
- **How did you ensure that you work as a team**
- **What were the major challenges and how did you overcome them**
- **Anything else that shows your team effort in achieving the objectives of this task**

Report:

- Compile your final report by concatenating the intermediate reports (for Task 1 to 6) and add the part related to this task (Task 7)
- Revise them to have a coherent story (no need to rewrite them)
- Add a section (half a page maximum) about teamwork and max half a page on value sensitive design and environmental impact (answering the questions above).

Rubric

Task 7 rubric [max 2 points]

Criteria	Ratings	Pts

This criterion is linked to a learning outcome Final report submitted on time	**1 PtsFull marks**The report is submitted on time. **0.5 PtsLate**The report is submitted late but before the 28/11/2025 EoD. **0 PtsNo marks**The report is not submitted by the end of the course (05/12/2025, EoD).	1 pts
This criterion is linked to a learning outcome Teamwork	**2 PtsFull marks**A half-a-page section on team work is submitted - it is clear and identifies the role of each team member **1 PtsOK**A half-a-page section on team work is submitted - it does not clearly outline the role of each team member **0 PtsNo marks**No section on teamwork.	2 pts
This criterion is linked to a learning outcome VSD	**3 PtsFull marks**All 3 aspects are in the report. **2 PtsGood**Reflection on two out of the three **1 PtsOK**Reflection on 1 out of the 3 **0 PtsNo marks**No reflection on VSD	3 pts
This criterion is linked to a learning	**3 PtsFull marks**Reflection on both AI and non-AI aspects **1 PtsOK**Reflection on one but not both. **0 PtsNo marks**No reflection on environmental impact	3 pts

outcome Environmental considerations		

Total points: 9