

Last updated on: 19 September 2021

Students are not supposed to read this document all at once. Kindly refer to "student tasks" document to keep track of what needs to be done every week. In any case, before each exercise, we will explain what needs to be done. This document explains the exercises and provides submission instructions. Also, it serves as a reference for students to ask any doubts.

## Exercise 1: Requirements collection

---

By the end of this exercise, students should have a good understanding of how to write user stories, draw use-case diagrams. Kindly note that students within teams are supposed to collaborate in finishing these tasks.

Task 1: Each team should collaborate in writing **at least six** user stories. You can also write more if you like. Next, classify which user stories correspond to functional requirements and which to non-functional ones. In case there are no non-functional user stories, then write at least three (3: functional + 3 non-functional).

Task 2: Each team should collaborate in creating a use case diagram for the authentication feature. You should use correct notations (lines, arrows, rectangle, ovals, etc) and have **at least one** actor and **two use cases**. Keep in mind that this diagram will be extended throughout the course, as the new requirements are introduced. You can use any tool for creating such a diagram. There are several options, you can use tools like Lucidchart (Google doc integration available) or simply draw it on a piece of paper and take a picture. A better choice would be to create a shared Google doc with your teammates as you can keep modifying the diagram (and class diagram that comes in a later exercise) as we make progress in the project.

Task 3: Write use case descriptions for each identified use case.

**Start and Deadline:** 29 September - 6 October

How to submit: For this exercise, you can prepare one document with answers to all three tasks. You can make a PDF if you prefer. In your project repository, there is a "Deliverables" folder. Kindly make a subfolder for exercise one and put this document there. Notify your assistant once you want him to take a look at it. You can spend about a week finishing this exercise. In the subsequent meeting, your assistant can give you feedback on it.

## Exercise 2: Responsibility-driven design

---

By the end of this exercise, students should understand how to identify the right domain entities from the requirements, and know why and how to prepare CRC cards. The following podcast from P2 on "Object-oriented design principles" could help with this exercise: <https://tube.switch.ch/videos/f74aedae>

Task 1: Discuss within the team and identify main domain concepts from the requirements. Make sure they represent domain concepts and not technical components. Create **at least two** CRC cards and show responsibilities and collaborators. You could prepare a shared document for your team where you will keep all the CRC cards. You can add/update CRC cards as the project progresses.

Task 2: Prepare UI mockups as the "User interface design" lecture comes at the end when most of the implementation work is finished. You could prepare a shared document for your team where you will keep all your current and future mockups for the application you are building. There are various tools to create mockups. The simplest way could be to use pen and paper! More sophisticated mockups can be created with tools, such as Balsamiq.

**Start and Deadline:** 13 October - 20 October

How to submit: For this exercise, you can prepare one document (digital or on paper and take a photo) with answers to both tasks. You can make a PDF if you prefer. In your project repository, there is a "Deliverables" folder. Kindly make a subfolder for exercise two and put this document there. Notify your assistant once you want him to take a look at it. You can spend about a week finishing this exercise. In the subsequent meeting, your assistant can give you feedback on your progress.

## Exercise 3: Project management

---

At the end of this exercise, students should be familiar with various project management-related tasks, especially, they should understand how to manage their workload and distribute it equally among all team members. They will also learn to track their own progress. You will also get familiar with the tools that are used to manage those tasks.

Task 1: Prepare a Trello board or use GitHub projects to add and manage identified user stories. Prepare a product backlog and sprint backlog. Add corresponding tasks against each user story, assign those to your team members. Share a link with us. Use your Trello board or GitHub projects extensively throughout the project.

**Start and Deadline:** 20 October - 27 October (Regardless of these deadlines, students will already be familiar with Trello and GitHub and they should start using any such project management tools earlier)

How to submit: There's nothing to submit for this exercise.

## Exercise 4: Modeling objects and classes

---

By the end of this exercise, students should have a good understanding of class diagrams and various notations used in a class diagram.

Task 1: Add/update CRC cards to reflect the new functionality.

Task 2: Draw a class diagram from CRC cards for the ongoing sprint. There should be **at least five** classes. You must use composition and aggregation, and show inheritance. You can use any tool to create a class diagram (Google docs has a Lucidchart integration) or simply use pen and paper! Kindly remember that you will keep updating this diagram as the project progresses. Therefore, you could use a shared document you created earlier.

**Start and Deadline:** 27 October - 3 November

How to submit: For this exercise, you can reuse the existing shared document (digital or on paper and take a photo) with answers to both tasks. You can make a new PDF with the latest changes. In your project repository, there is a "Deliverables" folder. Kindly make a subfolder for exercise four and put this PDF there. Notify your assistant once you want him to take a look at it. You can spend about a week finishing this exercise. In the subsequent meeting, your assistant can give you feedback on your progress.

## Exercise 5: Modeling behavior

---

By the end of this exercise, students will get familiar with another UML diagram: an activity diagram.

Task 1: Extend the existing use case diagram for the ongoing sprint.

Task 2: Update the existing class diagram to reflect the new functionality.

Task 3: Prepare an activity diagram for at least one functionality of the ongoing sprint. Make sure you use correct notations.

**Start and Deadline:** 3 November - 10 November

How to submit: For this exercise, you can reuse the already created document (digital or on paper and take a photo) with answers to all tasks. You can make a PDF if you prefer. In your project repository, there is a "Deliverables" folder. Kindly make a subfolder for exercise five and put this document there. Notify your assistant once you want him to take a look at it. You can spend about a week finishing this exercise. In the subsequent meeting, your assistant can give you feedback on your progress.

## Exercise 6: Software metrics and estimation

---

By the end of this exercise, students will learn a simple cost estimation model: Cocomo.

Task:

Tells us in the shared Google Sheet:

- ☐ How many lines of code your team has written
- ☐ Estimate and show the cost using use case points and the Cocomo model for the ongoing sprint

**Start and Deadline:** 8 December - 15 December