Group Project Assignment
Fog and Cloud Computing
Autumn 2024/2025 - IFI/UiO

1. Introduction

This document describes the group project to be developed by each group[1] (2 students) for the course Fog and Cloud Computing (IN5700 & IN9700) at UiO in Autumn 2024.

The project simulates a smart garbage collection system with the following components:

- a client (garbage truck) who carries a smartphone with WiFi capability,
- a central garbage site holding all the data somewhere in the cloud,
- two garbage cans, each placed on a different street.

The system layout consists of three streets: two horizontal streets (one running from right to left at the top, and the other from left to right at the bottom), each with a garbage can, and one vertical street that connects the two. The client (garbage truck) navigates these streets to collect waste from two garbage cans. The data related to the garbage collection is then stored at the garbage site. Students are expected to explore and compare the time required to complete the garbage collection task using different approaches, including cloud-based, fog-based, and no-garbage solutions. The videos on the course webpage provide a visual demonstration of the client's (garbage truck's) route and how the system operates.

2. Goal

In order to implement the system, three solutions are to be considered: cloud-based (with slow messages), fog-based (with fast messages), and no garbage to be collected from the two cans (no garbage). The final state of each solution is represented:

- Figure 2 represents the final state of a cloud-based solution;
- Figure 3 represents the final state of a fog-based solution; and
- Figure 4 represents the final state in which there is no garbage to be collected from both cans.

Each group should develop a single application using the OMNET++ simulator. Within this application, solutions are selected via options defined in the configuration file 'omnetpp.ini,' as shown in Figure 1: 'GarbageInTheCansAndSlow' (cloud-based solution with slow messages), 'GarbageInTheCansAndFast' (fog-based solution with fast messages), and 'NoGarbageInTheCans'. Students should also compare the solutions by analyzing their delay (the time taken to send and receive the messages). Additionally, each group is required to present the final state for each solution (cloud-based, fog-based, and no garbage) in their written report. The reports should follow the provided template and questions in the template should be answered accordingly.

The possible messages in the system and their meaning are as follows:

- "1-Is the can full?" and "4-Is the can full?" - These messages (each one) are lost exactly 3 times, and they are sent from the smartphone (inside a truck) to the cans. These messages are used to check if there is garbage in the cans that need to be collected.
- "02-NO/3-YES" and "05-NO/6-YES" – These messages are transmitted from the cans to the smartphone in response to messages 1 and 4. If there is no garbage to collect, the cans will send "2-NO" and "5-NO" messages. However, if garbage needs to be collected, the cans will send "3-YES" and "6-YES" instead.
- "7-Collect garbage" and "9-Collect garbage" - These messages are sent to the cloud when there is garbage to be collected from both cans. The sender of these messages varies depending on the solution: in the fog-based approach, the messages are sent by the cans, while in the cloud-based approach, they are sent by the smartphone. In the scenario where there is no garbage to collect, these messages are not generated.
- "8-OK" and "10-OK" – These messages confirm the previous two (messages 7 and 9), indicating that the

[1] The existing groups and their members are available on the IN5700 webpage.

garbage in both cans will be collected by the truck.

For communication with the cloud, there are two types of connections and messages: slow and fast. The connection between the cloud and the smartphone is classified as slow, with the messages exchanged between them referred to as slow messages. The connection between the cloud and the cans is classified as fast, and these messages are called fast messages. Additionally, all other messages exchanged between the remaining components of the system are also classified as fast messages.

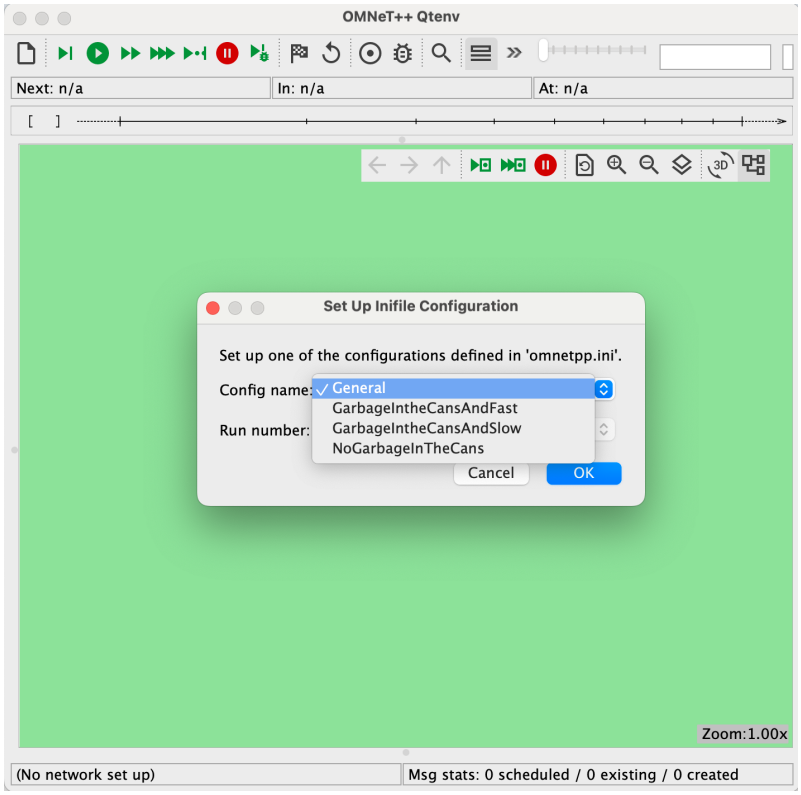
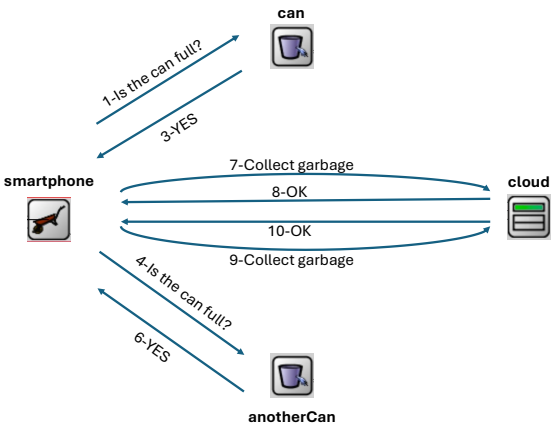


Figure 1-Options available in the file "omnetpp.ini".

The messages exchanged in the cloud-based solution (with slow messages) are as follows:



Schema 1.

In the cloud-based solution, the smartphone inside the truck sends messages 7 and 9 to the cloud and receives the corresponding acknowledgments (slow messages).

Please ensure that your simulation matches the one shown in the videos exactly, except for the delay values, which are allowed to differ. Each group should describe the technologies assumed for communication between the system entities and calculate the resulting delays accordingly. The final results should then be shown at the end of the simulation. The final state of the cloud-based solution is represented in Figure *2-Illustration* (in 2D) of the final state in the "cloud-based" (with slow messages) solution.Figure 2.

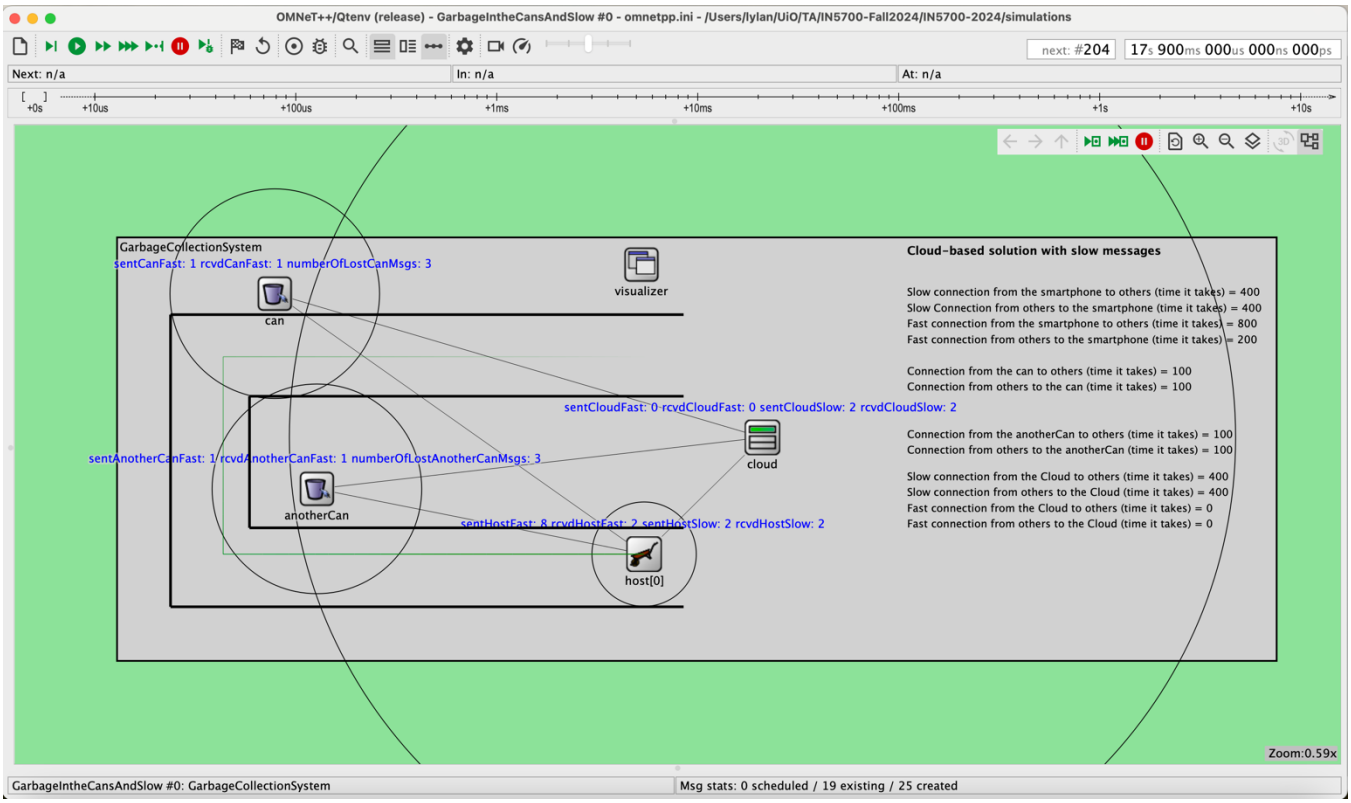
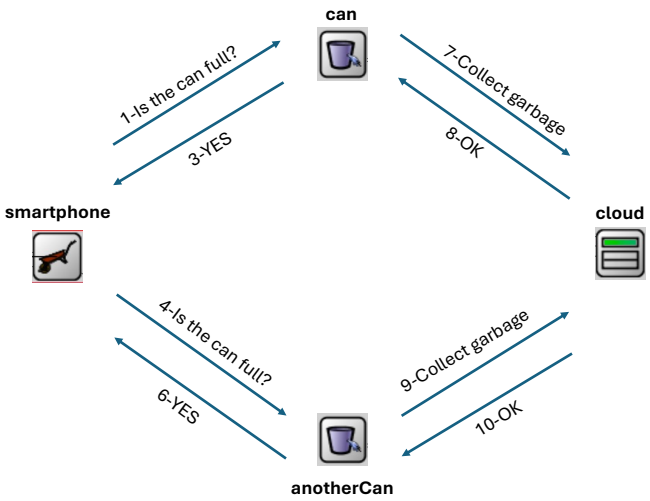


Figure 2-Illustration (in 2D) of the final state in the "cloud-based" (with slow messages) solution.

The messages exchanged in the fog-based version (with fast messages) are as follows:



Schema 2.

In the fog-based solution, the can and another can send messages 7 and 9 to the cloud, respectively, and receive the corresponding acknowledgments (fast messages). All other messages follow the same pattern as in the cloud-based solution. It is important to note that messages 1 and 4 must be lost exactly three times, with the cans successfully receiving them on the fourth attempt.

The final state of the fog-based solution is represented in Figure 3.

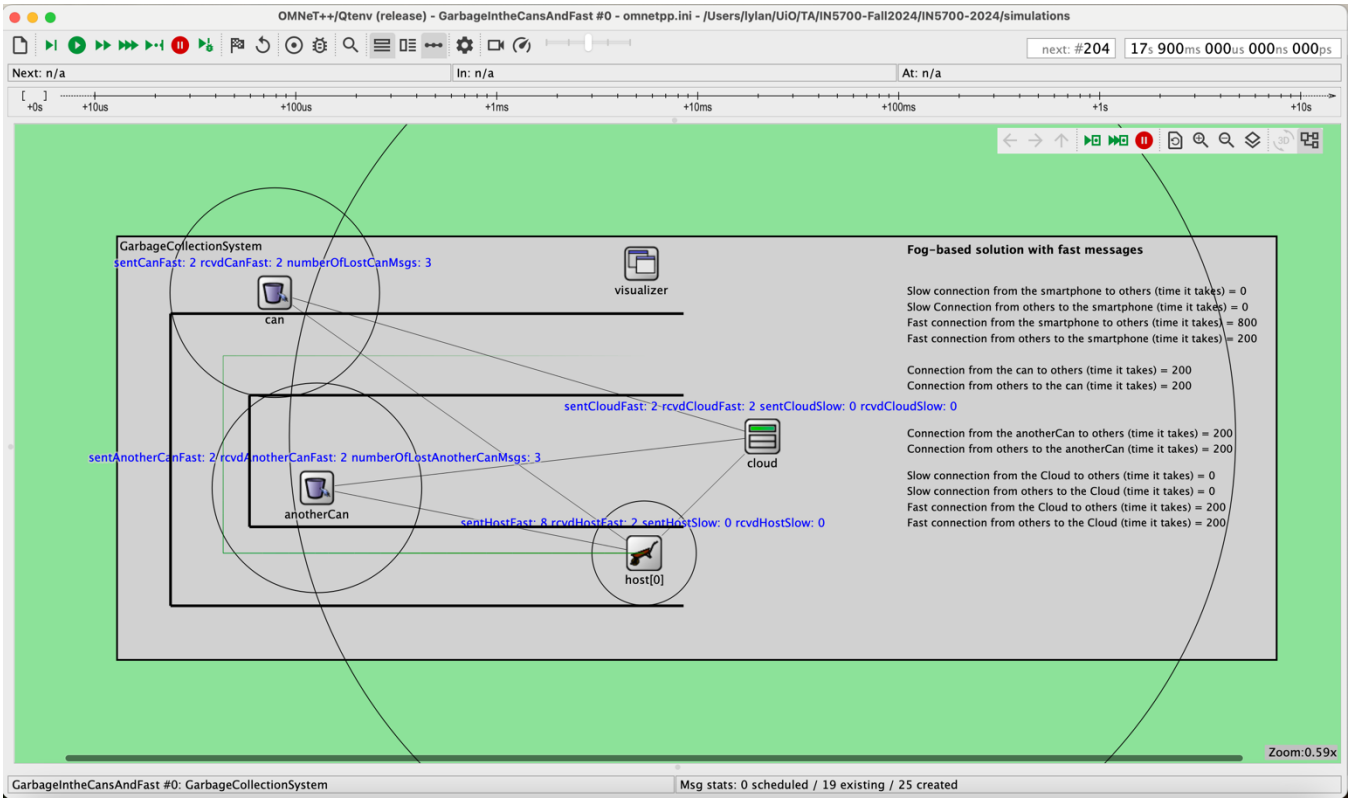
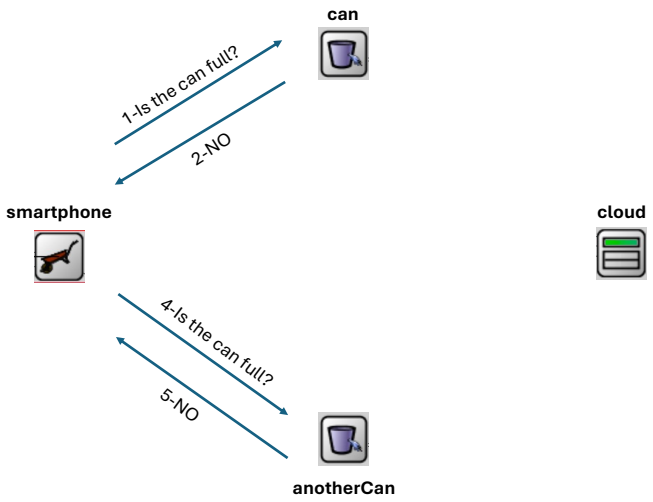


Figure 3-Illustration (in 2D) of the final state in the "fog-based" (with fast messages) solution.

In the no-garbage scenario, messages 3, 6, and 7-10 are not generated because there is no garbage to collect. The messages exchanged in this version are as follows:



Schema 3.

The final state of the cloud-based solution is represented in Figure 4.

Please note that in all solutions, the messages "1-Is the can full?" and "4-Is the can full?" are lost exactly three times. For a detailed overview, refer to the videos available on the IN5700 webpage:

- GarbageInTheCansAndSlow: Cloud-based solution with slow messages;
- GarbageInTheCansAndFast: Fog-based solution with fast messages; and
- NoGarbageInTheCans: No garbage collection scenario, where no messages are exchanged.

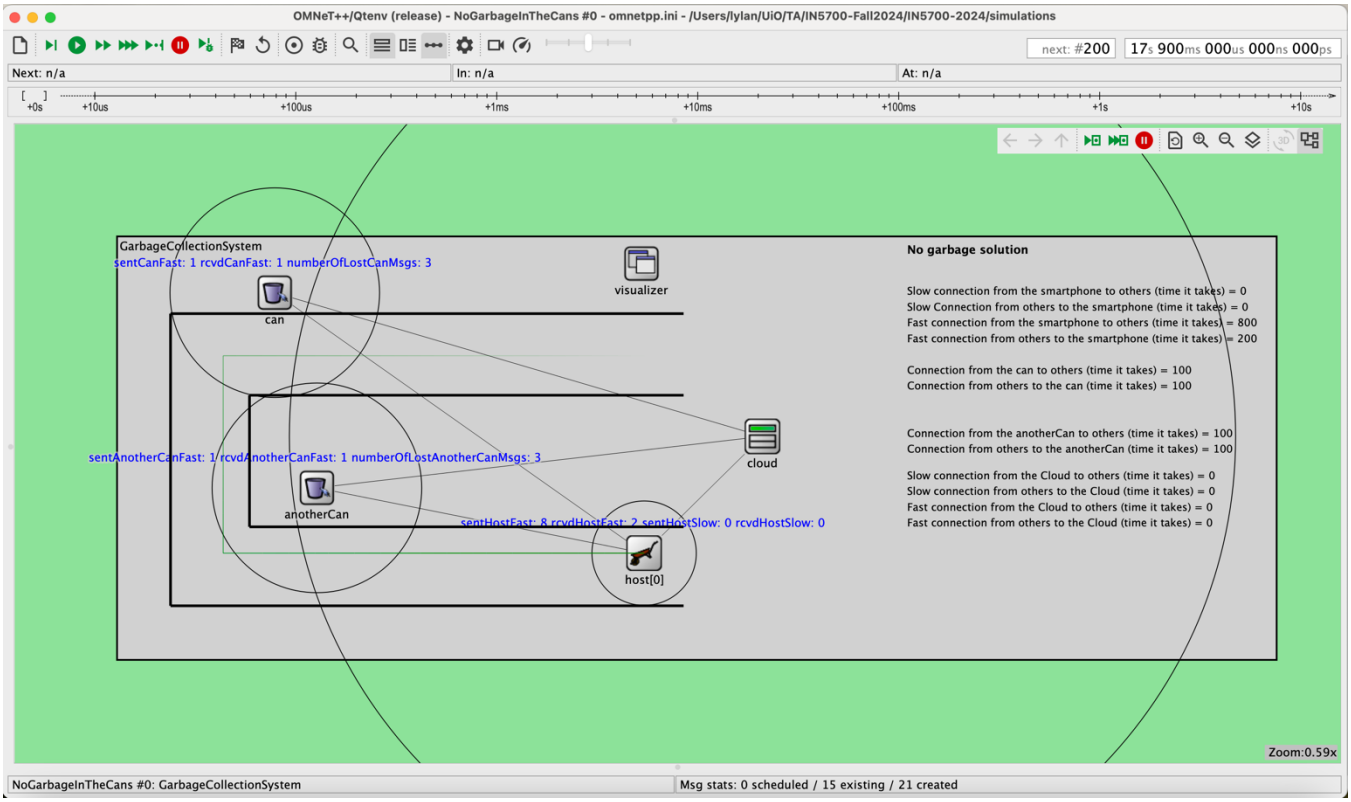


Figure 4-Illustration (in 2D) of the final state in the "no garbage" solution.

Students are required to implement the system exactly as shown in the videos. Each group must choose delay values with justification for the chosen values and calculate the final delays accordingly. Students should then compare all the solutions in terms of the delay taken by the messages.

3. Functionality

As already mentioned, there are 3 solutions to be considered: i) the cloud solution in which it is up to the smartphone to inform the cloud that the cans have garbage, ii) the fog solution in which both cans send the message directly to the cloud, and iii) the no garbage scenario. All these solutions are to be picked using the configuration file ("omnetpp.ini").

3.1 Cloud-based Solution

The cloud server receives messages sent by the smartphone informing it that the cans should be emptied. Once the cloud server receives the above-mentioned messages, it replies saying OK indicating that the cloud will take care of the issue.

3.2 Fog-based Solution

In this case, the fog-based solution, it is up to both cans to inform the cloud that they have to be emptied. So, the cloud receives two messages sent by both cans.

3.3 Results

The results to be obtained should be those illustrated in the videos (corresponding to Figure 2, Figure 3, and Figure 4, shown previously). Please note that the videos serve as the primary reference, and all the details presented in the videos should be followed during the simulation.

The written report to be provided along with the source code (see Section 9) should have no more than 5 pages (a template is provided) and it should address a comparison between the solutions (cloud-based, fog-based and no garbage to be collected) regarding the delays of the messages sent and received. Note that the template report has more sections (not mentioned here) that must also be provided.

3.4 OMNT++ Functionality

The videos mentioned above show which messages are supposed to be exchanged between the smartphone, both cans, and the cloud (in all the solutions).

Please, take into account the following aspects/suggestions:

- 3 messages being sent to both cans are lost;
- the movement of the smartphone is described in an xml file;
- use options to configure the application (as shown in Figure 1);
- the connections between the components of the simulation (smartphone, both cans, and cloud server) are distinct, depending on whether slow or fast messages are considered, in terms of their data rate and delay; take into account that all the values must be defined accordingly and the values shown in the videos (as well as in Figure 2, Figure 3, and Figure 4).

Obviously, each group should provide a project that has exactly the same messages and output that is illustrated in the videos. There are baseline and advanced functionalities as described below. Accordingly, each group reports the findings in the report. Please note that the values of the channels connecting the several components (smartphone, both cans, and cloud server) must have justification and should not be any arbitrary values.

4. Baseline Functionality

All the aspects should be considered except the movement of the smartphone; thus, the baseline functionality shows all the messages flowing among the components being all stationary.

5. Advanced Functionality

In addition to the baseline functionality, the movement of the smartphone (the client) should also be considered.

6. Development Environment

The simulation must use OMNeT++ 6.0.3 (https://omnetpp.org/). The simulator runs on Windows, Linux, and Mac. The project has been tested on the following systems:

- Windows 11 with OMNeT++ 6.0.3 and INET v4.5,
- Mac Sonoma 14.5 with M1 Max chip with OMNeT++ 6.0.3 and INET v4.5, and
- Windows 7 with OMNeT++ v5.6 and INET v4.

7. Suggested Development Stages

The faculty suggests the project be developed in several stages in the lab classes. Some extra work, done outside the labs, may be needed.

Start by installing all the software needed. Note that the basic aspects of the simulator will be explained in the lab classes. It is advised that students read the OMNeT++ Simulation Manual as well as the related documentation that is available online. Each time you add some functionality you should test it to see if the simulation still works.

8. Classification

The classification of the project will depend on the quality of the architecture of the simulator, the written code (i.e., how clear the code is, comments, size of the methods, etc.), and simulation functionality. In particular:

- the basic functionality has a maximum classification of 16 (scale from 0 to 20),
- the advanced functionality has a maximum classification of 20 (scale from 0 to 20).

Note that the above values are maximum; usually, taking into account the aspects mentioned above, makes the classification of the project to be inferior.

9. Deadlines and Deliverables

The deadline for delivering this project is the 8th of November 2024 (Friday) at 23h59. Please use the link https://devilry.ifi.uio.no/ to deliver your project.

This project must include the following deliverables:

- the source code of the simulation (the complete project folder),
- the videos of your simulation (similar to those shown on the webpage of IN5700), and
- a written report according to the template provided to students (also on the IN5700 webpage).

Please place all deliverables in a single folder, then compress it into a ZIP file. Name the ZIP file as follows: "**groupXX-FC-2024.zip**," where 'XX' represents your group number (available on the IN5700 webpage).

10. Addendum

For those who do not have another group member (i.e., just one person) because the number of students registered in the course is odd, the project is the same with the following exception: the advanced functionality does not include the movement of the smartphone. All the other aspects are to be evaluated. Accordingly:

- the basic functionality has a maximum classification of 20 (scale from 0 to 20).

All the rest is equally applicable.