

Agile report – Group 1

Strategy Planning

A major part of strategic planning has already been achieved in the first part of the project. The diagrams produced during this phase will guide the team throughout the process of system implementation and GUI design. Firstly, the use case diagram contains different users of the system and their use cases (user needs). It gives the gist of the system and what it should be capable of doing, as the use cases of these users must be fulfilled for the system to be functional. This allows the team to identify the tasks they need to do and divide them equally among the members.

Secondly, the class diagram, also designed during the first part of the project, provides a static view of all the classes necessary in the system. The diagram identifies all the attributes and methods in each class for the system to be able to provide functionalities to satisfy the user needs described in the use case diagram. This diagram will act as the blueprint of our system and will be used to design all the classes and their methods and attributes.

While the class diagram provides a static overview of the classes, the three sequence diagrams designed by the team show how these classes interact with each other using their methods and attributes, to produce a particular functionality and satisfy a use case described in the use case diagram. This diagram will assist the team in designing the GUI as it displays how the methods in the classes will be used to produce dynamic results in the working system.

Besides from these diagrams, the tasks during this phase will be divided in the following manner. One of the members of the team will be in charge of the GUI (Connor), another will take charge of testing (Jeno) and the other will take charge of the report (Dikshyanta). Each of them will be responsible for their assigned parts and would work with their teammates iteratively to progress that aspect. Conversely, they will also work with their teammates in the other two aspects that they will not be in charge of. The module practical every week will be used to convey updates and queries and work together. A WhatsApp group would supplement by allowing online meetings and text exchange.

Continuous Team Iteration

The team will use an agile approach to the project and will keep it flexible and iterative between planning and implementation. This is because the team understands that while designing the system, the team might come across an obstacle which might require the team to tweak the design without losing functionality. A sprint cycle will be used by the team. The entire project is divided into twenty sprint periods and each sprint period contains a small portion of work to be done. The whole sprint cycle is listed in the Work Plan section of this document. During a sprint period, the team will work together to achieve the objectives mentioned in the sprint. After completing a sprint, the team moves on to the next sprint. The process will continue until the team undergo the twenty sprints defined in the cycle and the project is expected to be completed. In practice, more sprints can be added as new objectives could be identified during the design of the system.

The use of such iterations will allow us to understand each other's ideas better. The process of continuously iterating as a team will allow all the teammates to understand the intended logic behind every part of the system and incorporate the other parts to synchronize with it. This will enable the team to produce a smooth system with a concrete and bug-free flow of logic.

Team Communication

The team will communicate both face-to-face and online during the implementation of the second part of the project. The team will meet every week on Monday, during the practical session of this module, where the team members will work together to achieve the objective defined for that sprint period in the sprint cycle. A stand-up meeting will take place before starting the practical session each week, where the team members will go over important tasks that have been finished, are ongoing or need to be started. This will allow each team member to understand the team's progress in different areas of the development process and enable the team to prioritize tasks in a more efficient manner. This meeting will also be used to communicate with the lecturer (James Lear) and discuss any ideas or confusions and clarify them.

The team will communicate online using Microsoft teams and WhatsApp. Teams will be used to host group meetings whereas WhatsApp will be used for text exchange. The team will also set up a GitHub repository so that the members can upload any changes, new files, etc. onto a common place and communicate these changes to the other teammates by writing notes about these changes in the update summary.

Simplicity

Our main objective is to practice simplicity throughout the project, based on the “maximising the amount of work not done” principle of Agile software development. Maintaining simplicity is a crucial objective to avoid our software structure becoming unnecessarily complicated. As we have developed our UML diagrams in the design phase, we have successfully understood and analysed the requirements of our system. In order to implement the system in the development phase, the team will utilise the principle of simplicity by focusing on the development of the simplest solution that might meet the previously identified system requirements, along with the prioritisation of delivering a Minimum Viable Product. The developed MVP then might be improved if necessary through incrementally building on it. Maintaining simplicity is an indispensable objective in order to deliver the simplest solution for this project, instead of wasting time resources on refining every fine detail of the product that might be not relevant to meet the requirements. Utilizing the SOLID principles in our project is also crucial task to avoid needless complexity and code repetition as well as maintaining the reusability of our software. Throughout development, we must ensure all five SOLID principles of object-oriented design going to be applied to our software system. Considering our software's structure, we must provide great attention to utilising the Single-Responsibility Principle (SRP) and the Open-Closed Principle (OCP) of SOLID to maintain a low-level coupling and high-level cohesion between software modules as well as implementing a system that open for extension but closed for modification.

Work Plan

The team will begin the development of the system as the design phase has been concluded. Aiming to start the implementation phase on 3rd January 2022. The work plan is based on 1 week-long sprints, that would provide an opportunity to discuss individual progression and any obstacles in development during the weekly lab sessions. These weekly lab sessions will also provide an opportunity to allocate short-term tasks and responsibilities among team members.

The developer team identified the following subtasks within the project that required to be delivered in order to successfully complete the project:

- Implementation of low-level class design (Sprint no. 1)
- Develop GUI in Java FX that is suitable to implement the functional requirements related to each of the different types of users that will interact with the system (Sprint no. 2 & 3)
- Connect class design to GUI, implementing logic that would result in meeting the system requirements

The above-mentioned high-level tasks are represented in the following product backlog, displaying the desired functionalities in detail as well as their priority, sprint number and their status of implementation. Delivering the system will also require thoroughly testing the product. Testing is going to be represented in an external document, including all the relevant test cases and the outcome of the process.

Product Backlog						
ID	As a..	I want to be able to..	So that..	Priority	Sprint	Status
1	Warden	Select a room and set its cleaning status to clean	I can register that the room has been cleaned	Must	Sprint no. 4	Done
2	Warden	Select a room and set its cleaning status to dirty	I can register that the room has not been cleaned	Must	Sprint no. 4	Done
3	Warden	Select a room and set its cleaning status to offline	I can register that the room is not available	Must	Sprint no. 4	Done
4	Warden	Select a room and view its status	I can identify whether the room is occupied or unoccupied	Must	Sprint no. 5	Done
5	Warden	Select a room and view its lease information	I can identify whether the room is leased to a student as well as the appropriate details of the agreement	Must	Sprint no. 5	Done

6	Warden	Select a room based on its room number	I can identify rooms assigned to a specific room number	Must	Sprint no. 5	Done
7	Hall Manager	Select a room and view its status	I can identify whether the room is occupied or unoccupied	Must	Sprint no. 6	Done
8	Hall Manager	Select a room and view its lease information	I can identify whether the room is leased to a student as well as the appropriate details of the agreement	Must	Sprint no. 6	Done
9	Hall Manager	Select a room based on its room number	I can identify rooms assigned to a specific room number	Must	Sprint no. 6	Done
10	Hall Manager	View total number of rooms	I can identify the number of rooms available	Must	Sprint no. 6	Done
11	Hall Manager	View the address of a specific hall	I can identify the address of a hall	Must	Sprint no. 7	Done
12	Hall Manager	View the phone number of a specific hall	I can identify the phone number of a hall	Must	Sprint no. 7	Done
13	Hall Manager	View the name of a specific hall	I can identify the name of a hall	Must	Sprint no. 7	Done
14	Hall Manager	View the number of a specific hall	I can identify the number of a hall	Must	Sprint no. 7	Done

15	Hall Manager	Create a new lease	I can register a new rental agreement in the system	Must	Sprint no. 8	Done
16	Hall Manager	Set lease duration	I can define the length of the rental agreement	Must	Sprint no. 8	Done
17	Hall Manager	Delete a specific lease	I can remove rental agreement from the system	Must	Sprint no. 8	Done
18	Hall Manager	Allocate lease to a student	I can allocate rental agreement to a specific student	Must	Sprint no. 8	Done
19	Root user	Access all the system features that have been implemented for Warden and Hall Manager	I can access all the system features that have been implemented for Warden and Hall Manager as a Super User	Must	Sprint no. 9 & 10	Done