**Project Overview**

Student Housing BV have their own different buildings where students (their clients) can stay during their study in the Netherlands. Every building consists of individual rooms for each client but also shared facilities such as kitchen, toilets, bathroom, open-area, storage space and more. For some time now they have started receiving complaints about certain issues and they would like to find a better way to make things work. Their vision is of an application in which clients can record and see agreements made between then with the possibility to see the house rules and file complaints anonymously. It will also save a lot of time and remove the need of sending an employee every so often that goes to the buildings to update house rules and gather any complaints.

The current presented problems are related to:

* Appointed persons not cleaning shared facilities
* Groceries not done or paid for shared items such as toilet paper, soap, etc.
* Garbage disposal is not done on time
* Unannounced parties, gatherings, etc.

It can be assumed that there are minor issues which have not been mentioned at all to the company.

The aim of this project is to find a software solution which will easy up the clients and decrease the amount of issues they face while also saving precious time of company employees and the trouble of having to manually work things out.

**Project Objectives**

Provide a fully working application that will solve most of the complaints and any minor ones while keeping in mind the current problems. Additional features are to be implemented to make the software better and more fitting for the users that it is made for. After discussing the information we have been provided with, the team came up with ideas in mind to resolve the issues and implement bonus features.

Required to have:

* Cleaning schedule that will include a client’s name and the facilities he has to clean in upcoming weeks.
* A grocery list with predefined shared products. Info whether they are available and whether it has been paid for them
* Garbage takeout schedule with a possibility of either mixing and displaying it on the grocery list schedule or a standalone window.
* Anonymous report screen so that each student can inform the company of any issues or complaints that they may be having.
* Events list in which everyone can create their own event be it a party/ gathering/ teambuilding/ etc. Each user will have a limit for a certain amount of time so that the system does not get over-spammed.
* House rules screen
* Admin (employee) window with the ability to see each event, scroll/read through all of the reports made, list apartments with their tenants (possibly add/remove ones) and the option to update house rules so that no employee has to manually go to each building and do it.

Optional to include:

* Preferably an option for each client to be able to re-schedule stuff/record agreements made between him and other member(s).
* Progress bar to the “Garbage takeout schedule” which using Arduino will determine how full is the trash can and indicate it 24/7.
* Adding/Removing apartments
* Added a database to the software
* More found throughout the workflow

**Constraints and expectations**

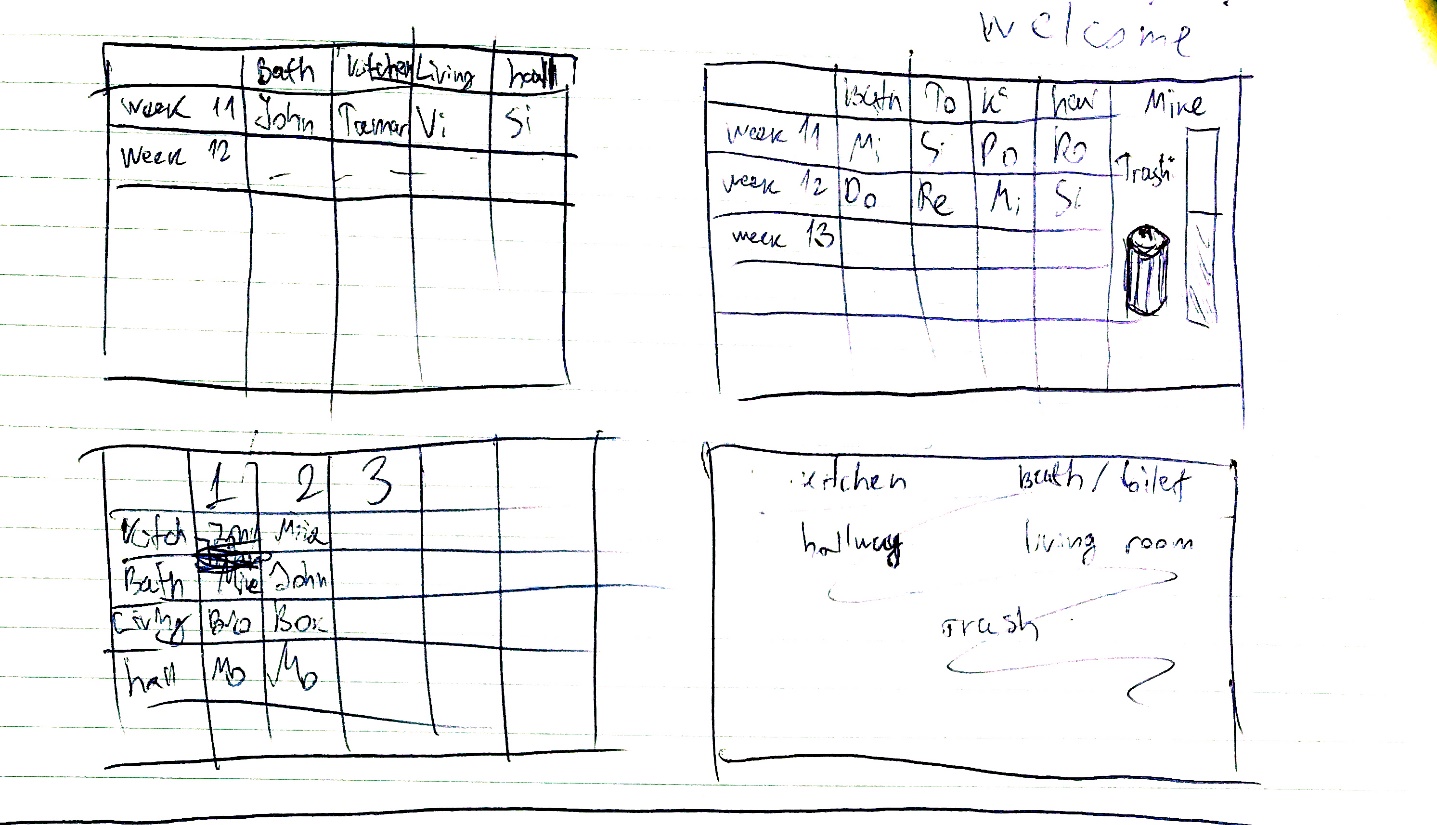
It is required of our team to keep the following constraints as agreed beforehand:

* The programming language is C# and the main application must be a Windows Form Application; an object-oriented programming must be applied for the application(s).
* Only theory covered during semester is expected to be applied for the application(s). For example, we do not have to connect our application(s) to a database and may assume an application is never closed to allow an employee to update the rules and gather any complaints.
* GitLab from Fontys ICT must be used for collaborating on the code base.
* Canvas must be used for determining and submitting the deliverables for the project.
* Weekly meeting(s) must be scheduled with the team’s assigned mentor to discuss the progression of the project, presence of all members is required.
* Minutes of meetings is delivered at latest 24 hours after the weekly meeting.

Active work split equally between all team members is expected. All members must be working on the deliverables. It is expected of the team to discuss, suggest and implement additional features besides the ones mentioned. The team should agree on implementing a feature and therefore also inform the mentor in such a way that it is acceptable. Good behavior towards one another and pro-active working with proper scheduling is highly recommended for better outcome in the end.

**Sketches**

Before starting to work on the interface we had to make some rough sketches with brainstormed ideas.

****

**User stories**

* As a tenant, I want to be able to see all events and gatherings happening at my housing unit so that I can schedule my time accordingly. I want to be able to tell others using said system whether I agree or disagree with certain organized events. If I wish to organize my own gatherings and inform fellow tenants, I need to easily be able to do so.
* As a tenant, I should be able to see the housing unit’s grocery list and check the availability of items.
* As a tenant, I want to be able to easily check the cleaning schedule. I need to be able to make agreements with the other tenants when I am too busy to handle said task.
* As an administrator, I want to be able to easily see reports so that I can go through as many as possible in a certain period of time.
* As an administrator, I want to be able to edit the house rules for every single housing unit so that I do not have to do it manually by going to each house’s address.

**Deliverables:**

* Week 1:

- Creating a project plan: Project analysis, initial sketches, wireframing;

- Discussing workflow with group members and brainstorming potential features for the application.

* Week 2:

- Implementation of the login system & creation of database tables to store our information

- Ability to add new/remove housing units and add/remove tenants to/from each unit.

- Creation of initial classes & connections with database

* Week 3:

- Implementation of cleaning & grocery sections for tenants

* Week 4:

- Addition of event section for tenants

- Handling the reports feature (Both tenant and administrator-side)

* Week 5:

- Polishing & debugging implemented components

- Addition of extra functionality to the application

* Week 6:

- Presentation of completed product