

**Class:** P-CB-S-03

**Subject:** ADA

**The group Member are:**

- Duarte Somsen **SN:** 4083148
  - Nour Khatib **SN:** 3962881
  - Omar Abou Dehn **SN:** 3560813
- 

## **Student Housing Project Analysis**

### Contents

1. Stakeholders:	2
2. Problem Description:	2
3. Functional requirement per end user.	3
Admin should be able to:	3
User should be able to:	3
4. Planning	4
5. Risk Analysis	6

## 1. Stakeholders:

- Student housing company which will pay for the app and administer it.
- Tenants (students), who are using the app that is provided by the land owners.
- The development team who are responsible for building the application.

## 2. Problem Description:

### **The Housing company is facing difficulties managing the student house**

The tenants are not following the appointed tasks; such as cleaning, taking out the garbage and doing groceries.

Some tenants are not contributing in the shared purchased items costs.

Tenants planning unannounced parties and events.

Increase of the spendage of energy due to tenants leaving the lights and/or heating on, when it is not necessary .

Difficulties for students in contacting the housing companies and vice versa in order to make complaints.

The involved parties in an agreement unable to keep track of the condition and details made in it, neither enforce it lawfully.

### 3. Functional requirement per end user.

Admin should be able to:

- Register the users and/or admins.
- Login/logout.
- Read complaints from users and reply to them.
- Communicate with the students making announcements.
- Get notified when room temperature reaches a specific threshold.
- Add/remove house rules.

User should be able to:

- Login/logout.
- View, add and remove products from the groceries list, and split the cost between tenants.
- Plan and respond to an event made by other students.
- Make a complaint anonymously to the admin.
- See and interact with the task schedule.
- See the common room temperature and receive notification when the spendage is above average.
- Make agreements with other tenants, and respond to agreements.

## 4. Planning

### Week 13

- ❖ *Create Git repository.*
- ❖ *Project analysis document.*

### Week 14

- ❖ *Create the wireframe*
- ❖ *Login/logout functionality: Determine the role of the current user (Student/Admin). Omar*
- ❖ *Create task distribution: Create functionality to split tasks among tenants. Duarte*
- ❖ *Create cost management functionality: to split the groceries cost between the tenants. Nour*

### Week 15

- ❖ *Create Events handling functionality: Creating events and a voting system for approval. Duarte*
- ❖ *Create communication functionality: E.g. The students make a complaint to the admin, and admin can contact the student(tenants). Nour*
- ❖ *Build the Agreements functionality: The student make/respond to agreements with other students.*

*Omar*

#### Week 16

- ❖ *Temperature monitoring system: notify concerned user in certain cases. Omar*
- ❖ *Creating the house rules list: modifiable by the admin. Duarte*
- ❖ *Modify the application based on the client feedback.*
- ❖ *Improve on the existing functionality.*

#### Week 17

- ❖ *Final improvements/touches.*
- ❖ *Deliver the final version of the application.*

#### Week 18

- ❖ *Present the application to the client.*

## 5. Risk Analysis

- **Connection problems.**

In case of internet problems, the development team will not be able to communicate properly and turn in new functionalities to the version control system (GitLab).

- **Sudden growth/change in requirements.**

The client may change his mind about a certain functionality or have a different opinion on how it should be implemented. This could lead to a waste of time and effort of the development team. To avoid that; the development team should discuss the wishes of the client and the software requirements in detail from the beginning.

- **Wrong time estimation**

If the development team lacks the experience, they may underestimate the time needed to deliver a certain functionality, which may result in a delay in the entire development process. To avoid this mistake; the team should build up experience and learn from similar past mistakes.

- **Hardware breakdown.**

The project is dependent on production hardware that can be essential to the creation of the project and without it, the project will fall e.g. laptops, Fontys servers, and gitlab...ect. a solution would be having alternative fast delivery hardware providers and enough saving money.

- **Miscommunication between group members.**

it can be receiving the information in a way that it is not intended to, or delivering the information in an ambiguous way, which mostly will lead to many meetings to elaborate the ideas repeatedly.

- **Group member leaving the team.**

The program can be dependent on a certain key person to make functionalities that he/she only knows how to make in the group. In that case losing this key person will lead at least for delay or maybe the inability for the

rest of the team to continue if the rest of the team couldn't learn fast enough or alter the program to avoid certain functionalities.

- **Productivity issues in the development team.**

If a member on our development team cannot keep up with what we decided to do. Like for example he cannot complete one of the features because we are not experienced enough to do it in a certain amount of time. We could solve this by having a good discussion about every feature of the program and designing it with the whole group so every member understands what to do and how to handle it.

- **Wasting time on unnecessary features.**

When we as a group decide to add a feature that is unnecessary to the project the would waste time and would have an effect on the time we have left to implement features that are necessary. We could handle this risk by deciding what are necessary features from the beginning the the design of the application so we do not waste time thinking or adding new features.

- **Unavoidable risks**

These are risks that we cannot keep track of or know what could happen and then we should solve it as soon as a problem like this comes to our attention. Like of example if Fontys closes due to underfunding and the study program stopped we could not finish the project.