



Project Report

Authors:

Rawan Abou Dehn 3560821

Femke Duijzings 3926443

Baian Khir Allah 3204367

Introduction

The main purpose of this document is to describe the main goal of the project, our process and results.

In the next chapters we will cover:

- Project background: the situation before we started the project and what we had to take into consideration.
- Problem statement: the problem our client faces.
- Process & Results: the process we went through individually and as a team and a description of our final result.
- Team Contribution: how everyone contributed to the project.
- Conclusion & Recommendations: whether our project is completed (compared to the requirements provided by the client) and recommendations for useful features that is beneficial to add in the future.
- Evaluation & Reflection: our individual opinion of how the whole process went, what we are proud of, would do again or would do differently next time.

Project background

Student Housing BV owns a building where students can stay during their study in the Netherlands. Their buildings are composed of rooms, where each student gets their own room, and shared facilities such as toilets, a bathroom, a kitchen, etc.

Students have to communicate about dividing tasks, as the shared facilities regularly need cleaning. These tasks can include cleaning the bathroom, cleaning the kitchen, doing the garbage and buying the groceries. There was no way yet to check who had which task and if their task was done or not. They received multiple complaints about students who did not complete their tasks, which was the main reason why they have asked our help.

Problem statement

As mentioned in the previous chapter, the main issue was students who did not complete their task(s). The issue they were facing was mostly caused by communication problems between students. They've asked us to help them create a software solution to arrange the day-to-day situation with tasks. This solution will not only help divide tasks and remind students to complete them, but also offers solutions to other issues, such as scheduling a party, students can properly view house rules at any time, students can also make agreements between each other and ask the housing company questions and file complaints at any time, anonymously. The most important role of this software solution is that there will be a way for the housing company to check if students are actually following the rules and completing their tasks. We are positive that this solution will make it easier for the housing company to manage their students.

Process & Results

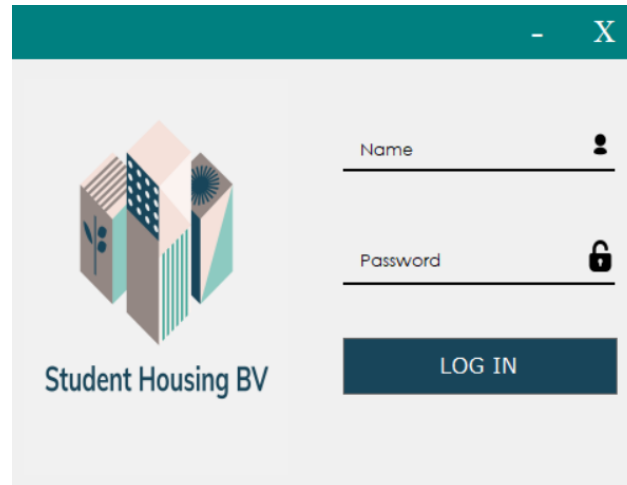
Product properties:

Our developed application has two main interfaces that play an important role, because they will display the information that the users need.

Product specification:

The first interface is the login:

- Users can login using their e-mail address and password.



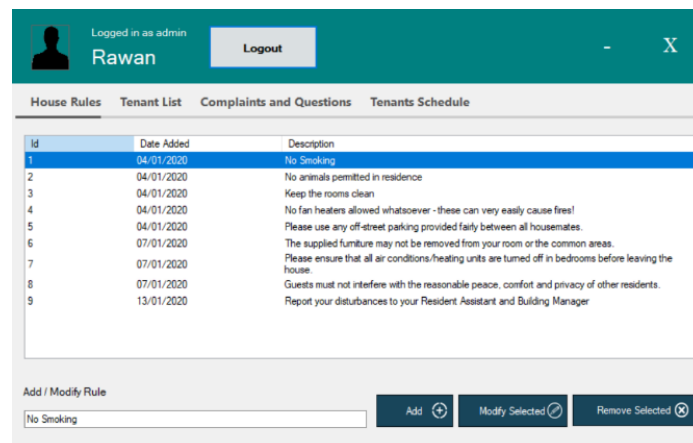
The login interface features a teal header bar with a close button (X). On the left, there is a logo for 'Student Housing BV' consisting of a stylized building icon. To the right of the logo, there are two input fields: 'Name' with a person icon and 'Password' with a lock icon. Below these fields is a dark blue 'LOG IN' button.

The actual function of each component in the interface of the administration App is assigned as follows:

House Rules

This page will display the house rules created by the Student Housing.

- Admin can modify a house rule by selecting one and pressing the modify selected button.
- Admin can similarly delete a house rule by clicking the delete selected button.
- Admin can add a new house rule by inserting it into the textbox and then pressing the add button.



The 'House Rules' management interface shows a user logged in as 'Rawan' with a 'Logout' button. Below the header, there are four tabs: 'House Rules', 'Tenant List', 'Complaints and Questions', and 'Tenants Schedule'. The 'House Rules' tab is active, displaying a table with the following data:

Id	Date Added	Description
1	04/01/2020	No Smoking
2	04/01/2020	No animals permitted in residence
3	04/01/2020	Keep the rooms clean
4	04/01/2020	No fan heaters allowed whatsoever - these can very easily cause fires!
5	04/01/2020	Please use any off-street parking provided fairly between all housemates.
6	07/01/2020	The supplied furniture may not be removed from your room or the common areas.
7	07/01/2020	Please ensure that all air conditions/heating units are turned off in bedrooms before leaving the house.
8	07/01/2020	Guests must not interfere with the reasonable peace, comfort and privacy of other residents.
9	13/01/2020	Report your disturbances to your Resident Assistant and Building Manager

At the bottom, there is a section 'Add / Modify Rule' with a text input field containing 'No Smoking' and three buttons: 'Add' (with a plus icon), 'Modify Selected' (with a checkmark icon), and 'Remove Selected' (with an X icon).

Tenant List

This page will display the tenants of Student Housing.

- Admin can add a student by inserting his/her info (name, email, password, floor and room numbers) and then clicking the add button.
- Admin can delete a student by selecting one from the list and clicking remove selected.

The screenshot shows the 'Tenant List' page. At the top, a teal header bar contains a user profile icon, the text 'Logged in as admin', the name 'Rawan', and a 'Logout' button. Below the header is a navigation bar with four tabs: 'House Rules', 'Tenant List' (active), 'Complaints and Questions', and 'Tenants Schedule'. The main content area features a table with columns: Student ID, Name, Email, Phone, Floor, Room, and Score. The table contains 10 rows of student data. To the right of the table is a form for adding a new tenant, with fields for Name, Email, Password, Floor, and Room. At the bottom left of the table area is a 'Remove Selected' button with a trash icon. At the bottom right of the form area is an 'Add New Tenant' button with a plus icon.

Student ID	Name	Email	Phone	Floor	Room	Score
4	Omar	omar@fontys.com		1	1	15
5	Pieter	pieter@fontys.com		1	2	38
6	Miley	miley@fontys.com		1	3	2
7	Mark	mark@fontys.com		1	4	64
8	Kelvin	kelvin@fontys.com		1	5	5
9	Ranim	ranim@fontys.com		1	6	12
10	Robin	robin@fontys.com		1	7	24

Complaint & Questions

This page is the channel in which the tenants and the housing company will communicate.

- Admin can reply to the coming questions and complaints from tenants by selecting one from the list and inserting the response in the textbox and then press the send reply button.
- Admin can export the list of questions and complaints as Excel.

The screenshot shows the 'Complaints and Questions' page. It has the same teal header and navigation bar as the previous page. The main content area displays a table with columns: Id, Date added, Type, Message, and Reply. The table contains 8 rows of data, with the first row highlighted in blue. Below the table is a 'Reply Selected With' text box. At the bottom right are two buttons: 'Send reply' with a paper plane icon and 'Export As EXCEL' with an Excel icon.

Id	Date added	Type	Message	Reply
1	04/01/2020	Question	How can I make changes to my tenancy agreement?	
2	06/01/2020	Complaint	The elevator does not work, when will it be repaired?	
3	06/01/2020	Complaint	People are not cleaning the shared facilities	
4	07/01/2020	Question	Why has my rent increased?	
5	10/01/2020	Complaint	My neighbors are organizing parties during the week very late at night	
6	10/01/2020	Question	How can I speak to my housing officer?	
7	13/01/2020	Question	Am I due to have my kitchen and bathroom upgraded?	
8	13/01/2020	Question	How can I report a repair?	

Tenants schedules

This page will display the tenants schedule and their tasks.

- If proven to be true, the admin can uncomplete a task based on a complaint that the task was in fact not completed.

Assignee	Date	Task	Status
Omar	1/16/2020 12:00:00 AM	Cleaning	Pending
Omar	1/19/2020 12:00:00 AM	Garbage	Pending
Pieter	1/21/2020 12:00:00 AM	Cleaning	Pending
Omar	1/22/2020 12:00:00 AM	Shopping	Pending
Pieter	1/24/2020 12:00:00 AM	Garbage	Pending
Omar	1/25/2020 12:00:00 AM	Cleaning	Pending
Mikey	1/26/2020 12:00:00 AM	Cleaning	Pending
Pieter	1/27/2020 12:00:00 AM	Shopping	Pending
Omar	1/28/2020 12:00:00 AM	Garbage	Pending
Mikey	1/29/2020 12:00:00 AM	Garbage	Pending
Pieter	1/30/2020 12:00:00 AM	Cleaning	Pending
Omar	1/31/2020 12:00:00 AM	Shopping	Pending
Mark	1/31/2020 12:00:00 AM	Cleaning	Pending

The actual function of each component in the interface of the student App is assigned as follows:

Schedule and Events

This page will display the tenants schedule, it also includes events organized by the tenants.

- Student can check their daily tasks and status.
- Student can create an event by setting a date, time, location and description and clicking the send request button.
- Student can see all events and vote on them with yes or no, If the majority of tenants voted yes, the event will go through, otherwise it will not.
- Student can organize an event by inserting date, time, location and description and clicking the send request button.
- Student can vote on an event yes or no.
- Student can view all upcoming events.

Assignee	Date	Task	Status
Omar	16/01/2020	Cleaning	Pending
Omar	19/01/2020	Garbage	Pending
Pieter	21/01/2020	Cleaning	Pending
Omar	22/01/2020	Shopping	Pending
Pieter	24/01/2020	Garbage	Pending
Omar	25/01/2020	Cleaning	Pending
Mikey	26/01/2020	Cleaning	Pending
Pieter	27/01/2020	Shopping	Pending
Omar	28/01/2020	Garbage	Pending
Mikey	29/01/2020	Garbage	Pending
Pieter	30/01/2020	Cleaning	Pending
Omar	31/01/2020	Shopping	Pending
Mark	01/02/2020	Cleaning	Pending
Pieter	02/02/2020	Garbage	Pending
Omar	03/02/2020	Cleaning	Pending
Mark	03/02/2020	Garbage	Pending
Mikey	04/02/2020	Cleaning	Pending

Create event

Date: 1/16/2020 Floor: Location:

Description: Send request

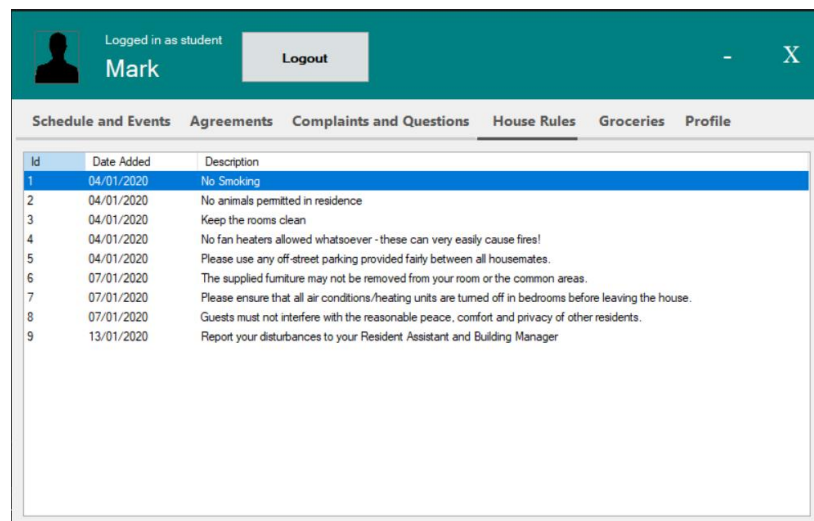
Event Id	Organizer	Date	Location	Description	Votes	Voter No
1	Mikey	18/01/2020	Floor 3 common area	My birthday party	50% YES	4
2	Omar	22/01/2020	First floor room 20	Hang out	100% YES	1

You can go through with an event when majority has voted yes

YES NO

House Rules

This page will display the house rules created by the Student Housing.

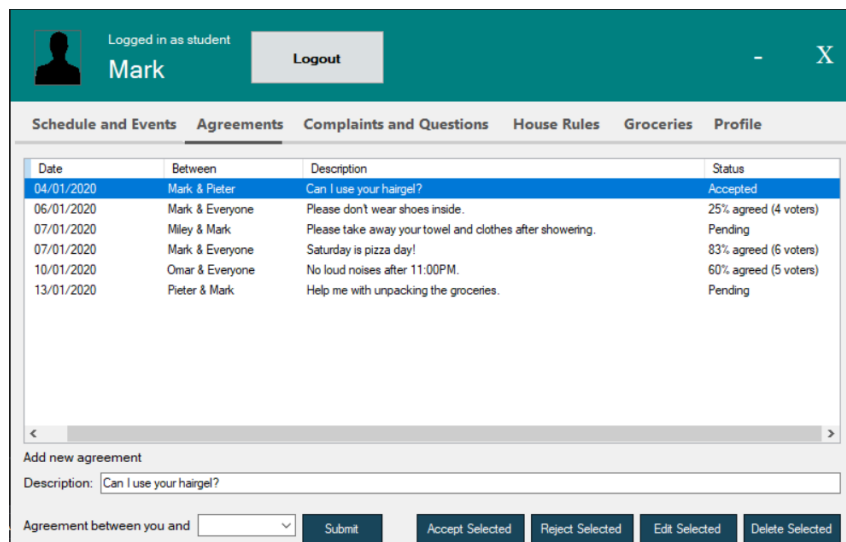


Id	Date Added	Description
1	04/01/2020	No Smoking
2	04/01/2020	No animals permitted in residence
3	04/01/2020	Keep the rooms clean
4	04/01/2020	No fan heaters allowed whatsoever - these can very easily cause fires!
5	04/01/2020	Please use any off-street parking provided fairly between all housemates.
6	07/01/2020	The supplied furniture may not be removed from your room or the common areas.
7	07/01/2020	Please ensure that all air conditions/heating units are turned off in bedrooms before leaving the house.
8	07/01/2020	Guests must not interfere with the reasonable peace, comfort and privacy of other residents.
9	13/01/2020	Report your disturbances to your Resident Assistant and Building Manager

Agreements

This page will display the agreements created by the tenants.

- Tenant can create an agreement between him and another tenant or with everyone by inserting a description, with who it is and pressing the submit button.
- Tenant can accept or reject an agreement if it involves him by pressing the accept or reject button, respectively.
- Student who created an agreement can edit or remove it by pressing the edit or remove button, respectively.



Date	Between	Description	Status
04/01/2020	Mark & Pieter	Can I use your hairgel?	Accepted
06/01/2020	Mark & Everyone	Please don't wear shoes inside.	25% agreed (4 voters)
07/01/2020	Miley & Mark	Please take away your towel and clothes after showering.	Pending
07/01/2020	Mark & Everyone	Saturday is pizza day!	83% agreed (6 voters)
10/01/2020	Omar & Everyone	No loud noises after 11:00PM.	60% agreed (5 voters)
13/01/2020	Pieter & Mark	Help me with unpacking the groceries.	Pending

Add new agreement

Description:

Agreement between you and

Complaints and Questions

This page is the channel in which the tenants and the housing company will communicate.

- Student can send a question or a complaint by selecting the type of message and inserting a message then clicking the add button.
- Student can remove a question or a complaint he previously sent.
- Student can view all the complaints and question he previously sent.

The screenshot shows the 'Complaints and Questions' page. At the top, a header bar indicates the user is logged in as 'Mark' with a 'Logout' button. Below the header, a navigation menu includes 'Schedule and Events', 'Agreements', 'Complaints and Questions' (which is active), 'House Rules', 'Groceries', and 'Profile'. The main content area displays a table of messages:

Id	Date	Type	Message	Reply
2	06/01/2020	Complaint	The elevator does not work, when will it be repaired?	
6	10/01/2020	Question	How can I speak to my housing officer?	
7	13/01/2020	Question	Am I due to have my kitchen and bathroom upgraded?	

Below the table, there is a 'Remove Selected' button and a form to 'Add a new message'. The form includes a 'Type' dropdown menu and a 'Message' text area, with an 'Add' button at the bottom right.

Profile

This page will display the info of the tenant.

- Student can adjust his personal details (name, age, password, email, phone) by changing the info in the textboxes and pressing the edit button.
- Student can view his personal details (name, age, password, email, phone, floor, room, balance, and score)
- Student can view his personal tasks and mark tasks as completed by selecting a task and pressing the complete selected task button.
- Student can deposit money to his account to pay for mutual groceries by inserting a number and clicking the deposit button.

The screenshot shows the 'Profile' page. The header and navigation menu are identical to the previous page. The main content area is divided into two sections: 'General Information' and 'My tasks'.

General Information

Name: Mark, Age: []

Password: []

Email: mark@fontys.com

Phone: []

Floor: 1, Room: 4

Balance: €0

Score: 64 Highest score! Good job!

[Edit]

My tasks

Date	Task	Status
1/31/2020 12:00:00 AM	Cleaning	Pending
2/3/2020 12:00:00 AM	Garbage	Pending
2/6/2020 12:00:00 AM	Shopping	Pending
2/9/2020 12:00:00 AM	Cleaning	Pending
2/12/2020 12:00:00 AM	Garbage	Pending
2/15/2020 12:00:00 AM	Shopping	Pending
2/18/2020 12:00:00 AM	Cleaning	Pending

[Complete Selected Task]

Add balance: [] [Deposit]

Groceries

This page will allow users to create organize and plan buying needed groceries.

- Student can add grocery items to list by selecting the amount and typing the name and then clicking the add item button.
- Student can remove an item from the list by selecting it and clicking the remove item button.
- Student can view all items in the grocery list.
- Student can mark a grocery list as complete when he buys all the items.
- Student can see the previous completed grocery list and who completed them.
- Student can get more info on the previous completed grocery list by selecting one and pressing the info selected button.

The screenshot shows the 'Groceries' page of an application. At the top, a teal header bar contains a user profile icon, the text 'Logged in as student Mark', and a 'Logout' button. Below the header is a navigation bar with tabs: 'Schedule and Events', 'Agreements', 'Complaints and Questions', 'House Rules', 'Groceries' (selected), and 'Profile'. The main content area is divided into two sections. The left section displays a table of the current grocery list with columns 'ID', 'Amount', 'Item(s)', and 'Added by'. The right section contains a form to 'Add a new item to the list' with input fields for 'Amount' and 'Item', and an 'Add Item' button. Below the form is a section titled 'Previous students who did groceries:' containing a table with columns 'ID', 'Date', and 'Student'.

ID	Amount	Item(s)	Added by
1	1	Olive oil	Mark
2	8	Toilet paper	Robin
3	1	Detergent	Miley
4	1	Chili sauce	Ranim
5	3	Wash powder	Mark
6	4	Handsoap	Kelvin

Buttons at the bottom: 'Remove item' (with a close icon), 'Mark grocery list as complete' (with a checkmark icon), and 'Info Selected'.

Team contribution

- **Rawan**

Worked on the GUI and separated the form into three forms and redesigned the app.

Solely created the following functionalities:

Mutual:

- Login and logout
- View schedule.

Admin specific:

- Add and remove tenants and view tenants list.
- Gather complaints and questions, view them and export them as Excel.
- View and adjust house rules.
- Make a task incomplete based on a complaint (If proven to be true).

Student specific:

- File a complaint or a question anonymously.
- View messages made by current user.
- Profile page: personal info/room/building number/tasks this week/Score.
- View house rules.
- Scoring system (Student with highest score).
- Mark a task completed.
- When the admin reverses a completed task based on a complaint (If proven to be true), the student who didn't do his task could be punished by getting a minus score.
- Create events and allowing users to vote on them.
- View events.

Regarding the challenges I faced in developing my part, Excel was a bit tricky I first created all the logic in the form but then I learned that the form should be as simple as possible so I had to learn how to do it in a class using a static method.

Creating the schedule algorithm was also tricky as the program had to assign tasks to tenants.

Other than that, everything programming wise went quite smoothly.

• **Baian**

I was making events and voting on them, adding schedules to each student which was the biggest challenge I had in the project where I had to get help from team mate in order to make it work knowingly that each of us working individually so I had to pass it on, after discussing potential solutions.

the final result of splitting tasks between students is pretty good after autocreation of schedule for each students it worked efficiently, another point was critical and time consuming is the voting on events unfortunately this part couldn't be done due narrowing time before finalizing, and the load that each mate has on shoulder also I had to do wireframes in early stages in order to imagine the functionality of the system, since it is the first huge project I work on, team playing was also challenging and hard to maintain. I also was responsible for testing and following up find out gaps and spot light on it mostly all the program works well with minimum number of exceptions as a result everything was similar to our expectations.

• **Femke**

I came across two challenges during the project. The first and most important: lack of knowledge. We had just started using classes and had to learn a lot of new things. I couldn't grasp the new content so quickly, and so for the first few weeks I couldn't contribute any programming to the application. I did attend every lecture and self-study, hoping to get a better understanding of the material. It eventually paid off and I was able to do my part of the project, which I will mention soon.

The second challenge was my laptop. From the start of the school year my screen had a defect, which resulted in the screen flickering a lot, often resulting in black screens and being unable to properly do my schoolwork. Because of this I also hadn't had the task of taking minutes of our meeting. There was no holiday long enough to have my laptop fixed, and so when the Christmas break finally arrived, I was able to have it fixed. The company that fixed my laptop had given the wrong diagnose and this delayed the process of fixing it by two

weeks. With the holidays in between, it delayed it even more. I was able to get my laptop back just before the break ended, but because of this I couldn't work on my parts during the holiday. This also resulted in a task of mine having to be taken because of the lack of time left.

Lastly, my contributions to this project. I've contributed to a few functions. I made the starting GUI, where I came with several ideas that were eventually used in our project, such as using tab pages to make the application easy to use and organized. I also made a login function, but in the end, we decided to implement it differently and I didn't have my laptop at the time, and so another team member took this task. I also re-made the agreements tab page and functionality, added a grocery list tab page and functionality, and made the function to add balance to a user's account.

The final result is okay for me. I was able to do a few functions in the end despite the challenges I faced. I wish I had been able to do a bit more.

Conclusion & Recommendations

Our project was successfully completed and we fulfilled all of the requirements and added some extra functionalities.

As there is always room for improvement, following are some essential and recommended functionalities.

Essential functionalities:

- Database.
- Connect balance functionality with bank.
- Encrypt passwords and have some restrictions on them/ auto generated passwords.
- Forget credentials functionality.
- Having multiple buildings.

Recommended:

- Rewarding system based on score.
- Separate management class into two (one for student and one for admin).
- Export complaints in different types.
- Allow student to add profile picture.
- Messaging functionality.
- FAQ questions.
- Make the application more appealing.
- Student should change password after first login.
- Notification functionality.
- Swapping tasks between tenants.
- Giving the tenants a status for when they are busy or away.

Evaluation & Reflection

- **Rawan**

The project went well, it was quite nice to follow the lectures and then right away apply what I learned.

I'm quite proud and satisfied with what I was able to achieve, I applied what I learned well and I also used some concepts that we have not learned yet such as Singleton, Inheritance and exporting to excel.

What could definitely be improved next time is the team work and the time management. I would also like to work with people who have similar work ethics and as serious as me when it comes to work.

- **Baian**

Despite of the pressure last week before submitting and exam, the project went smoothly we didn't have any conflicts between team mates that's the most importantly, and achieved targets which is making the requirements and adding some creativity on it, in the next project I would choose such a type of members based on the flexibility and qualifications they have, I would maintain the methodology we follow, which is division work between each other, and thinking always of wireframes, which helps to make up a vision about program, but I would look for more resources for my project, since the theory not covering all ideas I want to implement, namely exporting files, and I would look for more examples with a critical problem need to be solved, I would invest more time on searching and self-learning, which are the key to keep up, I am proud of program, as it fulfilled mostly all potential demands, that might be required at later point, that means to me, that I understand most of concepts in this term, moreover it is the first realistic application I've made so far, it left an impression of ability to accomplish big tasks with team working.

- **Femke**

I think the project overall went well. Our group was able to everything and create a stable application. Everyone completed the tasks they needed to do.

I would definitely try to do more in the next project. I would also try to start earlier on my tasks than this time.

I don't think I have anything to be proud of in this project. But I am glad to have finished my tasks without any difficulties in the end.