JANUARY 17, 2020

SOFTWARE SOLUTION FOR STUDENT HOUSES

STUDENTHOUSING BV

VAN BRABANT
OMER FARUK GOKBAK - 3782174
HRISTO PAVLOV - 3788733

Contents

Introduction

We have successfully finished our first semester's orientation phase, in which we were introduced to the basics of programming applications in C#, designing interfaces, and other ICT related subjects. Coupled with the knowledge we have gotten in the following weeks of the advanced phase, most notably in the area of Classes and Object-Oriented Programming, we can now tackle projects on a much larger scale. This report is meant to present the results on our latest project - Student Housing BV. He will serve as a great reference of how much we have improved in the IT sphere. To accomplish that, we are going to start off, by giving you some context about that project.

Background

We, Student Housing BV, are a housing company for students in the Netherlands. Every student gets his own room and shares the other facilities, such as toilets, kitchen and so on, with a couple of roommates. As a result of that, we've been getting complaints from our clients on various topics related to their daily chores with their new roommates, such as shared facilities not getting cleaned, garbage not being thrown off, unannounced events in the house etc.

In order to support our Windows Form Application in accomplishing such a task, we decided on using in conjunction with it, an SQL [2] server database and an Arduino Uno [1]. We wanted to implement multiple layers of communication in-between the clients themselves, as well as in-between the client and us, their housing agency. In order to accomplish that, we ended up working with a database purely due to the fact that with it, we would be able to store our data separately and have it available at our disposal, when the application starts, instead of losing it, after we stop the solution. Thanks to it all messages, complaints, announcements, rules etc. are saved on an online server, through which we could gain access to them and update them after we start the application. We also needed a technology capable of notifying our application when the garbage was getting thrown off. We picked up the Arduino due to a myriad of reasons, most notably, we were already familiar with it, thanks to our classes on Embedded Systems in the Orientation phase and we could easily obtain it from the ISSD desk. We are also loading the programs directly into the device, without the need of a hardware programmer to burn the program, which makes it one of the easiest technologies to use and work with. Coupled with the large community of people talking about it online, we would be able to easily find help about everything Arduino related. With that being said, let's continue with the problems we have encountered along the way.

Problem Statement

The company, Student Housing BV, offers accommodation facilities for students throughout their bachelor or master studies in the Netherlands. The building consists of 5 floors and 4 personal rooms and shared facilities (kitchen, toilet, bathroom, living room etc.) on each floor. They are faced with issues while managing the housing system. Issues are usually related to:

- The capacity of the building is 20 people and 4 people for each floor. It means that everyone has to keep clean shared facilities, but places are found dirty and messy. Absence a control mechanism, appointing a specified person on specified assignment for each day and announcing it publicly.

- Avoiding assigning the same task to the same person for twice in succession. Updating assignments regularly every single day.
- Shared items on each floor should be bought by a student who lives on that floor. However, it is not done even some people do not pay money and it damages the discipline.
- Garbage in the kitchen must be disposed of every night otherwise it causes the dirty smell. Sometimes it is not thrown a couple of days.
- Meeting with friends at home and organizing party without informing other floor-mates can be found annoying by others. Sometimes two organizations can conflict, and it causes a problem.
- Tenants are not able to check the updated rules or announcements.
- Tenants want to contact with administration office easily and inform about technical or personal issues
- Tenants also need a platform that they can share their complaints or opinions anonymously with others.

We have received these realistic issues above. We tried to solve all these issues with our software, and we believe that it will help to solve the problems of the company.

Process & Results

Process

Onur Ereren

As a team leader, I think I contributed to this project more than enough. The connection, creation and design of the database made by me [2]. Furthermore, I designed most of the user interface such as navigation menu, colours, buttons, fonts, data grid views and forms [4]. On the other hand, for the software perspective, I contributed to the Student GUI, Admin GUI, student timetable, messaging system, complaints, rules, announcements, login and registration system and so on.

There were many challenges that I faced. Firstly, the connection of the database to the C# environment was a bit hard. Before deciding the use MySQL for the database, I tried to use MSSQL [2] and PostgreSQL but It did not work out. Secondly, I had to learn SQL functions such as deleting, updating and creating and how to express them in the C# environment in order to make use of our app's functionalities. Moreover, I had to think about the design of the database which was the hardest part. Therefore, now, I know designing a database for large scale projects is very crucial.

Designing a pleasurable user interface was very difficult for me because there are many things that you should consider such as the colour combinations of the UI, button positions, font sizes and data views. Furthermore, windows forms have limited options in terms of user experience. I tried to find external libraries for more good looking UI but you have to pay for these software. Moreover, Designing UI and coding simultaneously can be challenging.

Omer Faruk Gokbak

As a team member, my contributions for this project were Arduino [1] and to connect it with C# application, designing GUI of checking tasks, task statues and besides some classes through the project. For the student users; weekly timetables and assigning specific tasks individually for each student and each floor, updating all timetables regularly, avoiding to assigning a task to more than one student in a day while updating, add a complaint anonymously to the database. On the other side, for the admin users; checking tasks whether they are done, checking the status of tasks weekly and able to find students who didn't complete their duties. Moreover, to show data from the database [3] to the C# application, it was a challenge for me, therefore, I have learned to design a DataGrid [4] table it is the best way to show your data in a C# application. Coding Arduino code and achieving the connection between Arduino and software smoothly, updating data in the database after Arduino checks and lastly informing the student with a message, who did not do his task at the end of the day.

Firstly, I have learned how to use classes effectively through advance theory and self-study classes and I have practiced all with coding during the project. Using the database part was one of the most difficult challenges for me. I have learned how to create a database [4] and use it in C# with my group leader's help, updating, inserting or removing data from the database [2]. It took 2-3 weeks to get used to working with a database. Besides, creating timetables for personally for each student and updating all timetable uniquely and without confliction. I have used the arrays and shifting method to assign tasks for the next student and through updating system. All tasks can be shifted easily for the next week for each student.

Secondly, decision of using Arduino in our project, at the beginning we have supposed to use different 5 Arduino for each floor, but Hristo coded Arduino to changing floor with buttons for our demo. I contributed to coding Arduino with him and connection of Arduino with our application via an independent visual studio project. When Arduino triggers, when someone has thrown the garbage, a message is delivered by Serial port to the app, C# code changes the status of the task as "DONE" on the database [2]. If Arduino doesn't send any message to the system until the night, then system checks all tasks and if any task is undone, it remarks them as "UNDONE" and sends a warning message to specified students, who was assigned to that task.

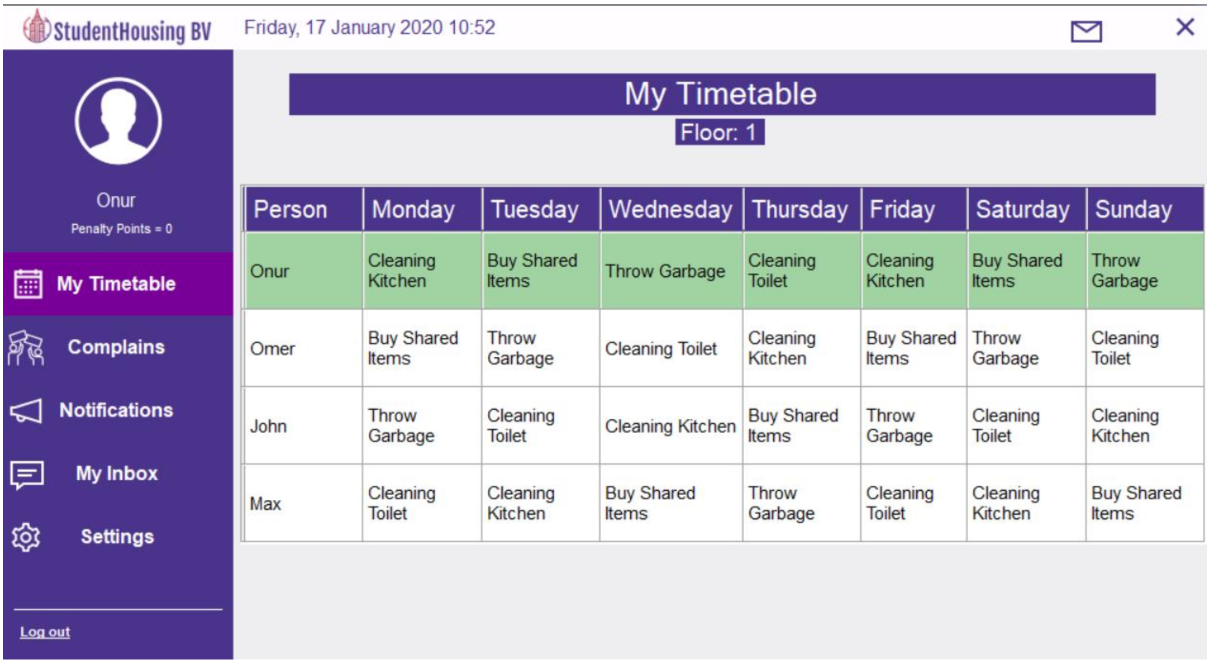
Hristo Pavlov

I have been working primarily on the back-end of the student interface and on the Arduino's code. Before submitting the interim version, I've created most of the student's windows forms applications and I coded the navigation of each form from one to another. Along the way I have been mostly working on the two classes I have been assigned to look after, namely the student and the shared rooms classes and I've kept updating them every week, as we've been learning more and more about object-oriented programming in the theory classes. After the interim version I did most of the code for the Arduino and merged it with Omer's work on that part of the project, I have also coded the entire back-end of the announcements & rules page and finished the back-end for the settings page. Unfortunately, some of my work on the interim version ended up getting scrapped over time. We ended up not needing the shared rooms class, which we agreed on having at the start of the project, as well as a specific set of properties, I have been working on, in my student class, which I had to revert back into auto-implemented properties. My biggest challenge during this project was when I had to essentially redo from scratch my code for the announcements and rules page, as I have initially made it in a way that functioned without having a database to read its data from, which did its job correctly, but was unable to store the announcements before starting the application and display them as I start it. I'm very satisfied with the state of our final version of the project. We managed to add a ton of functionalities both on the student and on the admin pages.

Results

In this section, a brief explanation of the functionalities of the app will be given which is divided into two sections namely functionalities for the admin and for the user.

App Functionalities for Student



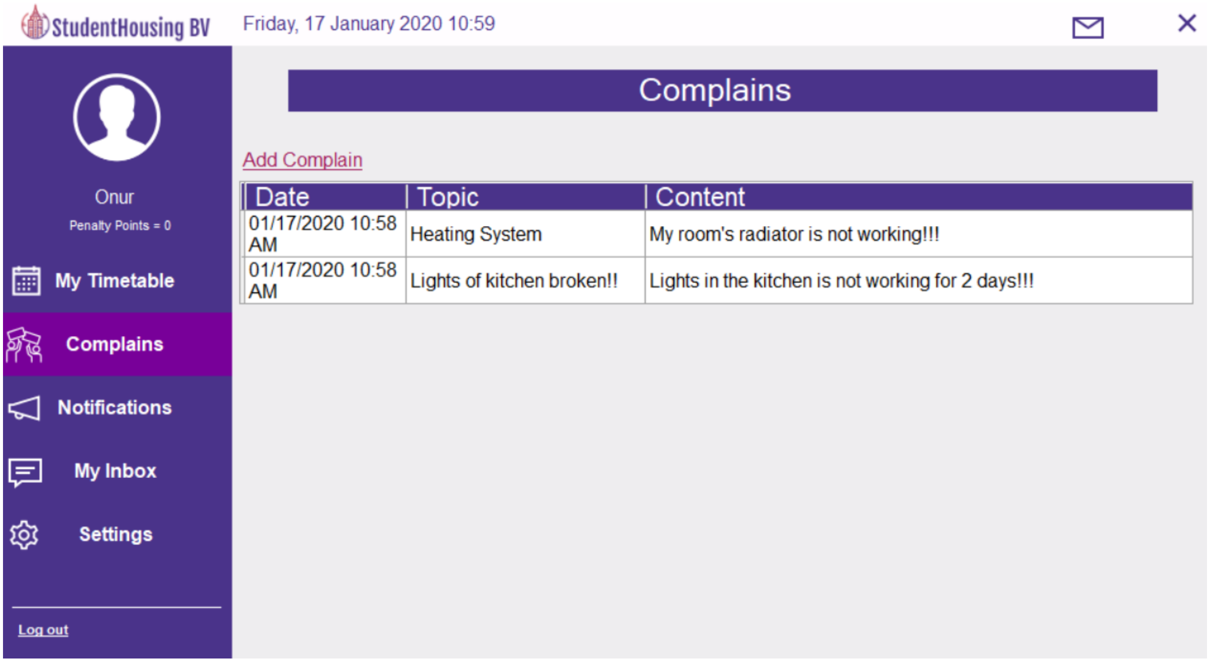
My Timetable

Floor: 1

Person	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Onur	Cleaning Kitchen	Buy Shared Items	Throw Garbage	Cleaning Toilet	Cleaning Kitchen	Buy Shared Items	Throw Garbage
Omer	Buy Shared Items	Throw Garbage	Cleaning Toilet	Cleaning Kitchen	Buy Shared Items	Throw Garbage	Cleaning Toilet
John	Throw Garbage	Cleaning Toilet	Cleaning Kitchen	Buy Shared Items	Throw Garbage	Cleaning Toilet	Cleaning Kitchen
Max	Cleaning Toilet	Cleaning Kitchen	Buy Shared Items	Throw Garbage	Cleaning Toilet	Cleaning Kitchen	Buy Shared Items

Figure [1]

It can be seen from the Figure [1] that there is a unique time table for each student for each day. Moreover, specific row highlighted for the specific student login.



Friday, 17 January 2020 10:59

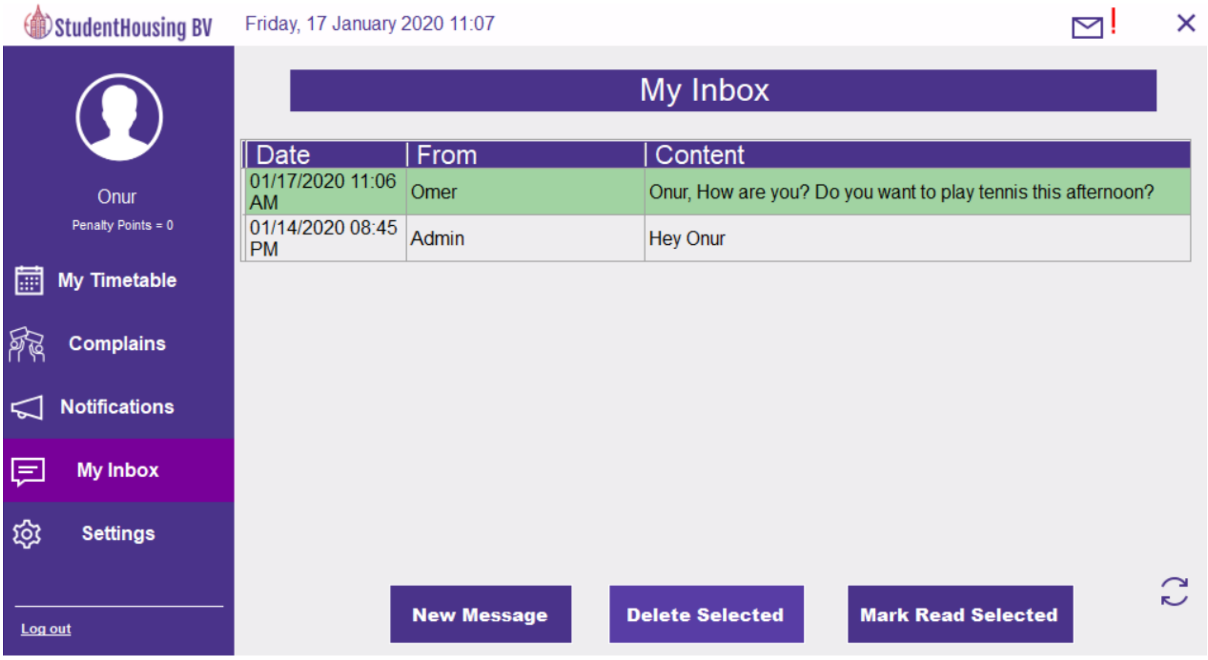
Complains

Add Complain

Date	Topic	Content
01/17/2020 10:58 AM	Heating System	My room's radiator is not working!!!
01/17/2020 10:58 AM	Lights of kitchen broken!!	Lights in the kitchen is not working for 2 days!!!

Figure [2]

It can be seen from the Figure [2] that students can send complaints anonymously.



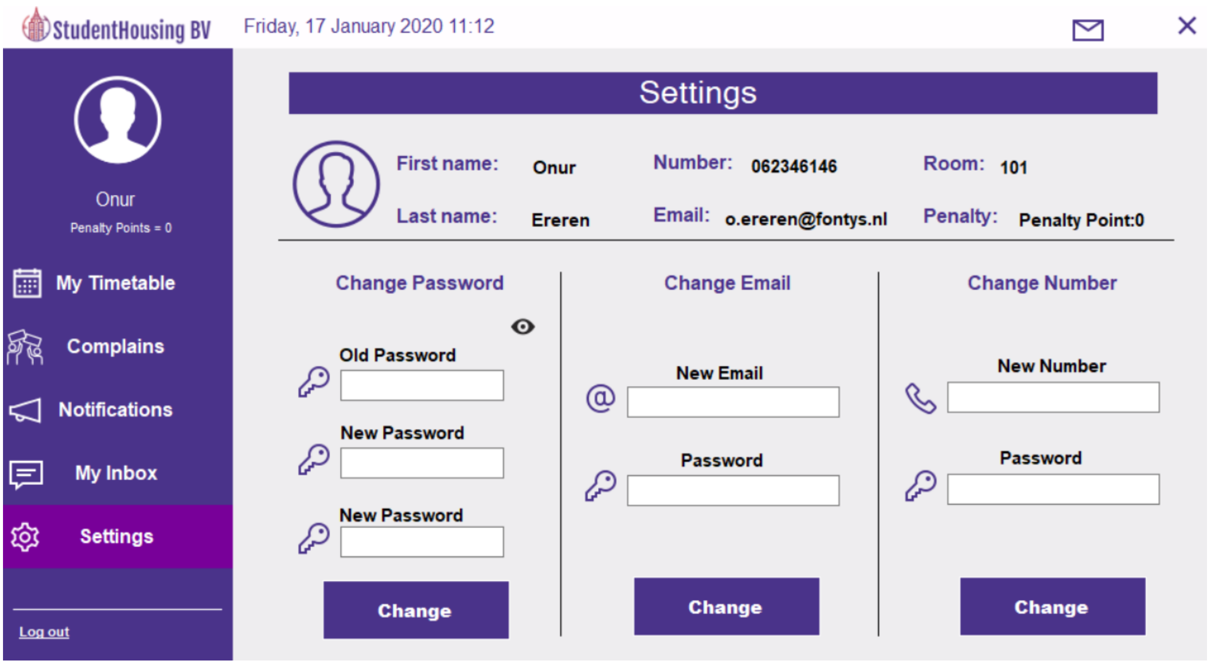
Friday, 17 January 2020 11:07

My Inbox

Date	From	Content
01/17/2020 11:06 AM	Omer	Onur, How are you? Do you want to play tennis this afternoon?
01/14/2020 08:45 PM	Admin	Hey Onur

Figure [3]

Students can send messages to admin, students and also receive messages from them as shown above.



Friday, 17 January 2020 11:12

Settings

First name: Onur

Last name: Ereren

Number: 062346146

Email: o.ereren@fontys.nl

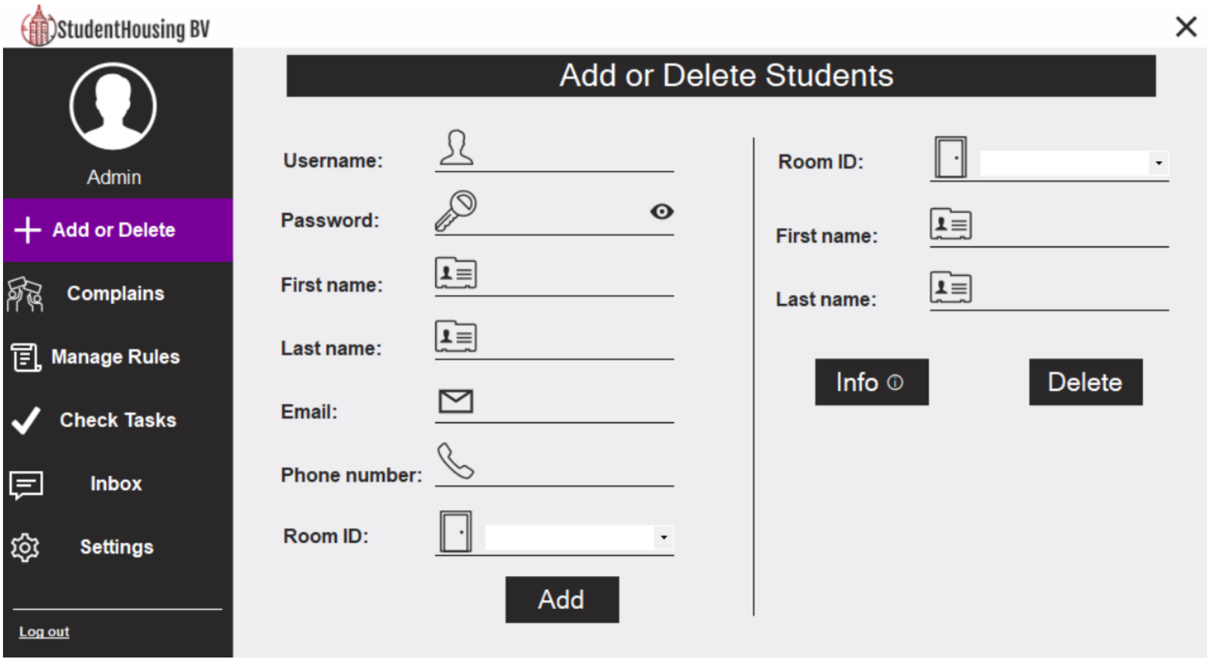
Room: 101

Penalty: Penalty Point:0

Figure [4]

Students can see their information and change their password, email and number in the settings page.

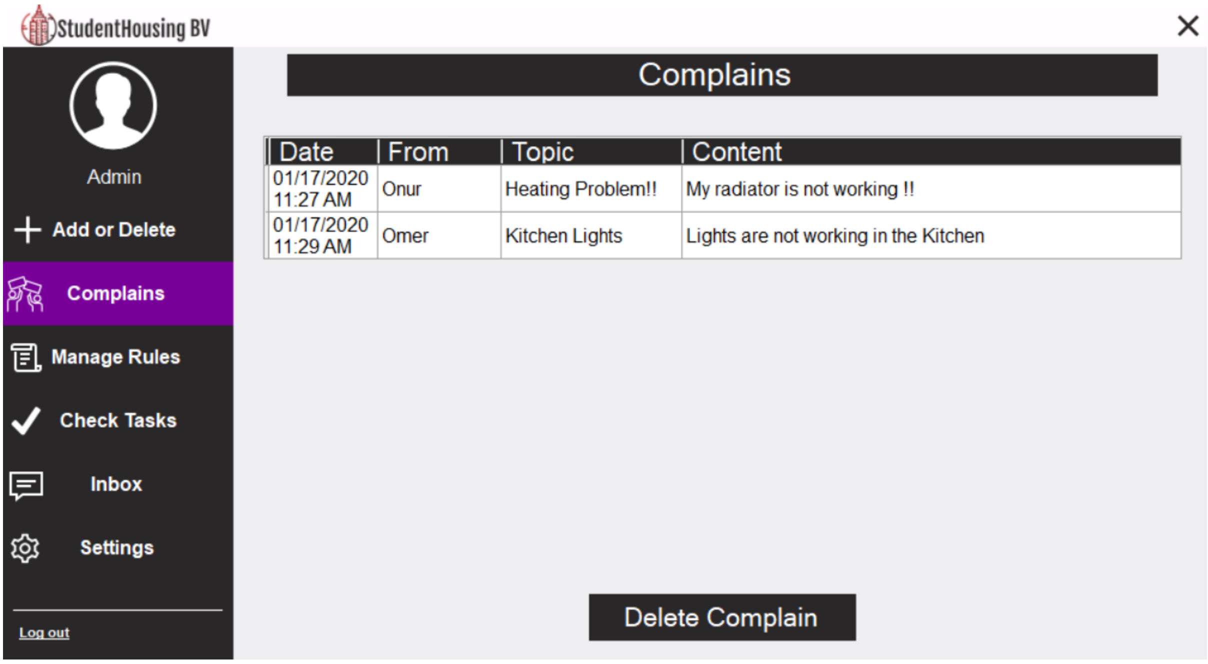
App Functionalities for Admin



Add or Delete Students

Figure [5]

Admin/Employee can add, delete and see information about the student.

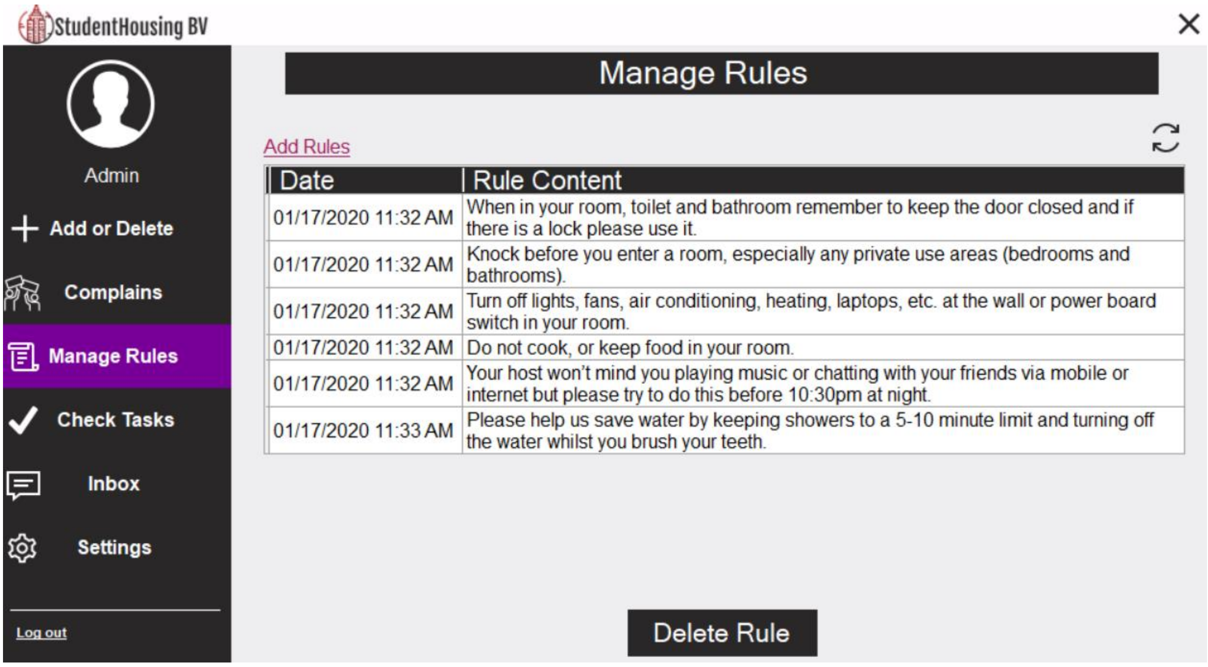


Complains

Date	From	Topic	Content
01/17/2020 11:27 AM	Onur	Heating Problem!!	My radiator is not working !!
01/17/2020 11:29 AM	Omer	Kitchen Lights	Lights are not working in the Kitchen

Figure [6]

Admin/Employee can see the complains and resolve them.



Manage Rules

Date	Rule Content
01/17/2020 11:32 AM	When in your room, toilet and bathroom remember to keep the door closed and if there is a lock please use it.
01/17/2020 11:32 AM	Knock before you enter a room, especially any private use areas (bedrooms and bathrooms).
01/17/2020 11:32 AM	Turn off lights, fans, air conditioning, heating, laptops, etc. at the wall or power board switch in your room.
01/17/2020 11:32 AM	Do not cook, or keep food in your room.
01/17/2020 11:32 AM	Your host won't mind you playing music or chatting with your friends via mobile or internet but please try to do this before 10:30pm at night.
01/17/2020 11:33 AM	Please help us save water by keeping showers to a 5-10 minute limit and turning off the water whilst you brush your teeth.

Figure [7]

Admin/Employee can set rules for the students.

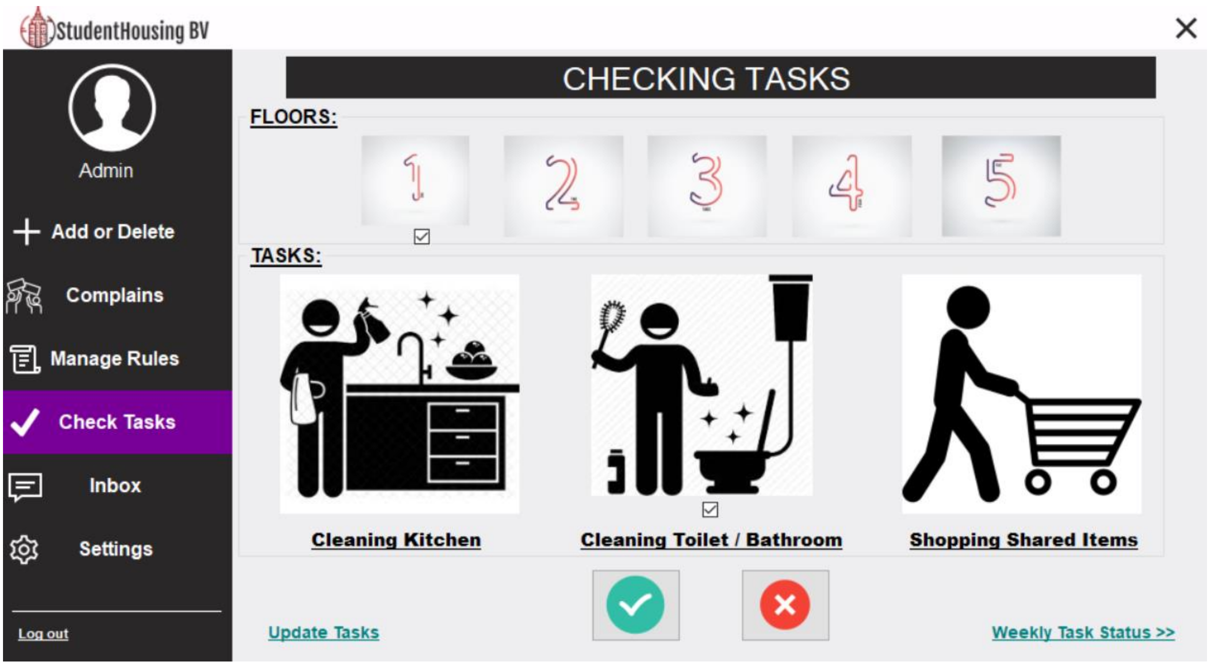
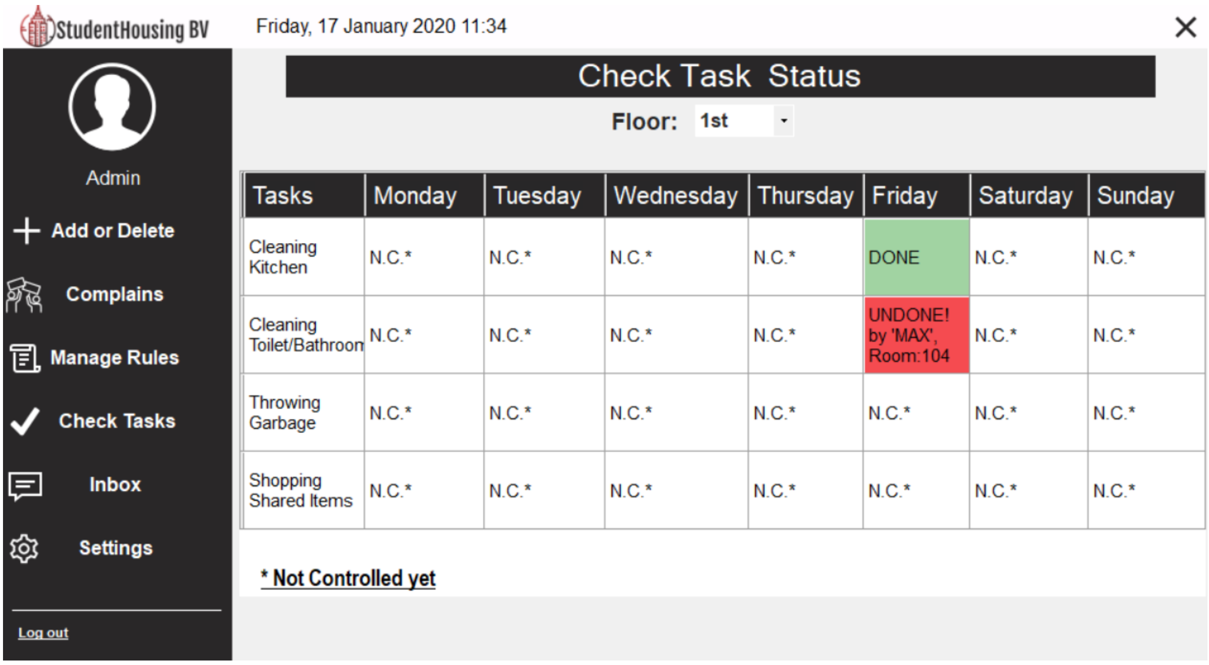


Figure [8]

Admin/Employee can check specific tasks for specific floors.



Friday, 17 January 2020 11:34

Check Task Status

Floor: 1st

Tasks	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Cleaning Kitchen	N.C.*	N.C.*	N.C.*	N.C.*	DONE	N.C.*	N.C.*
Cleaning Toilet/Bathroom	N.C.*	N.C.*	N.C.*	N.C.*	UNDONE! by 'MAX', Room:104	N.C.*	N.C.*
Throwing Garbage	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*
Shopping Shared Items	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*	N.C.*

* Not Controlled yet

Figure [9]

Admin/Employee can see specific tasks for each day that whether these tasks are done or not.

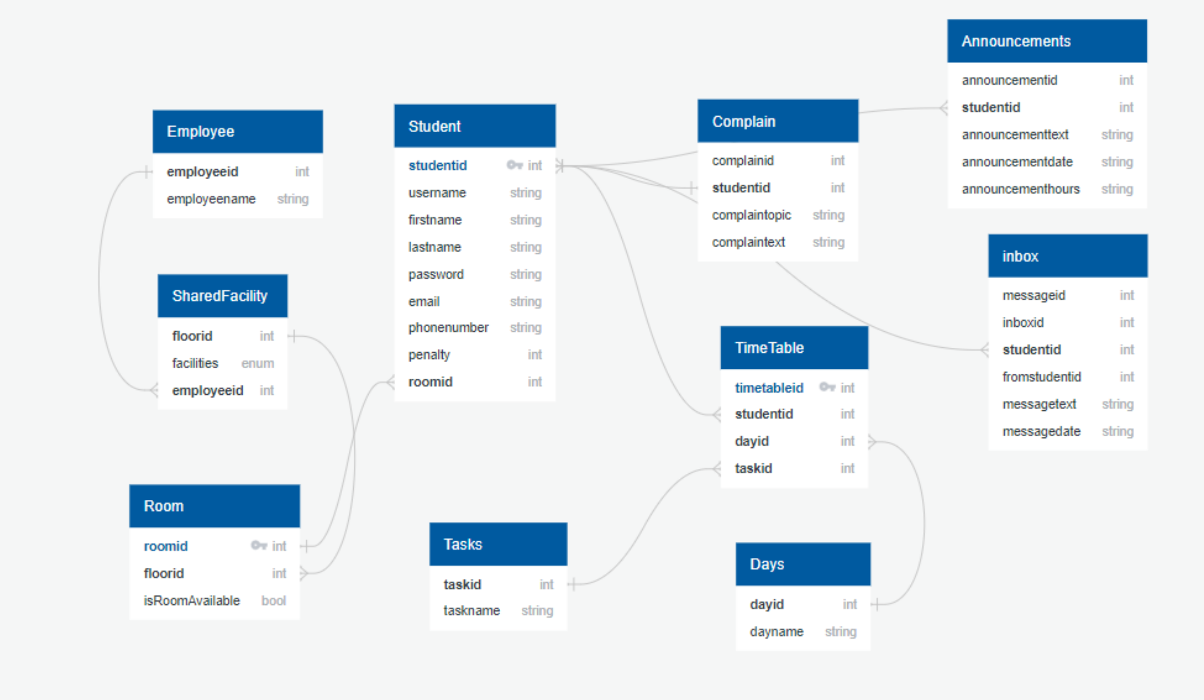


Figure [10]

Figure [10] represents our example database design [3].

Conclusion & Recommendations

Recommendations

A remarkable time spent on designing the user interface of the application. However, working with a designer for the UI would be good in order to focus more on the software for our team. Database design[2] could be improved in terms of database normalization. Furthermore, more functionalities could be added with the usage of Arduino such as setting a keypad to each floor so that safety of the properties will be improved also, we could place a sensor to the kitchen door that keeps tracks of persons who enters or leaves. In the current version of the application, there will be a one employee checks all the floor tasks. If there will be lots of floors/rooms, it can be difficult for that one employee. Therefore, assigning different employees on different floors could be implemented in the future version of the project.

Conclusion

In conclusion, the project is successfully completed by our team and all the requirements for the project have been finished. Moreover, our team added extra functionalities such as database, Arduino[1] and messaging system. Tasks equally distributed between each individual member. Furthermore, weekly meetings arranged in order to discuss the progression of the project. Finally, our team consider that our software application is extendable due to mainly usage of the databases.

Evaluation & Reflection

Onur Ereren

The project went very well because all the members of the team took responsibility and a project timetable prepared in order to follow the required deliverables. I learned that It is crucial to think and discuss at the beginning of the project that how the software should look like at the end. Furthermore, User requirements/needs of the app should be determined before starting the project.

I am happy that our team was able to finish the project on time with the completion of the requirements and even added extra functionalities.

Omer Faruk Gokbak

This project was a new experience for me both from using classes and database for a C# application. The practical part of the studies during the semester was really advantageous where one gets a chance for rising knowledge through the lectures and during self-studies. I have realized again the importance of teamwork, creativity, doing practice from theories which you have learned in lectures and added new ideas into the project. And this task becomes much easier if you have assigned to the task for each member in the group and everyone has an interdisciplinary team working for a cooperative goal. I have truly learned a systemic approach to keep simple coding, how to work in teams for short span of time, how to divide tasks within the group, how to filter the needs or requirements of the coding, how to be creative for my part or all during the project.

From now on I will use a database for all my projects. I find it helpful and easier if you need to study with data. Moreover, if you add a database in your project, you will never lose your data unless you remove them from the database even though you exit your application, otherwise when you close your app you lost all information. I would keep my code simpler for next projects and try to achieve my goal as using fewer methods.

Hristo Pavlov

Overall, I'm happy to say that the project proceeded smoothly throughout the weeks. We were always on schedule with our timetable and we all did our parts with little to no issues, aside from a couple of Git Hub merge incidents. For my next project I would definitely try spreading my work evenly across all parts of the code I'm covering, instead of trying to finalize one part at a time. That way I can minimize the amount of "lost" code, that may occur, if my future team were to change their mind on something, during our project's development. The part of this project, with which I'm most proud of, is my "CheckReservation" method in the "EditEventsGUI", within which I'm using the "String Builder" class to create a string composed of the first characters of every row in a list box. With it I can check whether a date (chosen by the calendar and the combo box on the form itself) is actually situated at the start of any of the rows as a scheduled date for an announcement (i.e. as a key to my dictionary) and not as part of the description of an announcement (i.e. the value of my dictionary, or as a simple part of the announcement's text which happened to be the same string as one, whom I would normally use for the date).

References

1- Arduino - Docs. (2020). Retrieved 17 January 2020, from https://www.arduino.cc/en/main/docs

2- MySQL: MySQL 8.0 Reference Manual. (2020). Retrieved 17 January 2020, from https://dev.mysql.com/doc/refman/8.0/en/

3- Welcome to phpMyAdmin's documentation! — phpMyAdmin 5.1.0-dev documentation. (2020). Retrieved 17 January 2020, from https://docs.phpmyadmin.net/en/latest/

4- Windows Forms. (2020). Retrieved 17 January 2020, from https://docs.microsoft.com/en-us/dotnet/framework/winforms/