Final Report

HouseMate

Georgi Nikolov: 3774163

Daniel Vaswani: 3782689

Khoa Doan: 3603407

[**1. Introduction**](#_e9brdwk0c6m1) **1**

[**2. Project Description**](#_dvk9oojrr8nu) **2**

[**3. Situation**](#_uvicv1om4140) **2**

[**4. Problems**](#_ue26njwxk9ir) **3**

[**5. Additional features**](#_wtgv0ik6ak0d) **3**

[**6. Evaluation of working process**](#_tamw4tmdcl3o) **3**

[6.1 Challenges](#_h7rfpv3tbi9d) 3

[6.2 How we overcame the challenges](#_buz94xj1x7zr) 4

[**7. Conclusion:**](#_uiqb407dkr7d) **4**

[7.1 What we have achieved](#_oz7alezhicb3) 4

[7.2 What we need to improve](#_rylzbkon0hzu) 5

[**8. Evaluation of deliverables:**](#_xobur9oc8nwf) **5**

[8.1 Documentation](#_uxah3xg2wtmp) 5

[8.2 UX/UI Design](#_njg2doid8hw) 5

[8.3 Prototyping and testing](#_uo472qwtikyd) 6

[8.4 Backend](#_aq5yb07siavw) 6

[8.5 Debugging](#_wjd1oqt2tzvc) 6

[**9. Recommendations**](#_la0qwtkwfiut) **6**

[**10. Evaluation of group work**](#_t322h89ogxfi) **7**

#### 1. Introduction

We are a small software company called HouseMate that aims at helping different housing agencies with the management of the buildings that they rent out to various clients. Therefore, in order to achieve this our product provides a means of communication and synchronization between tenants regarding the matter of maintenance of shared facilities such as kitchen, bathroom etc. Also, users of our application are able to announce events that they may have planned to their roommates, thus eliminating any potential unpleasant surprises and misunderstandings between each other.

In addition, every agency, that is concerned about its customers, should have an overview of what is happening within each residence. Our team solves that problem by providing an easy-to-use interface that enables employees to directly see the agreements made between tenants and gather any filed complaints up to a year in the past. In such a way, the responsible authorities can take an immediate and informed action to solve any major issues by updating the house policy, which is only one click away. Moreover, employees have the option to manage tenants by using a very straightforward process of adding or removing people from a particular residence in a matter of seconds.

The goal of our application is to alleviate the cohabitation between housemates and to assist the accommodation agencies as much as possible in coping with management hell. As a result, all stakeholders would be satisfied and we would contribute to the growth of this housing sector.

In the following paragraphs we present you a particular business case that we managed to solve with our product. First, there is a brief introduction to the problem, describing the current situation of our client agency. Consequently, you will become familiar with our project plan with a clear outline of the methodologies that we used and the challenges that we ran into during development.

Finally, there is a dedicated section that focuses on work and project evaluation describing what we as a team were able to accomplish, therefore explaining what we are most proud of and listing any points for improvement in the future.

#### 2. Project Description

The company Student Housing BV is having a problem regarding communication within and to their tenants. They chose to hire us,

HouseMate, to create a software solution to fix these issues. They needed a way to bridge the gap in information delivery.

#### 3. Situation

Students are finding it difficult to log their commitments to each other about chores and events. They also need to be able to make complaints. It could be complaints about the house itself or even about problems they are having as a result of ignorance from the other housemates. The landlord is not able to gauge the living environment within the house. This is why the application is needed.

#### 4. Problems

When there is no way to check if the students are following the rules, There is no incentive to make sure everything is working as it should, weather everything is stocked up, weather chores are done. Additionally the tenants are for the most part, not bothered to announce gatherings that they might organize.

#### 5. Additional features

Our vision for extending the functionality of the application includes designing, testing and deploying a robust database that ensures the fast, reliable and sustainable storage and accessibility of information regarding the residences of a particular housing agency.

Another objective is the implementation of a simple network between users as a means for rapid, real-time communication which guarantees a consistent flow of data.

As a final feature that our team strongly believes to prove useful in the future is the linking of our application to the Arduino microcontroller for the automatic garbage detection that would take care of notifying tenants once their housemate has done that job.

#### 6. Evaluation of working process

As the complexity of our product grew it was inevitable that we ran into difficulties that slowed our progress. However, through stable teamwork we managed to solve the pressing issues in a reasonable time interval.

##### 6.1 Challenges

Throughout development many challenges occurred which slowed our progress. Our first major issue was thinking of a way to share information between different forms in our application. In order to keep processes simple and quick we abandoned the idea of using relational databases, which came at the expense of data storage and reliability. Consequently, the initial design had a serious flaw, because it used too many static variables in the manager class, thus violating best practices of good object-oriented programming.

At a later stage, we had some version control problems using Git resulting in code conflicts. Also, we had extra branches of the repository that we worked in which made it a little bit more complex for merging and choosing the correct implementation of the desired features.

As for the group atmosphere and communication, every member of the team had strong opinions and suggestions about how to approach the problems. Therefore, we would often have long-winded discussions on what is best in terms of performance and simplicity which in turn would briefly slow our progress down.

##### 6.2 How we overcame the challenges

Our team managed successfully to resolve the issues that occurred during development through hard work and constant peer evaluation. This enhanced our group dynamic increased our skill at working collaboratively. Following is a brief overview on how we overcame the challenges.

First of all, we learned how to share information between forms using a common container class for all of them and having an instance of the manager class there as well. This allowed us to access the data from whichever form had access to the manager class without the use of static variables. As a result, our application became more OOP oriented and would be easily maintainable, scalable and new-feature-friendly in the future.

Concerning the Git issues we resolved them as soon as possible by ensuring the consistency of every member’s work. After a certain number of code conflicts that we resolved it became easier and flow of the process gained more speed.

As a result, the challenges increased our awareness of good coding practices and clean architecture paradigms.

#### 7. Conclusion:

At this point of the project, we have completely delivered all the requirements for the

project with the expected functions presented in the first week project definition. We

will list out what we have achieved and what we have not.

##### 7.1 What we have achieved

* The whole program is designed by dint of object-oriented paradigm application. The programming language we used is C# and the software is Visual Studio with Windows Form Application interface. We introduce different classes for different required tasks involved between students and employees.
* All the problems have been solved by an effective and consistent program. There will be no conflict between the employees and students as the program can be considered as a bridge to connect them together. From now on, students can assign tasks to each other and check to make sure whether or not the other students have finished their task. Employees can also manage the tenants more effectively by keeping track of all the complaints and then resolve them in time,...
* We have collaboratively and effectively finish the project using Git. A good goal we have achieved is knowing how to use Git and work on the project remotely. This is very important as Git is a very powerful tool for programmers.

##### 7.2 What we need to improve

For the program, we can make use of a database system to store and query data from the users. We are, however, still need more knowledge and practice to realize this goal. A database would be more effective and practical.

#### 8. Evaluation of deliverables:

After the last six weeks, our group has completely achieved all the goals that we had

planned to deliver. The points of view will be clarified below:

##### 8.1 Documentation

After many weeks of mentor meetings, we have deducted the right way to write the full documentation, which is the final report, meeting all the to-do requirements. The first part of the report is the introduction in which we give the whole picture of the problem we should tackle, the stakeholders involved in the project, the programming methodology we apply, the risks that we would face and many others elements contributing to the success of the program. The next part is project description including the whole picture of the interrelation between the problems and the solutions, what we are having and what we need to achieve. All the problems will be clarified in the problem section and possible solutions will also be introduced in the solutions part right below. After acquiring the basic foundation of a working version program, we will suggest some additional features to improve efficiency. Next we will evaluate what we have overcome and the challenges we came across during the project through the evaluation of working process. Deliverable evaluation will also elaborate on the deliverables we have introduced in the last weeks and what we have done to deliver these goals. The report will be closed with the conclusion part and the evaluation of group work in which all members in the team will express their opinion on the performance of the other teammates.

##### 8.2 UX/UI Design

For the best user experience, we include all the functions that we assume a student or an employee would need. We also include validations to make sure that during the interaction with the program, no potential error could arise and crash the whole program. There are other assumptions and constraints that we apply. For instance, only employee can add a new students or both student and employee can see the complaints,...The user interface has also been decorated to make it more user friendly and responsive. No unnecessary actions is included during the user interactions.

##### 8.3 Prototyping and testing

Having identified all the required features for the program, we then design a temporary prototype where the user can interact with the primitive interface without the code behind. By this way, we can test our idea before coding to make sure that we are on track and no additional time is allocated for coding the whole program again. The testing cycle involves initiating ideas, collecting feedback and improving on the program.

##### 8.4 Backend

Throughout the whole project, we have efficiently apply the power of C# and its object-oriented paradigm. By using different classes and methods, we abstracted out the whole program and make it as modular and collaborative as possible. Moreover, all the team members can work individually for their assigned classes without interrupting with the performance of the other functions. We also learn to use Git to remotely work simultaneously in the project. It was really fun and we did learn a lot during the project.

##### 8.5 Debugging

For this part, we have debugged the program at three points, before coding, during coding and after coding. Before coding, we decided to have the same naming techniques for the buttons, label and method to make sure they are connected with each other and no syntax error will occur. During the coding phase, we included the access identifier private to all the instance variables and public to the method for the best practice. Each of the teammates must ensure that their code is saved and corrected before pushing to Gitlab. After coding, we mostly dealing with validations and logic debugging. After all these phases, we can say that we have successfully debugged the whole program and make it a working version that we need.

#### 9. Recommendations

Although, the program we have designed is complete and meet all the requirements to solve the problems, there might be potential bugs or errors that can happen during the user interaction. Moreover, the program needs to store all the data in a database for better query and retrieving user information. In that case, you can contact our company for further improvement. Nevertheless, if your company decides to hire other companies, we can tell our tech advisor to visit and give a clear explanation about the code for future maintenance.

#### 10. Evaluation of group work

My name is Khoa Doan and my student number is 3603407. After the advanced project, I did learn a lot from working in а group. Moreover, because of the knowledge gap between me and my teammates, I always urge myself to learn faster and more effectively. I also learn to use Git in practice, which makes me feel very excited because I have never had any hands-on experience with a version control system before. After all, it was a good learning experience. The group dynamics was also effective.

Daniel: I feel that everything proceeded without any major hiccups, We planned everything meticulously and ensured that the final result would be bug-free and simple to use for just about anyone. There were no issues with people not contributing as they should. In addition to this, all of us had sufficient programming knowledge and were able to move forward without any gaps in our understanding of what needed to be done. The allocation of the workload was done equally. We did everything by breaking down the steps and creating interim deadlines amongst ourselves. With that said, I think that we learnt a fair bit about how software projects are done in the real world.

Working with Daniel and Khoa throughout the advanced project was a delightful experience. Even though we had our differences in terms of knowledge and skills we managed to collaborate in a professional way. Our group atmosphere was full of creativity and discussions regarding the best way we could implement the solution. We made sure that simplicity was our first concern and did not overcomplicate the project. Moreover, we had a friendly attitude towards each other and we were tolerable if a member did not understand fully the task he was supposed to do. In addition, the importance of sharing opinions and ideas was one of our primary focuses. Therefore, we successfully solved the problem of Student Housing BV and managed to deliver the final product on time. However, I regret we did not arrange more meetings in order to perfect our product. Nevertheless, I would enjoy working with them on future projects.