Delft University of Technology

Object-Oriented Programming Project
CSE1105

Software Project: GoGreen

Authors:

Aleksander Sinanev (asinanev) - 4830040
Daniel van den Boogaard (dvandenboogaar) - 4783433
Gian Marco Forghieri (gmforghieri) - 4922115
Mahmoud Alhumsi Alrefaai (malhumsialrefa) - 4838068

Contents

1. General: How did the project go?

1.1 Did you manage to stick to the planning (why (not))?

Overall, we managed to stick to the planning, despite being only 4. At the start of the project we had some issues with the setup which resulted in a slow start of the project for the team, but after a couple of weeks everything was going up to speed. Naturally, we had to combine working on the project with studying for other courses, which led to having some people work hard on the project for some period and study hard for other courses for another period, but everything was well coordinated.

1.2 How did the collaboration in the team go?

At the very beginning of the course, there was some reluctance in the team, but that was soon resolved. Afterwards, everything just flowed, there were a lot of ideas on what to implement and just enough enthusiasm. Granted not everyone was willing to work as a team, either because of personal issues or pure lack of commitment to the course and the program. Thanks to the effort of some of the team members we were able to push forward and start making good progress. Everyone had found his spot in the team and the work was flowing.

1.3 How did you communicate?

Our primary mean of communication was WhatsApp. We used this platform because all of us already had it, so it was the fastest to set-up. Another advantage was the instant notifications, which helped with resolving problems faster and more efficiently. We also used the suggested platform of communication by the university, namely Mattermost, to communicate with our TA.

1.4 How did version control help (if at all)?

At the very beginning, version control was kind of a problem for our team. But we soon realized the potential behind it. After a couple of weeks of playing around with it, most of us got somewhat comfortable and started using more and more of the available features. Thanks to version control most of our work was organized and systematic, it became easier for us to help each other out. Overall, we believe that using version control helped us a lot and the acquired skill that is working with version control will be an asset throughout our studies and careers.

1.5 What did you learn?

This course has made a huge impact on the skills that the group members originally had. A lot had to be learned and it wasn't always easy. The main things that we learned during the project are how to work with version control, how is a desktop application constructed, how to create a GUI(JavaFX, Scenebuilder), how to connect an app to a database hosted online, how to make the functionality of such an app, how to create client-server response/request communication and how to work in a group in order to create applications effectively.

2. Design decisions

2.1 What are some of the major decisions that you have taken as a team?

Our first major decision was at the very beginning of the project. The work and the features that had to be implemented had been divided based on the experience and desires of the team members. The main draft of the design of the application has been made in the second week, however other major decisions were made later as the team progressed since the team didn't have experience with any framework or library. Another major decision was switching from VertX to Spring, the lack of resources available for VertX was the thing that pushed the group towards the usage of Spring. Deciding to use Relational Database was made because it was the simplest and most effective solution possible. At the later stages, a decision has been made to have a UI polish/redesign because quite some UI elements on the home screen had to be scrapped as half of the team decided to quit. Which also brought along the redesign of the navigation bar, since there were tabs on the UI, which haven't been created. The login screen has also undergone a visual upgrade which was meant to be done earlier, however, hasn't been assigned due to the increased workload just by having fewer team members.

2.2 Which technological choices did you make and why?

We decided to use some tried and trusted technologies such as JavaFX and MySQL. We used JFoenix for the modern looks and Google APIs for the additional route information. We made the decision to use JavaFX over Swing, because JavaFX is more modern and has all the functionality of Swing, so we didn't miss out on anything. Furthermore, the abundance of resources available for this GUI library and the possibility to use tools such as SceneBuilder in order to simplify the workflow steered us towards choosing JavaFX for our GUI. We wanted our GUI to be more modern looking and the JFoenix library promised us just that. We decided to use JFoenix instead of other libraries, because of the ease of integration and the sheer number of tutorials available throughout the web. For our database solution we needed something that is easy to use, that is fast and that is not too complex, and MySQL provides us with the exact qualities we are looking for. Since we decided to use a map for the transportation features, we chose to get the best API possible, making Google Map API our choice. It provides us with very good documentation, accuracy and good reputation.

3. Points for improvement

3.1 How can your software be improved?

There are always things that one would like to change about their finished product. Same goes for our GoGreen app.
With regards to the GUI, one of the improvements can be that the middle circle is a progress bar, which shows how far you are to the next goal/achievement.
One of the security improvements is hashing the password. Since this is a simple app that will not be distributed it is not a high priority to make hashed passwords. But if it were an app that would be distributed, adding a hashed password would be a high priority and an improvement for our app.
An extension of current features is for the vegetarian meal a cuisine recommendation, where you will be shown one or more different vegan meals, either with the recipe inbuilt or with a link to the recipe.
An extension of the local food is the nearest farm to me functionality, where you will be encouraged to buy directly from the farm in order to stimulate the local food production and reduce the consumption of imported food.
One major feature we would like to add, but decided it would take too much time was making an Android version of the app. We decided against it as it would be an extra feature in which we have to sink hours and hours to make it happen. And at the point, the app was ready to make an android version there was too little time to make an Android version.

3.2 How can the process/collaboration be improved?

Collaboration could be improved by having more meetings, for example, one on Monday and one on Friday. This would increase our contact time; we would be more up to date on what our teammates are doing and how much progress they have made during the midweek. This might also help those team members that get stuck on their part of the weekly sprint. Resulting in possibly more work getting done and/or a more even spread of the work getting done. This will also relieve some of the weekend work so the weekend will be more of a weekend.
Other than more meetings, a more active communication from everybody should also have a positive result on the final product.

3.3 How can the course be improved?

An improvement for this course would be a more concrete description of what the report and presentation should contain. As for now, it is, “The full requirements of these can be found in the introduction slides”, which is fine but having it in the Rubric would make it more concrete and easier. Furthermore, specifying the criteria more elaborately in the Rubric, that the requirements for the demo need to be fully tested as well, which was only specified, later, thus many teams have received a reduced score for the first demo retroactively. The same would apply for explicitly stating that a friend system is required, as the rubric only contained the word friend once in a completely different context, therefore the interpretation of having a friend system is remarkably implicit.

4. Individual feedback

- **Alex**: This course was more enjoyable than I anticipated at the beginning and that is what helped me improve. I started the project with the mindset that I have the discipline and willingness to work hard, on the other hand, I was rather skeptical about my ability to integrate into a team and work my way through the final goal. This teamwork has helped me improve my social and programming skills in more ways than I thought it will. I'm happy that I ended up in a group with colleagues who are very open-minded and are willing to go over problems again and again until solutions are found. This meant that there were no conflicts during the project whatsoever. I'm pleased that I was able to work my way through some challenges, challenges which made me a better developer. At the start, I was lacking confidence, but as the project went further, I started to realize that I can only get better and pushing myself is the only way to do it. The loss of three team members meant that there was even more work to be done and that was the factor that made me work harder and learn more and more. Now, at the very end of the project, I feel way more comfortable while working on applications that utilize CRUD operations. I got some insight on how to build good quality GUI and how to connect everything together in order to make a finished product. I believe that I achieved my goal and even managed to acquire some useful qualities during that process.

- **Daniel**: In the first few weeks, I was relapsing back to my 'usual' high school attitude, which is known in Dutch as "zesjes cultuur", that means I was doing just enough to pass. Usually, this means only doing things on the day before the deadline or the same day. In the meetings, I did everything I got asked to do. I might have been a little bit silent and made relatively little comments about the work of others. Which did improve over the course as I started to know my teammates. But even at the end it still wasn't on par with the rest. Resulting in no real conflicts with anyone. Midway through I realized that using this mentality/attitude for this course, would not only result in me not passing the project, but that would also mean that my teammates had to do extra, and I would disappoint them. Which to be fair is not something that I would like. So, I knew I had to step up my game, which was amplified by leaving team members and team members that did, to my knowledge, too little. So instead of just doing what we decided to assign to me in the meetings I tried to do something extra or finish well on time. Working with vertx was somewhat annoying and therefore I was not really motivated to work on it. Once we switched to spring, everything was easier, and it motivated me to do more than when we had vertx. I glad we made the switch. What also did not help with the motivation when working with vertx, was that I worked together with David and I had to try to explain how vertx worked to him, while I did not fully understand vertx myself.

- **Gian Marco**: As mentioned in my personal development plan my strong points *are* being organized and staying on top of things and issues, making sure that deadlines are met and most of the rubric criteria are fulfilled. Furthermore, I am willing to put in a lot of effort and commitment, thus ensuring that things get done. My weak point was mostly not having prior experience in application development; therefore, I needed a lot of time to understand concepts and be able to apply it. My goal for this course was to improve my coding skills as well as general understanding on how an app should be built from scratch, and how different parts of the application are intertwined (GUI, Database, Server, Functions). I believe I was able to achieve my goal with pure effort and commitment since whatever task I assigned to myself, I had absolutely no idea how I am going to be able to complete it in the beginning. To be able to finish my first task which was creating a login screen with a database connection (which was just a serverless SQLite database), I really had to push through a lot of frustration with sheer willpower and just continue doing it because I wanted to make the deadline. After having repeated the same process with different tasks, I have realized that the most important aspect was to not let frustration (of not having a clue what/how to do) take over. During this course I have proven to myself multiple times, if I was just able to continue researching, looking up ideas and solutions (instead of completely abandoning the task) then I would be able to finish the task. I would say this experience and realization is my most important takeaway from this course. Furthermore, having a quite well functioning team (even though 3 members quit) helped with the process. I don't remember having actual conflicts. Of course, we had some misunderstandings, but we were all open-minded enough to take a step back and evaluate the matter in an objective and reasonable matter.

- **Mahmoud**: As a matter of fact, this course was more than a great experience for me, where the idea of coding as part of a team was vague to me, it turned out later to be a pleasant work to do, especially with great teammates, and reliable tools such as git. Despite my weak points were both coding and teamworking, I was able to be present and stay synchronize with others thanks to the Scrum framework. After putting the outline of our product. My contribution was mainly on the coding part of the transportation features and designing the UI. I considered here using some APIs to retrieve the route information with some map visualization, therefore, I spend some time on learning this properly. Fortunately, I was able also to get a free budget to be spent on Google API Services. So I have implemented this feature where the user can input from where to where he/she would travel and the program will show a map for the route with information about duration, distance, and CO2 reduction in both cases using bike or public transport instead of the car and commit to use one of these. Turning to UI design, where one of my strong points appeared, I took care of the aesthetic side of the UI and standardized the looking by using "JFoenix" library and some CSS code. Now, I am proud of our product thanks to all great-four team members. Also I am thankful for this experience which allows me learn a lot and step my communication skills up.

5. Value Sensitive Design

Chosen not obvious stakeholder: Car Manufacturers

Since the main value/design pattern for this project was given as sustainability that's in the core of our application, however to be able to achieve that we have created an app which focuses on user gratification, since it is the gamified, somewhat addictive approach, which is going to actually make the user login on a daily basis and complete activities which reduce their CO2 emissions and contribute towards the greater good which is in essence the goal of this application.

As required by the assignment, if we would consider the addition of another principal stakeholder such as car manufacturers, additional features which promote the interest of the car manufacturer should be created. Since there is an inherent strife between the interest of a general user, whose goal is to reduce CO2 emission as much as possible and a car manufacturer whose primary interest theoretically lies in making a profit. Of course, in our case, we shouldn't look too far ahead in the future as popularity of electric cars is rapidly growing making(technavio.com, 2018) the conflictive state between these stakeholders much less of pressing issue if at all. Nevertheless, our application encourages travel by bike or public transport rather than car, which still might cause a clash between the two parties, even if the car is electric, taking into account that the change of zero emission public transportation vehicles will most likely occur before all individuals acquire a zero-emission car(techcrunch, 2015).

To be able to decide what additional features could be made, consultation with car manufacturers and discussions about their plans on tackling the worldwide movement against greenhouse gas emissions, could prove to be beneficial as the features of our application could align with the plans of the car manufacturer, thus helping to promote their interests. Furthermore, the recruitment of scientists, whose research area concerns the emergence of mass efforts supporting the reduction of CO2 emissions and their effect on the sales of car manufacturers could provide further insight into how our software could balance the scales.

A few options/features we were able to come up with to potentially relieve the tension with the hypothetical stakeholder (car manufacturers) would be the creation of a tab which lists different electric cars, so that the user would have greater awareness of the available cars on the market and could also compare different brands and their unique properties as well as prices. This would greatly help the car manufacturer, by reducing their marketing and promotional costs as well as possibly giving a competitive edge over other traditional cars or other manufacturer's cars.

Another different feature, which could be combined with the one mentioned previously, would be a discount schema based on the CO2 emission prevention of the user. Therefore, certain amounts of points could be exchanged for % or flat reduction of the car price. Alternatively, it could also be that if the user purchases an electric car, for instance a Tesla, the user could get price reduction for Tesla solar panels or whatever product is in the range of the manufacturer and has relevance to the application and CO2 emission prevention.

References

Technavio, 2019, Technavio: Electric Car Market Size Available at: https://www.technavio.com/research/electric-car-market-size (Accessed 5 April 2019).

Techcrunch, 2015, Techcrunch: Urban Transportation Will Go All Electric Sooner Than You Think Available at: https://techcrunch.com/2015/05/29/urban-transportation-will-go-all-electric-sooner-than-you-think/ (Accessed 5 April 2019).

McKinsey, 2018, McKinsey: Fast Transit: Why Urban e-Buses Lead Electric-Vehicle Growth Available at: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/fast-transit-why-urban-e-buses-lead-electric-vehicle-growth (Accessed 5 April 2019).