#GoGreen Report

Group 83

Sami Farahi - 4723422

Lucas van de Geer

Shaan Hossain - 4908058

Alex Shulzycki -

Vanessa Timmer – 4728033

# Table of Contents

[Table of Contents 2](#_Toc5278676)

[Section 1: Product 3](#_Toc5278677)

[1.1 - Technological Choices 3](#_Toc5278678)

[1.1.1 - PostgreSQL 3](#_Toc5278679)

[1.1.2 – JSON (TODO: Alex) 3](#_Toc5278680)

[1.1.3 – JavaFX (TODO: Vanessa) 3](#_Toc5278681)

[1.1.4 – CalculatorAPI 3](#_Toc5278682)

[1.2 - Architectural Choices 3](#_Toc5278683)

[1.2.1 - Vegetarian Meal 3](#_Toc5278684)

[1.2.2 - Local Produce 4](#_Toc5278685)

[1.2.3 - Bike Ride 4](#_Toc5278686)

[1.2.4 - Public Transport 4](#_Toc5278687)

[1.2.5 - Temperature Adjustement 4](#_Toc5278688)

[1.2.6 - Solar Panels 4](#_Toc5278689)

[Section 2: Process 5](#_Toc5278690)

[Section 3: Reflection 6](#_Toc5278691)

[3.1 - Reflection on process 6](#_Toc5278692)

[3.2 - Reflection on product 6](#_Toc5278693)

[3.3 - Reflection on course 6](#_Toc5278694)

[Section 4: Individual feedback 7](#_Toc5278695)

[4.1 – Lucas van de Geer 7](#_Toc5278696)

[Contribution 7](#_Toc5278697)

[Weak points 7](#_Toc5278698)

[Strong points 7](#_Toc5278699)

[4.2 – Sami Farahi 7](#_Toc5278700)

[Contribution 7](#_Toc5278701)

[Weak points 7](#_Toc5278702)

[Strong points 7](#_Toc5278703)

[4.3 - Shaan Hossain 8](#_Toc5278704)

[Contribution 8](#_Toc5278705)

[Weak points 8](#_Toc5278706)

[Strong points 8](#_Toc5278707)

[4.4 - Alex Shulzycki 8](#_Toc5278708)

[Contribution 8](#_Toc5278709)

[Weak points 8](#_Toc5278710)

[Strong points 8](#_Toc5278711)

[4.5 - Vanessa Timmer 8](#_Toc5278712)

[Contribution 8](#_Toc5278713)

[Weak points 8](#_Toc5278714)

[Strong points 8](#_Toc5278715)

[Section 5: Value Sensitive Design 9](#_Toc5278716)

[5.1 – Sami Farahi 9](#_Toc5278717)

[5.2 – Lucas van de Geer 10](#_Toc5278718)

[5.3 – Shaan Hossain 11](#_Toc5278719)

[5.4 – Alex Shulzycki 12](#_Toc5278720)

[5.5 – Vanessa Timmer 13](#_Toc5278721)

# Section 1: Product

## 1.1 - Technological Choices

### 1.1.1 - PostgreSQL

Since we are all familiar with PostgreSQL from the Web and Database course we choose to use PostgreSQL.

### 1.1.2 – JSON (TODO: Alex)

To be added

### 1.1.3 – JavaFX (TODO: Vanessa)

To be added

### 1.1.4 – CalculatorAPI

We chose to work with the API from Brighterplanet, because they had all the aspects we needed for our project. We used the API especially for the Car vs Bike/Public transport features. Second for the vegetarian meal we decided to use fixed amount of points. Furthermore for the ‘Temperature adjustment’ we searched about how much CO2 you could save when changing the temperature of your house, from a high temperature to a lower temperature. And so we used the API again for calculating how much CO2 you would save per month when adjusting the temperature of your house. About ‘Local produce’ we found out that whenever you buy imported food (1 kilograms), in this case fruit and vegetarian food, you would emit 560 grams of CO2. However when you buy local food (1 kilograms), in this case fruit and vegetarian food, you would emit 256 grams of CO2. So we made a calculation where we would subtract these amounts from each other. As last about the ‘Solar panels’ we found that whenever you use solar panels, you would save 0.46 kilograms of CO2 per 1 kWh (Kilowatt-hour).

## 1.2 - Architectural Choices

### 1.2.1 - Vegetarian Meal

To be added

### 1.2.2 - Local Produce

To be added

### 1.2.3 - Bike Ride

To be added

### 1.2.4 - Public Transport

To be added

### 1.2.5 - Temperature Adjustement

To be added

### 1.2.6 - Solar Panels

To be added

# Section 2: Process

# Section 3: Reflection

## 3.1 - Reflection on process

To be added

## 3.2 - Reflection on product

To be added

## 3.3 - Reflection on course

To be added

# Section 4: Individual feedback

## 4.1 – Lucas van de Geer

### Contribution

To be added

### Weak points

To be added

### Strong points

To be added

## 4.2 – Sami Farahi

### Contribution

At the start of the project I had no clue of what I could expect from this project. Especially with the random groups, I was not so sure how I would work together with people I did not know at all. At our first meeting everyone was a little bit quiet which made sense, because no one knew each other. But after a while we started to talk. We actually had seven members in our group, but two of them never showed up, so we ended up having only five members. This was not really a problem for us, because we all did our part that had to be done. I figured out that I did not have to be afraid at all to let the others know my opinion about things, which was new for me. As I stated in my personal development plan. I did not really know what I could do in the beginning, because there weren’t many tasks. After our first demo, I realized that this project is actually really nice to do. I worked pretty well with the other member of the group, and the communication also went good. I decided that I would do the part of the API calculator. The TA told me that this could be the hardest part of the project, but I wanted to challenge myself, and see if I could figure out how it all works. The API calculator cost me a lot of time, I struggled a bit, but I managed to get it done. I was proud and relieved that I got it done, because it was a very important part of the project, and also one of the hardest. Overall, I think that my progress throughout this project was quiet good, I developed my programming skills and my lack of letting others know my opinion. We did a good job together as a group, and I think we ended up with a very good Java application.

### Weak points

I stated in my personal development plan that I sometimes don’t really speak up, and that I don’t really share my opinion. I think that I managed to improve these weak points, and worked on them during this project. I noticed that when I spoke up, or shared my opinion that there was nothing to be afraid about, which made me realize that I should do it more often.

### Strong points

Two of my strong points are that I can work well in a group and that I am always willing to help others. In the beginning, I did not really know what I could expect from the group, but I had no doubts that I would be able to work well together with the other group members. Whenever someone asked me if I could help them, I tried my best to help them. I also offered sometimes if someone needed any help, or if they had any question or something, they could just ask them.

## 4.3 - Shaan Hossain

### Contribution

To be added

### Weak points

To be added

### Strong points

To be added

## 4.4 - Alex Shulzycki

### Contribution

I wrote the server, both in terms of network communication, and setting up the framework to process requests, which includes authentication (with salted passwords). I also wrote the https client for the application-side, and made the server https compliant. I also had a hand in optimizing the database classes and rewriting some of them to make them more robust, and made sure that everyone was on the same page by writing up a standard and consistent JSON API we would use when communicating between server and client.

### Weak points

I don’t think that some of my explanations were clear enough in terms of what I was doing and how to use the frameworks which I set up, so I wrote up the api list and serverquery how-to guides (in /docs) to illustrate how everything works. I feel like sometimes there were times where I didn’t show enough initiative when I should have did, but I hope that I have learned my lessons and that for future projects I will take more responsibility when I need to.

### Strong points

I made my code as robust and universal as possible, which paid off immensely once we began adding functionality and actually making the whole thing work. I also made sure to thoroughly test (the parts that could be tested) my code and made sure that others knew how to do it as well (once they started writing within my serverquery framework)

## 4.5 - Vanessa Timmer

### Contribution

To be added

### Weak points

To be added

### Strong points

To be added

# Section 5: Value Sensitive Design

## 5.1 – Sami Farahi

## 5.2 – Lucas van de Geer

## 5.3 – Shaan Hossain

##### Design for Environmentalists

The main concept of our project is originally aimed at non-environmentalists. But we need to take into account that environmentalists are actually the ones who can influence our program the most. This is because all of our calculations are based on data that is inspired by their perseverance in making this planet a better place to live on. They are the ones who encourage and motivate researchers to keep investing time, money and effort into

The main focus to improve our application would be retrieving the data that is used to calculate the CO2-emissions. As for right now these calculations are just bare estimations and speculations. To polish these calculations some actual research will have to be done by us, either looking for the more specific data ourselves or looking for someone who has done some research about this. We have also taken far too few factors into account. The amount of CO2 that is saved by eating a vegetarian meal is not a constant but rather depends on many things such as, what are you replacing your meat with and in what season are you eating your vegetarian meal in. Leaving out all these factors might not be relevant all these non-environmentalists, but I can imagine that calculations with large errors might lead to conflicts with actual environmentalists since the number do not represent the environment correctly. The worst-case scenario would be that an ignorant client of our application thinks he is saving up more CO2 that he actually is.

Improving our sources of information and data can have many degrees of improvement. A minor improvement would be to simply extend our research in data and add some of these extra factors to the calculations. By doing this more extensive research we might be able to narrow down our error in CO2 calculation. This research can either be done online. Reading articles and academic papers is a good way to start. The first page after googling ‘reduce co2 footprints’ already shows some very promising articles and even a TedTalk on how to “cut your carbon footprint by 60%”.

For a bigger improvement we will need to dig deeper. We will have to confront an actual environmentalist and/or researcher. They can give us insight on how far the problem branches out and were we should stop finetuning our calculations. These researchers can provide us more accurate data specifically for our application. The Georgetown University seems to have made some progress into the research of the ‘Carbon Footprint’. Also the Dutch government agency RVO (Rijksdienst Voor Ondernemend Nederland) has a good overview of organizations who are doing research on sustainability and innovation. By digging deeper and trying to truly grasp this global issue we can make our application more accurate.

We must not forget the influence we can have on non-users as well. Yes, our application can even change the lifestyle of someone who will never be in touch with our application. Imagine 2 friends sitting at a bar. Friend X has been using our application whereas friend Y has never even heard about it. Friend X is telling Y about how he is eating vegetarian at least once week and goes to work by bike. He has been saving roughly 60kg of CO2 a month. Friend Y is surprised that such a small change in lifestyle can have such a big impact and he decides to change his lifestyle as well. Of course, this is a fictional story and unlikely to happen exactly like that. But the main point is that our application can influence many people, even if they will not use it.

Our application is still very barebones if you think about all the additional features that can be added. And because it is still in its alpha version there is a lot of room for improvement as well. But no matter what we will change to our application, we will always to take all of our stakeholders into account.

## 5.4 – Alex Shulzycki

## 5.5 – Vanessa Timmer