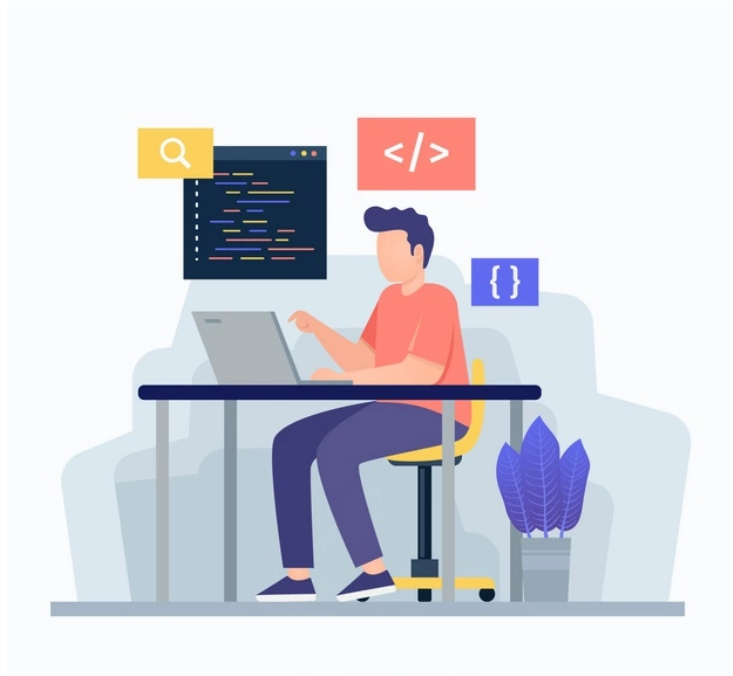


# Sprint 3

## Coding Journey



### 1. First steps

Setting up git and getting introduced to the commands. We made sure to use different branches for each page so everyone could work without having conflict. Before setting up our project we did testing to make sure everyone did understand the commands and branches. Check out our blog to see our group in action (some videos & pic).

#### 1. Dividing the work

Everyone shared their opinion on what they wanted to work. We as well took into consideration the coding skills of our team members. We had as well at that time other challenges and documentation we had to take care of.

At first, everyone was assigned one page each. Of course, some people work faster than others so we decided that whoever finishes sooner can take another page. Our goal was to finish the front-end by the end of November.

Ali took interest in the database together with Umberto. The rest were in charge of the making of the other pages. Later on, Ali took the lead into the database so Umberto was assigned some pages to help the group.

End result division :

Database & Back-end - Ali

Login/Sign up - Umberto

Reward page, Profile page, Location page, Home page - Iris

Challenge page, Sponsor page - Catherine

Contact page, Newsletter - Desislava

## 2. Progress

We had short meetings 2-3 times a week to check on everyone's progress. We always helped each other whenever someone was stuck. Iris and Catherine helped when there were problems in the CSS or JS for the other team members.

The newsletter went through three design transitions. Each one is becoming better and better. Desislava manages to add a backend to it. However unfortunately at the time, we did not manage to connect it to the original database.

### 1. Some difficulties:

- A lot of pop-ups were created in the challenge page and that took a lot of time to be made and fixed with JS.
    - Catherine spent a lot of time on improving them so they did work correctly.
  - When we had to connect the database to the website's point system, the pop-ups were no longer working because the page was reloading.
    - This was fixed by Ali.
- When we had to connect the homepage to the newsletter page, the CSS for both pages were broken.

At one point we tried to organise our files better. For example, by putting all rewards pages into one folder. However, what we did not take into consideration was that we had to change all path links. We of course used the shortcut commands but then other errors occurred. We decided that it would be best to go back to the previous version because of the time we had left.

### 2. Decisions:

Although Ali manages to fix most of the pop-ups with the database. There were two things we did not manage, so we had to make a decision. We had to decide if it was more important if:

- The claimed rewards to disappear *permanently* when used, but then we would not have the whole user interaction of starting the timer getting the confirmation.
- Or have the whole interaction working but when reloading the page the reward will appear again.

We decided that we should keep the user interaction because that was a big part of our concept. We decided to keep the game as well because it was something everyone liked.

### 3. Biggest mistake we made

We were working on the website the very last day of our submission . We wanted to use all the possible time we could to finish. We merged everything 2 hours before the deadline.

- We got a lot of errors. Things were no longer working correctly and there was stuff missing that we did put on Git. We panicked because of the time limit and should have stopped and submitted the work we had at the time.

#### **What we learned :**

- Rule number 1

We made a decision to have our finished website hosted on Hera at least 2 days before the deadline.

### 4. Changes:

#### 1. From QR-code to NFC -tag

We decided to change the concept a bit. Instead of using the QR-code we now use an NFC tag.

Why? In the beginning, we decided to use QR- code. However, during the process, we found out that the NFC tag was easier to implement and use than the QR-code. We came up with new steps and created a special menu that the barista will use to interact with the NFC tag.

We did some tests and researched the topic. We found out as well that more and more companies are now headed toward the use of NFC. Umberto bought some NFC -tags and we borrow an android phone from Fontys and started to experiment. The reason we borrowed an android phone from Fontys was that after some research we found out that the web NFC Api only worked on certain android phones.

- Using NFC technology in our concept makes our project unique, on the other hand, it gives us a negative feeling that we would not be able to completely implement this idea. Although this was the most difficult part of our project, we managed to finish the concept, and the functionality related to this part is working as desired.

## 5. Database & Back-end

1. MySQL 5.7.26
2. PHP

At the beginning of the project, we had no previous experience with back-end development in a PHP language, so a large portion of time was dedicated to learning, understanding, and testing smaller bits of functionality, as well as looking at alternatives to figure out what I thought worked best. Eventually, we would end up having to teach ourselves PHP programming language.

### 1. Login/Register/Forgot Password/User configuration

- One issue we had was using and matching passwords, the user's password needs to be encrypted due to security reasons. We fixed this issue by using [password\\_hash\(\)](#) function.
- During the forget password process, we needed to send a 5-digit code for the user's email to confirm user identities. Because of using Hera's as a host, we could not send this email to the user email.

### 2. Menu (NFC)

- The menu page was created so we could have a smooth interaction between the user, the barista, and the website. After adding a customer's order to the menu page, it's time to scan the loyalty card/NFC of the customer to claim points to the customer's profile. All these actions are taken by the barista. For using NFC technology in the browser(mobile version) we need to use [The Web NFC API](#). We have used JavaScript for reading data(customerID) from a loyalty card (NFC) and importing this data (customerID) to the database using PHP.

### 3. Points system

- For adding the point system to the website, we had to get the points the user already had plus the sum of points the user took recently. An issue we had was that: When we connected the database to the website's point system, the pop-ups were no longer working because the page was reloading. However, Ali did manage to fix this with some help from the teacher.

#### 4. Contact forum

- Because of using Hera's as a host, we could not send an email to Lucifer's email.

Quotes from our database developer (Ali):

“Although I was not able to complete some of the proposed functionality, the functionality related to the core purpose of the website is working as desired. I believe there is still a lot of potential for this website, and If we had more time we would be able to add even more .”

#### 6. What could have been better:

- A big issue we had was time management. Even though we had a lot of meetings and were setting deadlines there was a problem. The problem was that not everyone was finishing their work on time.
- Often some members had to help a lot of others, where they still had to work on their own pages. That means they had more pressure and more work to do.
- We often had different opinions on certain topics and we ended up spending too much time discussing some tasks when we could have worked on them the same day.

#### 7. What went good:

- Everyone did have a different opinion but we were always respecting each other's point of view.
- We always took into consideration the feedback of the teachers.
- We always tried to help each other.
- We were motivated to learn new things from each other.
- We did not have big problems during the front-end process.

#### 8. What we learned/ will improve:

- Have a clear concept always at the start
- Do not keep adding new ideas when there is a time limit. Can lead to more errors on other parts.
- No longer meetings than max 40 min.
- To stay on topic during meetings and not divert to redundant chatter.
- Keep the coding organized from the start.
- Always double-check for spelling mistakes.

Taking into consideration what we learned:

Ali managed to fix the database for the newsletter however when we did connect it other problems appeared. For the sake of not repeating our mistake from before we decided to leave it out of the website. However, we did not want all the effort to be left unseen. We decided to leave it as a comment in the code.

If we had more time:

We would have added a button for the newsletter that will be stuck to the right side of the screen.

We would test the website more to see where we can improve the security of the website against potential abuse of our point system.

**Thank you for reading our journey!**