|  |  |  |  |
| --- | --- | --- | --- |
| **CS102** | **Fall 2020/21** | Project Group | 1D |
| Instructor: | **Uğur Güdükbay** |  |  |
| Assistant: | Sinan Sonlu |  |  |

|  |  |  |
| --- | --- | --- |
| **Criteria** | **TA/Grader** | **Instructor** |
| Presentation |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Overall |  |  |

~ Sustainability Coach ~

SavioursoftheGreenWorld

Özgür Göker,Batuhan Balaban, Murat Ertan

|  |
| --- |
| Detailed Design Report  ( Version I )  23 December 2022 |

# Introduction

Sustainability coach is a program aiming to the users keep track of their sustainable habits by enabling them to record achievements given by the program. Our program has a website-like interface which requires signing up and signing in for further actions. After the login, the player will record the things they have done on the achievements page. On the news page, we can see the highest-ranking players. In the profile section, the player can update and maintain their information. In this document, we would like to extend the classes we will use and our final product environment. We have shrunk some features of our planned program as we have lost some team members along the way.

# Details

## Functioning

When a player starts the program, they will have to login using their name and surname with their password. Note that players with the same name and surname are different players given their passwords are different. No email verification for this project.

When the player interreacts with achievements, they will create an interaction object (Time Stamps) which will be stored on this player’s achievements list in their player object. In this way, all the data unique to the player will be stored, which could be followed by this player. The Statistics class will hold all the users and help calculate rankings given each player interaction Score. This class will also calculate the time difference to develop a time-based ranking reflected on the total score. All the generated data from the statistics class will be displayed in the GUI news section. Note that those classes do not hold GUI details as the GUI itself will adjust such. Other than that, there will be achievements and a challenge list for every player. The achievements list will hold the completed challenges. The Challenge list will contain the uncomplicated yet given challenges. If an achievement is completed once, the tree will be coloured red and if not, it will be blank under the achievements tab.

## Solution Architecture

By storing all the player interactions, the program will come up with rankings and conclusions which will incentivize the players.

The Player class is responsible for holding each player’s data. This class introduces tools to process such data accordingly. Each player’s game data is held by their ArrayList<Achievement>. Player objects will store passwords that make their account unique. We do not deploy further security measures for this project.

The achievement class is responsible for holding achievement objects under the player by aggregation. Each player has an achievement list. The game designs those achievements, but specific attributes that are special for the player will be held under their class-specific objects. This class will be used to load GUI-specific messages across the platform for the given player. Additionally, we may want to store trophy conditions if a trophy system is thought necessary by the supervisors.

The achievements class and its section under GUI will take user information in the addInteraction method and will add challenges to the user. If any challange is completed or unwanted by user it will be deleted by deleteInteractons method. Other than that, calculatePlayer Interactions method will calculate how many challenges and challenge parts the user is completed. From that information it will give feedback to the user. If the feedback is positive user will be awarded by points and badges. If the feedback is negative ,the user will have warned messages from the app. Other than giving feedback it will rank users between them. This will encourage users to do more challenges. Finally addPlayerNote method will take users’ notes and giving that the date. From that date, user can be reminded. On that time users’ notes will pop-up.

Interactions class will basically be a time-stamp for player input. Those objects will be responsible for determining the player’s score or habitual grade. This class holds date objects to time stamp the interaction, an integer value to be used in playing processing for quantitative data and finally, the String input that the player has given information about.

The statistics class holds all player objects that are registered to the system and is responsible for calculating player scores for ranking. We could extend the processing methods to increase different rewarding systems, but a primary, frequency-domain amplitude-based ranking system would be sufficient for three people working.

## Technologies and Environments

* We are using UMLET for UML diagram generation and viewing
* We are using APACHE NETBEANS IDE for GUI
* JAVA

## Back-end Classes

### Player Class

- Holds player information such as name surname, gender, status, achievements etc.  
- Adds profile image  
- Populate badge information of the player  
- Calculate players scores based on their achievements  
- Compares scores of the players

### Achievements Class

- Holds and sets achievements information

-Holds Stamp Objects to time hash interactions

### Stamp Class

- Holds player comments and the stamping date as well as numeric input.

### News Class

- Creates textfields for the news page  
- Set format, color etc., of the text

### Statistics Class

-Calculatenumeric score, penalty, habitscore

### CoachControl Class

-Set and hold badge information for every player  
- Loads News which are player comments and player rankings  
- Check if the player has pass the badge criteria.  
- Load profile name, gender, score, stats ...  
- Logs in the player  
- Logs out the player  
- Set frames visible or invisible  
- Loads achievement information  
- Gets player’s achievement information  
- Updates profiles  
- Ranks players  
- Gets stamps and uses them in order rank players or give badges  
- Find the player according to his/her name  
- Register player  
- Create coach tips for player  
- Deletes player accounts

## Front-end Classes

### treeComponent

-Tree GUI component on the achievements page

-Inherits from the JComponent

-Keeps circiles and lines in array lists

-Overrides paintComponent

-Chnages progress information to paint leaves of the tree

-Takes a 3x8 string matrix to label leaves of the tree.

### generateTree

-Takes a treeComponent object in the constructor.

-Adds circles and lines to the treeComponent object.

-GenrateSubBranch methods puts circles in the relevant subbranch.

-GenerateMainBranch puts circles in the main branch.

### TreePanel

-Adds createsTree component and manipulates its through generateTree class.

-Adds buttons to the panel.

-Contains actionListener classes of buttons.

-Changes messageMatrix of the tree using treeComponent’s changeMessage method.

-Chnages leaves of the tree using changeProgressChart method of the treeComponent.

-Creates panels of buttons and places that in the south of the panel using BorderLayout.

### SignUp and SignIn

-When constructed, calls rankPlayer method of the coachControl class so refreshed rankin in the news page.

-In the action listener of the sign in button, registerPlayer method of the coachControl is called.

-When sign in is clicked, player data is uploaded to the program from a file.

### PopUp

-This is the pop where player informs the program that achievement is completed.

-In the Completed button’s action performed method, it stamps the achievement.

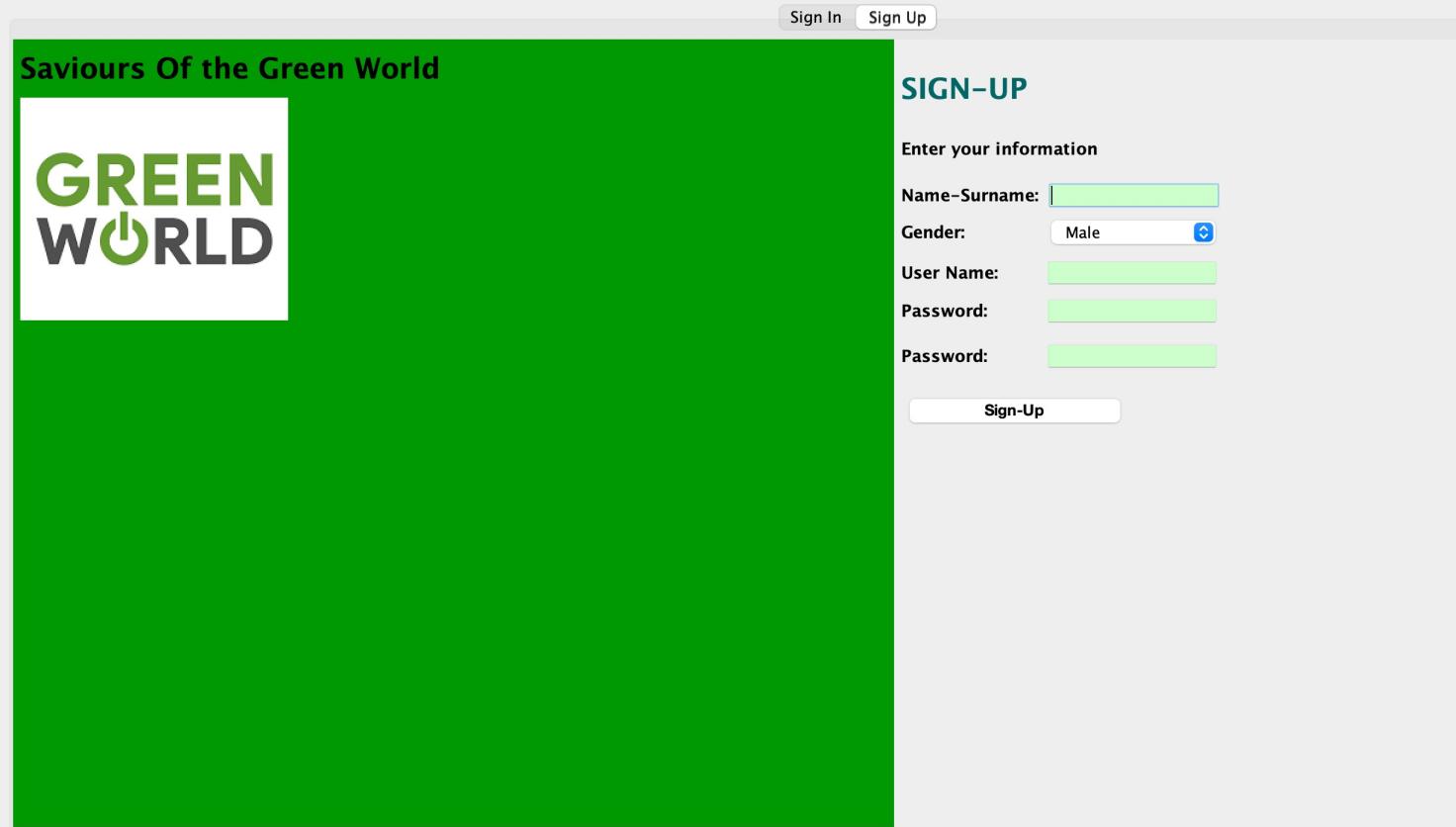
-In the combo box action performed method, it changes the pop up info.

### Profile and News

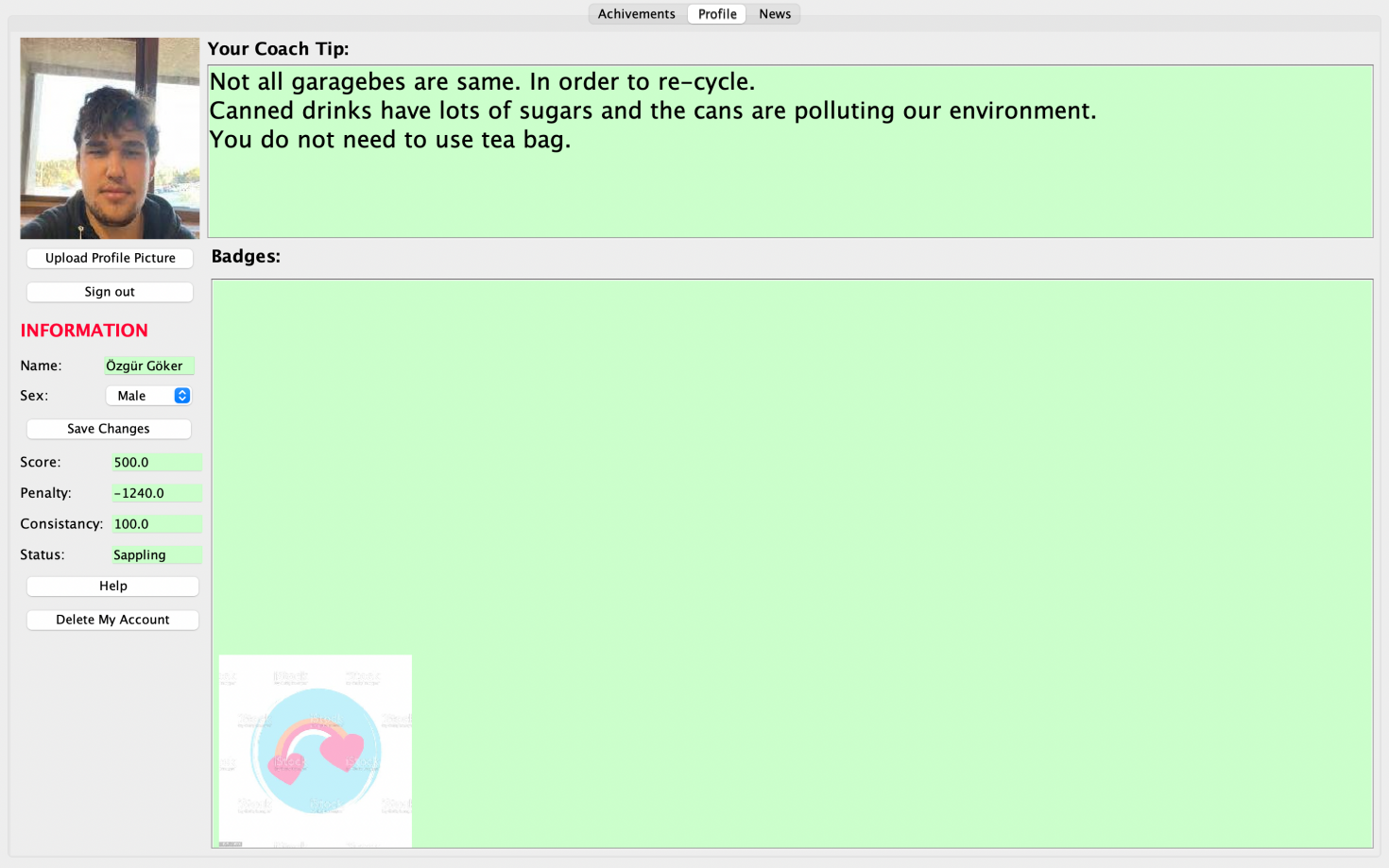
-Both takes coachControl object as argument

-In their action performed methods calls neccassary coachControl method.

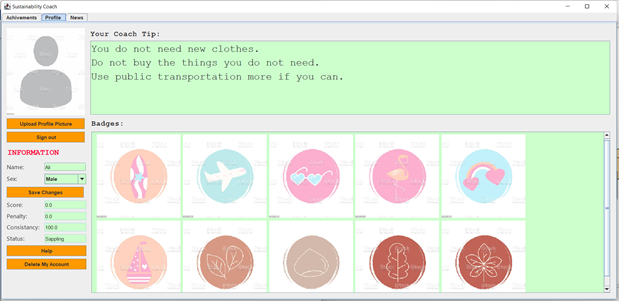
## Pictures from our app



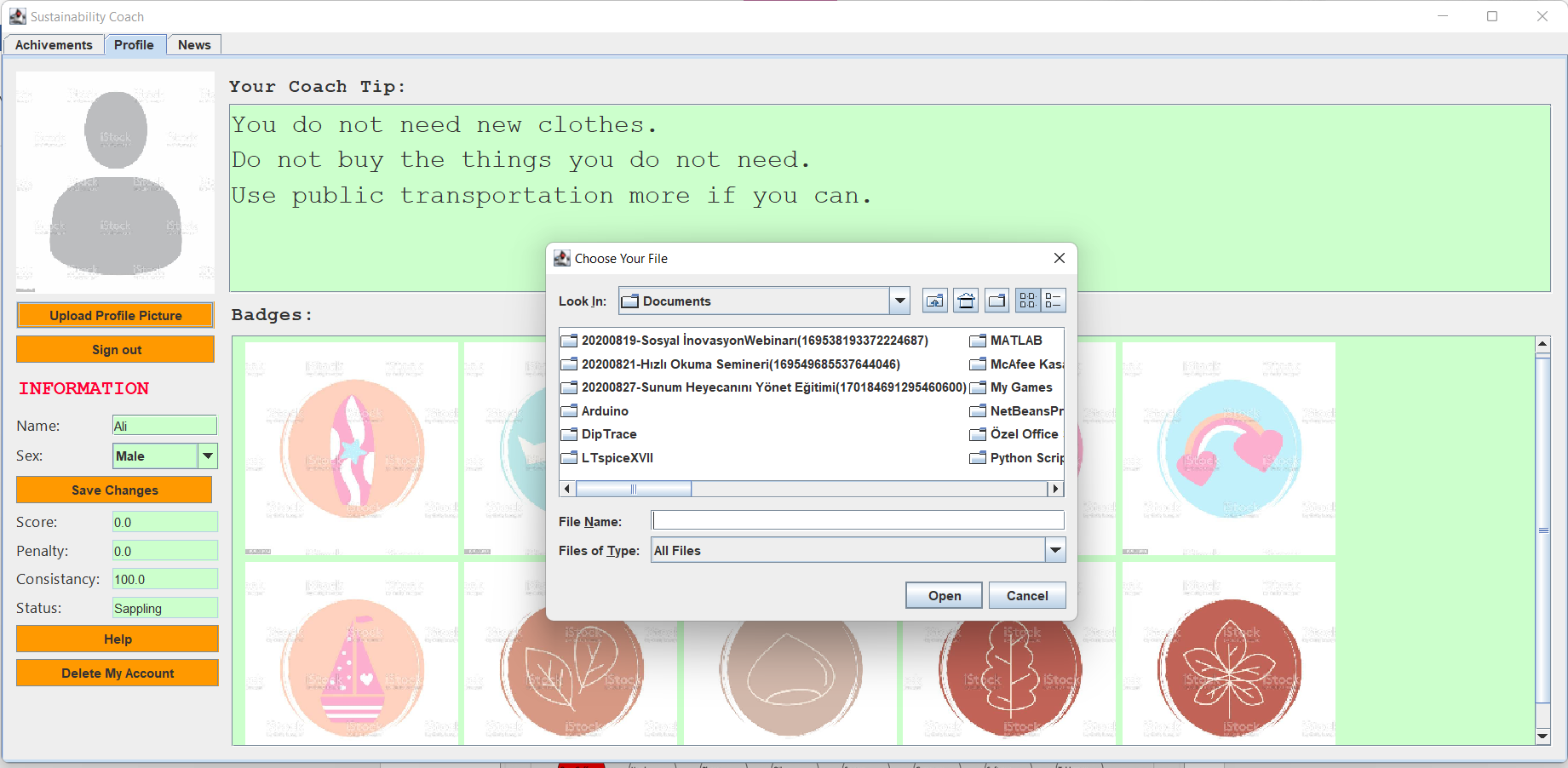
Sign-Up page (Makes Validation!) -> Tabbed page can move to sign-in (No Validation)



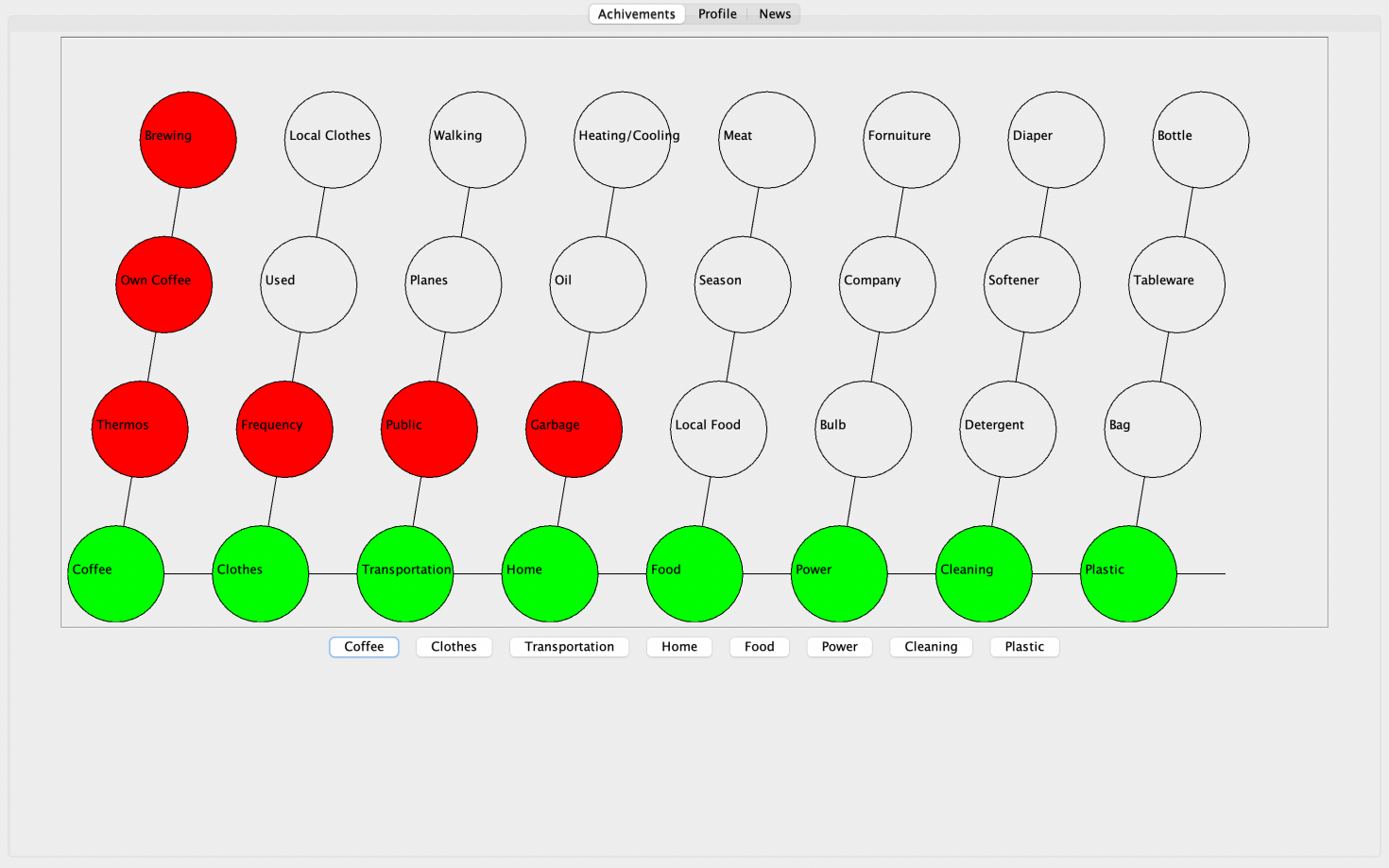
Profile Page



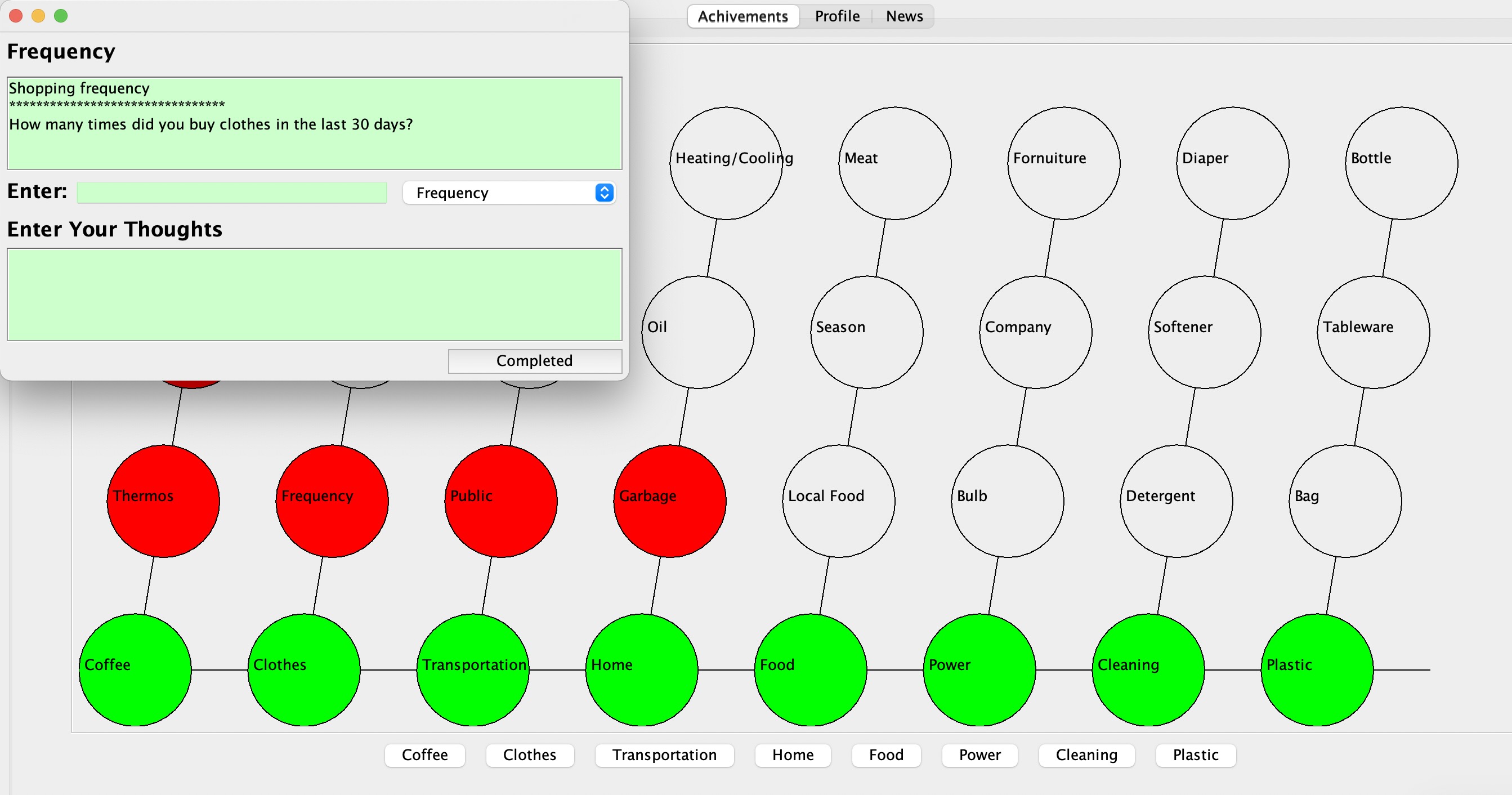
All badges visible



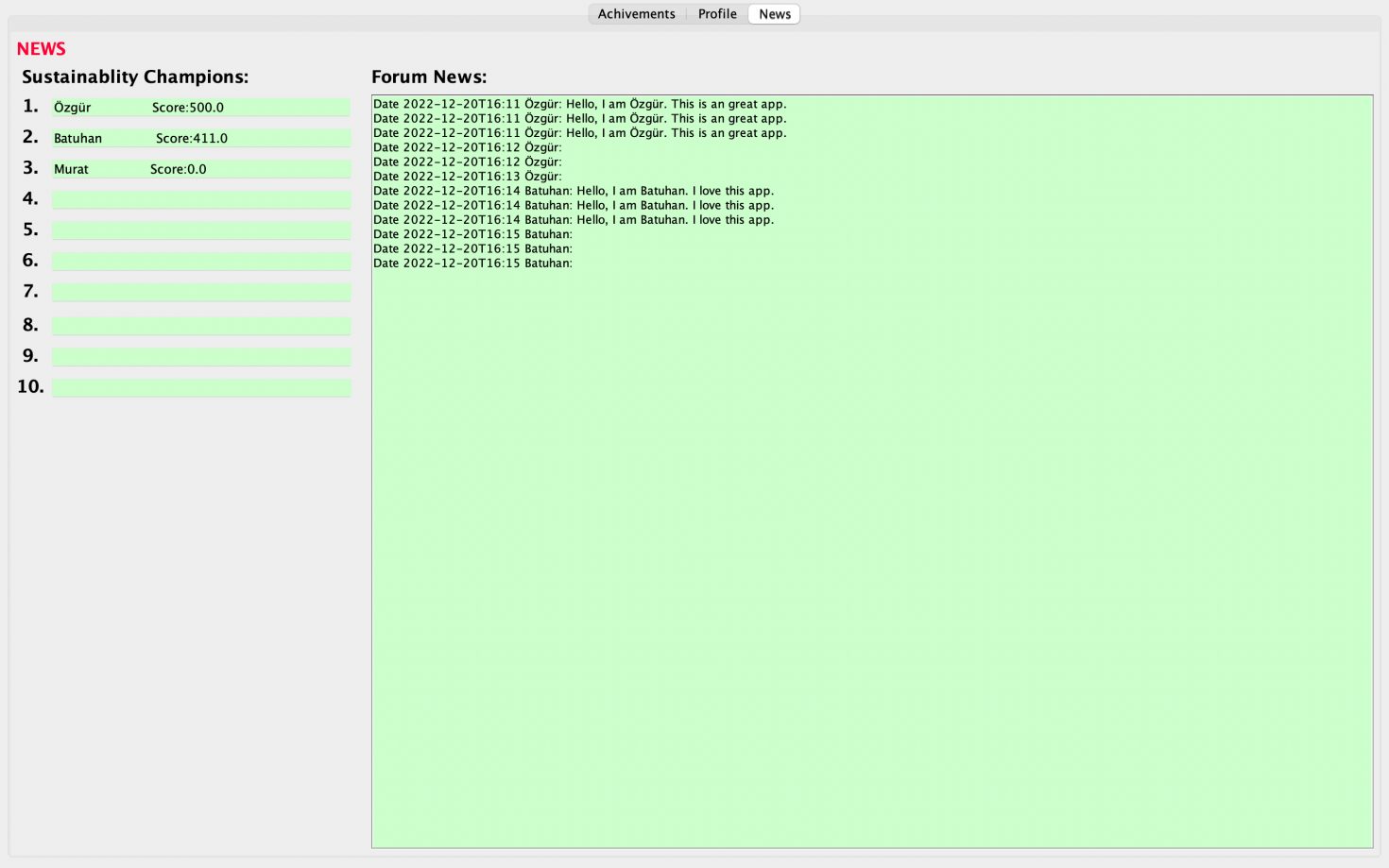
Uplading profile picture



Sample Achievements Page

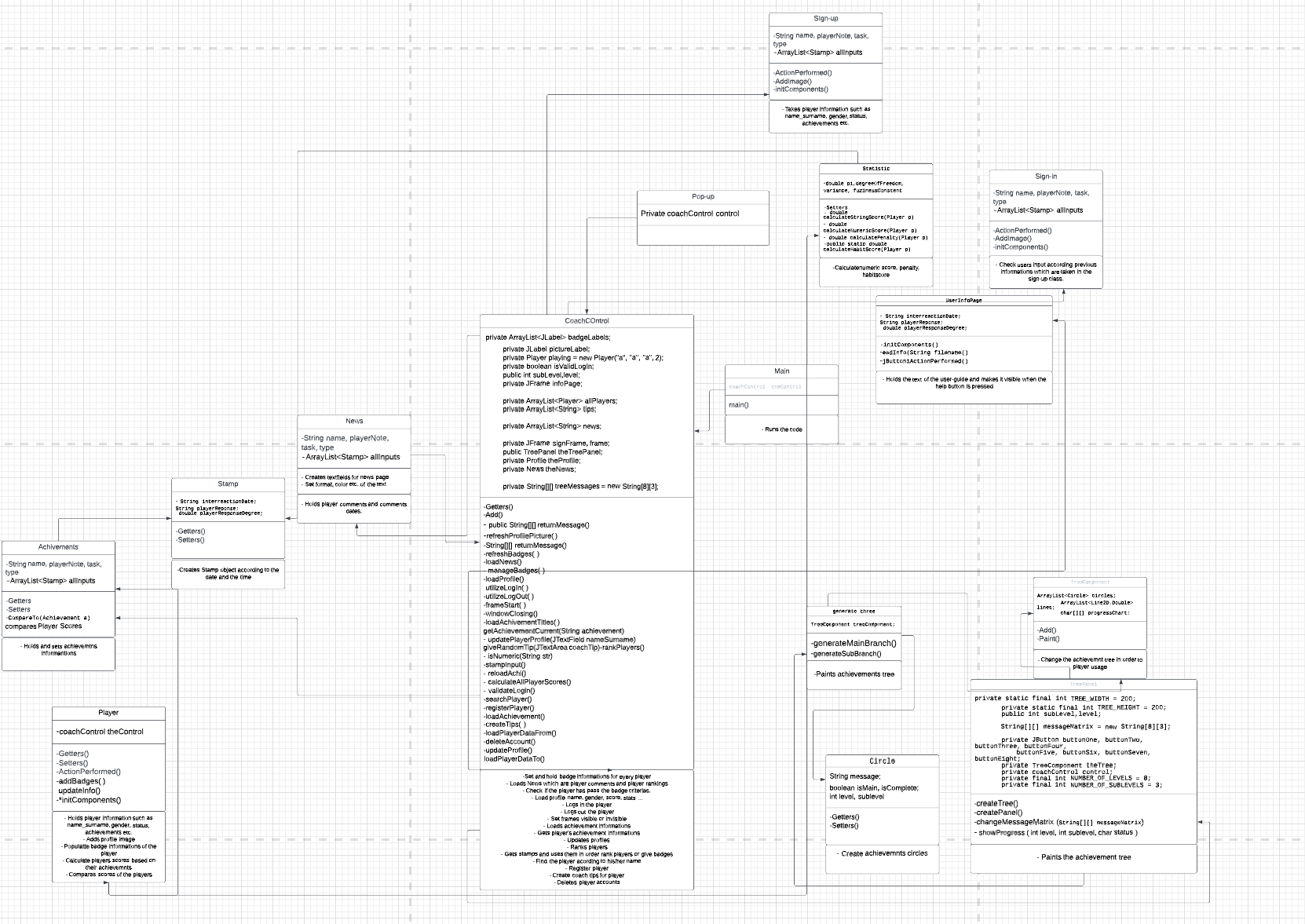


Time Stamping a Habit



Logs have shown for each Contribution to the game from each PLAYER.

## FINAL STATIC UML Diagram



[https://lucid.app/lucidchart/abbb80cc-bdbc-45cc-a399-b8c3e2a86459/edit?invitationId=inv\_62b277ee-5b54-4400-ad16-f0e831bebd3c&page=0\_0#](https://lucid.app/lucidchart/abbb80cc-bdbc-45cc-a399-b8c3e2a86459/edit?invitationId=inv_62b277ee-5b54-4400-ad16-f0e831bebd3c&page=0_0)

Given classes will process the player input collected from GUI. GUI classes are listed elsewhere as they have been in the GUI report. As we have lost two members, responsibilities are distributed as follows:

## What We Achieved in The Final

l We achieved gamified inputting player data.

l Reward players based on their contribution to the environment.

l Make an intelligent program that stores player progress.

l Mid-level model for the game design.

l Friendly user interface.

## What We Didn’t Do in The Final

l We grant achievements but do NFT-ise.

l We run basic algorithms for determining user statistics.

l We use Local and do not make applications available across networks simultaneously.

| We do not time stamp when the badges are initially gained. This was planned to take place as a more prominent component in the machine learning part which we excluded due to the adaption process.

| No sign-in validation message

| Not showing the badge granting date on click

| Game is endless, but the levels are drawn in red if they are completed once. We could add different graphical segments for different numbers of interaction

# Final Task Contrubution

## Final Task Contrubution

1. **Batuhan Balaban** 
   1. **Creating the GUI of** 
      1. **Achievements**
      2. **Profile pages.**
   2. **Coding the** 
      1. **Achievements**
      2. **Profile classes.**
   3. **Creating and coding the** 
      1. **TreeComponent,**
      2. **TreePanel**
      3. **Circle**
      4. **generateTree classes.**

1. **Murat Ertan**
   1. **Coding the** 
      1. **Player**
      2. **Stamp**
      3. **Achievement,**
      4. **CoachControl**
      5. **News classes.**
   2. **Creating bridges between the classes and combining all of the classes.**
   3. **Helping to create**
      1. **Powerpoint of the presentation**
      2. **Detailed Design of the document**
2. **Özgür Göker**
   1. **Creating the GUI of** 
      1. **Sign-in**
      2. **Sign-up pages.**
   2. **Helping coding the**
      1. **Player**
      2. **Achievements**
      3. **CoachControl classes.**
   3. **Makin the**
      1. **UML diagram of the project**
      2. **Making a video of it while using it**
      3. **Creating players for the presentation**
      4. **Creating powerpoint of the presentation**
      5. **Making the Detailed Design Document of this project.**

# Reflection

**Murat Ertan**

What did you dislike about project work?

  First, for me Project work means teamwork. For individual assignments, you have to be skilful in your field. But for teamwork, that is not enough. You have to have social skills in order to communicate and blend your ideas into the final product. That being said, you feel like you are really performing in the field rather than feeling under-education.

What was the most difficult aspect of it?

  I like project work all the time. The only measure of quality is how your teammates perform. I wouldn't change any aspect of the work as we did consistent steps until the final product.

What would you do differently if you had to start over again?

  I am OK with what we have achieved, we lost two team members along the way. We adapted. That was the most challenging aspect, but we minimised the risks. Project work is better when you select your teammates. The more you can't, the worse it is.

How much time did you spend on it?

  I have spent considerable time developing the project and planning. It was not a small deal, but it has been a pleasure.

Are you proud of what you have achieved?

  Even though we had to shrink our project, it has been fun to deal with it. Also, the amazing characters of our teammates have been cherished throughout the whole process. If only we had more teamwork-based courses.

**Batuhan Balaban**

What did you dislike about project work?

  The only thing I disliked was losing two of our teammates. We had to remove the forum part. I would like to see the forum part. Other than that, we implemented everything we wanted.

What was the most difficult aspect of it?

  Figuring out how to save data was stressful because at first, we did not design our code in a way that classes can be saved. We had to change certain things to make players savable.

What would you do differently if you had to start over again?

  I would not change anything. We used Java libraries for GUI instead of Android studio, which I believe was a good decision. We had a chance to practice GUI.

How much time did you spend on it?

 I spent, on average, 3.5 hours weekly. Around 50 hours in total.

Are you proud of what you have achieved?

  I am proud of the final result because we achieved our goals.

**Özgür Göker**

What did you dislike about project work?

 I dislike creating GUI in java because it is unnecessarily more energy and time-consuming.

What was the most difficult aspect of it?

  The most difficult part is that the understand the code of my teammates.

They are smart, wonderful, and great teammates. However, understanding other codes is really time-consuming because everyone has their own style of thinking and coding.

What would you do differently if you had to start over again?

  I would not make major changes. My teammates were great. Our plan and our ideas were great also. The only thing I would change is being more disciplined, but because my teammates our very disciplined, I try to adapt to them, and I won lots of good habits from this project.

How much time did you spend on it?

 Until the last three weeks, I do not spend that much time. Mostly 7-8 hours.

However last three weeks we work loads of hours. We nearly spent all of our evenings. I spent at least 30-40 hours on this project. My teammates spent more time than me. In total there are at least 120-130 hours of work time.

Are you proud of what you have achieved?

 I am very proud of what we have achieved. Obviously we would like to achieve more. We would like to implement Nft based badges and Machine learning. However, we lost 2 members.  It is %40 percent of our team. According to that, I am very happy and proud of what we achieved.

# Summary & Conclusions

Henceforth, we can conclude that Sustainability Coach is a tracker for sustainable habits. What is more important is it could be extended with NLP recognition and level adjusting etc. It is a diary that helps keep track of sustainable practices and makes the player more aware of their occupation. With this app, people have more power over their lives and the environment. It will help people to organise themselves and their environment. The program coaches sustainable living with its To-Do style solution architecture. We have laid the foundation which could be developed. The critical aspect here has been to develop the project idea and apply the learned skills to the concepts in parallel with the curriculum.