

University of Prince Mugrin  
College of Computer and Cyber Sciences  
Department of Software Engineering



## **SE323- Software Process and Modeling**

### **Project – Semester I (Fall 2022)**

RecycleMe- Recycling portal for unwanted items

#### **Team Members:**

Fatima Aljalmoud 4010096

Jana Aldubai 4010372

Salwa Shama 4010405

Samah Shama 4010403

Sana Shama 4010404

#### **Instructor:**

Dr. Khaled Khankan

18 December 2022

## **Chapter 1: Introduction**

1.1	System Overview.....	3
1.2	System Objective.....	3
1.3	System Scope.....	3
1.4	Proposed Solution (System Functions and Features).....	4

## **Chapter 2: Software Requirements**

2.1	Functional Requirements.....	4
2.2	Non-functional Requirements.....	4

## **Chapter 3: Software Analysis and Design Models**

3.1	Use Case Diagram.....	5
3.1.1	Use case description.....	7
3.2	Sequence Diagram.....	8
3.3	Overall Architecture.....	8
3.4	Component Diagram.....	9
3.5	Initial Set of Objects.....	9
3.6	Class Diagram.....	10

## **Chapter 4: Evaluating the Quality of Models**

4.1	Validation of Models.....	11
-----	---------------------------	----

## **Chapter 5: Conclusion and References**

5.1	Conclusion.....	12
5.2	References.....	13

## **Chapter 1: Introduction**

### **1.1 System Overview**

This system will be in the form of a mobile app that will be available to those who live in Medina. Using this system, materials can be easily recycled from reputable sources. The customer can place recycling orders and earn points for their transactions. Later, these points will be checked in the system and exchanged for offers. Besides, the customers will be able to see their levels on the leaderboard, in addition to their positive influence on the environment, which will raise people's social awareness and make it easier for them to achieve a high-quality environment. Further, the admin will be able to add the offers that will be exchanged with points by customers and generate a report about the app's statistics. For the application's delivery representative, the drivers will be able to pick up customers' orders to start the recycling life of the materials added by customers.

### **1.2 System Objective**

The app aims to change people's attitudes and behaviors towards the concept of recycling, making it part of their lifestyle for a more sustainable future.

The app facilitates the recycling and utilization of waste rather than the process of disposing of it and opens the door for users to contribute to reducing the negative impact on the environment. The app provides users with financial rewards in the form of points and recycling discounts, in addition to providing leader boards to make them more competitive. This idea creates a competitive environment and encourages them to recycle their waste.

### **1.3 System Scope**

All features of the system, including login, downloading a report, adding offers, the ability of customers to create orders and exchange points, alongside the system functionality of displaying orders to users, a leaderboard, and customer impact, are represented by the use case diagram, the software architecture diagram, the sequence diagram, and the component diagram.

In terms of the class diagram, it is limited to visualizing functionalities that are related to order processing, system users with authentication, and offering functionality.

## **1.4 Proposed Solution**

To assist in the process of recycling the items, the application provides five main functionalities. First, deliver the items that need recycling by sending a driver to gather them. Second, receive points for each order created to deliver the items. Third, exchange the points with the offers provided in the application by requesting an offer. Fourth, display the positive impact on the environment after recycling the items. Finally, display a leaderboard that shows the top 50 customers who's the most positive impact on the environment.

## **Chapter 2: Software Requirements**

### **2.1 Functional Requirements**

- The user shall be able to view orders with their details.
- The admin shall be able to download a report showing the statistics about the user and recycled item up to that day.
- The admin shall be able to add offers in the app.
- The customer shall be able to create an order.
- The customer shall be able to view the leader board that depends on the customer's impact on the environment. (User impact is measured by the amount of reduced CO2 per recycled item gram)
- The customer shall be able to view his/her own positive impact on the environment.
- The customer shall be able to view his/her points in the app.
- The customer shall be able to request offers by his/her collected points.
- The customer shall be able to add details in his/her order (Item's name, type, location, time, date, picture, statuses (sent, in progress, received), weight)
- The driver shall be able to select the customer's order.

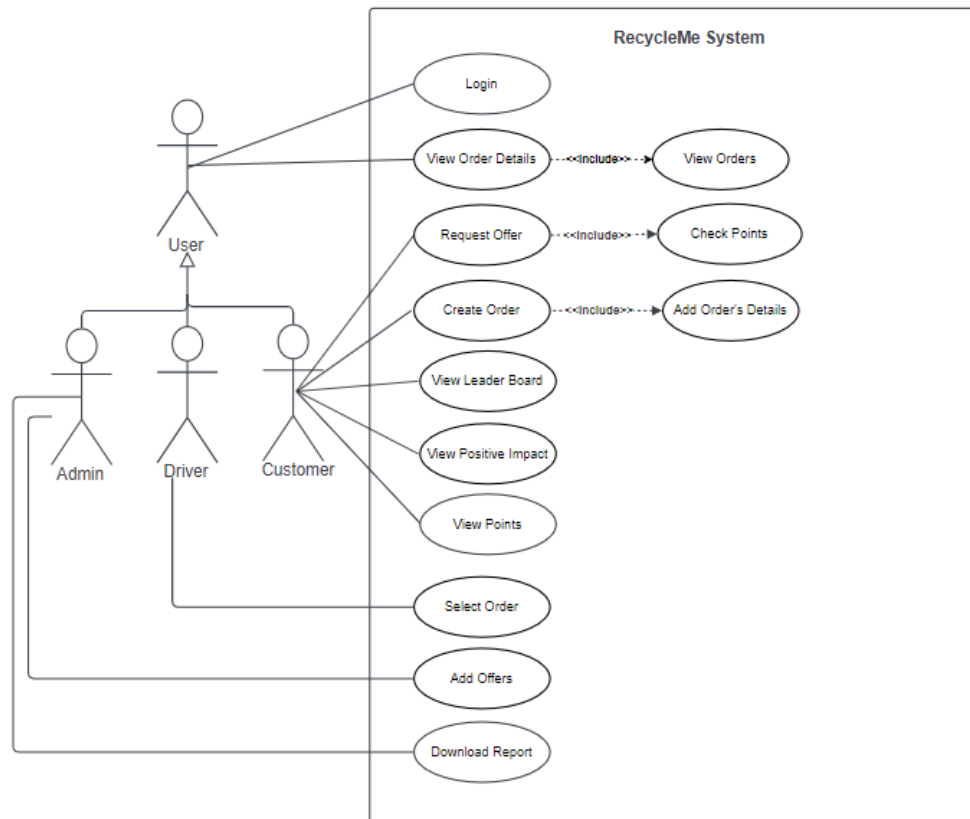
### **2.2 Non-functional Requirements**

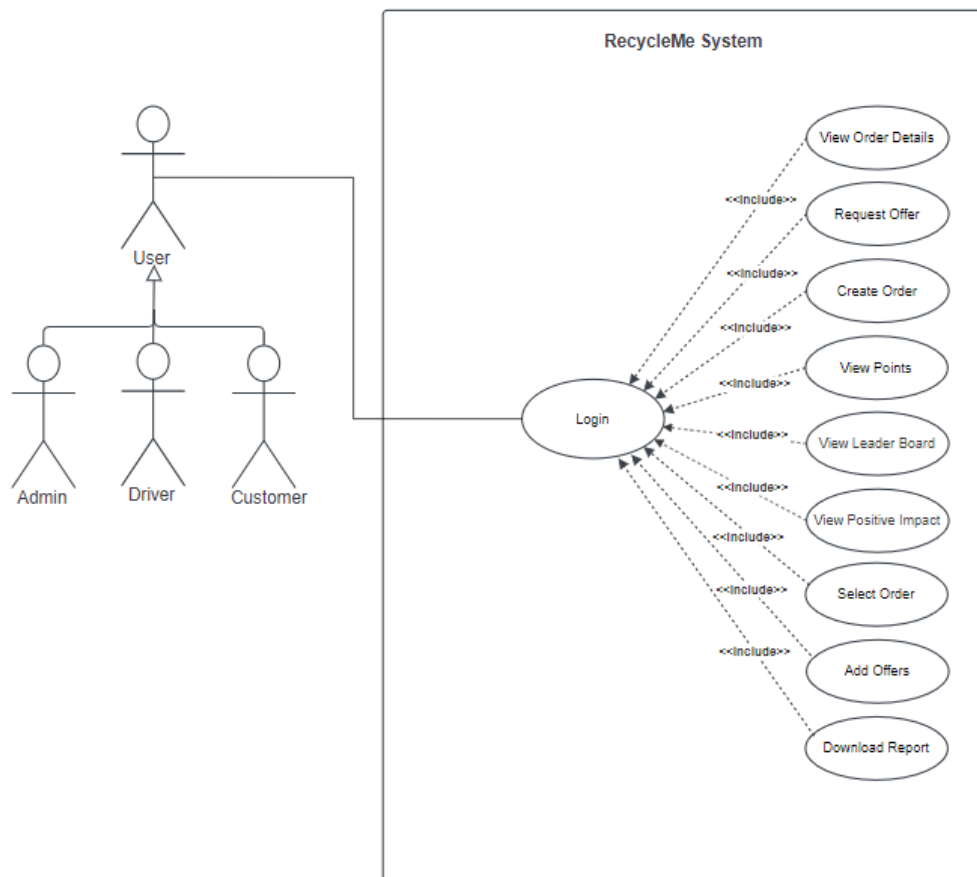
- The user shall be able to login after matching username and password. (security)
- The system shall be available 24/7. (availability)
- The order weight shall be more than or equal to 5kg. (Constrain)

- The system shall be able to respond quickly to any transaction in less than 10 seconds. (Performance)
- The user shall be able to use all the system functions after four hours of training. After this training, the average number of errors made by experienced users shall not exceed two per hour of system use. (Usability)
- The system shall be able to reject any exchanging points process with not enough points. (Constrain)
- The customer shall not be able to see other customers' orders. (Privacy)

## Chapter 3: Software Analysis and Design Models

### 3.1 Use Case Diagram

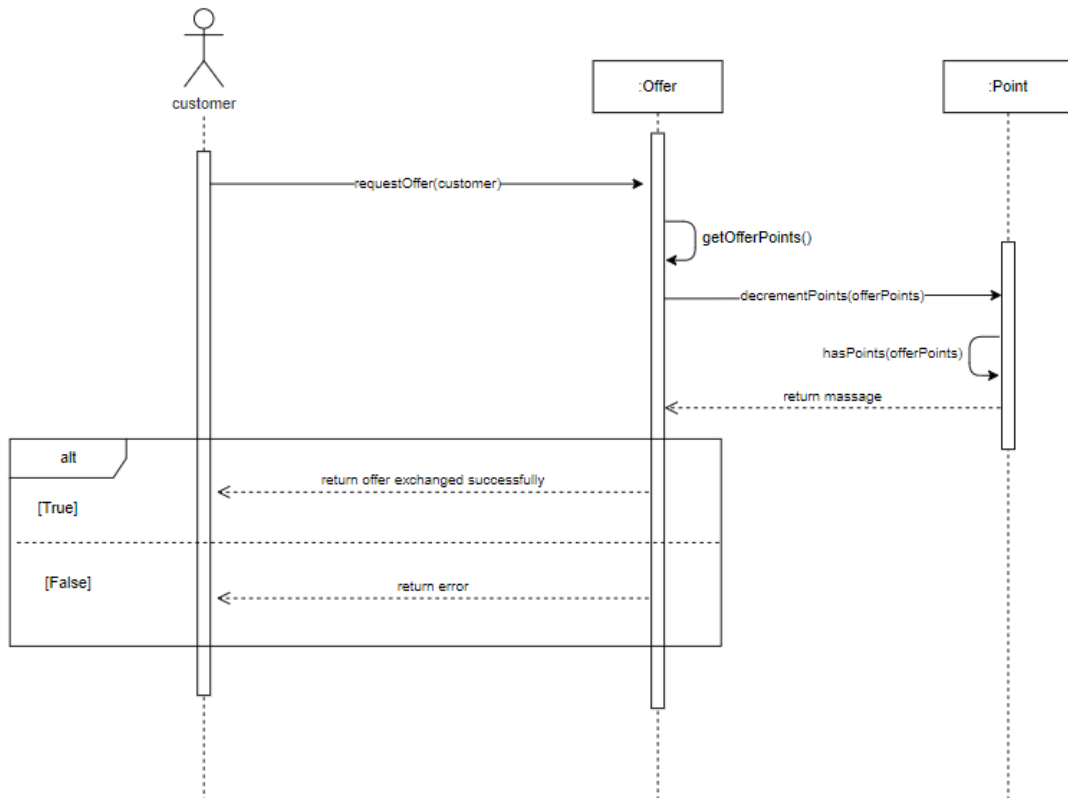




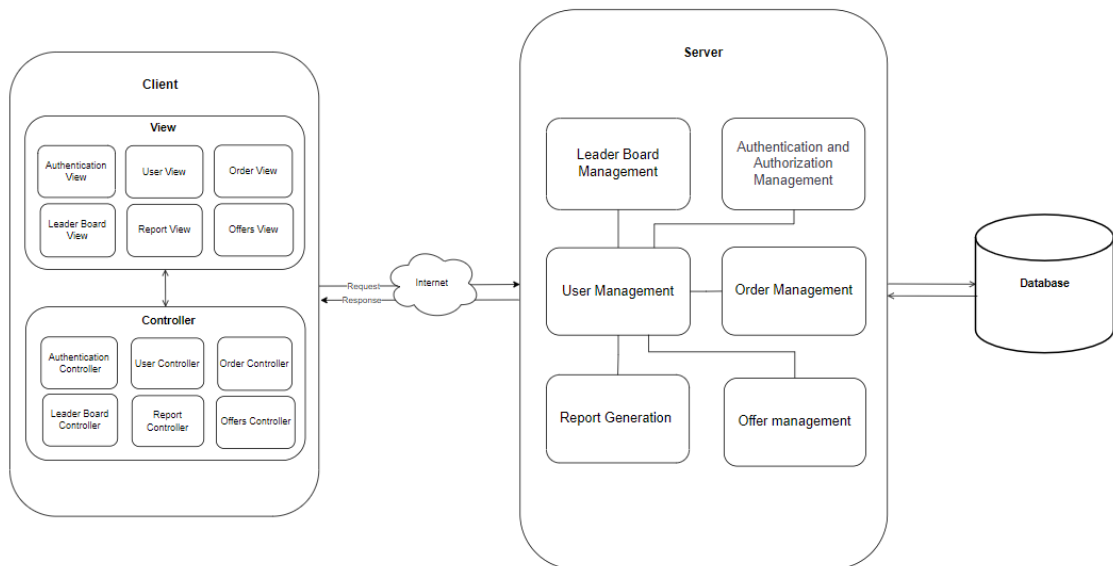
### 3.1.1 Use case description

Use Case Name: RequestOffer	
<b>Primary Actor:</b> Customer	<b>Use Case Type:</b> Essential (include checkpoints use case)
<b>Brief Description:</b> This use case allows the customer to exchange his/her points with any offer/promo according to his/her points calculated and provided by the system. Points are gathered according to the number of orders s/he made in the system.	
<b>Pre-Conditions:</b> <ol style="list-style-type: none"><li>1. The customer must be logged on to the system successfully.</li></ol>	
<b>Basic Flow of Events:</b> <ol style="list-style-type: none"><li>1. The system reads the offer points.</li><li>2. The system checks if the customer has enough points.</li><li>3. The system will decrement the customer points.</li><li>4. The system displays succussed massage.</li></ol>	
<b>Exceptional Flows:</b> <ol style="list-style-type: none"><li>2.1 The customer doesn't have enough points.<ol style="list-style-type: none"><li>2.1.1 Return failure massage.</li></ol></li><li>3.1 The system will not be able to decrement customer points.<ol style="list-style-type: none"><li>3.1.1 Return failure massage.</li></ol></li></ol>	
<b>Post-Conditions:</b> The system automatically notifies the customer with the response.	

### 3.2 Sequence Diagram

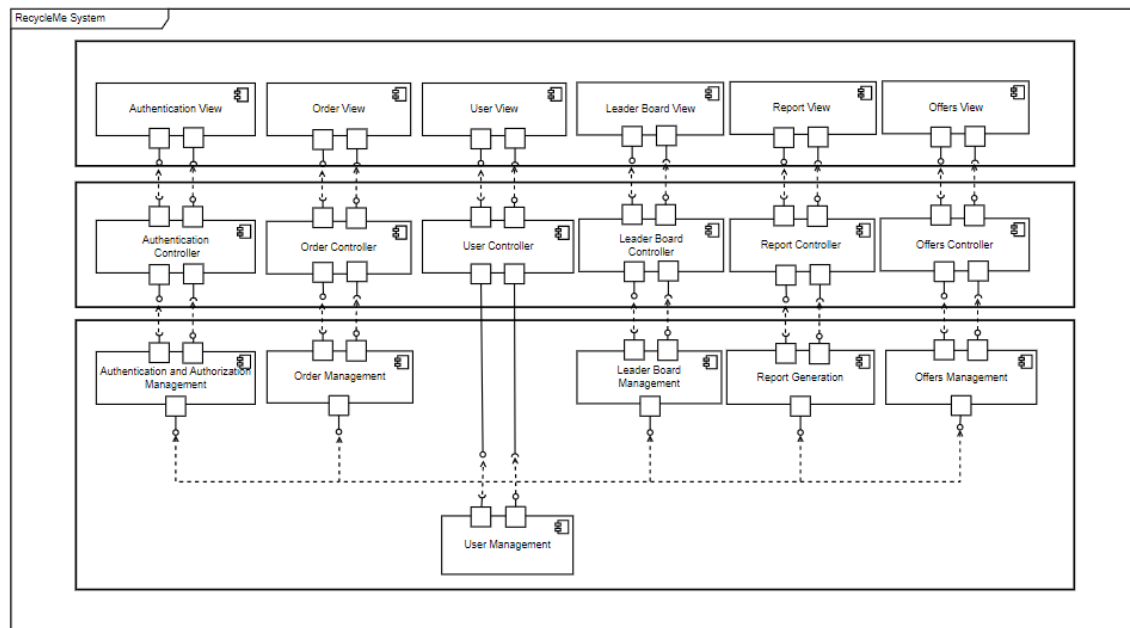


### 3.3 Overall Architecture

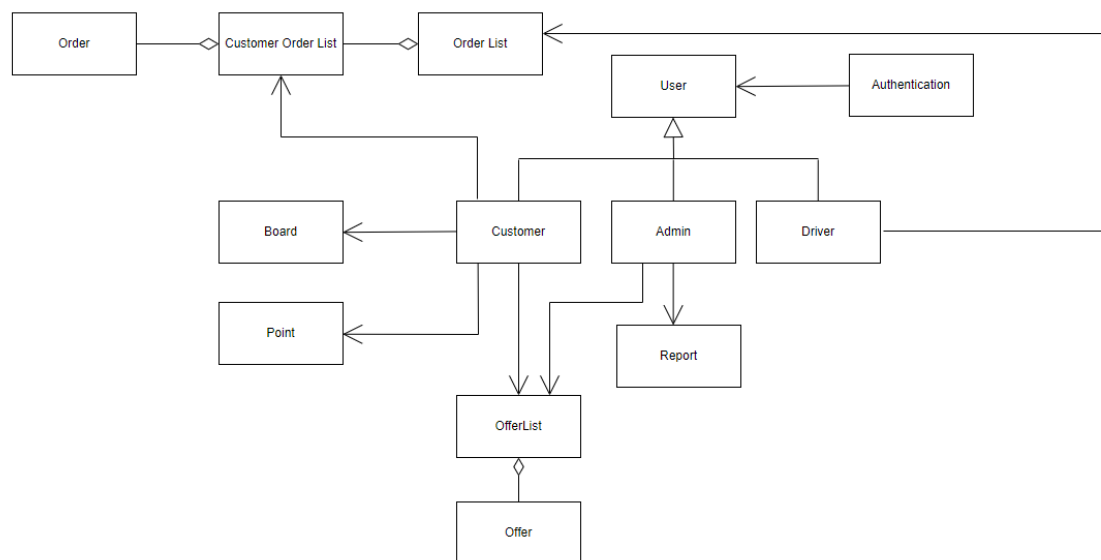




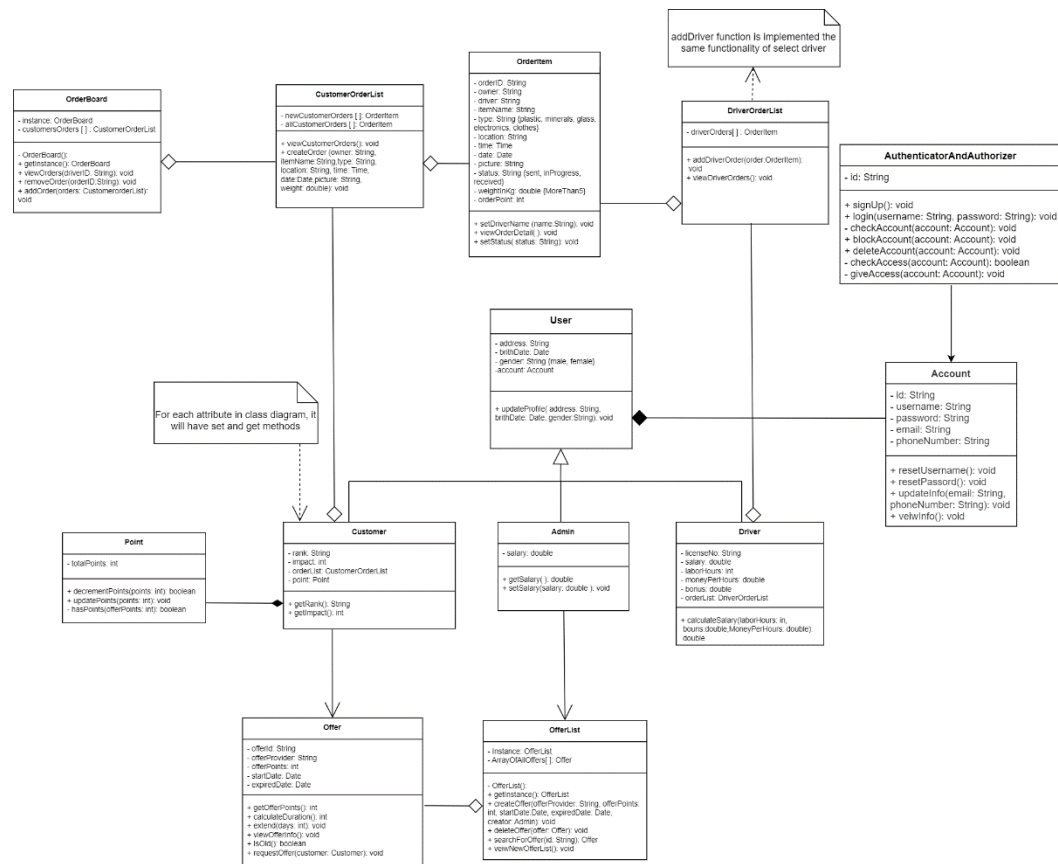
### 3.4 Component Diagram



### 3.5 Initial Set of Objects



## 3.6 Class Diagram



## **Chapter 4: Evaluating the Quality of Models**

### **4.1 Validation of Models**

- we captured an error that generates by changing in use case name without reflection this change on table description name.
- we made sure the use case name is verb and the actor name is single nouns representing a role.
- we made sure the system boundary is existing.
- we made sure the direction of arrows for generations and including are correct.
- we made sure the objects and methods are appear in a class digram with the correct return type.
- there is was heavy class that is user class and during validation we sprete it to to classes point and user.
- in the components diagram doesn't reflect the complete relationship between components as software arch.
- Change the flow in the sequence without reflection in the table description that generates confliction between a use case diagram and sequence.
- we made sure the name of the class is nouns, the visibility of the methods, data type of attribute and the logical relationships between classes.
- in the component diagram, we check the name of the component if it reflects the same name that appears in the overall architecture the same thing applied to the relationship.

## **Chapter 5: Conclusion and References**

### **5.1 Conclusion**

All of the models were based on requirements that were elicited from by the course instructor during explaining the suggested ideas, plus some requirements techniques were followed which are brainstorming with team members and looking at similar analogy apps such as RecycleSmart, Recycling Coach, and so on.

Moreover, this report provides some analysis and design models which are a use case diagram, sequence diagram, and initial set of objects in terms of the analysis stage, while component diagram and class diagram with singleton design pattern was provided in the design stage. In addition to analysis and design models, overall architecture was designed as a bridge between these two stages. All of these artifacts were implemented to design software solutions to solve real-life problems in terms of facilitating the recycling of unwanted items through the RecyclingMe mobile application to be our project in the SE323 course.

In addition, the report provides main points that are related to the objective of the project and which is achieved by following: to meet our main goal, which is encourage people to adopt the recycling attribute, one feature has been considered which is give them some discount codes, that will provide from our partners, to exchange the points that are collected from the previous recycling orders with these codes.

If given the opportunity, we will continue working on the class diagram, which due to time limitations in the first version of the system eliminated some functionality from our scope. The features include a leader board that will be displayed alongside the point's features to encourage people to recycle; additionally, more features will be available to the admin account, such as a downloaded report with statistics about the user and their orders .In terms of new ideas that we have in mind, we would like to add some notifications and calendar features to our app to enhance the user experience in recycling management.

## References

- Fakhroutdinov, K. (2013, November 27). *Online shopping*. UML graphical notation overview, examples, and reference. Retrieved December 18, 2022, from <https://www.uml-diagrams.org/examples/online-shopping-uml-component-diagram-example.html?context=cmp-ex>
- Fakhroutdinov, K. (2018, April 21). *UML component diagrams*. UML graphical notation overview, examples, and reference. Retrieved December 18, 2022, from <https://www.uml-diagrams.org/component-diagrams.html>
- Fowler, M. (2005, February 3). *Bliki: Ballandsocket*. martinowler.com. Retrieved December 18, 2022, from <https://martinfowler.com/bliki/BallAndSocket.html>
- StuartReid. (2015, July 22). *Algorithmic trading system architecture - Stuart Gordon Reid*. Turing Finance. Retrieved December 18, 2022, from <http://www.turingfinance.com/algorithmic-trading-system-architecture-post/#prettyPhoto>
- Unhelkar, B. (2005). *Verification and validation for quality of Uml 2.0 models*. Wiley-Interscience.
- YouTube. (2017). *Singleton Pattern – Design Patterns (ep 6)*. YouTube. Retrieved December 18, 2022, from [https://www.youtube.com/watch?v=hUE\\_j6q0LTQ](https://www.youtube.com/watch?v=hUE_j6q0LTQ)