
Software Requirements Specification

for

BeEcological

Prepared by:

- Jacopo Fabi, 0243938
- Danilo Dell'Orco, 0245513

Table of Contents

1. Introduction	1
1.1 Aim of the Document.....	1
1.2 System Overview	1
1.3 Operational Settings	1
1.4 Related Systems.....	2
2. User Stories.....	3
3. Functional Requirements.....	4
4. Use Cases: Overview Diagram.....	6

1. Introduction

1.1 Aim of the Document

This document describes the features, interface, system, and constraints of **BeEcological**. It also describes how to interact with application, and how to use that. This document is applicable to both the developers and stakeholders and is used to define the system.

1.2 System Overview

In industrialized countries the landfill is still the most widespread system for waste disposal, but it is a system that is harmful to public health and the environment. Pollution is an irreversible problem, and in order to limit it there are special centers for the recycling of bulky waste, called "ecological islands".

The "BeEcological" is a software which helps people to find ecological islands and to book for unload their waste. For every registered unload, the BeEcological users get points based on the type and amount of waste discharged. The points obtained are called EcoPoints, a virtual currency that can only be used within the application. EcoPoints are cumulative for each unload and can be converted into bonuses or discounts for the user.

The ecological island owners can provide their center information using the BeEcological interface for center owners. These informations will be displayed to the user to organize his unload. The center owner can also accept or refuse the booking requests for his ecological islands.

An administrator uses the BeEcological interface for administrators in order to administer the system and keep the information accurate. The administrator can, for instance, verify ecological island owners and manage user information.

1.3 Operational Settings

- Install and link Java JDK 11 to a Java IDE.
- Install and link JavaFX 11 for the standalone part.
- Move the 'libProject' folder to " C:\ " so that you don't have to change the IDE build path
- Install MySQL with the following configuration:
 - Port: 3306
 - Username: root
 - Password: root
- Execute "/db/beEcological-build.sql" to setup and initialize the database.
- Execute "/db/beEcological-populate.sql" to populate the db.
- Install Apache TomCat 9.0 to host and run the server for the web application
- Install a browser, like Google Chrome or Mozilla Firefox.

1.4 Related Systems

The BeEcological system takes inspiration from booking software like Booking.com or Trivago, but applied to the waste sector.

1.5 Definitions

Term	Description
Unloader	Person registered in the system and logged in.
Center owner	Person registered in the system and logged in, who has been verified by an administrator as the owner of an ecological island.
Administrator	Person registered by default in the system, who verify the ecological islands.
Visitor	Person that visit the system, also Unloaders and Center Owners are Visitors, but they are registered.
Personal Data	Name, Surname, Email, Phone Number
Login Credentials	Username and Password to access the system.
Center Information	Center name, City, Address, Opening and closing hours, Phone Numbers, E-Mail
Booking Information	Center name, Unloader Username, Date and Time
Unload Data	Booking Information, Quantity and type of waste unloaded, EcoPoints received.
Unload History	List of active bookings and registered unloads with the relative Unload Data.
EcoPoints	Points obtained by an unloader from a registered unload. They can be used only in BeEcological shop, to redeem prizes.

2. User Stories

1. As a visitor, I want to search for ecological islands in my city, so that I can choose where to unload the waste. **[Fabi]**
2. As a visitor, I want to know the opening and closing hours of a center, so that I know when I can unload the waste. **[Dell'Orco]**
3. *As a visitor, I want a FAQ section, so that I can easily know how to use the application.*
4. *As a visitor, I want a customer support area, so that I can easily contact an administrator for any problem I have.*
5. *As an unloader, I want to book for an unload, so that I can deposit my waste avoiding the queue.*
6. As an unloader, I want to know when my unload is registered, so that I can see how much EcoPoints* I received. **[Dell'Orco]**
7. *As an unloader, I want to know my unload history, so that I can keep track of my activity.*
8. As an unloader, I want to see a list of prizes available, so that I can decide how to spend my EcoPoints. **[Fabi]**
9. As an unloader, I want to know how much EcoPoints I have, so that I can choose the prizes I want to redeem. **[Dell'Orco]**
10. *As a center owner, I want to provide my center information, so that people are always updated about it.*
11. As a center owner, I want a list of all the booking request for my center, so that I can confirm or decline them. **[Fabi]**
12. As a center owner, I want to register a successful unload, so that the unloader can receive his EcoPoints.
13. *As a center owner, I want to add a photo of my ecological island, so that the visitors can be more interested in my center.*
14. *As an administrator, I want to verify ecological island owners, so that I can keep the correct information about the centers.*
15. *As an administrator, I want to insert a list of prizes available, so that the unloaders can see and redeem them.*

*User stories formatted as *user-stories-text* are those used for software design, but that are not considered for evaluation.

3. Functional Requirements

- **Homepage**

- The system shall display a homepage to welcome visitors. The homepage shall provide the possibility to search for ecological islands, explain how the system works, give the possibility to login, and give a help area.
- The system shall display a homepage for center owners. The center owner homepage shall display center information and shall provide the possibility to change or add the center photo.

- **Login-Registration**

- The system shall provide the possibility to visitor to register. Registration request can be as an unloader or as a center owner and requires both personal data* and login credentials*. Registration for a center owner requires also the center information.
The information entered during the registration must be saved in a database.
- The system shall provide the possibility to unloaders, center owners and administrators to login in the system, entering their login credentials.
- The system shall provide to unloader and center owner the possibility to restore the forgotten password sending an email with a new random password.
- The system shall notify the unloader of successful registration by sending an email with the following message: "Hi, <username>, welcome to BeEcological. Your registration has been successful".
- The system shall notify the center owner of successful registration by sending an email with the following message: "Hi, <username>, welcome to BeEcological. Your center, <center_name> has been verified, the registration is completed".

- **Booking**

- The system shall provide the possibility to the visitor to search ecological island entering his name, city or address. **[Fabi]**
- The system shall display to the visitor the center information* of the selected ecological island.
- The Google Maps API shall provide a map to see the position of the selected ecological island.
- The system shall provide to the unloader the possibility to make a booking request for an unload to the selected center, in a specific day and time. **[Fabi]**
- The system shall display to the center owner a list of the incoming booking request for his ecological island.
- The system shall provide to the center owner the possibility to manually add a new booking for his center.
- The system shall display to the unloader a list of all the booking requests, organizing them according to their status.
- The system shall display to the unloader a list of all the registered unloads, with the relative amount of EcoPoints earned. **[Dell'Orco]**

- The system shall provide to the center owner the possibility to confirm a booking request for his center, registering the booking information*. **[Dell'Orco]**
- *The system shall notify the center owner when an unloader makes a booking request, sending an email with the following message: "Hi, <center_owner>, the user <username> asks for an unload at <center_name> at <date-hour>".*
- *The system shall notify the unloader when the center owner confirmed or declined the booking request, sending an email with the following message: "Hi, <username>, your booking for the unload at <center_name> at <date-hour> is <declined/confirmed>".*
- **Store**
 - The system shall display to the unloader a list of prizes available where each item is associated with its cost in EcoPoints. **[Dell'Orco]**
 - *The system shall provide to the unloader the possibility to redeem the selected prize, sending an email containing the relative coupon and the following message: "Hi, <username>, your prize has been redeemed correctly. Attached you will find the coupon!".*
- **Unload**
 - The system shall provide to the center owner the possibility to register the unload data* of a completed unload, entering quantity and types of waste unloaded. **[Fabi]**
 - *The system shall notify the unloader when the owner registered the unload, sending an email with the following message: "Hi, <username>, your unload at <center_name> at <date-hour> has been registered. You received <amount_of_EcoPoints> EcoPoints".*
- **Account Information**
 - *The system shall provide to the unloader and center owner the possibility to see his personal data and login credentials (password must be shown with asterisks).*
 - *The system shall provide to the unloader and center owner the possibility to change the current login credentials.*
 - *The system shall provide to the unloader the possibility to see his current amount of EcoPoints.*
 - *The system shall provide to the unloader the possibility to see his unload history*.*
 - *The system shall provide to the center owner the possibility to add a photo for his center, which will be displayed to visitors.*
- **Help & FAQ's**
 - *The system shall provide to the visitor the possibility to read the FAQ's customer support.*
 - *The system shall provide to the visitor the possibility to request online help, sending an email to a random administrator.*
- **Administration**
 - *The system shall display to administrators all the registration request. The administrator can confirm or decline the registration of an unloader/center owner, verifying that the information entered are real.*

- The system shall provide to administrators the possibility to configure for each type of waste the amount of EcoPoints received, entering EcoPoints for a kilo of waste.
- The system shall provide to administrators the possibility to insert prizes redeemable by unloaders, with the relative cost of EcoPoints.

*Functional requirements formatted as *functional-req-text* are those used for software design, but that are not considered for evaluation.

4. Use Cases: Overview Diagram

