Free Second-Hand Clothing Distribution System

Requirements Determination and Use Case Analysis / System Proposal / Analysis Phase   
(Homework No.2)

Project team: Team 02

Instructor: Dr. Araz Yusubov

Submitted in partial fulfillment of the requirements of the INFT 2303: Systems Analysis and Design course project

|  |  |
| --- | --- |
| GitHub repository | <https://github.com/ADA-SITE-INFT2303-2023-Spring/sys-dev-project-team-02> |
| Version date | Version information |
| 26/03/2023 | Initial draft |
| 01/04/2023 | Final Version |

|  |  |
| --- | --- |
| Other documents in the package | |
| Use Case diagram | Use case diagram is submitted as a drawio extension |
| Homework 1b | Edited version of the System Reqeust file |

# Table of contribution

|  |  |  |
| --- | --- | --- |
| Team member | Contribution to this homework (NOT the project) | Estimated % |
| Farid Mammadli | User Story Description + Use case Diagram, Use Case Descriptions, Bonus 2 | 10% + 20% + 10%  total: 40% |
| Nargiz Heybatova | Use Case Analysis, User Story Description, Use Case Descriptions, Bonus 2, Functional requirements, Nonfunctional Requirements | 4% + 5% + 13% + 5% + 5% + 4%  total: 36% |
| Atlas Hamzali | Introduction, Definitions, Functional Requirements, Nonfunctional Requirements, Bonus 1 | 4% + 4% + 5%+ 3% + 15%  total: 31% |
| Azer Shukurlu | Design constraints, External Actor Description, References | 5% + 10% + 5%  total: 20% |

# Table of Contents

[Table of contribution 2](#_Toc131283855)

[Table of Contents 3](#_Toc131283856)

[1. Introduction 3](#_Toc131283857)

[Definitions 3](#_Toc131283858)

[Requirements Definition 4](#_Toc131283859)

[Functional Requirements 5](#_Toc131283860)

[Nonfunctional Requirements 6](#_Toc131283861)

[Design Constraints 7](#_Toc131283862)

[2. Use Case Analysis 7](#_Toc131283863)

[External Actor Descriptions 9](#_Toc131283864)

[User Story Descriptions 9](#_Toc131283865)

[Use case Diagram 10](#_Toc131283866)

[Use Case Descriptions 12](#_Toc131283867)

[Use Case 1 12](#_Toc131283868)

[Use Case 2 12](#_Toc131283869)

[Use Case 3 13](#_Toc131283870)

[Use Case 4 14](#_Toc131283871)

[Use Case 5 15](#_Toc131283872)

[Use Case 6 16](#_Toc131283873)

[Use Case 7 16](#_Toc131283874)

[Use Case 8 17](#_Toc131283875)

[Use Case 9 18](#_Toc131283876)

[Use Case 10 19](#_Toc131283877)

[Use Case 11 20](#_Toc131283878)

[Use Case 12 21](#_Toc131283879)

[Use Case 13 21](#_Toc131283880)

[Use Case 14 22](#_Toc131283881)

[Use Case 15 22](#_Toc131283882)

[Use Case 16 23](#_Toc131283883)

[References 24](#_Toc131283884)

# Introduction

This is part of the System Proposal for a hypothetical project ***Free Second-Hand Clothing Distribution System*** submitted for partial fulfillment of the requirements of the Systems Analysis and Design course in the School of Information Technologies and Engineering at ADA University, Baku, Azerbaijan.

With the help of this system, people will be able to make donations, gather, and give used clothes to people who need them. By promoting the use of second-hand clothing, the system aims to reduce the amount of waste produced and advance sustainable fashion.

The Free Second-Hand Clothing Distribution System enables both suppliers and receivers to browse and receive the clothing they want. Suppliers may quickly share details about the pieces of clothing they intend to contribute. The platform will also have a contact system to help exchanges among suppliers and receivers, as well as databases to keep records of the supply of given clothes.

The criteria for the Free Second-Hand Clothing Distribution System will be explained in this document, together with any functional and non-functional needs and constraints, that were established throughout the development phase. Moreover, it will offer a comprehensive use case analysis that will be used to guide the design and implementation of the system. What this system will need to perform, in general terms and according to the functional needs of the system. What tasks consumers do wish to complete while utilizing this system. To guarantee that the platform satisfies the demands of its own users and admins, the goal of this document is to offer a straightforward and comprehensive overview of the system's functions and characteristics.

**Bonus 1:** Get extra 15% points for revising and SUBSTANTIALLY improving the Homework 1B document.

(Homework is pushed to the GitHub and also submitted to the blackboard)

## Definitions

|  |  |
| --- | --- |
| Term | Definition |
| GDPR  CCPA  FAQ  API | The General Data Protection Regulation (GDPR) is a legal framework that sets guidelines for the collection and processing of personal information from individuals who live and outside of the European Union (EU).  The California Consumer Privacy Act (CCPA) is a state-wide data privacy law that regulates how businesses all over the world are allowed to handle the personal information (PI) of California residents.  A list of questions and answers relating to a particular subject, especially one giving basic information for users of a website.  An application programming interface (API) is code that enables two software programs to communicate. An API defines how a developer should request services from an operating system (OS) or other application, and expose data within different contexts and across multiple channels. |

# Requirements Definition

* The system shall… A required, must have feature
* The system shall have permission for users to create profiles and register properly.
* The system shall have permission for users to search the selection of used clothing.
* The system shall have permission for users to browse particular categories of clothing.
* The system shall monitor the amount of inventory and provide authentic supply updates.
* The system shall provide admins with the control over user profiles and supplies.
* The system shall have permission for users to add clothes to the basket securely.
* The system should… A desired feature, but may be deferred till later
* The system should allow users to provide their suggestions for improving the service that can be a new feature or functionality that the user thinks would be useful, or a suggestion for making existing features more efficient or user-friendly
* The system should allow users to provide their negative comment or criticism about the system that can be related to a problem the user encountered while using the system, or a general dissatisfaction with the system's performance or feature.
* The system should allow users to provide their feedback on the donation process or experience. It could include feedback on the payment process, the donation form or landing page, or the verall experience with the system.
* The system should track the user activity and generate reports will be used to diagnose and fix problems, monitor system health, or provide insights for system.
* The system should have an intuitive user interface which is simple to access and operate.
* The system may… An optional, nice-to-have feature that may never make it to implementation.
* The system may report on the user behavior.
* The system may include a discussion panel for the users to communicate with each other.
* The system may have a ranking system based on the amount of donations made by the users.
* The system may offer users the option to evaluate the customer services provided by the system.

## Functional Requirements

**Process-oriented functional requirements:**

* The system shall have permission for users to create profiles and register properly.
* The system shall have permission for users to sign in and out of their accounts.
* The system shall have permission for users to change their usernames and password on the system if they lose them.
* The system shall have permission for users to modify their private information.
* The system shall have permission for users to browse clothes by their names, sizes, and accessibility.
* The system shall have permission to remove their orders if users decide otherwise.
* The system shall have permission for users to rank and comment on clothes.
* The system shall have permission for admins to adjust the system's inventory and accessibility details
* The system shall have permission for admins to monitor all transactions through the system.
* The system shall have permission for admins to control user reviews

**Information-oriented functional requirements:**

* The system shall contain the details about the donations
* The system shall contain the accounts information
* The system shall contain feedbacks on the donations
* The system shall contain the “Raise an Issue” Functionality reports (suggestion, complaints, reports)

## Nonfunctional Requirements

**Operational Requirements**

* The system should be accessible twenty-four hours a day, seven days a week, with minimal interruption for repairs and upgrades.
* All admins and users should find the system simple to use.
* To handle increasing traffic levels and customers, the system should be adaptable.

**Performance Requirements:**

* The system should be capable of managing a massive amount of transactions.
* The system should reply to user queries efficiently and with low delay.
* The system should be able to manage enormous amounts of data.
* Posting and Deleting must be performed fast and correctly by the system.

**Security Requirements:**

* The system should have effective identification and authorization processes to preserve user information from illegal access.
* The system should apply encryption to preserve confidential material.
* The system should have a backup in place to guarantee that information is not destroyed in the case of any incident.
* The system should adhere to applicable security regulations, such as the GDPR or the CCPA.

**Usability Requirements:**

* The system should have a colorful and user-friendly interface.
* The system should give users detailed guidance and information.

**Cultural requirements:**

* The system should operate in numerous languages and countries.
* The system should consider cultural variety while also adapting to the choices and demands of various populations.
* The system should be open and available to all users, no matter what their ethnicity or religious views are.
* There should be no cultural prejudices promoted by the system.

**Political Requirements**

* The system should adhere to the rules and standards that guide second-hand clothing distribution, including those affecting safety and wellness.
* The system should adhere to the rules and ensure that delivery employees are employed with equality and integrity.
* The system should support recycling and trash reduction in order to improve ecological sustainability.
* The system should not support any political motive or discrimination.

## Design Constraints

**Technical Constraints**

1.Project’s app should be available in web application

2.The system should have limitations from external services and API requirements.

3.The user should have high bandwidth for internet connectivity to show better performance.

**Legal Constraints.**

1.All files (like image, video) that belong to clothes should be protected by copyright by Intellectual Property Law.

**Organizational Constraints**

1.Deadlines should be set not late by each stakeholder that can impact designers research, prototype and test design ideas.

2.Designers often have to balance user needs with organizational goals, which can limit design choices of clothes.

**Talent Constraints**

Every designer’s skill and expertise should be known to complement one another. It enables managers to provide the right people with clothes.

# Use Case Analysis

Register as an Admin

**Description:** Admins team members are added to the system by the other admins. All the necessary information about personalized admin accounts is displayed on the system’s database.

Register as a Customer Support

**Description**: Customer Support accounts are registered to the system to assist users. Customer Support registrations are assigned by the admins team.

Register as a User

**Description**: User accounts are registered to the system to either use the facilities and get needed help.

Donate Clothes

**Description**: Donors can donate their second-hand clothes through the system by adding all the information needed (size, color, condition, picture).

Remove Donation

**Description**: Users and Admins have the ability to request the removal of a donation if they are not satisfied with it for any reason.

Search Clothes

**Description**: Users can search for the needed clothes and check the related information (size, color, condition, location).

View Inventory

**Description**: All the ranks (user, Customer Support, admin) can view the inventory levels of specific categories of clothes.

Request Existing Donation

**Description**: Users can request iteams available on the system. If needed, a Customer Support team is asked for assistance with the requesting procedure.

Request Special Donation

**Description**: Users request specific clothing items they need that are not present in the system. As soon as there is any clothing that matches the criteria provided by the user, the system sends notification to the user’s account providing container box location, clothing details, and donor account information.

Raise an Issue

**Description**: Users can provide all the complaints, suggestions, and reports related to the donation and the distribution experience (reports mainly cover the condition of container boxes, for ex: damaged, not working, etc) with the assistance of Customer Support.

Provide Donation Feedback

**Description:** Users provide their feedback on the quality of the provided clothes (ex: condition is same/not same as described by the donator).

Assist User

**Description:** Customer support answers users’ questions and assists in the process to help users to overcome any issues they may be experiencing and continue using the service ensuring its long-term success.

Analyze Feedback

**Description:** Customer support analyzes the feedback the user provided to understand the issue or suggestion being raised, identify the root cause of the problem, and provide any necessary context or additional information to the admin team.

Forward Feedback

**Description:** Customer support forwards the feedback to the admin team so that the issue can be addressed and resolved by the admin team in a proper manner.

Work on Feedback

**Description:** Admin team gets the report and works on it for the improvements to ensure the maintenance of a high level of customer satisfaction and loyalty.

Manage User Account

**Description:** Admin team can delete or edit information on user profiles. There is a possibility that a user account is also deleted completely if suspended for a behavior against the rules.

## External Actor Descriptions

Human actors:

**System admins:** Individuals who manage the system and its operations.

**Donee:** Individuals who receive free second-hand clothing from the system

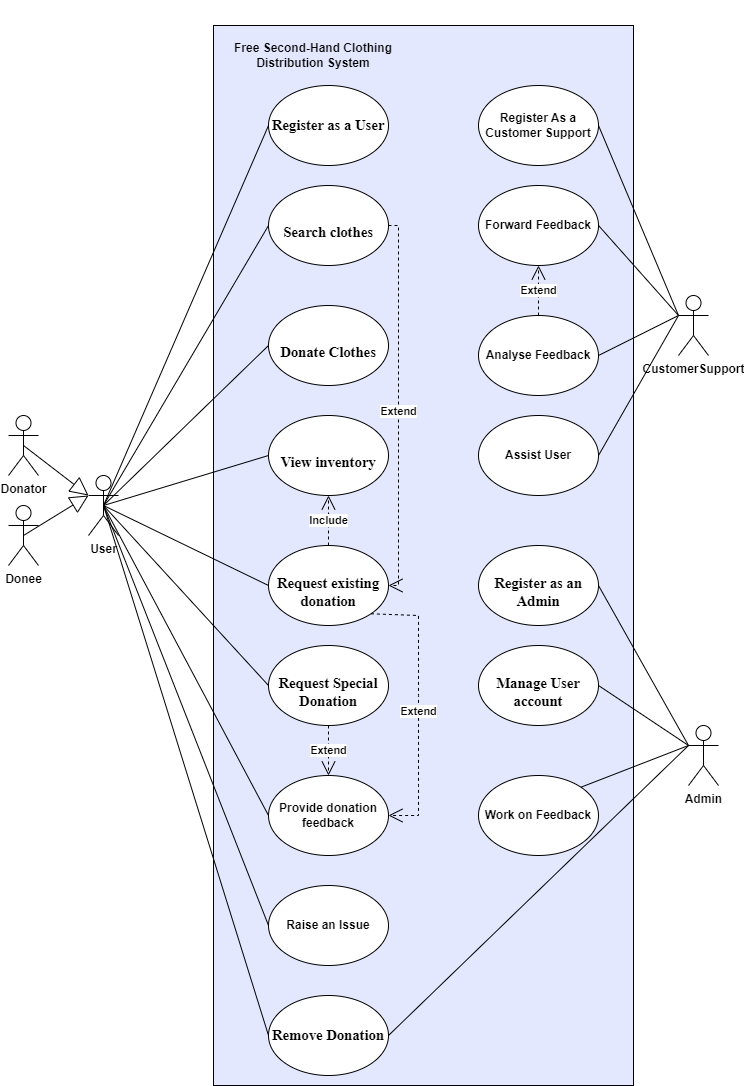
**Donor:** Individuals who donate second-hand clothing to the system

**Customer Support:** Individuals who help manage the work of the system by sorting, organizing, and distributing clothes.

## User Story Descriptions

|  |  |  |
| --- | --- | --- |
| User story name | Description | Release |
| Register Admins | Each of the admins team members will be added to the system by the other admins. All the necessary information about personalized admin accounts will be displayed on the system’s database | R1 |
| Register Customer Support | Customer support accounts will be assigned to the system to answer questions, analyze feedback and forward to the admin. | R1 |
| Register User | User accounts are assigned to the system to import/export donations. Users are also capable of reporting any issue or providing any suggestions that users may want to see on the system. | R1 |
| Donate Clothes | Users donate their clothes by adding all the information needed (size, color, condition, picture). They add their clothes physically to the boxes, and virtually to the system. Admins supervise if the donation is valid and follows the rules determined by the admins. | R1 |
| Remove Donation | Users and Admins have the ability to request the removal of a donation if they are not satisfied with it for any reason. | R1 |
| Search Clothes | Users use the search panel to search for the needed clothes providing the related information (size, color, condition, location). If there is a problem with the search functionality, Customer Support will be informed for reporting to the admin team (if needed). | R1 |
| View Inventory | All the ranks (user, Customer Support, admin) have access to display the inventory of a specific clothes container. By that functionality, all the information related to the clothes are provided including the location information of the container. | R2 |
| Request Existing Donation | Users request for the clothes that are available on the system. If needed, a Customer Support team is asked for assistance with the requesting procedure. | R1 |
| Request Special Donation | Users provide some information about clothes they need from the system, but are not present yet. As soon as there is any clothing that matches the criteria provided by the user, the system sends notification to the user’s account providing container box location, clothing details, and donor account information. Customer Support team assistance is ready for the process if needed. | R2 |
| Provide Donation Feedback | After a request made by users, users are free to provide feedback for the clothes (ex: condition is same/not same as described by the donator). If there is a need for assistance, Customer Support is requested to help in the process. These feedbacks are added to the feedback board of the personal accounts who donated the particular clothes. | R3 |
| Raise an Issue | Users provide all the complaints, suggestions, and reports (reports mainly cover the condition of container boxes, for ex: damaged, not working, etc) with the help of Customer Support. Customer Support get the text message from the users (may contain a picture description of a problem / or visualization of the suggested functionality) and check its validity before forwarding to the admins. | R2 |
| Assist User | Customer support answers users’ questions and assists in the process to help users to overcome any issues they may be experiencing and continue using the service ensuring its long-term success. | R2 |
| Analyze Feedback | Customer support analyzes the feedback the user provided to understand the issue or suggestion being raised, identify the root cause of the problem, and provide any necessary context or additional information to the admin team. | R2 |
| Forward Feedback | Customer support forwards the feedback to the admin team so that the issue can be addressed and resolved by the admin team in a proper manner. | R2 |
| Work on Feedback | Admin team gets the report and works on it for the improvements to ensure the maintenance of a high level of customer satisfaction and loyalty. | R2 |
| Manage User Account | Admin team can delete or edit information on user profiles. There is a possibility that a user account is also deleted completely if suspended for a behavior against the rules. | R1 |

## Use case Diagram



## Use Case Descriptions

### Use Case 1

|  |  |
| --- | --- |
| **Use Case Number:** | UC-01 |
| **Use Case Name:** | Register as an Admin |
| **Actor(s):** | Admin |
| **Description:** | Each of the admin team members will be added to the system by the other admins. All the necessary information about personalized admin accounts will be displayed on the system’s database. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - Actor registering as an Admin. |
| **Pre-condition(s):** | * Admins announced Admin recruitment. * System has been set up and configured. * System is running and is open for admin registration requests. * Registration panel has been accessed on the website via URL |
| **Main (Success) Flow:** | 1.Actor requests for being admin on the system  2.System requests information about the actor such as name, surname, date of birth, address, phone number, email address.  3.Actor provides the information the system requested  4.Verification is done by the system to check if the corresponding information is provided   * If information is invalid System displays a message. Return to Step 2   5.Admin account registration details are sent to the mail address.   * If the information does not satisfy the needs to be in an admin team, then return to the step 1   6.Approval mail is sent to the actor  7.System requests the login information  8.Actor provides the information required for logging in  9.System approves the information provided   * If the information provided does not match to any account, then return step 8   10.System displays confirmation of registration |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the registration process. The following steps would occur:  1. Actor selects option to cancel during registration  2. System requests confirmation to cancel  3. Actor confirms  4. System returns to main screen |
| **Post Condition:** | Alternate Flow #1: Actor did not complete registration. System does not store the Actor’s information.  Main (Success) Flow 11: The system authenticates the admin's account details and grants access to the admin features, tools and settings (system configurations, managing accounts, and permissions) . |
| **Requirements:** | 1 – The Actor shall register using his personal information  2 – The Actor shall be able to cancel registration during the process if needed  3 – The system shall verify all information provided by the for registration as an admin.. |

### Use Case 2

|  |  |
| --- | --- |
| **Use Case Number:** | UC-02 |
| **Use Case Name:** | Register as a Customer Support |
| **Actor(s):** | Customer Support |
| **Description:** | Customer support accounts will be assigned to the system to answer questions, analyze feedback and forward to the admin. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - Actor registering as a Customer Support. |
| **Pre-condition(s):** | - System has been set up and configured.   * System is running and is open for Customer Support registration requests. * Actor has accessed website via URL |
| **Main (Success) Flow:** | 1. The Actor selects the option to register as a “Customer Support”.  2. The system requests personal information i.e. name, surname, email, phone number.  3. The Actor provides the required information.  4. The system verifies the required information is provided.   * If not provided, then return to the step 2   5. Customer Support account registration details are sent to the mail address.   * If the information does not satisfy the needs to be in an Customer Support team, then return to the step 1   6.System requests the login information  7.Actor provides the information required for logging in  8.System approves the information provided   * If the information provided does not match to any Customer Support account, then return step 8   9. System displays confirmation of registration. |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario, the actor can cancel the registration process.  1 The actor selects the option to cancel during registration.  2. The system requests confirmation to cancel.  3. The actor confirms.  4. The system returns to the main screen. |
| **Post Condition:** | Alternate Flow #1: Actor did not complete registration. System does not store the Actor’s information.  The actor's information is stored in the system, and the Customer Support dashboard can be accessed.  Main (Success) Flow 11: The system authenticates the actor’s account details and grants access to the Customer Support features, tools and settings (answer questions, analyze feedback and forward to the admin.) . |
| **Requirements:** | 1. The actor shall provide personal information to the system.  2 – The actor shall be able to cancel registration during the process if needed  3 – The system shall verify all information provided by the user for registration as a Customer Support. |

### Use Case 3

|  |  |
| --- | --- |
| **Use Case Number:** | UC-03 |
| **Use Case Name:** | Register as a User |
| **Actor(s):** | User |
| **Description:** | User accounts are assigned to the system to import/export donations and provide feedback to them. Users are also capable of reporting any issue or providing any suggestions that users may want to see on the system. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - Actor registering as a User. |
| **Pre-condition(s):** | * System has been set up and configured. * System is running and is open for user registration requests. * Actor has accessed website via URL |
| **Main (Success) Flow:** | 1. The Actor selects the option to register as a user.  2. The system requests personal information i.e. name, surname, email, phone number.  3. The actor provides the required information.  4. The system verifies the required information is provided.   * If not provided, then return to the step 2   5. User account registration details are sent to the mail address.  6.System requests the login information  7.Actor provides the information required for logging in  8.System approves the information provided   * If the information provided does not match to any moderative account, then return step 6   9.System displays confirmation of registration. |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the registration process. The following steps would occur:  1. Actor selects option to cancel during registration  2. System requests confirmation to cancel  3. Actor confirms  4. System returns to main screen  Alternate Flow #2: If the system match the account details with the existing account on the system during the success scenario 4, then these steps occur:  1.System notifies the user that there is an account exist using the details provided for the registration  2.System asks if the user wants to log in to his account  3.If the log in credentials are correct, then the system directs to the account.   * if the credentials are not correct, then go to step 5 in the success scenario. |
| **Post Condition:** | Alternate Flow #1: Actor did not complete registration. System does not store the Actor’s information.  Main (Success) Flow 11: The system authenticates the user account details. User accounts are created on the system. |
| **Requirements:** | 1 – The actor shall register using his personal information  2 – The actor shall be able to cancel registration during the process if needed  3 – The system shall verify all information provided by the user for registration as an user.. |

### Use Case 4

|  |  |
| --- | --- |
| **Use Case Number:** | UC-04 |
| **Use Case Name:** | Donate Clothes |
| **Actor(s):** | User |
| **Description:** | Users donate their clothes by adding all the information needed (size, color, condition, picture). They add their clothes physically to the boxes, and virtually to the system. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - User donates clothes. |
| **Pre-condition(s):** | * System has been set up and configured. * System is running and open. * User has accessed website using URL * User authorized himself and signed into the account. |
| **Main (Success) Flow:** | 1.User selects the option “donate clothes”  2.System requests information about the clothes such as size, color, condition.  3.User provides the information the system requested  4.Verification is done by the system to check if the corresponding information is provided   * If information is invalid System displays a message. Return to Step 2   5.System requests pictures for each clothing in a specified file format (jpeg)  6.System verifies the picture   * if the file format is not matched, then return to the step 5   7.Donation acceptance details are sent to the mail address  8.System displays the donation. |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the donation process. The following steps would occur:  1.User selects option to cancel during donation  2.System requests confirmation to cancel  3.User confirms  4.System returns to main screen |
| **Post Condition:** | Alternate Flow #1: User did not complete the donation process. System does not store the Users’ donation.  Main (Success) Flow 8: The system stores the information about the donation recently added. The donation is displayed and accessible by the other users on the system. |
| **Requirements:** | 1 – The User shall donate using donation information details  2 – The User shall be able to cancel donation during the process if needed  3 – The system shall verify that all information provided for the donation is approved |

### 

### Use Case 5

|  |  |
| --- | --- |
| **Use Case Number:** | UC-05 |
| **Use Case Name:** | Remove Donation |
| **Actor(s):** | User, Admin |
| **Description:** | Users and Admins will be able to remove the donations from the system if they are not satisfied with it. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - User and Admins can remove the donation from the system donates clothes. |
| **Pre-condition(s):** | * System has been set up and configured. * System is running and open. * Actors have accessed website using URL * The Actor authorized himself and signed into the account. * Donation should exists on the system |
| **Main (Success) Flow:** | 1.Actor selects the option “remove the donation”  2.System asks for the confirmation  3.Actor confirms the process  4.System removes the donation from the system |
| **Alternate Flows:** | Alternate Flow #1: After Step 3 in the success scenario, if the donation is not posted by the user, the following steps would occur:  1.User selects option to confirm to remove the donation  2.the system warns the user that the user does not have an authorisation to remove  3.System forwards the user to the main screen |
| **Post Condition:** | Alternate Flow #1: User was not able to remove the donation. System keeps the donation on the system.  Main (Success) Flow 4: The actor deletes the donation from the system. The system does not display the donation on the page anymore. |
| **Requirements:** | 1 – The actor shall be able to remove donation  2 – The unauthorized users shall not be able to remove others’ donations. |

### 

### Use Case 6

|  |  |
| --- | --- |
| **Use Case Number:** | UC-06 |
| **Use Case Name:** | Search Clothes |
| **Actor(s):** | User |
| **Description:** | Users use the search panel to search for the needed clothes that also provides the related information (size, color, condition, location). |
| **Priority (Release)** | R1 |
| **Trigger:** | External- Users access the clothes search panel and search based on some criterias. |
| **Pre-condition(s):** | * System has been set up and configured. * System is running and open. * User has accessed website using URL * User authorized himself and signed into the account. |
| **Main (Success) Flow:** | 1. User accesses the clothes search panel.  2. User enters search criteria (size, color, condition, location).  3. System verifies the criteria provided by the user   * if the details are not satisfied, return to the step 2   4. System returns search results based on specified criteria.  5. The results are successfully displayed on the page. |
| **Alternate Flows:** | Alternate Flow #1: No search results found.  1. System displays message to User: "No search results found. Please try again with different search criteria."  2. the system returns to step 2 in the success scenario. |
| **Post Condition:** | Main flow #4: User displayed all the desired results based on the search criteria |
| **Requirements:** | 1. The user shall provide required criteria on the search panel. 2. The system shall display all the donations based on the searching filters |

### 

### 

### Use Case 7

|  |  |
| --- | --- |
| **Use Case Number:** | UC-07 |
| **Use Case Name:** | View inventory |
| **Actor(s):** | User |
| **Description:** | Users have access to display the inventory of a specific clothes container. By that functionality, all the information related to the clothes are provided including the location information of the container. |
| **Priority (Release)** | R2 |
| **Trigger:** | External - User displays the content of the container boxes |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - User has accessed website using URL  - Users can view the inventory levels of specific categories of clothes.  - User authorized himself and signed into the account. |
| **Main (Success) Flow:** | 1. User selects the option to “view inventory”  2. System displays the available containers on the map  3. User selects the container to display  4. System displays the inventory consisting of donated clothes in it.  5. User reviews all the details about the corresponding donations. |
| **Alternate Flows:** | Alternate Flow #1: The selected container is empty. After step 3 in the success scenario, the following steps would occur:  1.System notifies the user about the situation  2.System suggest to look for the other containers available  Alternate Flow #2: Specific area for the available containers, after step 2. The following steps would occur:  1. User search for the specific area for the containers   * if there is no location matching the area provided by the user, display the message: “area is not found, search for the other area” and return to step 2 in the main success scenario.   2.System displays all the available containers related to the area provided by the user.  3.Goes to the success scenario 3. |
| **Post Condition:** | Main (Success) Flow 4: The inventory of the specified donation container is displayed to the user.  Alternate Flow #2: Specific area containers are listed by the system |
| **Requirements:** | 1 System should list all the available containers  2.User shall provide a location area for the containers  3. System should display the empty containers  4. Users shall be able to display all the container inventories. |

### Use Case 8

|  |  |
| --- | --- |
| **Use Case Number:** | UC-08 |
| **Use Case Name:** | Request existing donation |
| **Actor(s):** | User |
| **Description:** | Users can request existing donations displayed on the system based on their needs. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - User request for clothes |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - User has accessed website using URL  - User authorized himself and signed into the account.  - User searches for the related product based on the specific criteria  - User displays the results |
| **Main (Success) Flow:** | 1.User selects for the desired donation  2.System displays the details about the donation  3.User selects “request” option  4. The system asks for the confirmation for the request  5. Request confirmation details are sent via the mail address |
| **Alternate Flows:** | Alternate Flow #1: After Step 3 in the success scenario System will display the option to Cancel the request. The following steps would occur:  1. User selects option to cancel during the request  2. System requests confirmation to cancel  3. User confirms  4. System returns to main screen |
| **Post Condition:** | Alternate Flow #1: User did not complete the request. System does not store the Request activity.  Main (Success) Flow 5: User successfully requests the donation.  Alternate Flow #2: After success scenario 5, the donation is archived. The system does not display the donation on the main page and search results. |
| **Requirements:** | 1 – The user shall be able to cancel request during the process if needed  2 – The User shall be able to request for the desired donation.  3 – The system shall verify if the requested donation is archived on the system. |

### Use Case 9

|  |  |
| --- | --- |
| **Use Case Number:** | UC-09 |
| **Use Case Name:** | Request Special Donation |
| **Actor(s):** | User |
| **Description:** | Users provide some information about clothes they require from the system, and as soon as there is any clothing that matches the criteria provided by the user, the system sends notification to the user’s account providing container box location, clothing details, and donor account information. |
| **Priority (Release)** | R2 |
| **Trigger:** | External - User submitting a request for a clothing |
| **Pre-condition(s):** | 1. - System has been set up and configured. 2. - System is running and is open. 3. - User has accessed website using URL 4. - User authorized himself and signed into the account. |
| **Main (Success) Flow:** | 1. User navigates to the "Request Clothing" page 2. System requests user to provide information about the required clothing, i.e., size, color, fabric, and style 3. User provides the required information 4. System verifies that all the necessary information is provided  * If information is missing or invalid, System displays a message. Return to Step 2  1. System searches the inventory for a clothing item that matches the user's criteria 2. If a match is found, System sends a notification to the user's account (box location, clothing details, and donor account information)   ● User can accept or decline the clothing item   1. user accepts the clothing item |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the request process. The following steps would occur:   1. User selects option to cancel the request process 2. System requests confirmation to cancel 3. User confirms 4. System returns to the main screen   Alternate Flow #2: in the success scenario 6, if there is not any match about the provided criteria, The following steps would occur:   1. system notifies the user that there is not exact match 2. System asks the user for the later notification about the clothes 3. User selects option to receive mails on the related clothes  * if not, then the system returns to the main page  1. the confirmation message is displayed on the screen 2. related matches are notified to the user if new donations are added   Alternate Flow #3: In the success scenario 7, users can decline the request. The following steps would occur:   1. User selects option to decline the request 2. System requests confirmation to cancel 3. User confirms intent 4. System returns to the main screen |
| **Post Condition:** | 1. System provides the matches on the specified search functionalities. 2. System notifies the users about the newly added matches on the system |
| **Requirements:** | 1 - The user shall provide information about the required clothing, 2 - The system shall verify that all the necessary information is provided, 3 - The system shall notify the user once a match is found, 4 - The user shall be able to accept or decline the clothing item. |

### Use Case 10

|  |  |
| --- | --- |
| **Use Case Number:** | UC-10 |
| **Use Case Name:** | Provide Donation Feedback |
| **Actor(s):** | User |
| **Description:** | After a request made by users, users are free to provide feedback for the clothes (ex: condition is same/not same as described by the donator). These feedbacks are added to the feedback board of the personal accounts who donated the particular clothes. |
| **Priority (Release)** | R3 |
| **Trigger:** | External - Users provide feedback for the requested donations |
| **Pre-condition(s):** | * System has been set up and configured. * System is running and is open. * User has accessed website using URL * User authorized himself and signed into the account * User requested for the donation |
| **Main (Success) Flow:** | 1. User opens the feedback panel after the request is completed 2. System asks the user to provide some feedback not more than 50 words 3. User provides the feedback 4. System verifies that the constraint is met.  * if the word count is higher, then return to the step 2  1. User submits the feedback 2. The feedback is uploaded to the banner of donator automatically 3. system successfully completes feedback mechanism |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the feedback process. The following steps would occur:  1. User selects option to cancel the process  2. System requests confirmation to cancel  3. User confirms  4. System returns to main screen |
| **Post Condition:** | Alternate Flow #1: User did not complete the feedback providing process. System does not store the Users’ feedback.  Main (Success) Flow 6: The feedback is displayed on the dashboard of the donor |
| **Requirements:** | 1 – The user shall provide feedback not more than 50 words.  2 - The user shall be able to cancel the process if needed  3 - The feedback should be displayed on the donor’s account. |

### 

### 

### Use Case 11

|  |  |
| --- | --- |
| **Use Case Number:** | UC-11 |
| **Use Case Name:** | Raise an Issue |
| **Actor(s):** | User |
| **Description:** | Users can mention complaints, suggestions, and reports (reports mainly cover the condition of container boxes, for ex: damaged, not working, etc) to the system. Users provide a text message (may contain a picture description of a problem / or visualization of the suggested functionality). |
| **Priority (Release)** | R2 |
| **Trigger:** | External - User raises an issue(suggestion, report, complaint) |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - User has accessed website using URL  - User authorized himself and signed into the account.  - User must have interacted with the system. |
| **Main (Success) Flow:** | 1. User opens the “Raise an Issue” panel.  2. System requires user to provide a header to the functionality the issue is about  3. System verifies that the header does not contain more than 5 words   * if the requirement is not satisfied, then return to the step 2   4. System asks the user to provide a text description of the issue.  5. Users submit the issue.  6. The issue is accepted by the system |
| **Alternate Flows:** | Alternate Flow #1: After Step 2 in the success scenario System will display the option to Cancel the process. The following steps would occur:  1. User selects option to cancel the process  2. System requests confirmation to cancel  3. User confirms  4. System returns to main screen  Alternate Flow #1: After Step 4 in the success scenario users provide a visual description of the corresponding issue. The following steps would occur:  1. User selects option to “upload image/video”  2. System accepts the visual description  4. Go to the step 6 in the success scenario |
| **Post Condition:** | Alternate Flow #1: User did not complete the process. System does not forward the User’s report to the volunteers.  Main (Success) Flow 6: Feedback is provided and will be used to improve the system. |
| **Requirements:** | 1 – The user shall provide a header to the issue  2 - the user shall be able to provide a text description of the problem  3 - the user shall be able to provide a visual description of the problem |

### 

### Use Case 12

|  |  |
| --- | --- |
| **Use Case Number:** | UC-12 |
| **Use Case Name:** | Assist User |
| **Actor(s):** | Customer Support |
| **Description:** | Customer support answers users’ questions and assists in the process. |
| **Priority (Release)** | R2 |
| **Trigger:** | External - Customer support assists users. |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - Actor has accessed website using URL  - Customer support authorized himself and signed into the account.  - User asked a question or requested an assistance |
| **Main (Success) Flow:** | 1.Customer support gets the request  2.Actor answers the question  3.Customer Support assists the user during the process. |
| **Alternate Flows:** | Alternate Flow #1: After Step 1 in the success scenario Customer support will be able to cancel the request in these cases:  1.The request is not made about the system  2.The request contains some unaccepted terms (curse)  3.The same request is made several times (although answered for the first time) |
| **Post Condition:** | Alternate Flow #1: Customer support did not accept the request. User is not answered and assisted.  Main (Success) Flow 3: Customer Support assists the user by accepting his request. |
| **Requirements:** | 1 – The Customer support shall be able to cancel the request  2- User shall get the answer to his question  3- User shall get assistance on the issue raised. |

### 

### Use Case 13

|  |  |
| --- | --- |
| **Use Case Number:** | UC-13 |
| **Use Case Name:** | Analyze Feedback |
| **Actor(s):** | Customer Support |
| **Description:** | Customer support analyzes the feedback the user provided. |
| **Priority (Release)** | R2 |
| **Trigger:** | External - Customer support analyzes the user request. |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - Actor has accessed website using URL  - Customer support authorized himself and signed into the account.  - User provided feedback (complaint, report, suggestion) |
| **Main (Success) Flow:** | 1.System send the feedback to the customer support  2.Customer Support gets the feedback  3.Customer support analyzes the user request  4.Actor verifies the request  5.System accepts the request as analyzed. |
| **Alternate Flows:** | Alternate Flow #1: During the step 3 in the success scenario Customer support will be able to remove the report in these cases:  1.The feedback is not made about the system  2.The feedback contains some unaccepted terms (curse)  3.The feedback request is made several times (although answered for the first time)  4.The suggestion is already a process working on the system  5.The Complaint and Report is insufficient |
| **Post Condition:** | Alternate Flow #1: Customer support removed the feedback. The report is not forwarded to the admin team  Main (Success) Flow 5: the request is marked as analyzed |
| **Requirements:** | 1 – The Customer support shall be able to remove the report  2- the actor shall analyze the report |

### 

### Use Case 14

|  |  |
| --- | --- |
| **Use Case Number:** | UC-14 |
| **Use Case Name:** | Forward Feedback |
| **Actor(s):** | Customer Support |
| **Description:** | Customer support forwards the feedback to the admin team. |
| **Priority (Release)** | R2 |
| **Trigger:** | External - Customer support decides to forward the request. |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - Actor has accessed website using URL  - Customer support authorized himself and signed into the account.  - User provided feedback (complaint, report, suggestion)  - Customer support analyzes the report  - Customer support marks the report as analyzed |
| **Main (Success) Flow:** | 1.Customer support chooses the option to forward the request  2. System gets the request  3. System asks for the confirmation  4.Actor verifies the process  5. System sends the report to the admin team  6. System displays the confirmation message. |
| **Alternate Flows:** | Alternate Flow #1: During Step 3 in the success scenario System will display the option to Cancel the forwarding process. The following steps would occur:  1. User selects option to cancel the process  2. System requests confirmation to cancel  3. Customer services confirms  4. System returns to main screen |
| **Post Condition:** | Alternate Flow #1: Customer support canceled the feedback forwarding process. The report is not forwarded to the admin team  Main (Success) Flow 4: Admin team is warned about the issue. |
| **Requirements:** | 1 – The Customer support shall be able to forward the report  2- The report shall be able to cancel forwarding process  3-admin team shall accept the request. |

### Use Case 15

|  |  |
| --- | --- |
| **Use Case Number:** | UC-15 |
| **Use Case Name:** | Work on Feedback |
| **Actor(s):** | Admin |
| **Description:** | Admin team gets the Report and works on it for the improvements |
| **Priority (Release)** | R2 |
| **Trigger:** | External - admin team works on the issue provided by the user. |
| **Pre-condition(s):** | - System has been set up and configured.  - System is running and is open.  - Actor has accessed website using URL  - Admin authorized himself and signed into the account.  - Customer Support forwarded the feedback to the Admin |
| **Main (Success) Flow:** | 1.Admin gets the feedback  2.System displays the report provided by the user  3.Admin analyzes the report  4.Admin make changes on the system  5.Admin publish the updates  6.System displays the new changes made on the system |
| **Alternate Flows:** | Alternate Flow #1: During Step 3 in the success scenario Admin removes the suggestion. The following steps would occur:   1. Admin analyze if the suggestion will meet the business requirements 2. Admin verifies that the report does not satisfy the business requirements 3. Actor removes the report |
| **Post Condition:** | Alternate Flow #1: Admin removed the suggestion report from the system.  Main (Success) Flow 6: The system displays the new changes made on the system based on the reports |
| **Requirements:** | 1.Admin shall be able to remove the suggestion report  2.Admin shall make changes on the system based on the reports  3.System shall display the changes made by the administration team |

### Use Case 16

|  |  |
| --- | --- |
| **Use Case Number:** | UC-16 |
| **Use Case Name:** | Manage User account |
| **Actor(s):** | Admin |
| **Description:** | The Admin team is authorized to delete or edit information on user profiles. User accounts can also be deleted completely if suspended for any behavior against the rules. |
| **Priority (Release)** | R1 |
| **Trigger:** | External - Admin team supervises a user account |
| **Pre-condition(s):** | 1. - System has been set up and configured. 2. - System is running and is open for donation. 3. - The system has been accessed using URL 4. - Admin authorized himself and signed into the account. |
| **Main (Success) Flow:** | 1. Team displays the list of users’ profiles 2. Admin team members selects a user profile 3. System displays the user's profile information and options to edit or delete 4. Admin team member selects edit or delete 5. System verifies the user's selection and asks for confirmation 6. System executes the selected action (edit or delete) 7. System displays a confirmation message |
| **Alternate Flows:** | Alternate Flow #1: After Step 5 in the success scenario System will display the option to Cancel the process. The following steps would occur:   * Admin selects option to cancel the process * System requests confirmation to cancel * Request is confirmed * System returns to the main screen   Alternate Flow #2: After Step 4 in the success scenario, the User account is selected to be edited. The following steps would occur:   * System displays the selected user's profile information * Admin team member modifies the user's profile information * The action is submitted to the system * System verifies the modified information   Alternate Flow #3: After Step 4 in the success scenario, the User account is selected to be deleted. The following steps would occur:   * the details of user information, behavior, and history is displayed * Admin team choose the delete option * System asks for the confirmation of the request * Request is confirmed * System displays a confirmation message   Alternate Flow #4: After Step 2 in the success scenario, the User account is decided to be suspended. The following steps would occur:   * Suspend option is selected * The reason for the action is explained by the admin team members * The descriptive text message about the reason to suspend is sent by the email * user account history and settings are deleted from the system * system request the confirmation of the process * the request is confirmed * system deletes the user account record. |
| **Post Condition:** | Alternate Flow #1: Process is not completed. System does not take any action on the User account.  Alternate Flow #2: User account details are updated on the system  Alternate Flow #3: User account record is deleted from the system  Alternate Flow #4: User account is suspended and deleted from the system |
| **Requirements:** | 1 - Admin team members shall be authorized to edit or delete user profiles,  2 - The system shall edit the user account details  3 - The system shall delete a user account completely  4 - The system shall delete a user account completely if suspended for any behavior against the rules. |

# References

1.Cockburn, A. *Writing effective use cases - www-public.imtbs-tsp.eu*, *Writing Effective Use Cases*. Humans and Technology. Available at: <https://www-public.imtbs-tsp.eu/~gibson/Teaching/Teaching-ReadingMaterial/Cockburn00.pdf>.

2.Author links open overlay panelEunsuk Hur *et al.* (2020) *Rebirth fashion: Secondhand clothing consumption values and perceived risks*, *Journal of Cleaner Production*. Elsevier. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S0959652620329966#preview-section-recommended-articles>.

3.*Use case document template* (2023) *Project Management Docs*. Available at: <https://www.projectmanagementdocs.com/template/project-documents/use-case-document/>

4.UXPin (2023) *7 constraints in design and how to overcome them*, *Studio by UXPin*. Available at: <https://www.uxpin.com/studio/blog/constraints-in-design/#:~:text=Design%20constraints%20are%20limitations%20or,these%20limitations%20before%20every%20project>.