**Kingdom Of Saudi Arabia**

**Ministry Of Education**

**Taibah University**

**College Of Computer Science and Engineering**

**المملكة العربية السعودية**

**وزارة التعليم**

**جامعة طيبة**

**كلية علوم وهندسة الحاسب الآلي**

**Logo

Description automatically generated**

**Saving Grace System**

**A Project Report Submitted to Fulfil the Requirements of System Modelling and Analysis**

**By Group (Group-No.):**

**Name: (SID:)**

**Supervised By**

**Dr. Nada Alamoudi**

Contents

[1. Project Purpose 1](#_30j0zll)

[2. Functional Requirements 1](#_1fob9te)

[3. Use Case Description 1](#_3znysh7)

[4. Non-Functional Requirements 6](#_tyjcwt)

[4.1 Availability 6](#_3dy6vkm)

[4.2 Reliability 6](#_1t3h5sf)

[4.3 Maintainability 6](#_4d34og8)

[4.4 Portability 6](#_2s8eyo1)

[5. Use Case Diagram 7](#_17dp8vu)

[6. Class Diagram 8](#_3rdcrjn)

[7. Sequence Diagram 9](#_26in1rg)

[8. Conclusion 10](#_lnxbz9)

# Project Purpose

Nowadays, poor’s percentage is increasing due to war and pandemic situations. Therefore, a system which helps in providing food for the poor is necessary. This system can help those in need to get foods. With the help of this system, restaurants and families with additional foods can provide these foods to those in need. Non-governmental organizations can play the role of a mediator between restaurants or donors and the poor. By communicating with donors through this system and taking the available quantities for needy people or families with low incomes. The product also aims to reduce food waste and donate leftover food available

# Functional Requirements

The functional requirements are shown as follows.

* Each of the users of the system which are the user, the restaurant or the organization can create a new account in the system by inserting the details.
* Each of the users of the system which are the user, the restaurant or the organization can login to the system to interact with it.
* For the user and the restaurant to donate food, each of them can make a contact with the organization to have further details.
* The user and the restaurant can send a donation request to the organization to donate food.
* After the organization has received the donation request, the organization can handle the request by accepting or rejecting it.
* The organization can manage the beneficiaries which are going to benefit from the donated food by adding, updating or deleting them.

# Use Case Description

The following tables show the functional requirements of the system for the user and the organization.

*Table 1 : Create Account Functional Requirement*

|  |  |
| --- | --- |
| Use Case Name | Create Account |
| Actor | User, restaurant and organization. |
| Description | This feature allows the users to log into the system using a new account. |
| Pre-Conditions | Each one of the actors must be holding an email address in order to register. |
| Post-Conditions | The account has been added to the database. |
| Flow of Events | The user completes the registration form.  The register button is pressed by the user. |
| Special Requirement | All of the inserted data must be stored in a secured database. |
| Exceptions | When entering invalid data, an error notice appears. |

*Table 2 : Login Functional Requirement*

|  |  |
| --- | --- |
| Use Case Name | Login |
| Actor | User, restaurant and organization. |
| Description | By entering the username and password, the users can gain access to the system. |
| Pre-Conditions | The database contains the account. |
| Post-Conditions | The system is accessible to users, restaurants and organizations. |
| Flow of Events | The username and password are filled by the user, the restaurant or the organization.  The login button is pressed by the user, the restaurant or the organization. |
| Special Requirement | The sent data to the server for checking must be secured. |
| Exceptions | When entering the wrong username and password, an error message appears. |

*Table 3 : Contact Functional Requirement*

|  |  |
| --- | --- |
| Use Case Name | Make a Contact |
| Actor | User, restaurant and organization. |
| Description | This feature allows organizations to communicate with users and restaurants to collect the food. |
| Pre-Conditions | The system requires any of the actors to be logged in. |
| Post-Conditions | Organizations can interact with users and restaurants for further details. |
| Flow of Events | The actor goes to the contact interface.  The actor types in the desired message.  Message is sent by the actor. |
| Special Requirement | - |
| Exceptions | When the actor overlooks the message box, an error message appears. |

*Table 4 : Send Donation Request Functional Requirements*

|  |  |
| --- | --- |
| Use Case Name | Send Donation Request |
| Actor | Restaurant and User. |
| Description | This function allows the users and restaurants to send a request in order to donate food. |
| Pre-Conditions | Actors must be logged in. |
| Post-Conditions | The request is sent to the system to be handled by the organizations. |
| Flow of Events | Actor inputs the food type and its quantity.  Actor presses the send button. |
| Special Requirement | - |
| Exceptions | If the actor misses some of the fields, an error must be shown. |

*Table 5 : Handle Donation Request Functional Requirement*

|  |  |
| --- | --- |
| Use Case Name | Handle Donation Request |
| Actor | Organization. |
| Description | This function allows the organization to handle the donation request in terms of accepting or rejecting it based on their availability. |
| Pre-Conditions | The request must be sent by the users or the restaurants. |
| Post-Conditions | The request is processed to whether being accepted or rejected. |
| Flow of Events | Organization clicks on the request sent.  Organization accepts or rejects the request. |
| Special Requirement | - |
| Exceptions | - |

*Table 6 : Add Beneficiary Functional Requirement*

|  |  |
| --- | --- |
| Use Case Name | Add Beneficiary |
| Actor | Organization. |
| Description | The organization can use this function to add beneficiaries to the system who are going to benefit from the food donated. |
| Pre-Conditions | The organization must first log into the system. |
| Post-Conditions | Beneficiary has been entered into the database. |
| Flow of Events | The beneficiary information form is completed by the organization.  The add beneficiary button is pressed by the organization. |
| Special Requirement | - |
| Exceptions | Because the beneficiary data is invalid, it will not be added. |

*Table 7 : Update Beneficiary Functional Requirements*

|  |  |
| --- | --- |
| Use Case Name | Update Beneficiary |
| Actor | Organization. |
| Description | The organization can use this function to update the details of a beneficiary. |
| Pre-Conditions | The beneficiary data must be entered into the database. |
| Post-Conditions | The beneficiary will be updated in the database. |
| Flow of Events | The new beneficiary information form is filled out by the organization.  The update beneficiary button is pressed by the organization. |
| Special Requirement | - |
| Exceptions | While updating the beneficiaries in the database, wrong data entered in the fields which causes an error. |

*Table 8 : Delete Beneficiary Functional Requirements*

|  |  |
| --- | --- |
| Use Case Name | Delete Beneficiary |
| Actor | Organization. |
| Description | The organization can delete a beneficiary from the database using this function. |
| Pre-Conditions | The beneficiary data must be entered into the database. |
| Post-Conditions | The beneficiary will be removed from the database. |
| Flow of Events | The organization selects the appropriate beneficiary to be deleted.  The delete beneficiary button is pressed by the organization. |
| Special Requirement | - |
| Exceptions | - |

# Non-Functional Requirements

## Availability

Users should be able to access the system at any time.

## Reliability

Under different workloads, the system should maintain the same degree of performance.

## Maintainability

The system documentation should be well-written in order to aid in system maintenance in subsequent stages of development.

## Portability

The system will be compatible with a wide range of web browser versions.

# Use Case Diagram

Diagram

Description automatically generated

*Figure 1 : Use Case Diagram*

# Class Diagram

Diagram

Description automatically generated

*Figure 2 : Class Diagram*

# Sequence Diagram

The following figure shows the sequence diagram of the send request function by the user.

A picture containing graphical user interface

Description automatically generated

*Figure 3: Sequence Diagram 1*

Also, the following shows the sequence diagram of the handle request function.

A picture containing diagram

Description automatically generated

*Figure 4: Sequence Diagram 2*

Furthermore, the following shows the sequence diagram of the add beneficiary function.

Graphical user interface, diagram

Description automatically generated with medium confidence

*Figure 5: Sequence Diagram 3*

# Conclusion

Due to war and pandemic crises, the poor's percentage is rising. As a result, a system that assists in the provision of food for the poor is required. This technique can assist those in need in obtaining food. Restaurants and families with extra food can use this system to provide these goods to those in need. Non-governmental organizations (NGOs) can act as a middleman between eateries and the disadvantaged. By using this technique to communicate with donors and taking the available quantities for needy people or low-income households. The product also seeks to reduce food waste by donating any leftovers.