***SOFTWARE REQUIREMENTS SPECIFICATION***

**1.0 Introduction**

This document outlines the Software Requirements Specification (SRS) for the food bank management system. It provides a comprehensive overview of the software's purpose, scope, context, and constraints. The system is designed to facilitate the registration and tracking of individuals receiving food assistance which ensures efficiency, accuracy, and security in the process.

**1.1 Goals and objectives**

The primary goal of the system is to streamline the food bank registration and distribution process. The software aims to enhance operational efficiency by automating the user registration, verifying their eligibility, preventing duplicate/multiple entries, and also generating reports for administrative use. By implementing this system, the food bank can ensure fair distribution of resources while minimizing manual workload and errors.

**1.2 Statement of scope**

The food bank management system is designed to register the individuals, verify their eligibility, track food distribution, and manage the staff’s access. The system will accept inputs such as scanned ID information or manually entered details, process the data to check for duplicate entries or eligibility issues, and generate unique registration slips for users. Outputs will include warning messages for duplicates or underage users, reports for administrators, and verification logs for audit purposes. The system is not responsible for the physical distribution of food but will provide tracking mechanisms for accurate record-keeping.

**1.3 Software context**

The software operates within a food bank environment where efficiency and accuracy are critical. It integrates with existing hardware, such as barcode scanners and database management systems, to ensure seamless data collection and verification. The system will be used by different roles, including the ICD staff, administrators, and kitchen staff, each with their own distinct permissions and functionalities. Strategic considerations include ensuring real-time synchronization across multiple registration points, compliance with data protection regulations, and adaptability to large-scale events.

**1.4 Major constraints**

Several constraints impact the design and implementation of this system:

* Hardware Compatibility: The software must be compatible with barcode scanners capable of reading PDF417 codes on government-issued IDs.
* Internet Connectivity: The system requires a stable internet connection for real-time synchronization and database access.
* User Access Control: Different user roles (staff, administrators, kitchen staff) must have clearly defined permissions to prevent unauthorized data modifications.
* Scalability: The system must handle high volumes of registrations, particularly during large-scale distribution events.
* Security and Compliance: Personal data must be protected in accordance with data privacy regulations, ensuring confidentiality and preventing misuse.
* Error Handling: The system must include mechanisms for addressing duplicate entries, incorrect ID scans, and system downtime scenarios.

**2.0 Usage scenario**

**This section provides a usage scenario for the software. It organized information collected during requirements elicitation into use-cases.**

**2.1 User profiles**

**1. ICD Staff:** The leading role is to take the people’s information either by scanning IDs or entering them manually.

* Scanning people’s IDs’ or entering the information manually.
* This system will provide real-time warning messages if there is a duplicate use or uses under 18.
* The system will show if the use has been blocked and how long.
* For each user, the system will create unique registration slips.

**2. ICD Admin:** The key role is mange the system and the database information.

* The system will provide only an admin page.
* The admin will have access to all database information.
* The admin will have the ability to add staff or delete staff member accounts.
* The admin will be able to block and unblock the violated uses based on the situation.
* The admin will download daily reports of all the users that have been served.

**3. ICD Kitchen Staff:** The key role is to scan the people slipping and serve them the food.

* Scanning people’s slips to mark them as served.
* Making sure that the people only served one time using the warning messages from the system.

**2.2 User stories**

**All the user stories defining the use-cases for the software are presented using the user’s own words.**

**1. User Story 1 the quick registration via ID scanning:** In the prosses the staff will use a 2d scanner to extract all the important information from the barcode in the back of the ID card. The system will extract the information to the appropriate labels like last name and mail address. If all the information is right and in the correct format then the staff member will right the user slips to the appropriate labels, print it and hand it to the user.

**2. User Story 2 warning system and banded individuals:** In this prosses the system will take the information and compare it to the existing information in the database. If the address looks the same or the age is under 18 a warning message will appear. For the duplicate address, the system will prompt the staff to decide if they want to block the individual or not. If the staff want to block them then they will be marked as blocked in the database. If the staff examen the situation and find that the home is two families or an apartment building, they have the option of not blocking the user. On the other hand, if the individual is under 18, they will be blocked, and they are not going to be served.

**3. User story 3 kitchen staff:** In this prosses the kitchen staff will have their page with a field that they will put the user slips number in to mark them as served. This process is the main process to validate that the individual will be served one time. If the individual claims that they lost the slip the kitchen staff will have search access to the database. The staff will ask the individual for their id and write the last 4 number of the id, and they will be able to see the user slip number and if they been served or not.

**4. User story 3 monitoring and maintaining the system:** In this prosses the administrator will have the ability to block and unblock individual base on the evidence that they provide. The administrator will have full access to the database so he/she will be able to edit individual information in the database. Also, at the end of the day the admin will have the ability to generate reports of the people that have been served. The admin will be able to create new staff users or admin users as well as deactivating user account.

**2.3 Special usage considerations**

**Special requirements associated with the use of the software are presented.**

1. **Real-time precise synchronization:** with multiple registration point at the same time it is challenging to prevent duplication.
2. **Hardware compatibility:** The computer must have access to the internet. Also, the scanner must have the ability to scan PDF417 which is found in the back of state issued ids.
3. **Handling large-scale events:** For example, turkey drive is large even the ICD do during the thanksgiving week which involve drive around and donating. So, the system needs to be scalable for those kinds of events.
4. **Verification using another form of documentation such as passport:** One of the options the individual has is if they do not have valid id card they can show their passport as form of validation. Due to that we must have a scanner that will read the DMZ.

**3.0 Data Model and Description**

**This section describes information domain for the software**

**3.1 Data Description**

**Data objects that will be managed/manipulated by the software are described in this section.**

**3.1.1 Data objects**

**Data objects and their major attributes are described.**

**3.1.2 Relationships**

**Relationships among data objects are described using CRC cards. No attempt is made to provide detail at this stage.**

**3.1.3 Complete data model**

**An UML Class model for the software is developed**

**3.1.4 Data dictionary**

**A reference to the data dictionary is provided. The dictionary is maintained in electronic form.**

**4.0 Functional Model and Description**

**Description of major software functions along with UML Use Case, sequence, and communication diagrams.**

**4.1 Description for Function n**

**A detailed description of each software function is presented by completing a use case template. Section 4.1 is repeated for each of n functions.**

**4.1.1 Use case name**

**Unique name for function is defined.**

User registration and verification

**4.1.2 Actors**

**Entities that produce of consume the information associated with the fucntion.**

* **ICD staff:** Put individual information into the system including individual slips numbers.
* **ICD Kitchen staff:** Take the individual slips number and put them to the system and mark them as served.
* **ICD Administrator:** Monitor the system for any wrong information or any information that extracted wrong from scanning or staff input.
* **Database system:** The database system will save and keep track of all information input and will do crucial role by examining the age restriction and duplicate addresses.

**4.1.3 Precondtions**

**A detailed description of the input and output interfaces for the function is presented.**

* The registration system must be operational and connected to the database.
* The scanner must be connected to the computer and functional.
* The database must have a existence list of all band users.

**4.1.4 Triggers**

**A detailed description of when the function will be utilized by the system.**

* The registration prosses must trigger when the ICD staff scan individual ID or enter the info manually.
* If the ID for the individual comes back as blocked or underage this should trigger and alert the ICD admin.

**4.1.5 Scenario Description**

**Describe the flow of events needed to accomplish the use case.**

* The ICD staff will scan the individual id to extract the important info or enter the information manual.
* The system will check the individual Age and banding status.
* If there is no issue found for the user, then the system will add the user to the database. Also, the staff will provide them with a unique registration slip.
* If there is an issue detected the system will display warning message for the staff accordingly.
* When approved the user will now be able to go to the kitcken and collect their food.

**4.1.6 Post Conditions**

**Any design constraints that will impact the subsystem are noted.**

* The individuals are registered, and they have been added to the system database.
* If flagged their data will be logged to the ICD administrators for review.
* If denied they will not issue registration slip.

**4.1.7 Exceptions**

**Describes how the system should respond to unusual circumstances.**

* **Internal issue with the system itself:** If the server we turn off or the host of the server has some minutes in their systems. Also, if there are any networking problems and the system lost connections to the internet. Moreover, how the data will be handled if they are not synchronized.
* **Scanning problems:** if there is for whatever reason problem with the code in the back of the id. Or scanner drivers have problems and need update.
* **Duplicate entry problem:** sometimes even if there is a different space the system will not consider it as the same address which is going to be a problem.

**4.1.2.1 Use Case Name**

**Duplicate Address Warning**

**4. 1.2.2 Actors**

* **ICD Staff:** Inputs individual information into the system and registers participants.
* **ICD Administrator:** Monitors flagged entries and approves/rejects registrations in case of duplicates.
* **Database System:** Saves and tracks all registration information and compares new entries to existing addresses to detect duplicates.

**4. 1.2.3 Preconditions**

* The registration system must be operational and connected to the database.
* The system must have an updated list of all previously registered addresses.
* The duplicate check function must be enabled to compare addresses in real time.

**4.1.2.4 Triggers**

* The function is triggered when an ICD staff member enters or scans an individual’s ID for registration.
* If the new registrant’s address matches an existing address in the database, an alert should be triggered for review.

**4.1.2.5 Scenario Description**

* The ICD staff enters or scans the individual’s ID.
* The system checks the individual's address against existing records.
* If a duplicate is detected, the system flags the registration and alerts the ICD administrator.
* When Admin receives that flag they will have to examine it and ether approve it or denied it.
* If an individual approved, they will be added to the system they will be given the unique registration slip.
* If individuals denied they will add to the system as block uses and they will not be given registration slip.

**4.1.2.6 Post Conditions**

* The system ensures that multiple registrations at the same address are flagged for manual review.
* If a registration is approved despite a duplicate address, it is logged for tracking.

**4.1.2.7 Exceptions**

* **Address Formatting Issues:** If the system fails to recognize small variations in address formatting (e.g., “Street” vs. “St.”), the duplicate check may not work properly.
* **Legitimate Multiple Registrations:** Some households may have multiple eligible registrants; the system should allow administrators to approve these cases manually.
* **Manual Entry Errors:** If ICD staff enter an address incorrectly, it may not be flagged as duplicate even when it should be.

**4.1.3.1 Use Case Name**

**Banned Individual Detection**

**4.1.3.2 Actors**

* **ICD Staff:** Inputs individual information into the system and initiates registration.
* **ICD Administrator:** Monitors flagged individuals and approved/rejects their registration attempts.
* **Database System:** Maintains a list of banned individuals and checks all new registrations against this list.

**4.1.3.3 Preconditions**

* The banned individual list must be preloaded into the system database.
* The registration system must be functional and connected to the database.
* The scanning device must be operational.

**4.1.3.4 Triggers**

* The function is triggered when an ICD staff member scans an individual’s ID or manually enters their details.
* If the individual’s information matches a banned entry, an alert is triggered.

**4.1.3.5 Scenario Description**

* The ICD staff scans or manually enters the individual's information.
* The system cross-checks the individual’s details with the banned list.
* If a match is found, the system flags the registration and notifies the ICD administrator.
* The administrator reviews the case and either confirms the ban or overrides it.
* If confirmed, the registration is denied, and no slip is issued.
* If overridden, the individual is added to the database and provided a registration slip.

**4.1.3.6 Post Conditions**

* If an individual is banned, their information is logged for review, and they cannot proceed with registration.
* If an override occurs, the administrator’s decision is recorded in the system.

**4.1.3.7 Exceptions**

* **False Positives:** If an individual is flagged by mistake, there should be an option to override and allow registration.
* **Name Variations:** If a banned individual registers under a slightly different name, the system may fail to detect them.
* **Manual Entry Errors:** If ICD staff misspell a name or enter incorrect details, the system may not recognize a banned individual.

**4.1.4.1 Use Case Name**

**Admin Management and Data Reporting**

**4.1.4.2 Actors**

* **ICD Administrator:** Manages system settings, reviews flagged entries, and generates reports.
* **Database System:** Stores and retrieves registration and flagged data.

**4.1.4.3 Preconditions**

* The administrator must have access to the system with the necessary permissions.
* The database must contain up-to-date registration and flagged records.

**4.1.4.4 Triggers**

* The function triggered when an administrator logs in to the system and selects a management option.
* The function is triggered when an administrator generates a report for registered individuals or flagged entries.

**4.1.4.5 Scenario Description**

* The ICD administrator logs in to the system.
* The administrator accesses the **Admin Dashboard**, where they can:
  + Manage system settings and update rules for registration.
  + Review flagged registrations and take necessary actions.
  + Generate reports on registrations, duplicate addresses, or flagged individuals.
* If generating a report, the administrator selects a report type.
* The system retrieves data from the database and exports the report in the requested format (Excel, CSV, or PDF).

**4.1.4.6 Post Conditions**

* Changes made by the administrator (e.g., adding users, updating banned lists) take effect immediately.
* Generated reports are stored in the system for later reference.

**4.1.4.7 Exceptions**

* **Unauthorized Access:** If a non-admin user attempts to access these functions, they should be restricted.
* **Data Corruption or Loss:** If the database experiences errors, backup mechanisms should be in place.
  + **Large Data Exports:** The system should optimize large data exports to prevent performance issues.

**4.2 Software Interface Description**

**The software interface(s)to the outside world is(are) described.**

**4.2.1 External machine interfaces**

**Interfaces to other machines (computers or devices) are described.**

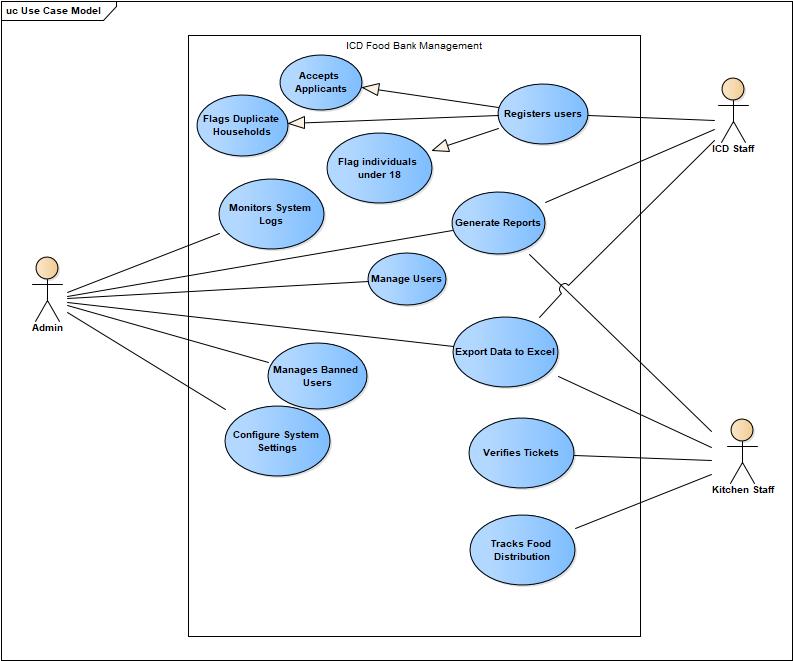
**4.2.2 External system interfaces**

**Interfaces to other systems, products or networks are described.**

**4.2.3 Human interface**

**An overview of any human interfaces to be designed for the software is presented.**

**4.3 Use Case Diagram & description**



**Use Case Description**

**This section provides a detailed description of the key use cases for the ICD Food Bank Management. Each use case defines the interactions between the system and its actors, outlining the essential functionality required to streamline registration, verification, reporting, and security within the food distribution process.**

**4.3.1 Use Case: Register Users**

**Introduction:**

**The Register Users use case ensures that individuals can register for food distribution in a structured and efficient manner. The system verifies user eligibility and prevents duplicate household registrations.**

**Use Case Diagram:**

* **Diagram Name: Register Users**
* **Diagram Number: UC-001**

**Main Actors:**

* **ICD Staff – Registers individuals by scanning IDs and verifying eligibility.**

**Flow of Events:**

1. **ICD Staff scans the user’s ID or passport.**
2. **System checks for duplicate registrations using household data.**
3. **If eligible, the system generates a unique ticket for the user.**
4. **If not eligible (duplicate household or flagged user), the system denies registration.**
5. **The user receives the ticket for food collection (if eligible).**

**Postconditions:**

* **Successful registrations result in a ticket being printed.**
* **Ineligible users are denied registration and flagged if necessary.**

**4.3.2 Use Case: Verify Tickets**

**Introduction:**

**The Verify Tickets use case ensures that each ticket is valid and used only once before food is distributed.**

**Use Case Diagram:**

* **Diagram Name: Verify Tickets**
* **Diagram Number: UC-002**

**Main Actors:**

* **Kitchen Staff – Scans tickets and verifies their validity.**

**Flow of Events:**

1. **Kitchen Staff scans the ticket.**
2. **The system verifies ticket validity and checks if it has been used.**
3. **If valid, the ticket is marked as used, and food is given.**
4. **If invalid or already used, the system denies food distribution.**

**Postconditions:**

* **Valid tickets allow food collection.**
* **Invalid or duplicate tickets prevent users from claiming food twice.**

**4.3.3 Use Case: Manage Users**

**Introduction:**

**The Manage Users use case allows Admins to add, update, or remove users and modify eligibility criteria.**

**Use Case Diagram:**

* **Diagram Name: Manage Users**
* **Diagram Number: UC-003**

**Main Actors:**

* **Admin – Oversees user records and system control.**

**Flow of Events:**

1. **Admin logs into the system.**
2. **Admin searches for a user profile.**
3. **Admin updates user details, removes a user, or modifies eligibility criteria.**
4. **The system saves changes and updates the database.**

**Postconditions:**

* **The user list is updated based on admin changes.**

**4.3.4 Use Case: Generate Reports**

**Introduction:**

**The Generate Reports use case allows authorized users to create custom reports on registration, verification, and food distribution trends.**

**Use Case Diagram:**

* **Diagram Name: Generate Reports**
* **Diagram Number: UC-004**

**Main Actors:**

* **ICD Staff, Kitchen Staff, Admin**

**Flow of Events:**

1. **Users select report type (e.g., registration trends, food distribution logs).**
2. **The system retrieves relevant data and generates a structured report.**
3. **User downloads or views the report.**

**Postconditions:**

* **A report is successfully generated and available for review/export.**

**4.3.5 Use Case: Export Data to Excel**

**Introduction:**

**The Export Data to Excel use case allows users to download system records for further analysis and record-keeping.**

**Use Case Diagram:**

* **Diagram Name: Export Data to Excel**
* **Diagram Number: UC-005**

**Main Actors:**

* **ICD Staff, Kitchen Staff, Admin**

**Flow of Events:**

1. **User selects the data type to export (registrations, distributions, flagged users).**
2. **System processes and formats the data into Excel format.**
3. **User downloads the Excel file.**

**Postconditions:**

* **A structured Excel file is generated and available for download.**

**4.3.6 Use Case: Flag Duplicate Households**

**Introduction:**

**The Flag Duplicate Households use case ensures that only one person per household can register for food distribution.**

**Use Case Diagram:**

* **Diagram Name: Flag Duplicate Households**
* **Diagram Number: UC-006**

**Main Actors:**

* **ICD Staff**

**Flow of Events:**

1. **The system checks the household address during registration.**
2. **If a duplicate household is detected, the user is flagged and prevented from registering.**

**Postconditions:**

* **If no duplicate is found, registration proceeds.**
* **If a duplicate exists, registration is denied, and the user is flagged.**

**4.3.7 Use Case: Flag Individuals Under 18**

**Introduction:**

**The Flag Individuals Under 18 use case ensures that users under 21 years old are flagged and prevented from registering.**

**Use Case Diagram:**

* **Diagram Name: Flag Individuals Under 18**
* **Diagram Number: UC-07**

**Main Actors:**

* **ICD Staff**

**Flow of Events:**

1. **System extracts date of birth from scanned ID.**
2. **If the user is under 18, registration is denied.**
3. **If the user is 18 or older, registration continues.**

**Postconditions:**

* **Users under 18 cannot register for food distribution.**

**4.3.8 Use Case: Tracks Food Distribution**

**Introduction:**

**The Track Food Distribution use case records the number of meals served to ensure fair distribution.**

**Use Case Diagram:**

* **Diagram Name: Track Food Distribution**
* **Diagram Number: UC-008**

**Main Actors:**

* **Kitchen Staff**

**Flow of Events:**

1. **Kitchen Staff marks users as served after ticket verification.**
2. **The system logs the meal distribution count.**

**Postconditions:**

* **Each served meal is recorded for tracking purposes.**

**4.3.9 Use Case: Manage Banned Users**

**Introduction:**

**The Manage Banned Users use case allows Admin to flag and prevent access for individuals violating policies.**

**Use Case Diagram:**

* **Diagram Name: Manage Banned Users**
* **Diagram Number: UC-009**

**Main Actors:**

* **Admin**

**Flow of Events:**

1. **Admin searches for a user profile in the system.**
2. **Admin flags the user as banned and confirms the action.**
3. **The system prevents the banned user from registering in the future.**

**Postconditions:**

* **Flagged users cannot register for food distribution.**

**4.3.10 Use Case: Configure System Settings**

**Introduction:**

**The Configure System Settings use case allows Admin to update system rules and registration policies.**

**Use Case Diagram:**

* **Diagram Name: Configure System Settings**
* **Diagram Number: UC-010**

**Main Actors:**

* **Admin**

**Flow of Events:**

1. **Admin accesses system settings.**
2. **Admin modifies parameters (e.g., flagging rules, ticketing rules).**
3. **System applies changes and updates policies.**

**Postconditions:**

* **System configurations are updated as per admin settings.**

**4.4 Sequence Diagrams**

**Used to model the class interactions needed for the use cases.**

**4.5 Communication Diagrams**

**Used to model the message passing structure of the system functions.**

**5.0 Behavioral Model and Description**

**A description of the behavior of the software is presented.**

**5.1 Description for software behavior**

**A detailed description of major events and states is presented in this section.**

**5.1.1 Events**

**A listing of events (control, items) that will cause behavioral change within the system is presented.**

**5.1.2 States**

**A listing of states (modes of behavior) that will result as a consequence of events is presented.**

**5.2 State Transition Diagrams**

**Depict the manner in which the software reacts to external events.**

**5.3 Activity Diagram**

**Depict the manner in which the software reacts to internal events.**

**6.0 Restrictions, Limitations, and Constraints**

**Special issues which impact the specification, design, or implementation of the software are noted here.**

**7.0 Validation Criteria**

**The approach to software validation is described.**

**7.1 Classes of tests**

**The types of tests to be conducted are specified, including as much detail as is possible at this stage. Emphasis here is on black- box testing.**

**7.2 Expected software response**

**The expected results from testing are specified.**

**7.3 Performance bounds**

**Special performance requirements are specified.**

**8.0 Appendices**

**Presents information that supplements the Requirements Specification**

**8.1 System traceability matrix**

**A matrix that traces stated software requirements back to the system specification.**

**8.2 Product Strategies**

**If the specification is developed for a product, a description of relevant product strategy is presented here.**

**8.3 Analysis metrics to be used**

**A description of all analysis metrics to be used during the analysis activity is noted here.**

**8.4 Supplementary information (as required)**

**Return to** [APM Documents](http://www.rspa.com/docs/index.html)