New Delhi, India

05 September, 2024

SOFTWARE Requirement SPECIFICATION

*EmpowerU Management Information System (MIS)*

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reasons of Changes | Version |
| iTM | 05 September 2024 | The first draft of SRS developed. | v1 |

Table of Contents

[1.0 Introduction 3](#_Toc176542997)

[1.1 Purpose 3](#_Toc176542998)

[1.2 Scope 4](#_Toc176542999)

[1.3 Definition, Acronyms and Abbreviations 4](#_Toc176543000)

[1.4 References 5](#_Toc176543001)

[2.0 Overall Description 6](#_Toc176543002)

[2.1 System Perspective 6](#_Toc176543003)

[2.2 System Functions 6](#_Toc176543004)

[2.3 User Classes and Characteristics 7](#_Toc176543005)

[2.4 Operating Environment 7](#_Toc176543006)

[2.5 Design and Implementation Constraint 9](#_Toc176543007)

[2.6 Assumptions and Dependencies 9](#_Toc176543008)

[3.0 Specific Requirements 11](#_Toc176543009)

[3.1 Functional Requirements 11](#_Toc176543010)

[3.1.1 Data Structure 11](#_Toc176543011)

[3.1.2 Geographical Coverage Entities 11](#_Toc176543012)

[3.1.3 User 12](#_Toc176543013)

[3.1.3 System Features 14](#_Toc176543014)

[A.1 HOME 17](#_Toc176543015)

[A.2 ABOUT 18](#_Toc176543016)

[A.3 INFORMATION KIOSK 18](#_Toc176543017)

[A.4 CONTACT 19](#_Toc176543018)

[A.5 LOGIN 20](#_Toc176543019)

[A.6 AT A GLANCE 20](#_Toc176543020)

[A.7 REGISTRATION 21](#_Toc176543021)

[A.8 REFERRAL 26](#_Toc176543022)

[A.9 GRIEVANCE 26](#_Toc176543023)

[A.10 CASH TRANSFER 27](#_Toc176543024)

[A.11 EXPENDITURE 28](#_Toc176543025)

[A.12 IMPORT LOG 29](#_Toc176543026)

[B.1 ADMIN 30](#_Toc176543027)

[3.2 Non-Functional Requirements 39](#_Toc176543028)

[3.2.1 Security Requirement 39](#_Toc176543029)

[3.2.2 Usability and External Interface Requirement 39](#_Toc176543030)

[Annexure A: Data Dictionary 41](#_Toc176543031)

[Annexure B: Data Flow Diagrams (DFD) 44](#_Toc176543032)

[Annexure C: System Architecture 47](#_Toc176543033)

[Annexure D: Software Architecture 48](#_Toc176543034)

[Annexure F: System Screens 49](#_Toc176543035)

# 1.0 Introduction

## 1.1 Purpose

In Lamwo District, Northern Uganda, the combination of socio-economic pressures and a high influx of refugees has created a challenging environment for the local residents. The district faces widespread child vulnerability worsened by severe food insecurity, malnutrition, and limited access to healthcare, education and other essential services. The arrival of refugees has intensified these issues, placing additional pressure on already limited resources and infrastructure. Families, including both the host and refugee communities, struggle with inadequate financial resources, which hampers their ability to address basic needs and invest in sustainable livelihoods.

The EmpowerU Programme was developed as a social protection initiative led by UNICEF in collaboration with the Development Response to Displacement Impact Project (DRDIP) and funded by the Dutch government. This programme aims to enhance the welfare of children and foster sustainable income-generating skills among vulnerable communities in Lamwo District. It provides a comprehensive package of cash transfers over six months, alongside services such as protection, health, education, and nutrition. Additionally, DRDIP offers savings and skills training to both beneficiaries and non-beneficiaries, reducing pressures on the working population and social services.

Currently, the project data is collected and managed either offline or in various other platforms. The lack of an integrated system makes it difficult to monitor progress, assess impact, and ensure transparency and accountability in cash disbursements and other program activities.

To effectively manage the programme data, establishing a robust Management Information System (MIS) is essential. The Management Information System (MIS) will ensure, all project information, which has to date been collected and managed offline, in different places, is brought together to facilitate real time monitoring and reporting on the project. Hence, an **EmpowerU MIS** application will be developed under the supervision of UNICEF and its partners and will have a centralized database for storing, accessing, and analyzing the programme's data. It will streamline data collection and reporting processes, which will enhance the real-time monitoring and analysis of the data, and will improve transparency in cash disbursements and other program activities.

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the EmpowerU MIS application. It describes the functional and non-functional requirements for the development of the application. This is a critical document in the software development lifecycle and serves as a communication bridge between stakeholders, developers, and testers. It is complaint with ISO/IEC/IEEE 29148:2018 standard.

The document comprises of the system scope, overview, and features along with the system design. It comprehends the system architecture by following the Data Flow Diagrams, System and Software Architecture, and System Screens provided in the Annexures.

## 1.2 Scope

The EmpowerU MIS will serve as an online application to centralize and streamline data management for the EmpowerU Programme. The application will be developed using the open-source software technologies to ensure easy adaptability, usability, and scalability. The data will be stored centrally in a database to ensure consistency, security and accessibility. The design and development team will work closely under the guidance of UNICEF during the development and implementation cycle. The application will have role-based access to create and manage the underlying database and its elements. It will be designed to cater the needs of the key stakeholders. It will have responsive web design to support display on various screen sizes including that of computers, tablets, and mobile devices.

The main scope of the application will specifically involve:

* Review and collate existing data and information, data collection tools, templates and reports.
* Identify gaps in data collection and management and modifications that might be needed.
* Design and develop a comprehensive MIS application to help track, monitor and report on the EmpowerU processes.
* Support the integration of the EmpowerU MIS into the National Single Registry system to ensure alignment with national data management practices.
* Build the capacity of key stakeholders to use and manage the MIS application.

Additionally, the following tasks will be covered in this project -

* Training and capacity building with the stakeholders
* Technical and application administration training
* Develop technical documentation
* Develop test cases and user manual
* Maintenance and support

## 1.3 Definition, Acronyms and Abbreviations

|  |  |  |
| --- | --- | --- |
| # | Item | Description |
| 1 | SRS | Software Requirement Specification |
| 2 | MIS | Management Information System |
| 3 | CSV | Comma Separated Value |
| 4 | UNICEF | United Nations Children's Fund |
| 5 | DRDIP | Development Response to Displacement Impact Project |
| 6 | DFD | Data Flow Diagram |
| 8 | GB | Giga byte |
| 9 | GeoJSON | Geographical JavaScript Object Notation |
| 10 | GIS | Geographical Information System |
| 11 | HTTPS | Hypertext Transfer Protocol Security |
| 12 | IEC | International Electrotechnical Commission |
| 13 | IEEE | Institute of Electrical and Electronics Engineers |
| 14 | ISO | International Organization for Standardization |
| 15 | JPG | Joint Photographic Experts Group |
| 16 | Mbps | Megabits per second |
| 17 | GEG | Girls Empowering Girls |
| 18 | SSD | Solid State Drive |
| 19 | UI | User Interface |
| 20 | UI/UX | User Interface/User experience |
| 21 | vCPU | Virtual Central Processing Unit |
| 22 | VM | Virtual Machine |
| 23 | FAQ | Frequently Asked Questions |

## 1.4 References

|  |  |  |
| --- | --- | --- |
| # | Item | Link |
| 1 | GEG MIS Portal | https://geg.kcca.go.ug/ |

# 2.0 Overall Description

## 2.1 System Perspective

The EmpowerU MIS application will be a one-stop shop to register, manage, track and monitor the household and its beneficiaries details of the EmpowerU programme. The data will be stored centrally in a database to ensure consistency, security and accessibility. It will be accessed and managed by users who will have controlled access based on their roles and permissions. The users will have to login into the system to access and manage the data. The application will use a data-driven, evidence-based approach to monitor, manage, and evaluate the program's various components, including cash transfer, skills training, grievance and other services. The application will have a centralized database to store and manage the eligible household’s data, each with a maximum of four children per household. This will ensure accurate and efficient management of cash transfers, savings, and skillset training activities. The application will present interactive visualizations to present data in a user-friendly manner that will enable stakeholders to track the program's impact in real-time.

The application will integrate data from various sources, including existing tools and manual records, to provide a consolidated view of the program’s progress and impact. The application will allow secure login, data entry, visualization, and reporting capabilities. It will enable real-time monitoring and analysis capabilities that will ensuring comprehensive oversight and effective management of the household’s data.

## 2.2 System Functions

The EmpowerU MIS will comprise of the two applications, the User Interface and the Admin which will be accessed by valid login credentials. The application will comprise of various independent modules. These modules will have interconnection to ensure the user navigation is seamless from one module to another.

The **User Interface** will allow users to register beneficiaries by household, manage grievances, referrals, cash transfer processes, and expenditures whereas the **Admin** application will allow to perform administrative tasks such as managing the underlying database elements which will include user management, role-based access control, classifications, resource management, program cycle and report generation. The application will manage data from various sources, and the data will be updated both using CSV formatted files and web-based data entry forms.

Following will be the core features of the MIS application:

* Allows users to register and manage beneficiaries by households and ensure accurate data collection and tracking.
* Allows to manage grievances and complaints raised by beneficiaries.
* Allows to refer beneficiaries to other services or support programs.
* Allows to manage the disbursement of cash transfers, including tracking and recording transactions for each beneficiary of the household.
* Allows to monitor and record program-related expenditures to ensure budget adherence and financial transparency.
* Allows to import existing data using CSV file imports and web-based data entry forms.
* Provides data visualizations and analytics for better tracking and analysis.
* Allows to download and print visualizations of underlying data in various formats.
* Allows uploading and updating data using industry-standard formats like CSV.
* Information Kisok module will present resources by category.
* Contact module will provide the contact details of the support team and include an enquiry form to send questions or feedback that will facilitate improvement in the application.

## 2.3 User Classes and Characteristics

The following will be the category of users with their expected requirements from the application:

* Administrators
  + Manage the users, roles, and permissions and ensure secure access to the data.
* Specialists
  + Register and manage households and beneficiaries into the system.
* Program Manager
  + Approve registered households and beneficiaries based on the specified criteria.
* Program Coordinator
  + Coordinate with the cash transfer process of the beneficiaries.
* Public users
  + View the content pages like About and Information Kisok.

## 2.4 Operating Environment

The application will be hosted on the cloud-based servers and it will be accessible via web browsers on desktop and mobile devices. It will be compatible with major browsers such as Chrome, Firefox, Safari, and Edge. Following will be the requirements to deploy the platform:

* Reliable and scalable web/application server such as Apache, Tomcat.
* Database server to host the database such as MongoDB.
* Cloud infrastructure such as AWS, Other cloud service.
* Operating system such as Ubuntu and other software components to host the application.
* Version control system to manage code such as GitLab.
* SSL/TLS certificate to secure data transmission between users and the web server.
* Public IP and domain.
* Firewall to protect the network and systems from unauthorized access and attacks.
* Scripts to take regular backups and a robust disaster recovery plan.
* Stable internet connection.

It will need an internet connection to be deployed on the web servers called as Virtual Machines (VMs). The following are the VM details.

|  |  |
| --- | --- |
| # of VMs | Purpose |
| 1 | To host the user interface and the admin application |
| 1 | To host the central database |

The application will be deployed in the cloud on the Virtual Machines (VMs). Each VM will have the following specifications:

Processor 8 vCPU

Memory 16 GB

Storage 500 GB SSD

Internet Connection 100 Mbps recommended

Ports80, 27017, 6379, 443, 21, 22

Web servers shall be protected by the firewall. A public IP and dedicated domain will be needed. A web master shall be assigned the task of managing, trouble shooting and maintaining the web server. There shall be adequate uptime of the web server. A decent internet connection will be required for the web server to manage the expected network traffic.

*Note: The above specifications are recommended requirements, though the application will be able to be deployed on lower specification web-based instance with options of elastic storage and processor. Port 80 would be required only during the development process*

## 2.5 Design and Implementation Constraint

The application will be developed using the following open-source technology and tools that will allow easy extensibility and scalable communication interfaces. The application will use the HTTP protocol for communication over the internet. Below will be the technology stack that lists the following programming environment and tools. See **Annexure B: Software Architecture**.

Operating Tool Ubuntu (RHEL) v 22.04

Front-end Language Typescript v4.0, JavaScript, HTML v5.0, CSS v3.0

Visualization Library eCharts v5.5.1, Leafletv1.9.3

Front-end Framework Angular v16

Back-end Language Node.js v18.1.0, PHP v7.4

Back-end Framework Express v4.19.2, Laravel v8.11.2

Database Tool MongoDB v5.0

Server Type Apache v2.4\*, Docker v20.10.7, Docker composer v1.27.0

Data will be imported from Comma Separated Value (CSV files).

The application will be designed to handle concurrent users and large volumes of data without performance degradation. The application can be integrated with various existing databases and external data sources through APIs. The application will load within a few seconds and provide near real-time updates to ensure timely access to data.

## 2.6 Assumptions and Dependencies

It is assumed that all required data from various sources, managed by humanitarian organizations and government agencies, will be available and accessible. The data provided by different sources will be accurate, up-to-date, and formatted to meet the application’s standard data entry requirements. Users are expected to have reliable internet access and modern web browsers that support the application functionalities. Cooperation from all relevant stakeholders, including data providers and end-users, is crucial for providing necessary inputs and feedback during the development and deployment phases.

The functionality of the application will depend on timely and continuous data feeds from various sources. Any disruption or delay in data provision can impact the platform’s accuracy and usefulness.

The technology stack used in the development of this application will include stable and actively maintained open-source libraries and frameworks. Any major changes or discontinuation of these technologies could affect the project. Successful deployment and adoption of the application will depend on adequate training and support for end-users, including humanitarian agencies, government officials, and program stakeholders.

# 3.0 Specific Requirements

## 3.1 Functional Requirements

### 3.1.1 Data Structure

The MIS will be developed to store and manage the household wise beneficiaries data. The data structure will be comprised of the following elements-

**Program Cycle**: One cycle of the EmpowerU program in which a group of people are registered as beneficiaries.

**Area**: Geographical structure referred as one unit. Example Country, City, Division, District, Parish, Village.

**Time Period**: Reported time period of the data values for each IUS combination. Example 2017, 2016-2017. The time period represents the day, week, month, quarter, year and year range.

**Beneficiary**: The participants of the program or program cycle generally the people who are benefitting from the program.

**Expenditure**: Usage of the cash transfers by households, including the funds allocation towards savings and livelihood activities.

### 3.1.2 Geographical Coverage Entities

The application will be developed to store and manage data at various geographical levels or administrative boundaries. The geographical level will be termed as Area. The admin application has module to manage areas at following level -

* National - This level pertains to the country that is Uganda. The application will accommodate data storage and management for beneficiaries at the national level.
* Sub-national - At this level, the application allows for data management at sub-national levels within the country. It supports the storage and organization of data specific to different regions, districts, divisions, parishes, and villages.

This will enable efficient data handling and analysis of data at both the national and sub-national levels, ensuring that information is appropriately organized and accessible according to the geographical context. As explained in the data structure section the data values will have an area dimension which will refer to the geographical area for which the data is entered. Each area will have the following details –

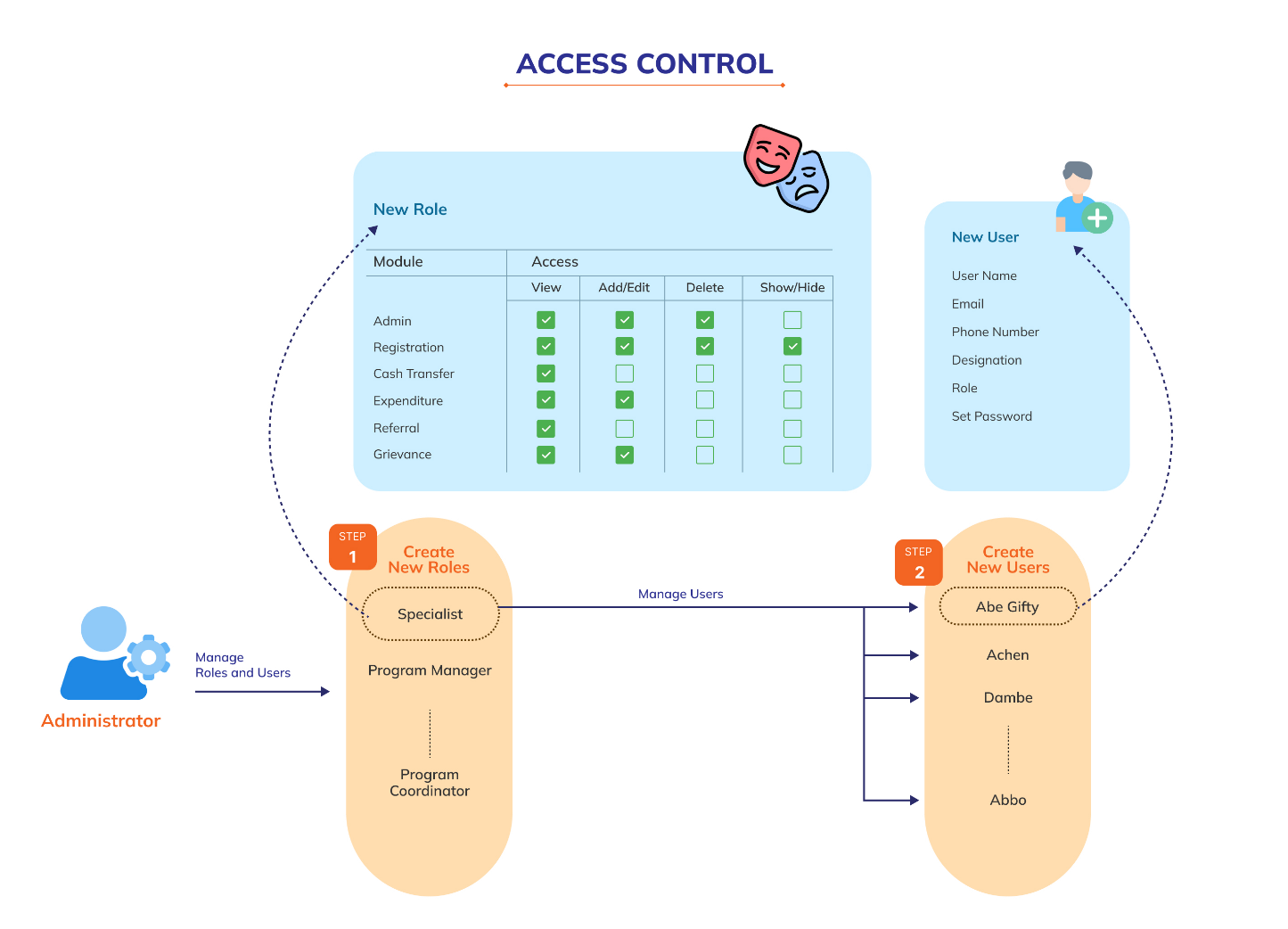
* Area ID
* Area name
* Area Level
* Area Parent ID

### 3.1.3 User

The application will have role-based access to the various modules and will allow to create and manage various users and assign them roles. The application will have public users as well who will visit the home page and view the reference materials uploaded in the information kiosk, also view the contact us information. The access to the registration, grievance, referral, cash transfer and admin modules will be restricted based on use role and permissions. The administrator will create further users for data entry and approval of households or beneficiaries. These users will login with their respective credentials.

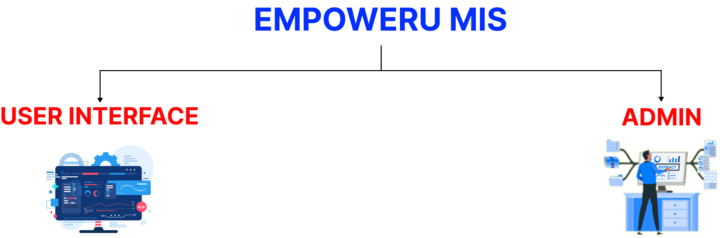
The following will be the Roles that could be assigned to the users of the MIS application:

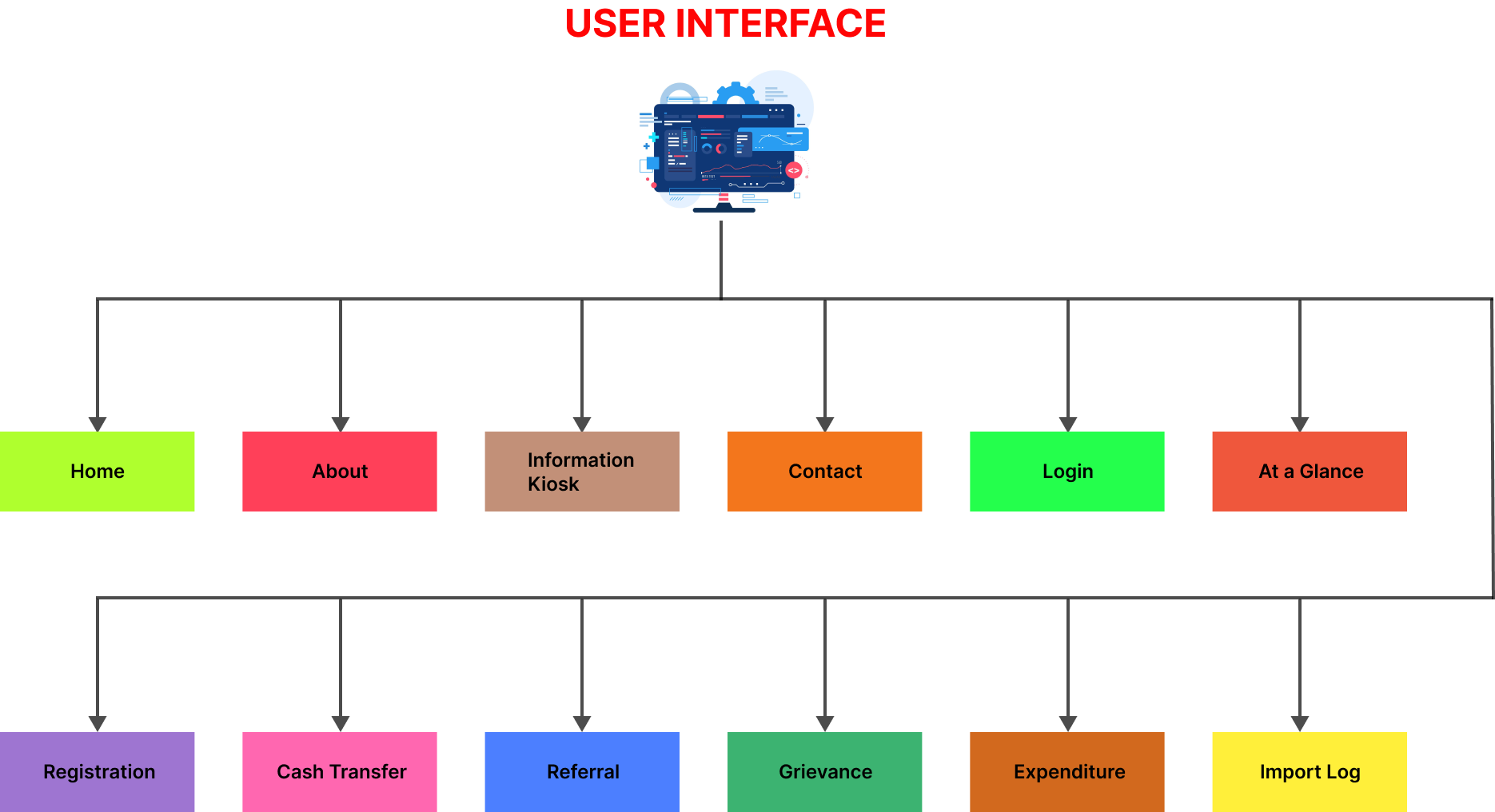
* Administrators
* Specialists
* Program Manager
* Program Coordinator

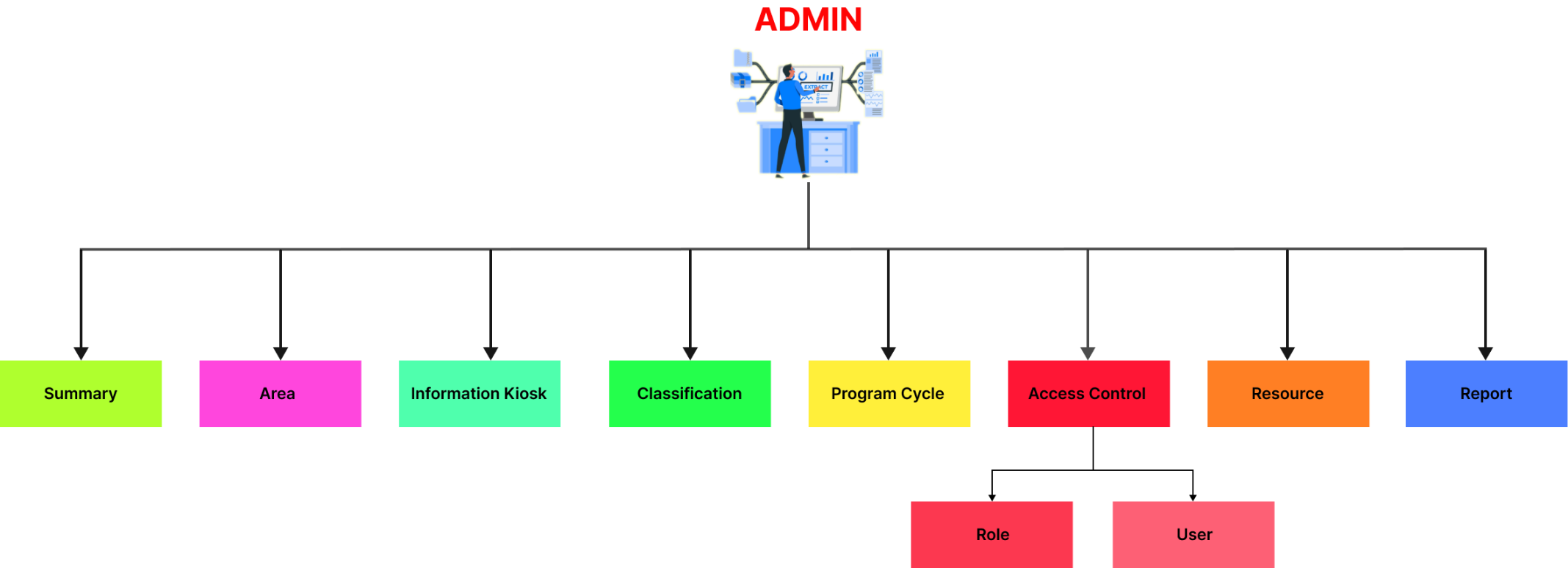
Below diagram depicts the access control of the EmpowerU MIS application:

### 3.1.3 System Features

The application will be comprised of the following two applications -

1. User Interface
2. Admin

The below diagram depicts the modules and submodules of the user interface application.

The below diagram depicts the modules and submodules of the admin application.

1. **USER INTERFACE**

The User Interface application will comprise of the following modules and sub-modules. See Annexure **B: Data Flow Diagram (DFD)** for input – process – output flow details of the system and See **Annexure C: System Architecture** for interaction between modules.

1. Home
2. About
3. Information Kiosk
4. Contact
5. Login
6. At a Glance
7. Registration
8. Cash Transfer
9. Expenditure
10. Referral
11. Grievance

### A.1 HOME

This module will be the landing page of the application that will show the key facts related to EmpowerU program. It will follow the branding guidelines of the program and will also contain partner’s logo. Below is the detailed functional requirement of this module -

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show EmpowerU logo at the top left corner of the homepage. |
| REQ 2 | Show menu options in the top header section - Home, About, Information Kiosk, Contact and login. |
| REQ 3 | Show related key facts along with respective pictures with animation. |
| REQ 4 | Show ‘About’ section on the next scroll which will give brief about the EmpowerU. |
| REQ 5 | Show the partner’s logo in the next section and provide the link to the partner’s website when clicking on the logo. |
| REQ 6 | Provide all menu options in the footer section with contact details and social media links - Facebook and Twitter. |
| REQ 7 | Show copyright message in the footer section. |
| REQ 8 | Show iTech Mission logo on the right side of the footer. |

### A.2 ABOUT

This module will be accessible by the public users and will present the static text which lists the objectives and functionalities of the EmpowerU Information System.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show About menu on top of the home page. |
| REQ 2 | Open the respective content page when the about option will be selected. |
| REQ 3 | Show the information about the EmpowerU information system and its functionalities. |
| REQ 4 | Show the main menu on the About page to allow navigation back to the home page. |

### A.3 INFORMATION KIOSK

This module will also be accessible to the public users which will provide a repository of the reference literature about the program. It will allow to view and download the reference documents, reports, studies, website links.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the list of Resources available along with an option to download the relevant documents. |
| REQ 2 | Each resource will have its image, name, date, highlights/description and link to download document or visit the website. |
| REQ 3 | Allow searching on the Information kiosk page. |
| REQ 4 | Provide option to sort the list of resources by date of creation or in alphabetical order. |
| REQ 5 | Show the main menu on the Resources page to allow navigation back to the home page. |

### A.4 CONTACT

This module will allow to access the contact details of key team members and stakeholders. It will display the necessary contact information, such as names, positions, phone numbers, email and addresses of the stakeholders and team members. There will be option to post query or contact the support desk.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the EmpowerU MIS logo at the top left corner of the home page. |
| REQ 2 | Show the language option at the top right corner of the home page. |
| REQ 3 | Show the user icon to the right side of the language option. |
| REQ 4 | Show the following menu options in the header section from top left to right – Home, About, Information Kiosk and Contact. |
| REQ 5 | Show Contact banner image on top of the page. |
| REQ 6 | Provide the address, contact number and email of the regional office to the right side of the page. |
| REQ 7 | Provide text field to enter name, email and message to the left side of the page. |
| REQ 8 | Provide submit button at the bottom of the form to send the query message. |

### A.5 LOGIN

This module will be the first access point to get into the user interface and admin application. Users with valid credentials will be able to enter their email and password for authentication. There will be CAPTCHA to determine if an online user is a human and not a bot. The platform administrator will be responsible for creating and managing other users of the platform.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Users with valid role and permission will be able to login by using pre-created email ID and password. |
| REQ 2 | Provide option to enter email Id and Password. |
| REQ 3 | Provide option to enter a captcha for security measure. |
| REQ 4 | Provide a login button to confirm the login process. |
| REQ 5 | Show error message when invalid email ID or password is entered. |
| REQ 6 | The login button should not be enabled until the entered captcha matches the system generated captcha. |
| REQ 7 | Provide option to reset password in case the users have forgotten their password. |

### A.6 AT A GLANCE

This will be the first view after successfully login. The summary data on registration, skillset, grievance, referrals, complaints, cash transfer and information of refugees and host communities by location will be presented in visually oriented interactive visualizations with options to filter data by Programme Cycle, Area and Time period.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show this module by default after successful login and provide option to navigate to other modules of the system. |
| REQ 2 | Show EmpowerU MIS logo at the top left corner of the dashboard. |
| REQ 3 | Provide dropdown to select Programme Cycle, Area and Time period in the header section of the dashboard. |
| REQ 4 | Provide the count of Users and Households using infographics. |
| REQ 5 | Show the summary information on Skillset training, Cash transfer, Referral and Grievance in their own separate widget with the help of various infographics or visualizations. |
| REQ 6 | Provide option to read more about the summary Information presented in each widget. |
| REQ 7 | Provide option to view household details. |
| REQ 8 | Provide option to download each widget separately or download the dashboard data in JPEG format. |

### A.7 REGISTRATION

This module will be the core module of the platform. This module will be accessible to all the users of the platform but the actions will be based on their roles and permissions. This module will allow authorized users to register and manage household data.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the households and their related information by status in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Show list in Pending status by default and provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide the option to perform following action in the Pending state –   1. Provide options to add and edit an element to the list. 2. Provide options to delete an element of the list. When deleting the element, provide the following validations –  * Show a confirmation box to confirm the deletion. * Delete all the information related with the deleted element after getting a confirmation.  1. Provide options to show/hide a household and its details with following validations –  * Show a confirmation box to confirm the action. * Show/Hide the household and its details from everywhere in the application.  1. Provide option to approve and reject a household with following validation –  * Show confirmation box to confirm the approval or rejection. |
| REQ 4 | Allow adding a new element and its details with the following validations-   * Location should be selected and it cannot be blank. * Country Of Origin should be selected. * District should be selected and it cannot be blank. * Settlement should be selected and it cannot be blank. * Sub County/RWC III should be selected and it cannot be blank. * Parish/RWC II should be selected and it cannot be blank. * RWC I/Village/Zone should be selected and it cannot be blank. * Name of respondent should be entered. * Respondent’s telephone number should be entered and it cannot be blank * Respondent’s ID number should be entered. * Household Code should be entered and it cannot be blank. * Enumerator Name should be entered and it cannot be blank. * Supervisor Name should be entered and it cannot be blank. * Date of interview should be selected and it cannot be blank. * Number of household members should be entered and it cannot be blank. * Name of household member should be entered and it cannot be blank. * Gender should be selected and it cannot be blank. * Date of birth known status should be selected and it cannot be blank. * Date of birth should be selected. * Age should be selected. * Birth certificate should be uploaded and it cannot be blank. * Birth registration should be selected. * Relationship with household head should be selected and it cannot be blank. * Mother's living status should be selected. * Father's living status should be selected. * School enrolment should be selected. * Class grade should be selected. * School non-enrolment status should be selected. * Other reason should be entered. * Pregnancy status should be selected. * Expected due date should be selected. * Chronic illness status should be selected and it cannot be blank. * Chronic illness should be selected. * Chronic illness (Other) should be entered. * Chronic illness treatment should be selected and it cannot be blank. * Chronic illness no-treatment reason should be selected * Chronic illness non-treatment reason (Other) should be entered. * Disability status should be selected and it cannot be blank. * Disability status should be selected. * Main Source of Livelihood should be selected and it cannot be blank. * Support on participation in program or service provided by external sources should be selected and it cannot be blank. * Supported By should be selected and it cannot be blank. * Major material of floor of household should be selected and it cannot be blank. * Type of toilet in household should be selected and it cannot be blank. * Meals 2–5-year-old children ate should be selected and it cannot be blank. * No food in past 30 days should be selected and it cannot be blank. * Frequency of no food in past 30 days should be selected and it cannot be blank. |
| REQ 5 | Provide option to save household registration record. |
| REQ 6 | Provide the option to view detailed profile of each household in the action dropdown. |
| REQ 7 | Provide the option to **Verify NIN** of the head of the household in the action column with the following validations –   * The Verify NIN option should be disabled for the participant who do not have an NIN number. * Check ID, First Name, Last Name and Other Name (if exist) from NSR when clicked on Verify NIN * Show message on the successful verification of NIN.   Show message on the failure of NIN verification. |
| REQ 8 | Provide an option to Approve in the action dropdown for each household. |
| REQ 9 | If approved, move the household to approve list from pending and allow performing following action –   * View Profile * Enable/Disable * Cash Transfer * Expenditure * Referral * Grievance * Edit * Delete |
| REQ 10 | Provide an option to filter household data based on various criteria. |

#### **A.7.1 Import Data**

This option will be available to import bulk household details in the database. The Import Data option will allow to:

* Browse and upload data template
* Import the data template into the database

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Provide option to browse and upload data template with data in CSV (Excel) formatted file. |
| REQ 2 | Import the data entry records with the following validations:   * All the validations applied for each field while adding a household using form date entry will also be implied while importing the same details into the database. * Combination of all the columns cannot be duplicate. * Overwrite the old data with new data when a duplicate entry will be found. * The details should not contain any special character. |
| REQ 3 | Show error message “Invalid File Format” if the above requirements are not met. |

#### **A.7.2 Export Data**

This option will be available to download the data file to enter household details for bulk data records at once.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Provide option to download household registration template in CSV (Excel) formatted file. |
| REQ 2 | Provide an option to download the template with required columns. |

### A.8 REFERRAL

This module will be developed to add and manage referrals provided by the service providers to the participants. The service provider will have the option to refer a participant for services like medical, protection and education based on the details during the interactive counseling session.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the grievances by households and their related information in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide the option to perform following action –   1. Provide options to add and edit an element to the list. 2. Provide options to delete an element of the list. When deleting the element, provide the following validations –  * Show a confirmation box to confirm the deletion. |

### A.9 GRIEVANCE

This module will be developed to manage grievances of the participants. There will be option to add a grievance reported by the participants related to the caretaker, service provider or cash transfer. There will also be options to edit, delete and update the status of the grievance. The program officer will have the option to register the grievance and its details. The grievance view will be restricted based on the user roles. The users with appropriate role will have options to edit, delete and update the status of the grievance. The services including the cash transfer will depend on the status of an open grievance.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the grievances by households and their related information in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide the option to perform following action –   1. Provide options to add and edit an element to the list. 2. Provide options to delete an element of the list. When deleting the element, provide the following validations –  * Show a confirmation box to confirm the deletion. |

### A.10 CASH TRANSFER

This module will allow to manage and track the cash transfer process of registered households based on their registration and status in the respective programs. The cash transfer flow will follow a step-by-step process that will include –

* First Verification
* Second Verification
* Approval
* Payment

Each cash transfer step will go through an approval process by authorized users.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the households and their payment information by program cycle in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide dropdown option to select Installment. |
| REQ 3 | Provide dropdown option to select Cash Transfer Status. |
| REQ 4 | Provide options to search, filter and navigate the existing list. |
| REQ 5 | Provide option to Approve/Disapprove payment of household by authorized users. |
| REQ 6 | Provide option to download cash transfer template in CSV (Excel) formatted file. |
| REQ 7 | Provide option to browse and upload data template with data in CSV (Excel) formatted file. |
| REQ 8 | Import the data entry records with the following validations:   * All the validations applied for each field while adding a household using form date entry will also be implied while importing the same details into the database. * Combination of all the columns cannot be duplicate. * Overwrite the old data with new data when a duplicate entry will be found. * The details should not contain any special character. |
| REQ 9 | Show error message “Invalid File Format” if the above requirements are not met. |

### A.11 EXPENDITURE

This module will allow authorized users to manage household expenditures. It will track expenses from funds received in installments through the EmpowerU program. It will have options to add, edit, and view detailed records of household spending.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the households and their expense information in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide options to search, sort and navigate the existing list. |
| REQ 4 | Allow adding a new element and its details with the following validations-   * Household should be selected and it cannot be blank. * Amount received from program should be entered. * Amount spent on education should be entered. * Amount spent on healthcare should be entered. * Amount spent on healthcare should be entered. * Is household has started any income-generating activities should be selected should be selected as Yes or No. * Income generating activity type should be selected. * Is household has Acquired Assets should be selected as Yes or No. * Type of Asset should be selected. |

### A.12 IMPORT LOG

The module will allow authorized users to view the import logs that are generated after the import process. Every bulk import operation of data will generate a log, which will report on the success and failure of the imported data records. User can access these logs to gain insights into the outcome of the import process, including details about successful imports and any encountered failures. While the administrator will be able to view and manage the all-import logs, the data entry user will only be able to manage their own import logs. Option will be available to download or delete the log.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the list of logs by pages along with pagination options to customize the view. |
| REQ 2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Show the list of Log name, date and time, user and status (successful or failed). |
| REQ 4 | Provide the option to view the complete details of the log including the number of records successfully imported and failed. |
| REQ 5 | Provide the option to download and delete an existing log. |

1. **ADMIN**

The admin application will comprise of the following modules and sub-modules.

1. Area
2. Classification
3. Program Cycle
4. Access Control

* Role
* User

1. Resource
2. Report

### B.1 ADMIN

This module will consist of six sub-modules – Area, Classification, Program Cycle, Access control, Resource and Report. This module will allow the administration and control of the core and foundational elements of the application Below is the brief description of the submodules of the admin module.

#### **B.1.1 AREA**

This sub-module will allow authorized users to manage the geographical areas. User will have the options to export and import area templates, add, edit, view, search, sort and show/hide the geographical areas.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the areas in a table grid by pages and provide option to set the number of areas per page. |
| REQ 2 | Provide options to search, sort and navigate the existing area list. |
| REQ 3 | Provide options to add and edit an area and its details. |
| REQ 4 | Provide option to delete an area and its details with following validations:   * Show the count of associated records and ask the user to confirm deletion. * Delete all the associated records after confirmation. |
| REQ 5 | Provide option to show/hide an area and its details with following validations:   * Show/Hide the area and its details in both the applications. * Show/Hide all the associated records. * Do not allow data entry and data import for the hidden area. |
| REQ 6 | Allow adding a new area and its details with the following validations:   * Parent Areais mandatory when adding area under a root area. * Area ID is mandatory and cannot be blank.   *A standard ISO format for ID can be followed for example UGA is the Area ID for Uganda which is referred to as Level 1. UGA001 is the Area ID for the first sub-national area of Uganda which is referred to as Level 2.*   * Show Map Areas will allow to refer to the Area ID from the pre-uploaded Map file data. * Area Nameis mandatory and cannot be blank. * Area Level should be auto-generated based on the Parent Area selected. For no Parent Area the Area Level should be 1. |
| REQ 7 | Provide an Add button to confirm adding a new area and its details. |

**Export Area**

A bulk area list can be imported in the database. The Export Area option allows to:

* Download empty area template
* Download area template with data (if data exists)

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Provide option to download an empty area template in XLSX/XLS (Excel) formatted file. |
| REQ 2 | Provide an option to download the template with the columns: Area ID, Area Name, Area Level, Parent Area ID.   * Parent Area ID refers to the Area ID of the parent. |

**Import Area**

The bulk area list can be imported in the database. The Import Area option allows to:

* Browse and upload area template
* Import the area template into the database

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Provide option to browse and upload area template with data in XLSX/XLS (Excel) formatted file. |
| REQ 2 | Import the area list with the following validations:   * All the four columns are mandatory and their sequence has to be AreaID, AreaName, AreaLevel, AreaParentID. * AreaID and AreaName are mandatory and cannot be blank. * AreaID and AreaName cannot be duplicate. * AreaID and AreaName cannot contain any special character. * AreaLevel should be numeric. * AreaParentID should be listed in the AreaID column. |
| REQ 3 | Show error message “Invalid File Format” if the above requirements are not met. |

#### **B.1.2 CLASSIFICATION**

This sub-module will allow authorized users to manage classifications within the database. User will have options to add new classifications, edit, delete, update status, and show the existing list of classifications based on their respective classification type.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Provide option to select classification from following options:   * Referral * Grievance * Vulnerability * Relationship * Skill |
| REQ 2 | Show the existing list of records and related information in a tabular grid based on the selected master list. Provide option to view the list by pages along with pagination option to customize the view. |
| REQ 3 | Provide options to search, sort and navigate the selected classification. |
| REQ 4 | Provide options to add and edit element to each classification. |
| REQ 5 | Allow adding a new record in selected classification and its details with the following validations:  In case adding a new record under selected classification, provide the following entries and validations –   * Provide option to enter new element. * No duplicate element should be added. |
| REQ 6 | Provide the Add button in the add new element window to confirm adding the element. |

#### **B.1.3 PROGRAM CYCLE**

This sub-module will allow the authorized users to add and manage the program cycles of the EmpowerU program. Below is the functional requirement of this module.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of program cycles in a table grid by pages and provide option to set the number of records per page. |
| REQ 2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide options to add and edit a program cycle and its details. |
| REQ 4 | Provide option to delete a program cycle and its details with following validations:   * Show the count of associated records and ask the user to confirm deletion. * Delete all the associated records after confirmation. |
| REQ 5 | Provide option to show/hide a program cycle and its details with following validations:   * Show/Hide the program cycle and its details in both the applications. * Show/Hide all the associated records. * Do not allow data entry and data import for the hidden program cycle. |
| REQ 6 | Allow adding a new cohort and its details with the following validations:   * Name/Number of program cycle should be entered and cannot be blank. * Duration of program cycle should be entered in MM/YYYY-MM/YYYY format and cannot be blank. * Maximum number of beneficiaries allowed should be entered and cannot be blank. * Total budget should be entered and cannot be blank. * Installments amount per beneficiary should be entered and cannot be blank. * Total number of installments should be entered and cannot be blank. * Last Date of Registration should be entered and cannot be blank. |
| REQ 7 | Provide an Add button to confirm adding a new program cycle and its details. |

#### **B.1.4 ACCESS CONTROL**

This sub-module will further consist two sub-modules – Roles and Permission and User. This sub-module which will allow authorized users to manage the roles and permissions and users of the application. You will have the options to add, edit, view, search, sort and show/hide the elements.

###### **B.1.4.1 Role**

This sub-module will allow to create and manage user role. You will have the options to add, edit, delete and show/hide the elements. Below is the functional requirement of this module.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the roles and their permissions in a tabular grid. Provide option to view the list by pages along with pagination option to customize the view. |
| REQ.2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide option to add and edit an element. |
| REQ.4 | Provide an option to show/hide an existing element. |
| REQ 5 | When adding a new element provide, the following entries and validations -   * Role should be entered and cannot be blank. * Select permissions from the list of permissions. At least one permission should be selected and cannot be blank. |
| REQ 6 | Provide the Add button in the add new element window to confirm adding the element. |

###### **B.1.4.2 User**

This sub-module will allow administrator to create and manage the users of the system and assign roles and permissions to them. The system administrator will be the first user of the system and will be able to create and manage further users. Below is the functional requirement of this module.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the users in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide options to add and edit an element to the list. |
| REQ 4 | Provide an option to show/hide an existing element. |
| REQ 5 | In add new element show the option to enter   * User Name should be entered and cannot be blank. * Select one of the Area from the list. * Select one of the Role from the below options –   + Specialist   + Program Manager   + Program Coordinator * Email ID should be entered and cannot be blank. * Passwords should be entered and cannot be blank. |
| REQ 6 | Provide the Add button in the add new element window to confirm adding the element. |

#### **B.1.5 RESOURCE**

This submodule will allow authorized users to manage create and manage the reference materials. It will allow to create and manage the resources including the relevant URL, Video links or to download reference documents. These resources will be available to access by the public on the home page.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the resources and related information in a tabular grid. Provide option to view the list by pages along with pagination option to customize the view. |
| REQ 2 | Provide options to search, sort, and navigate the existing list. |
| REQ 3 | Provide option to add, edit and delete an element. |
| REQ 4 | Provide an option to show/hide an existing element. |
| REQ 5 | Allow adding a resource with the following validations -   * Title should be entered and cannot be left blank. * Description should be entered with maximum 200 characters allowed and cannot be left blank. * Type should be selected from the list with options of Document or URL with default to be set as Document. * If the Type selected is Document, then allow to browse and upload the document file. * If the Type selected is URL, then allow to Enter URL and it cannot be left blank. * Allow to browse and upload an Image associated with the resource. If no image is uploaded show a default image. |
| REQ 6 | Provide the Add button to confirm adding the element. |

#### **B.1.6 REPORT**

This sub-module will allow to authorized users to generate customized reports based on selected parameters. Below is the functional requirement of this module.

**Functional Requirements**

|  |  |
| --- | --- |
| REQ 1 | Show the existing list of the report templates in a table grid by pages along with pagination options to customize the view. |
| REQ 2 | Provide options to search, sort and navigate the existing list. |
| REQ 3 | Provide options to add and edit an element of the list. |
| REQ 4 | Provide option to Execute a report template. |
| REQ 5 | Provide option to view the status of execution of the report. |
| REQ 6 | Provide option to download report in XLSX and CSV format. |
| REQ 7 | Provide option to add a report template. |
| REQ 8 | Provide option to enter report name. |
| REQ 9 | In add new element show the option to enter –   * Report Name should be entered and cannot be blank. * Report Type should be selected as Registration, Skillset, Grievance, Referral or Cash Transfer and cannot be blank. * Column(s) of the dataset should be selected and it cannot be blank. * There should be option to filter data by selected columns |
| REQ 10 | Provide the Add button in the add new element window to confirm adding the element. |

## 3.2 Non-Functional Requirements

### 3.2.1 Security Requirement

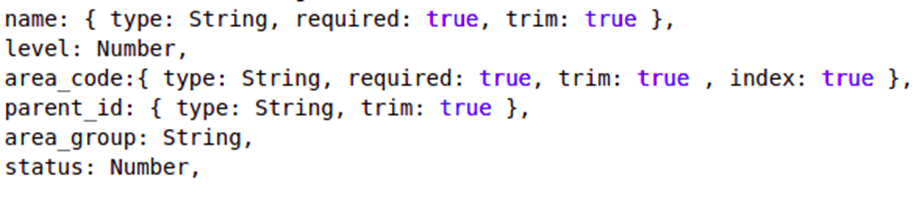
The application will enforce HTTPS for all data transmissions and will implement data encryption at rest and in transit. It will be ensured that all vulnerabilities of the web application are identified and corrective measures are taken. The web application environment including the scripting language, web server software, and the operating system are mentioned in this document. Configuration and coding will be done in the applications to manage the identified vulnerabilities. A manual testing will be conducted on the applications to ensure that no known vulnerabilities exist in the system. Below are some of the known vulnerabilities that will be handled in the applications:

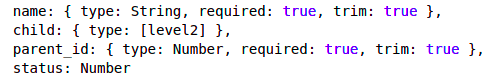
* Displaying the passwords between client and server in clear text
* Session hijacking
* Cross-site request forgery (CSRF) attack to load a page containing malicious request
* Upload malicious (.exe) file
* Brute force attack
* Not maintaining audit trails
* Runtime/Server error
* View the authenticated page from the cache of the browser
* Server version discloser in header response

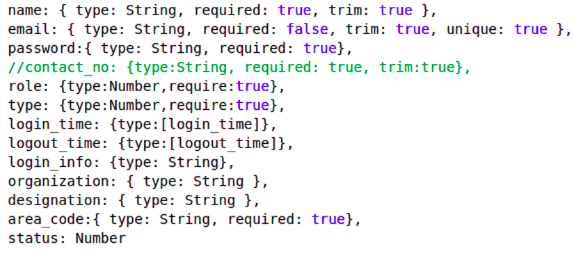
### 3.2.2 Usability and External Interface Requirement

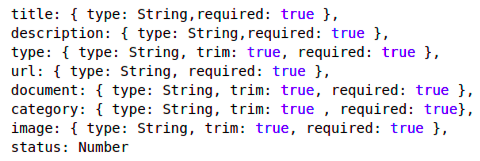
The application will be compatible with the commonly used internet browsers including Google Chrome, Mozilla Firefox, Microsoft Edge and Apple Safari. The application will be developed using open-source software development technologies and frameworks. The application will be easy to navigate and will have a menu-driven approach to access the modules based on the user permissions. The application will be responsive and compatible with both desktop and mobile devices. It will be modern design principles for accessibility. There will be no specific hardware interface requirements as the application will be cloud-based. The application will be integrated with the underlying database via RESTful APIs and will allow exporting data in common formats like CSV and Excel. The applications will use REST APIs for communication between the frontend and backend.

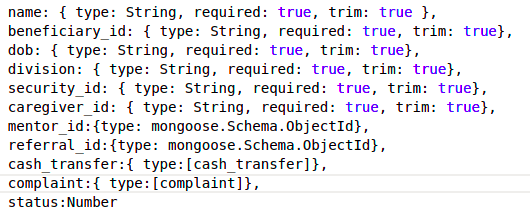
# Annexure A: Data Dictionary

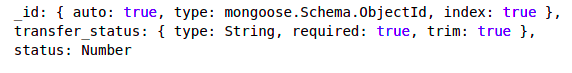
**Manage Areas**

**Master Lists**

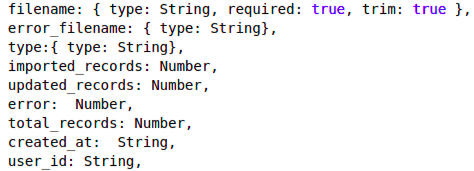
**Users**

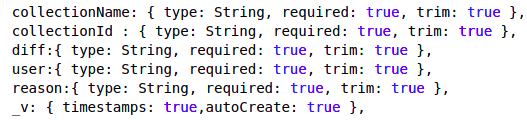
**Resources**

**Beneficiary**

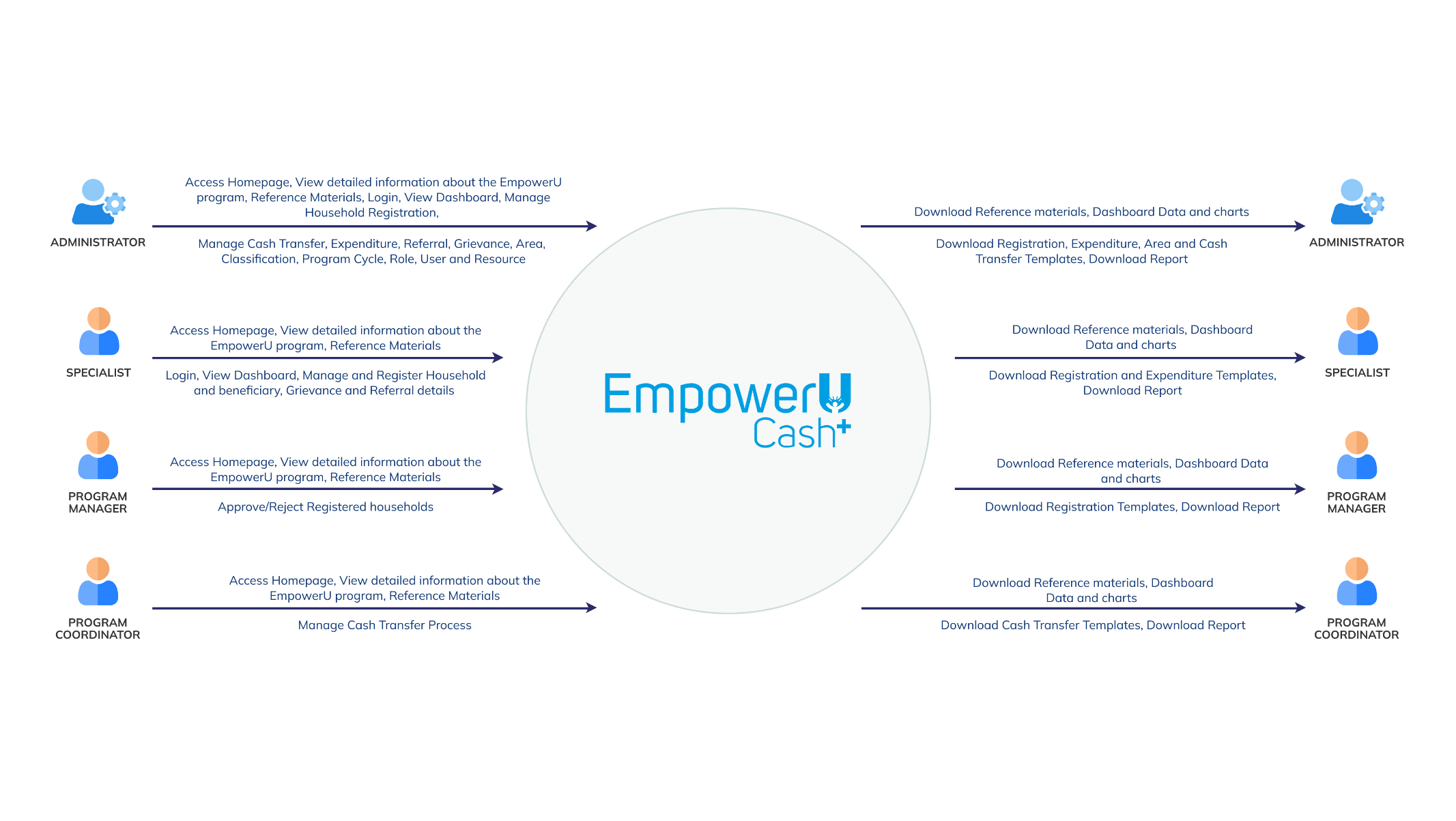
**Cash Transfer**

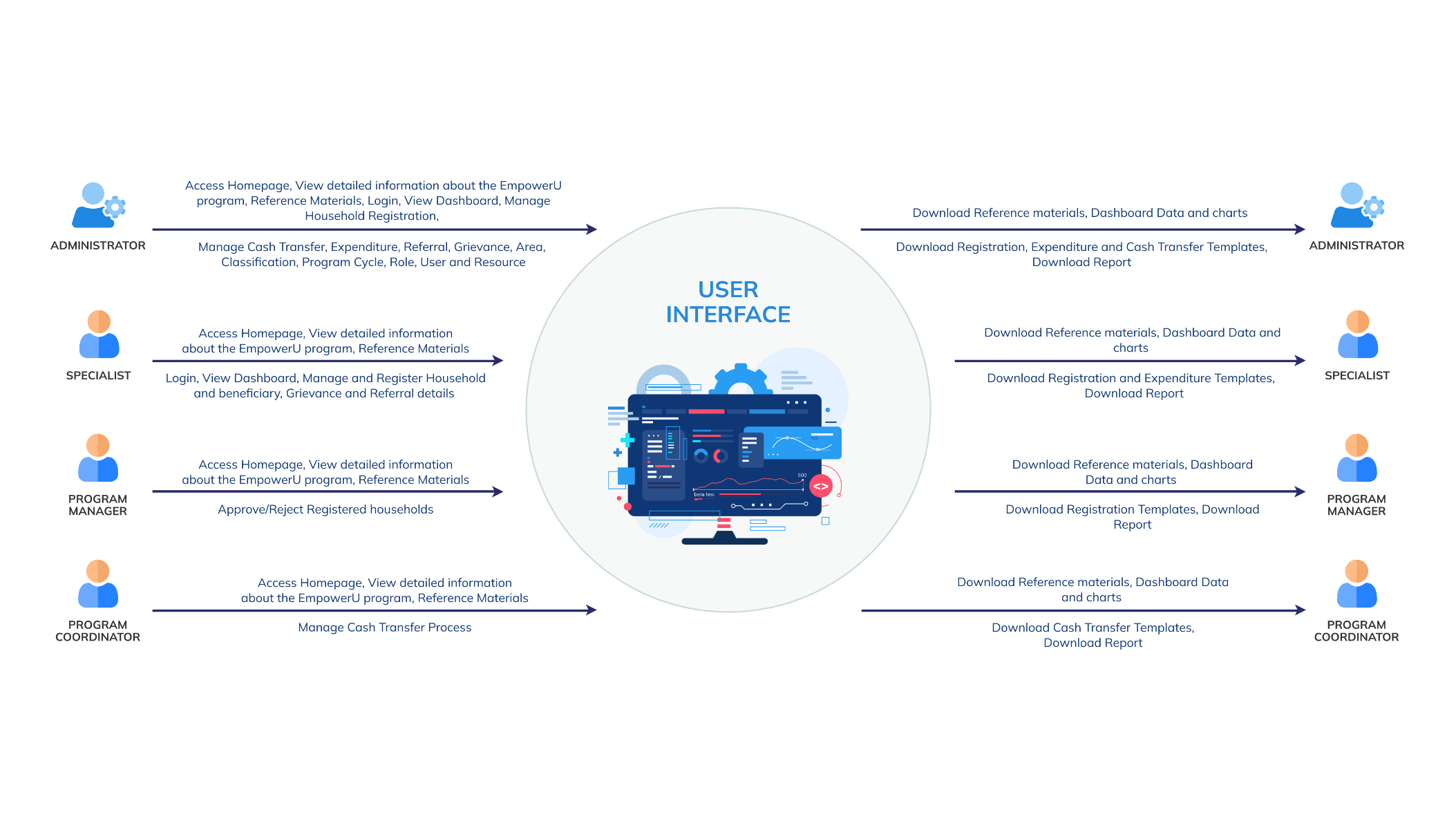
**Complaint**

**Logs**

**Audit**

# Annexure B: Data Flow Diagrams (DFD)

**Level** **0**

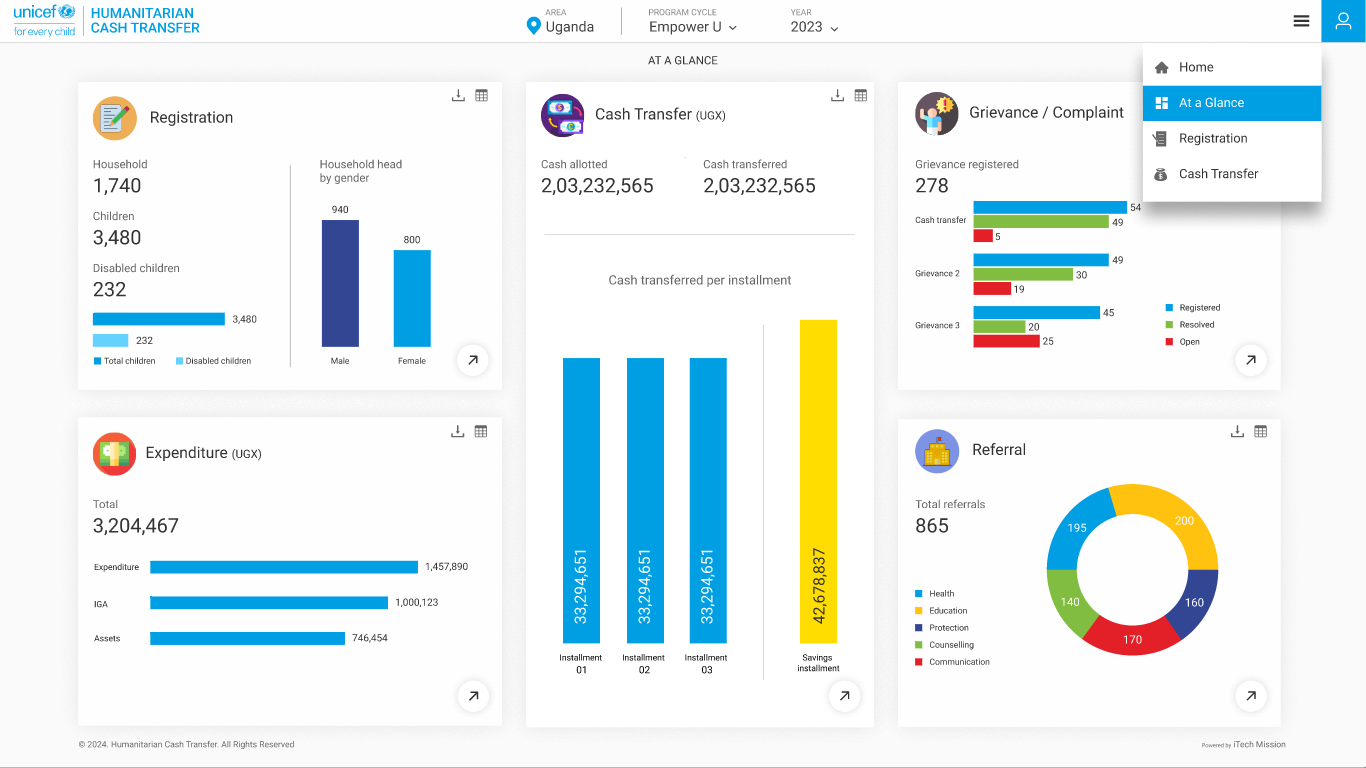
**Level** **1 -User Interface**

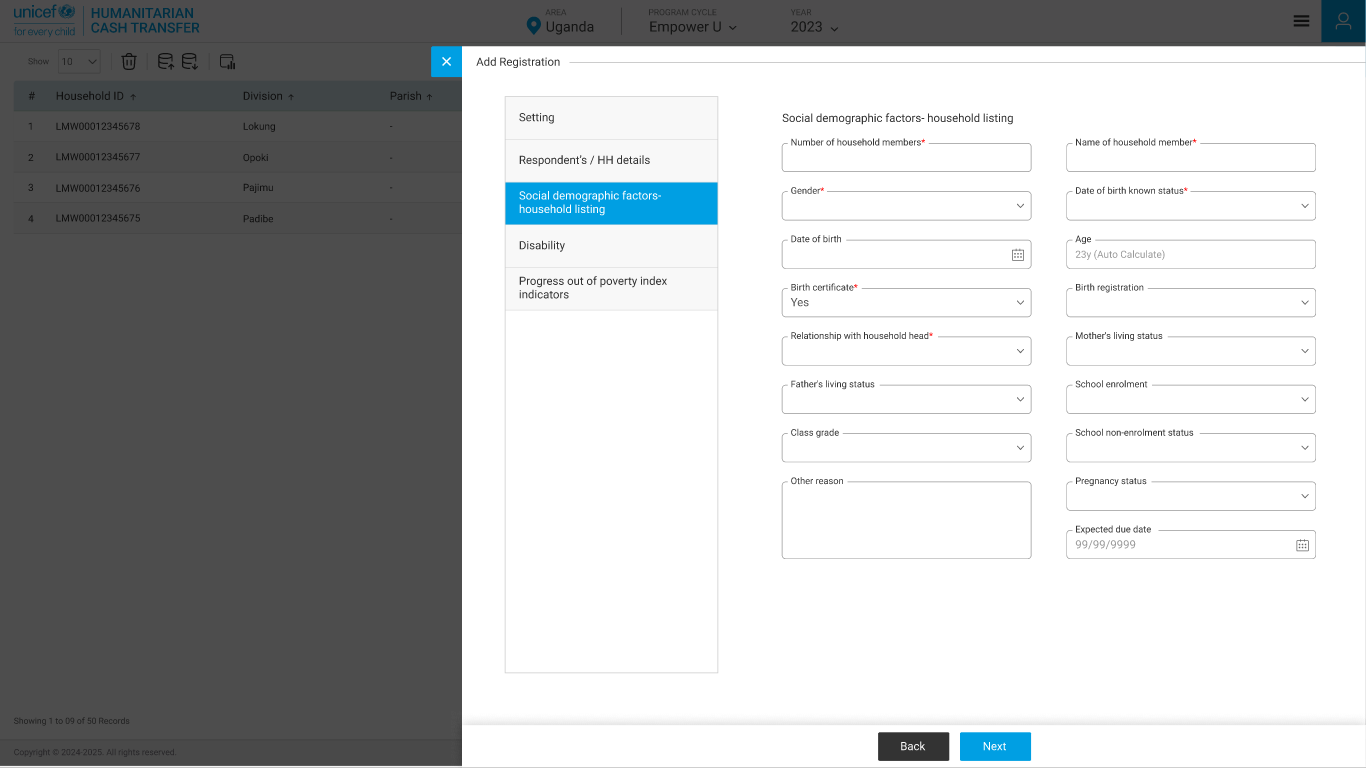
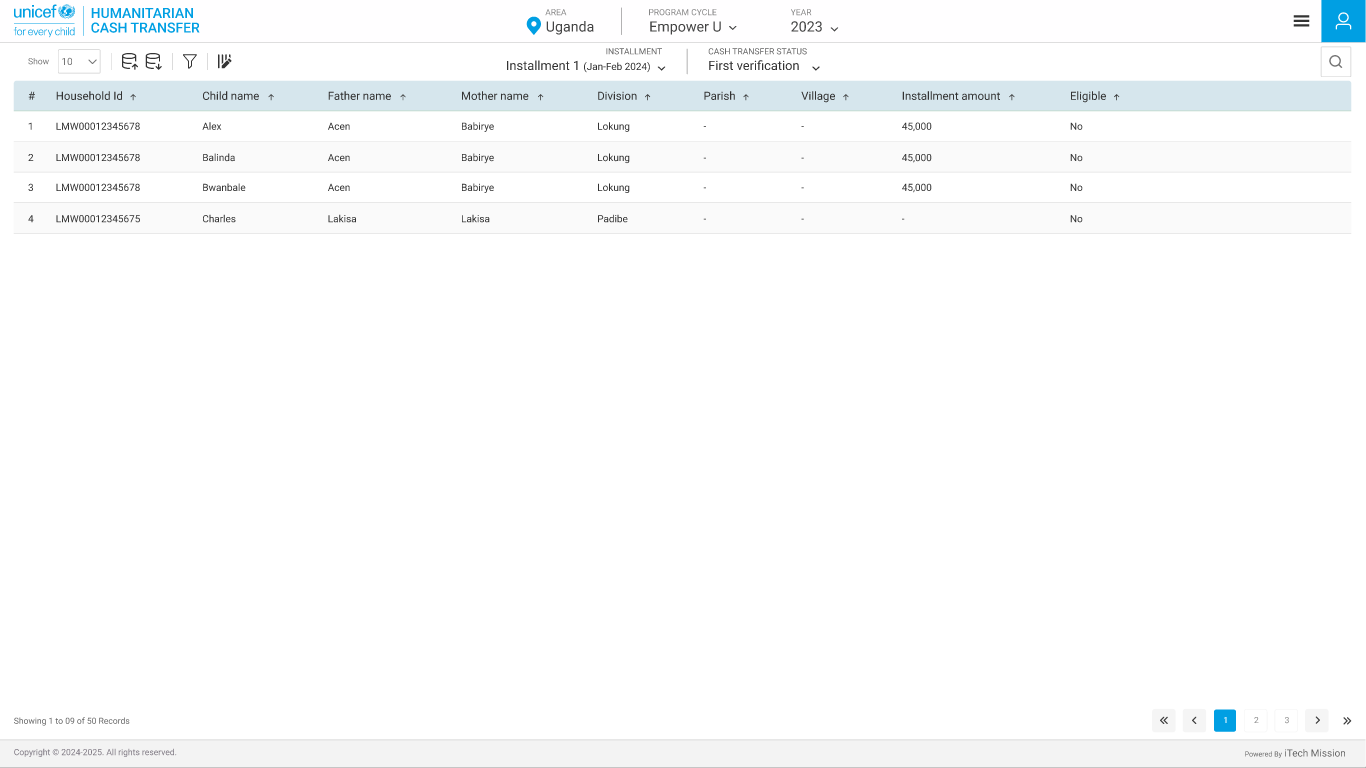
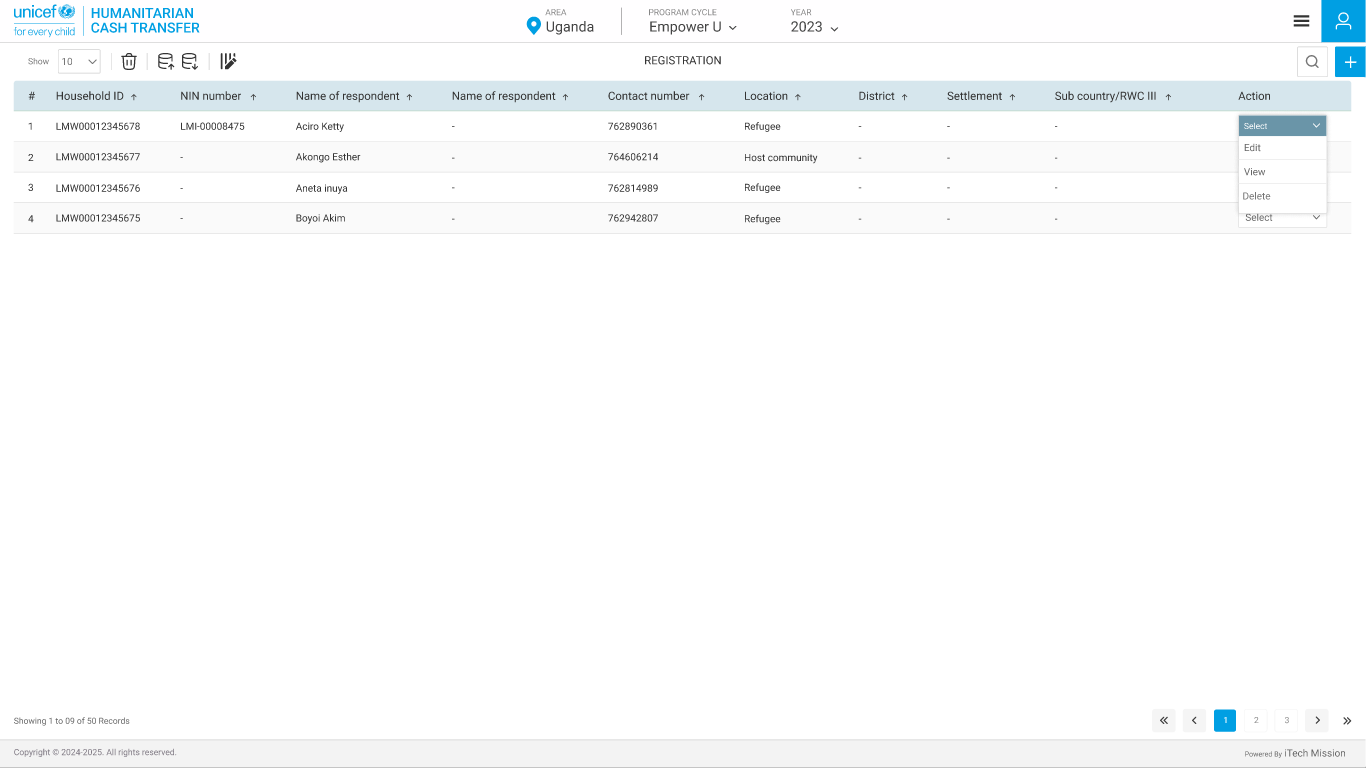
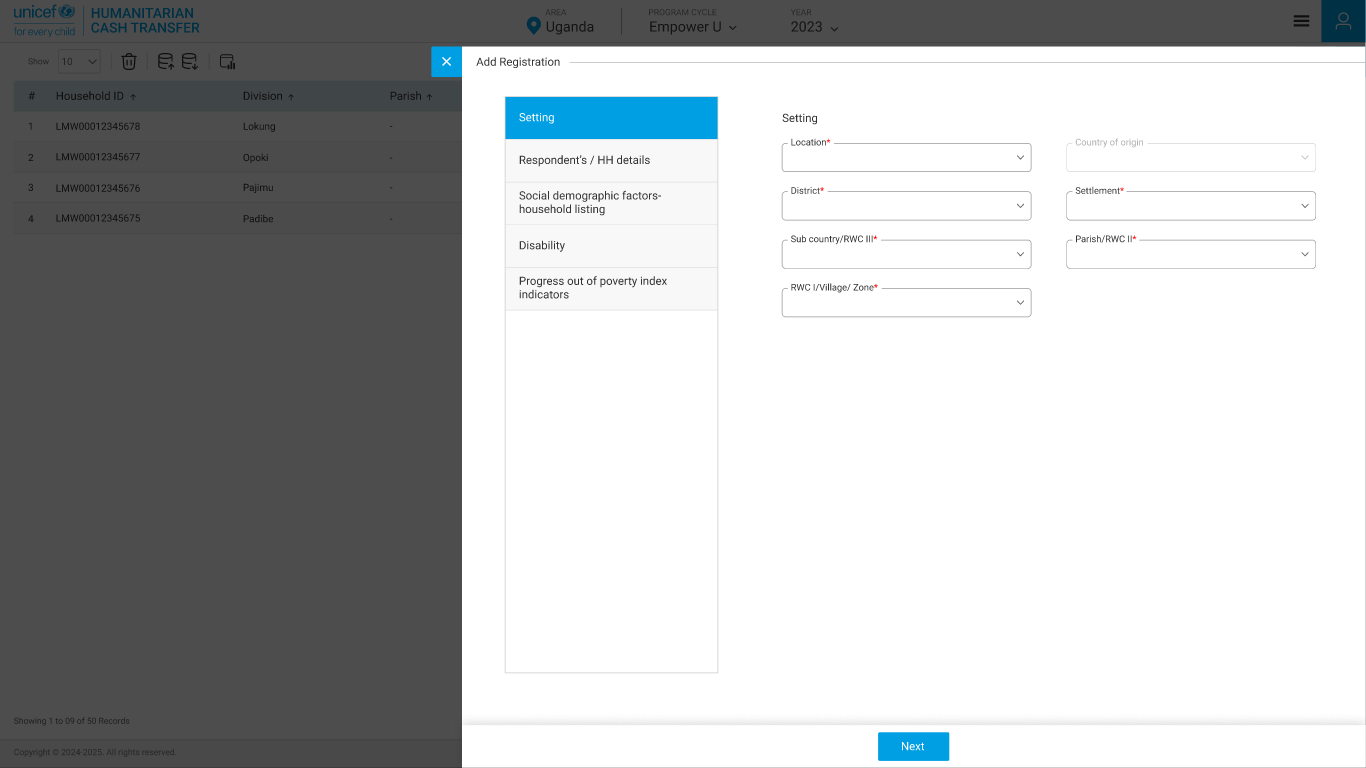
**Level 2 – Admin**

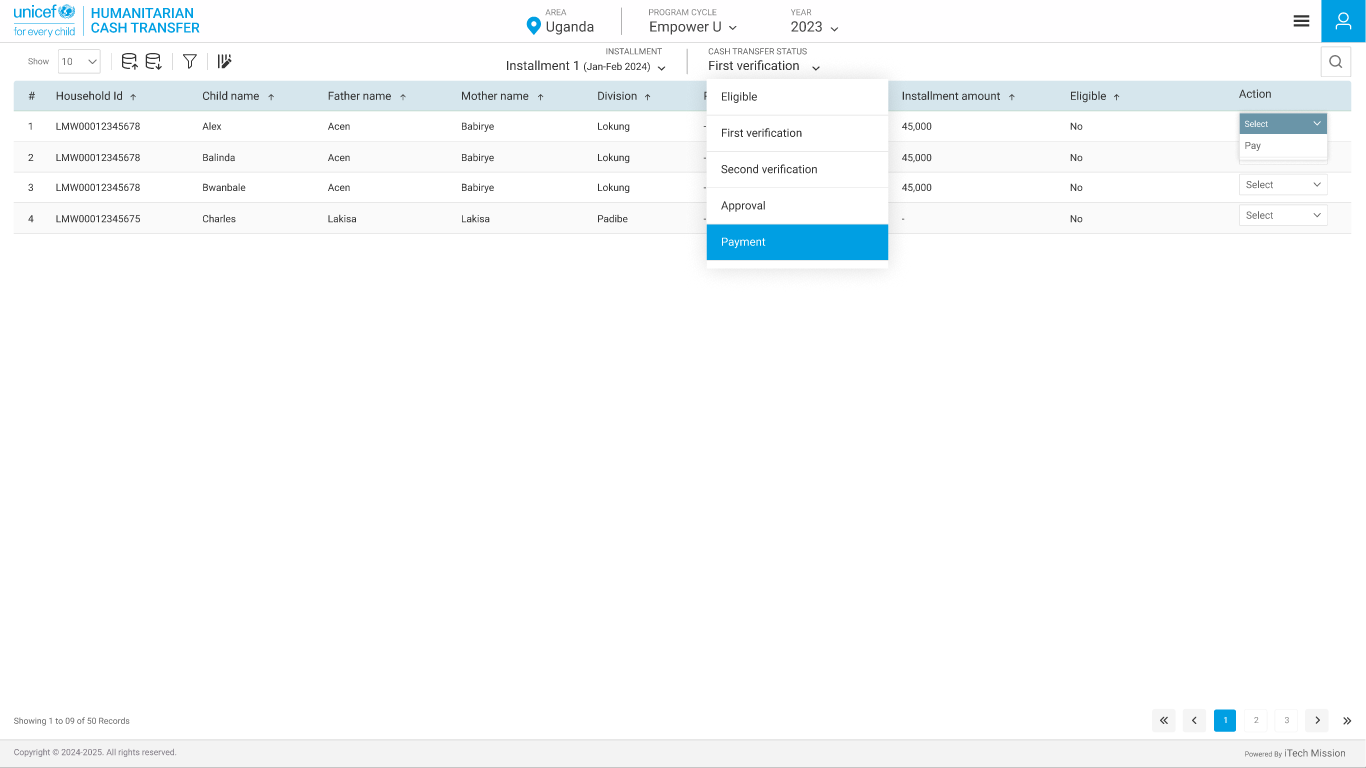
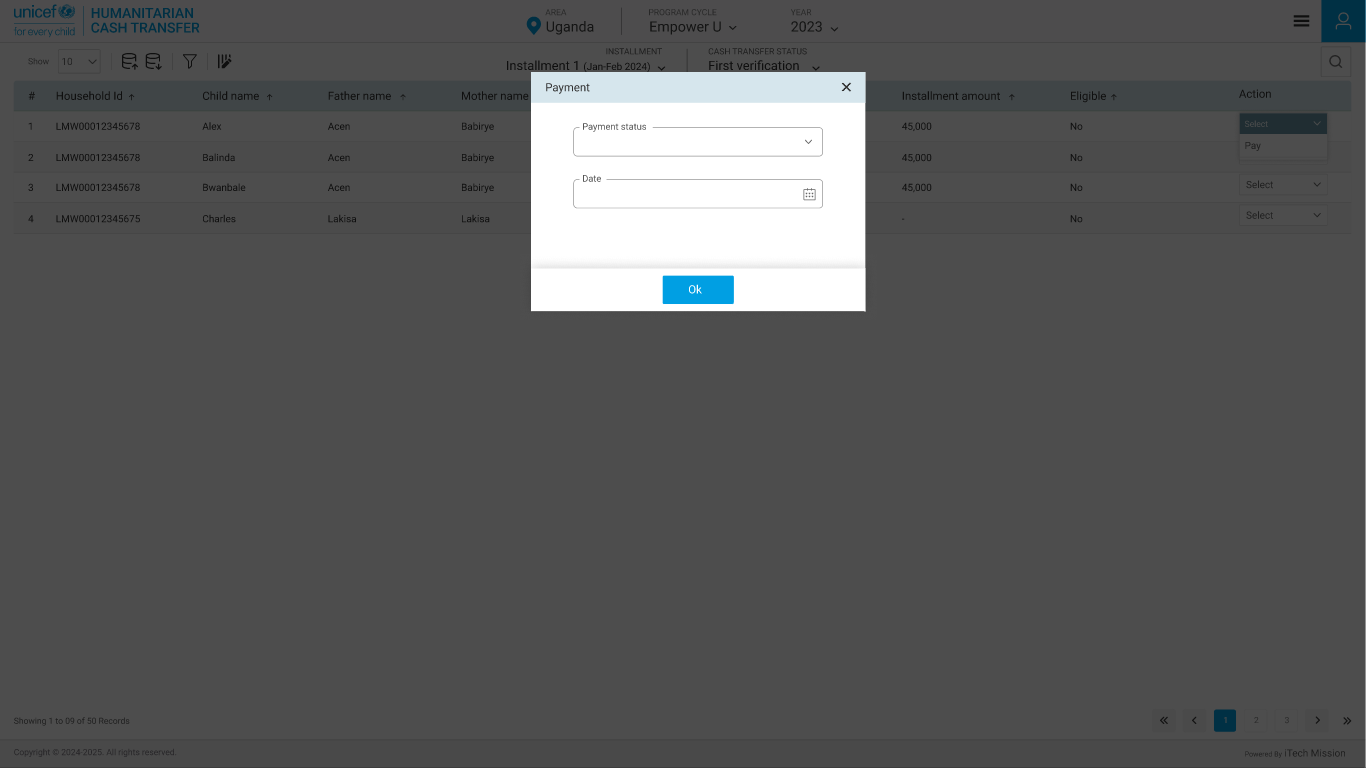
# Annexure C: System Architecture

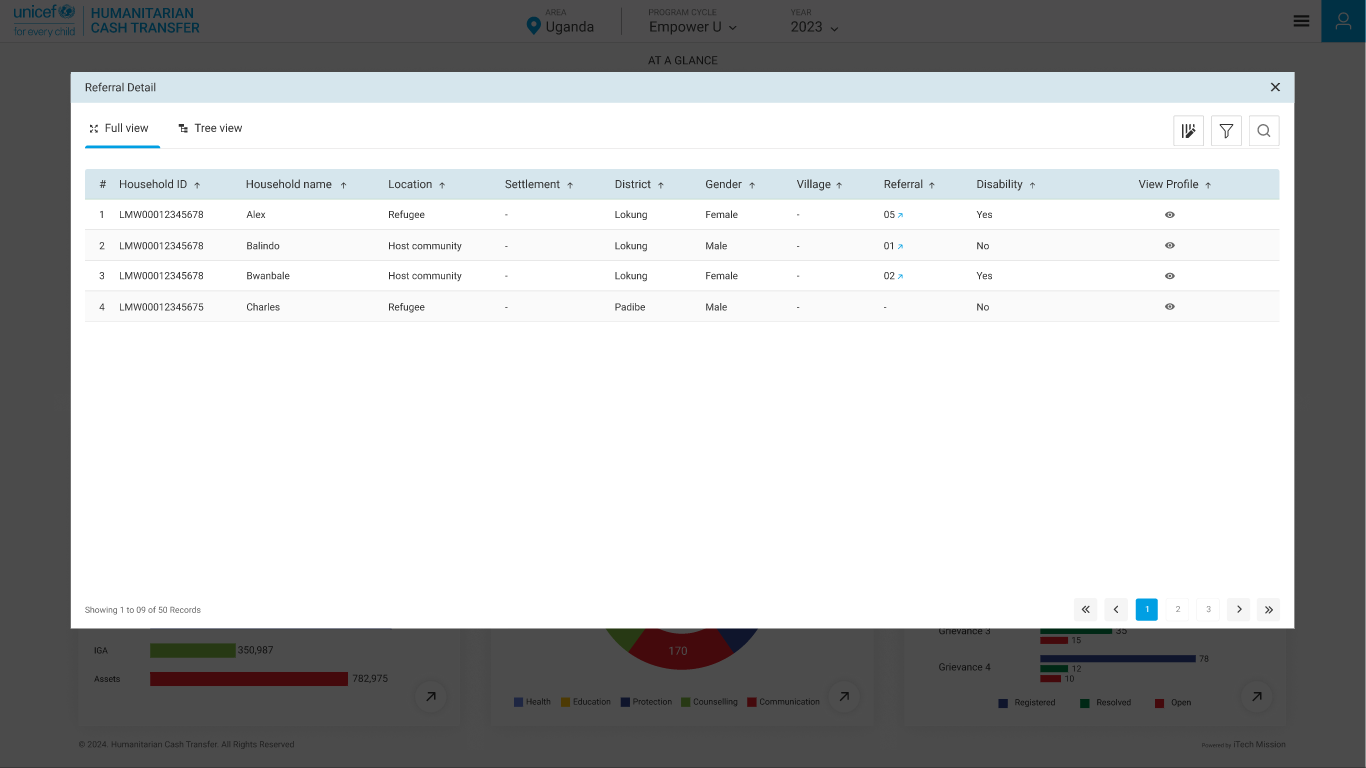
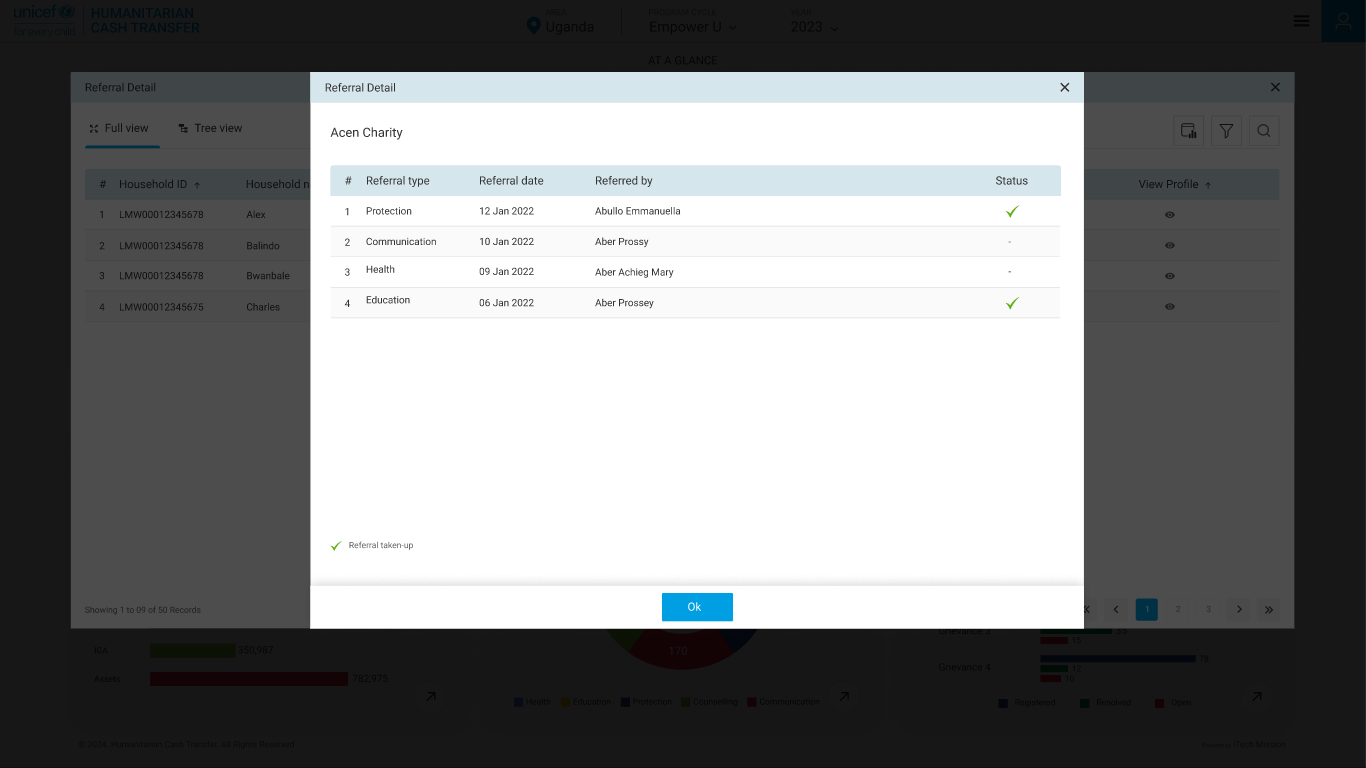
# Annexure D: Software Architecture

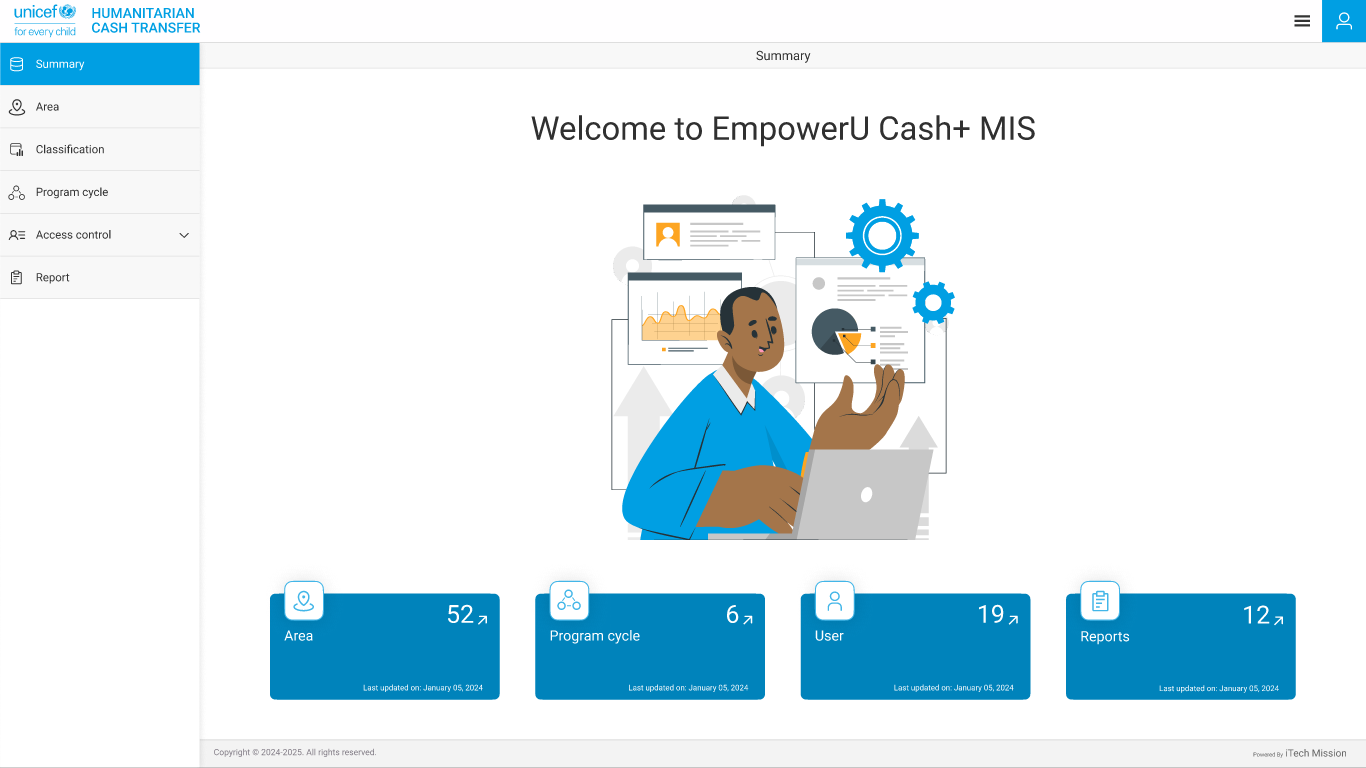
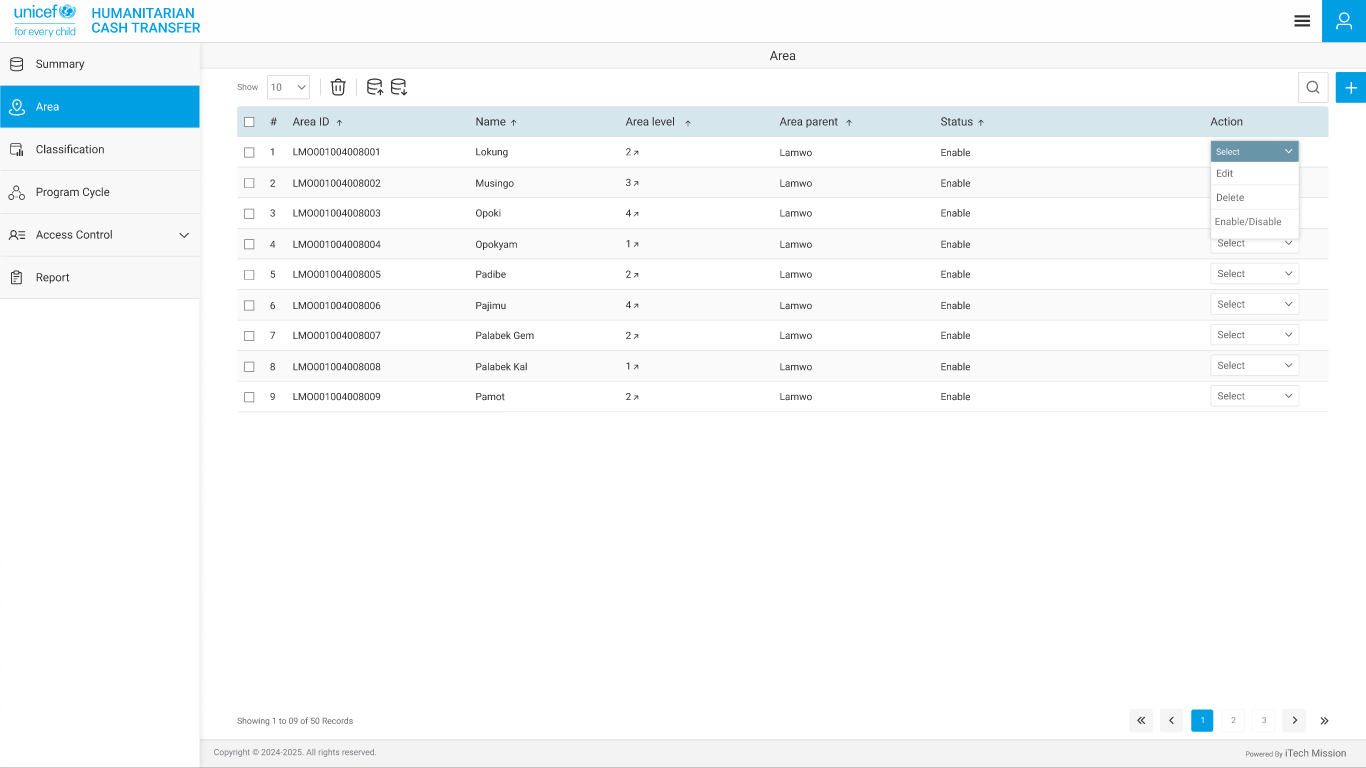
# Annexure F: System Screens

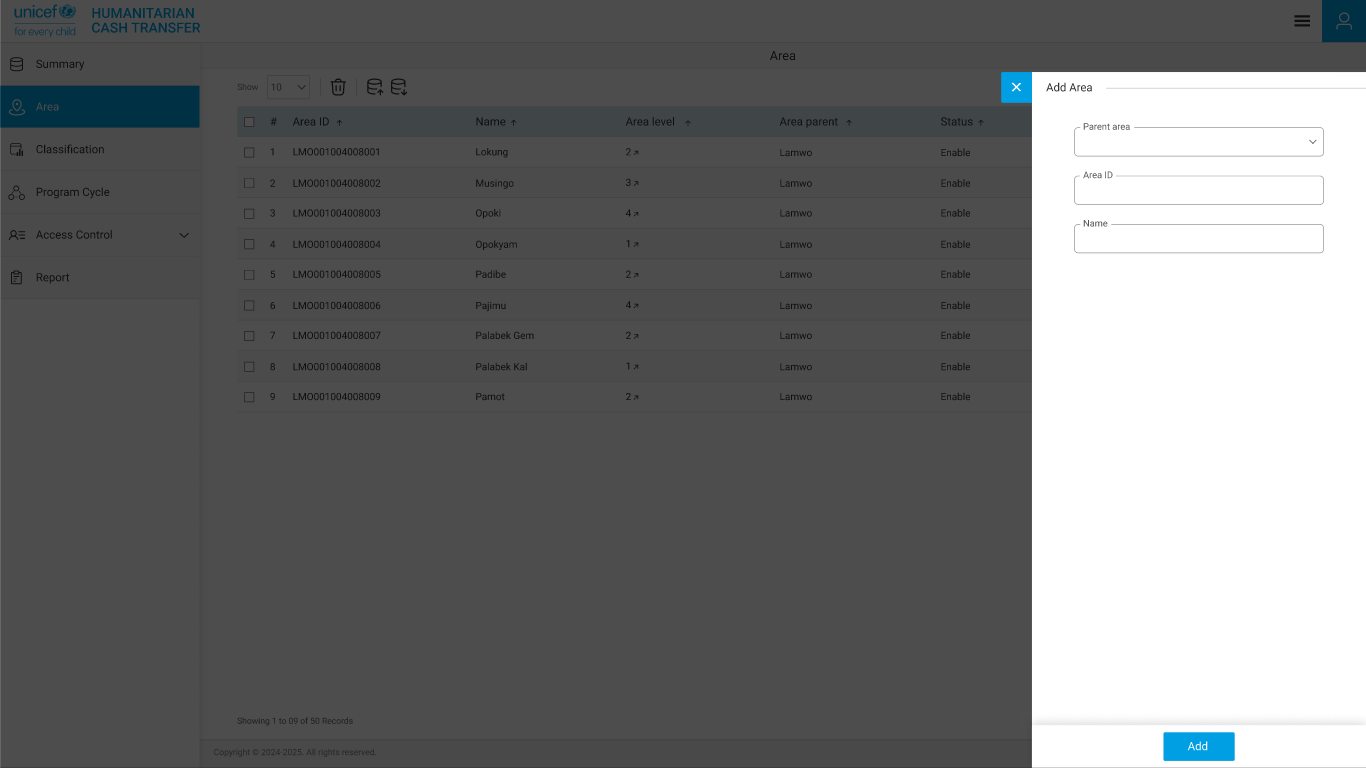
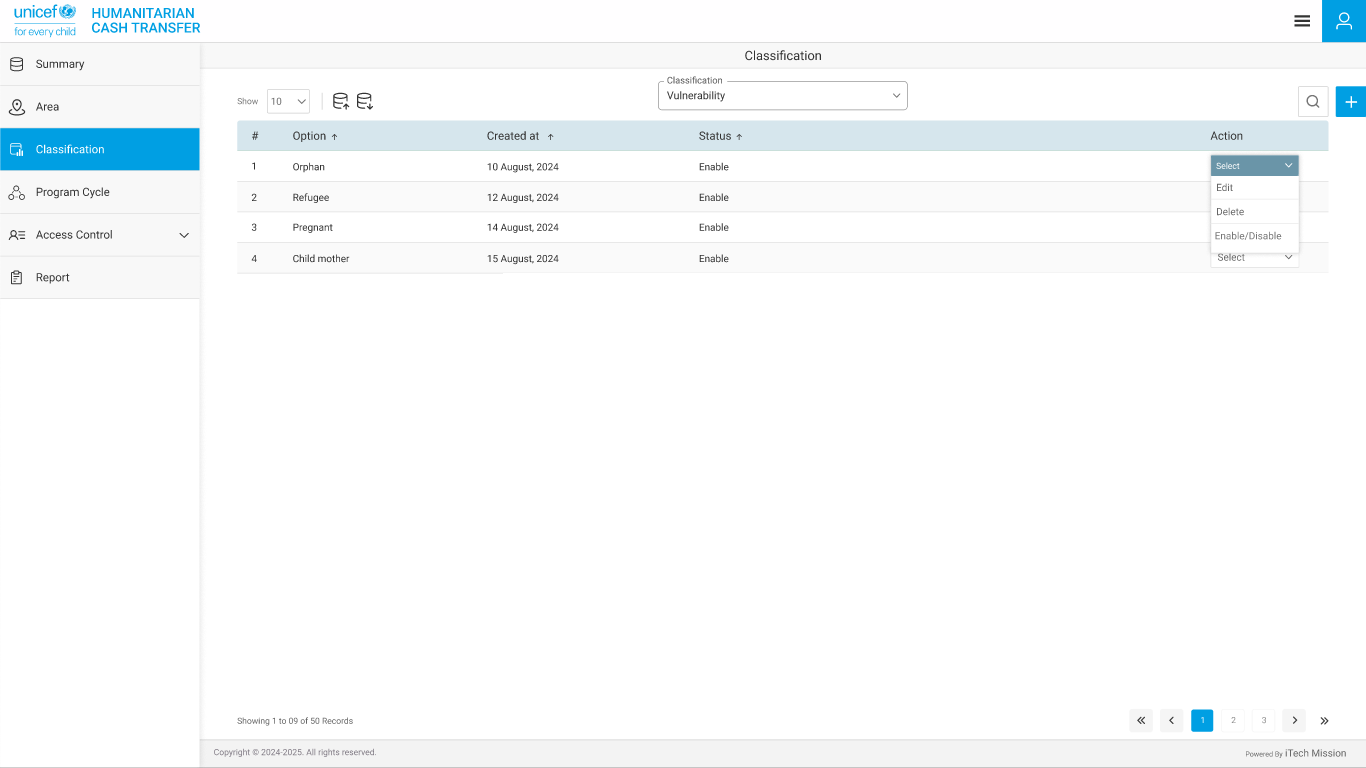


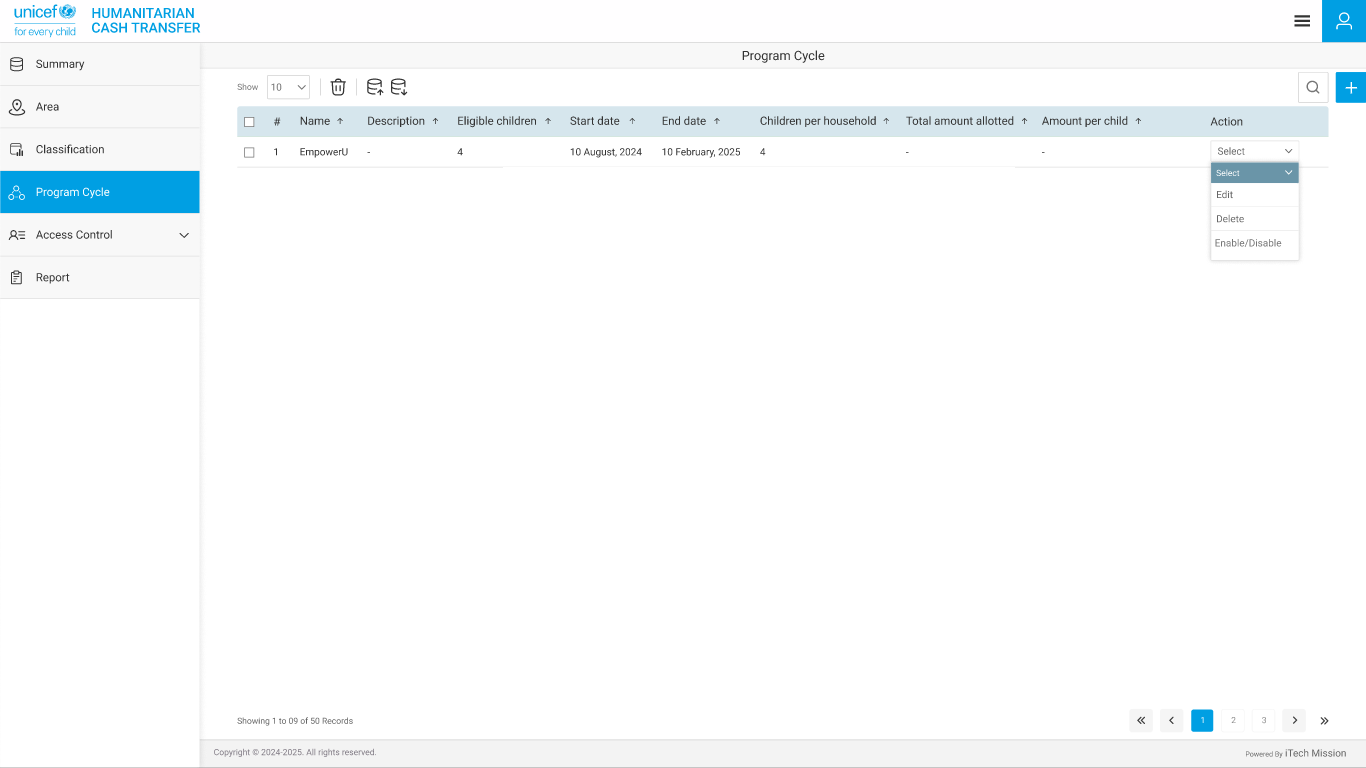
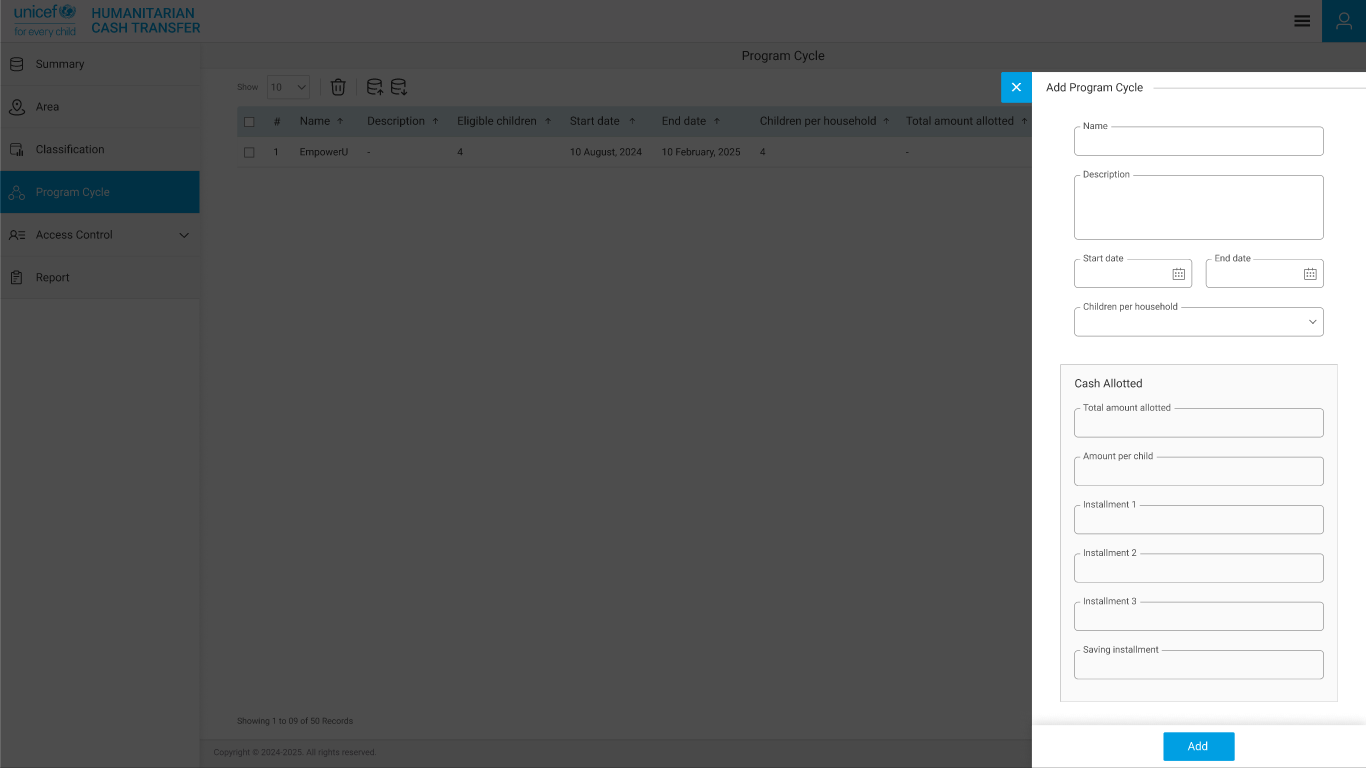












\*\* End of the Document \*\*