

IHub SD Learnership

Requirements Document

*Problems.Space*

Team Members:

Date Submitted: To be confirmed

**TABLE OF CONTENTS**

[1 SYSTEM OVERVIEW 1](#_Toc99700793)

[1.1 Problem Description 1](#_Toc99700794)

[1.2 Project Assumptions 1](#_Toc99700795)

[2 FUNCTIONAL REQUIREMENTS 2](#_Toc99700796)

[2.1 Business Use Case Model 2](#_Toc99700797)

[2.2 Use Case Glossary 3](#_Toc99700798)

[2.3 Use Case Narratives (User Stories) 5](#_Toc99700799)

[2.3.1 Package A: [Student] 5](#_Toc99700800)

[2.3.2 Package B: [Institution] 7](#_Toc99700801)

[2.3.3 Package C: [Funder] 9](#_Toc99700802)

[3 NON-FUNCTIONAL REQUIREMENTS 13](#_Toc99700803)

[3.1 Interface Requirements 13](#_Toc99700804)

[3.2 Performance Requirements 14](#_Toc99700805)

[3.3 Security Requirements 14](#_Toc99700806)

[3.4 Operational Requirements 14](#_Toc99700807)

[4 DATA REQUIREMENTS 15](#_Toc99700808)

[4.1 Preliminary List of data and attributes 15](#_Toc99700809)

[4.2 Domain Class Diagram 17](#_Toc99700810)

[5 Preliminary Schedule of Responsibilities 18](#_Toc99700811)

[6 Researching the Requirements 19](#_Toc99700812)

[10 User Interview Questions 19](#_Toc99700813)

# SYSTEM OVERVIEW

## Problem Description

Problems.space is a groundbreaking online support community, redefining the way individuals navigate life's challenges. With a user-friendly interface that accommodates both anonymous and identified posts, the platform encourages users to share a diverse array of problems, from relationship intricacies to profound addiction challenges and school related matters. Its dual-pronged approach combines community-driven support with access to evidence-based solutions curated by experts, providing a comprehensive resource for users seeking resolutions.

The platform fosters multifaceted user interaction, allowing individuals to engage in dynamic exchanges with solution providers. Users can express gratitude, discuss solution effectiveness, and establish ongoing connections for continued support, emphasizing the importance of building lasting connections within the community. Problems.space also integrates a meticulous feedback loop, enabling users to rate solutions and provide constructive feedback, ensuring the platform remains a reliable source of valuable and impactful help.

Beyond serving as a diverse space for connections, Problems.space stands as a gateway to evidence-based advice, enhancing users' ability to find effective solutions. The platform transforms the act of seeking and providing help into a rewarding experience, creating a positive and uplifting atmosphere. It embodies digital altruism, where users actively contribute to each other's well-being, making Problems.space a transformative space where challenges find nuanced and comprehensive solutions. Welcome to a platform where the journey of self-discovery and mutual support takes center stage.

## Project Assumptions

# FUNCTIONAL REQUIREMENTS

## Business Use Case Model

## Use Case Glossary

|  |  |  |
| --- | --- | --- |
| **Package Id: A Package Name: Student** | | |
| **Use Case Id** | **Use Case Name** | **Actors** |
| A0100 | Register Student | Student |
| A0200 | Maintain Student details | Student |
| A0300 | Apply for funds | Student |
| A0400 | Track funding | Student |
| **Queries/Reports** | | |
| A0500 | View financial statement | Student |
| A0600 | View fund request | Student |

|  |  |  |
| --- | --- | --- |
| **Package Id: C Package Name: Funder** | | |
| **Use Case Id** | **Use Case Name** | **Actors** |
| C0100 | Register Funder | Funder |
| C0200 | Maintain Funder details | Funder |
| C0300 | Update fund request | Funder |
| C0400 | Upload student list | Funder |
| C0500 | Update funding status | Funder |
| C0600 | Employee | Funder |
| C0700 | Maintain funds | Funder |
| **Queries/Reports** | | |
| C0800 | View applications | Funder |
| C0900 | View list of potential bursars | Funder |

## Use Case Narratives (User Stories)

### Package A: [Student]

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0100 | Register Student | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | Students can set up their accounts. Relevant information such as student number, ID number, nationality, name and surname, email address, cellphone number, postal address, institution name and bursary name for the student will be kept. | |
| **Pre-Conditions** | The student must have a link of the web-based system to the device they will be using. | |
| **Triggers** | The student taps the link supplied on their device and then clicks the register icon on the home screen. | |
| **Post-Conditions** | For the student, an account is created. For each student, an identifying number is generated and stored alongside the other data in the database. Following registration, the student receives an email. | |
| **Basic Flow of Events** | 1. Select student icon on the registration screen. 2. System displays the registration screen with fields into which relevant details must be entered. 3. Student enters their details. 4. The student clicks the register icon on the screen. 5. System validates entered details. 6. The system generates the identifying number for the student and store it in the database. 7. The student receives an email. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0200 | Maintain Student details | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | Allows students to manage their student account details by navigating to a screen where they can manage their information. | |
| **Pre-Conditions** | A student user must be logged in. | |
| **Triggers** | If the system determines that the student's information is inadequate, or if the student presses the manage details button on the interface. | |
| **Post-Conditions** | The student's information is updated in the database, and the revised information is shown on the interface for all users that require it. Student users can go back to their own account page. | |
| **Basic Flow of Events** | 1. The system displays the maintain details screen. 2. Student updates their details 3. Student presses the save button. 4. The student’s information is updated. 5. The system shows the updated information on the screen. 6. Student press a button that takes him or her to their own account page. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0300 | Apply for funds | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | Allows students to apply for funding by navigating to a screen. where they can provide the relevant information needed for applying for funds. | |
| **Pre-Conditions** | A student user must be logged in. | |
| **Triggers** | Students without bursaries can use the apply for funds button in the interface to apply for a bursary of their own choosing. | |
| **Post-Conditions** | Student has completed an application for bursary refund and the relevant information is kept in the database for the funder to view. | |
| **Basic Flow of Events** | 1. System displays the apply for funds screen with fields into which relevant details must be entered. 2. Student enters their details. 3. Student chooses the bursary to apply for. 4. The information is kept on the database. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0400 | Track funding | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | Students have the ability to keep track of the application they made for whether they are funded or not for a certain academic year. | |
| **Pre-Conditions** | A student user must be logged in. | |
| **Triggers** | Students need to press the track funds button in the interface. | |
| **Post-Conditions** | Students have seen whether they are funded or not. | |
| **Basic Flow of Events** | 1. System displays the track funding screen with application statuses showing. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0500 | View financial statement | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | The student can review the financial statement of himself or herself and determine whether they are still owing the institution money or not. | |
| **Pre-Conditions** | A student user must be logged in. | |
| **Triggers** | The student clicks the view financial statement button. | |
| **Post-Conditions** | A report compiled by the system on the financial statement of the students is displayed on the interface. | |
| **Basic Flow of Events** | 1. The system displays the financial statement on the screen. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| A0600 | View fund request | |
| **Primary Business Actors** | | **Other participating Actors** |
| Student | |  |
| **Description** | The student has the ability to view if the institution has requested funds on behalf of themselves and if the funder has replied in terms of releasing the funds or not. | |
| **Pre-Conditions** | A student user must be logged in. | |
| **Triggers** | Student clicks the fund request button on the interface. | |
| **Post-Conditions** | A report on whether funds are requested or not is shown on the screen of the interface. | |
| **Basic Flow of Events** | 1. The system displays if the funds have been requested or not. | |

### Package B: [Institution]

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0100 | Register Institution | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | Institutions can set up their accounts. Relevant information such as the name, physical and postal addresses, telephone number, and email address of the institution will be kept. | |
| **Pre-Conditions** | The institution must have a link of the web-based system to the device they will be using. | |
| **Triggers** | The institution taps the link supplied on their device and then clicks the register icon on the home screen. | |
| **Post-Conditions** | For the institution, an account is created. For each institution, an identifying number is generated and stored alongside the other data in the database. Following registration, the institution receives an email. | |
| **Basic Flow of Events** | 1. Select institution icon on the registration screen. 2. System displays the registration screen with fields into which relevant details must be entered. 3. Institution enters their details. 4. The institution clicks the register icon on the screen. 5. System validates entered details. 6. The system generates the identifying number for the student and store it in the database. 7. The institution receives an email. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0200 | Maintain Institution details | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | Allows the institution to maintain institution related details by navigating to a screen where the institution can manage, they are institution details. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | If the institution wishes to update its information, the institution selects an edit details button on the interface. | |
| **Post-Conditions** | The institution details are updated in the database. The institution users can go back to their own account page. | |
| **Basic Flow of Events** | 1. The system displays the maintain details screen. 2. Institution updates their details 3. Institution presses the save button. 4. The institution’s information is updated. 5. The system shows the updated information on the screen. 6. Institution presses a button that takes him or her to their own account page. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0300 | Provide student results | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | The institution provides the academic results of the previous or current academic year of each student that is attending that institution. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | The institution selects the edit result button on the interface. | |
| **Post-Conditions** | Each student's academic results are updated in the database and available for users that can use the relevant information. | |
| **Basic Flow of Events** | 1. System displays the screen with fields into which relevant details must be entered regarding student results. 2. Authorized personnel uploads result for the student 3. System updates the results on the database | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0400 | Request student funds | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | The institution requests funds for each student That is attending that institution for a particular year from their founder. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | Institution selects the request student funds button on the interface. | |
| **Post-Conditions** | The requesting of funds for each student is updated in the system and is available for viewing. by students that want to see if funds are requested or not. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0500 | Upload financial statement | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | The institution uploads the financial statement of each student. The financial statement is a detailed statement of the student's financial performance. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | Institution selects the upload financial statement button on the interface. | |
| **Post-Conditions** | The financial statement of each student is updated on the system and is kept in the database for viewing by students that need to see their financial statements. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0600 | Employee | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | The people that will be responsible for doing different activities for the institution. This will include uploading financial statements, providing academic results for students, authorized personnel updating the institutions information. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | Institution needs to select the employee button in the interface. | |
| **Post-Conditions** | A list of employees will be kept in the database that can provide certain functions for an institution. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| B0700 | View list of students funded | |
| **Primary Business Actors** | | **Other participating Actors** |
| Institution | |  |
| **Description** | This is a list of students that are funded for a particular academic year that is provided by the funder. This is so that the institution can allow students to register. | |
| **Pre-Conditions** | An institution user must be logged in. | |
| **Triggers** | Institution selects the view student funded button on the interface. | |
| **Post-Conditions** | A report compiled by the system on the list of funded students is displayed on the interface. | |
| **Basic Flow of Events** | 1. The system displays a list of students that have funds. | |

### Package C: [Funder]

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0100 | Register Funder | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | Funders can set up their accounts. Relevant information such as the tax number, name, email, telephone number, physical address, and the name of the bursary that will be offered by the funder will be kept. | |
| **Pre-Conditions** | The funders must have a link of the web-based system to the device they will be using. | |
| **Triggers** | The funder taps the link supplied on their device and then clicks the register icon on the home screen. | |
| **Post-Conditions** | For the funder, an account is created. For each funder, an identifying number is generated and stored alongside the other data in the database. Following registration, the funder receives an email. | |
| **Basic Flow of Events** | 1. Select funder icon on the registration screen. 2. System displays the registration screen with fields into which relevant details must be entered. 3. Funder enters their details. 4. The funder clicks the register icon on the screen. 5. System validates entered details. 6. The system generates the identifying number for the funder and store it in the database. 7. The funder on receives an email. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0200 | Maintain Funder details | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | Allows the funder to maintain funder related details by navigating to a screen where the funder can manage, they are funder details. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | If the funder wishes to update its information, the funder selects an edit details button on the interface. | |
| **Post-Conditions** | The funder details are updated in the database. The funder users can go back to their own account page. | |
| **Basic Flow of Events** | 1. The system displays the maintain details screen. 2. Funder updates their details 3. Funder presses the save button. 4. The funder’s information is updated. 5. The system shows the updated information on the screen. 6. Funder presses a button that takes him or her to their own account page. | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0300 | Update fund request | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | The founder responds to the request that was made by the institution. They provide whether they will be paying the funds or not. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | The update request fund button is pressed in the interface. | |
| **Post-Conditions** | Updated fund request from the funder's point of view. is kept in the system and relevant users can see the information. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0400 | Upload student list | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | Funder uploads a list of students that will be funded by them for a particular academic year. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | The upload student list button is pressed in the interface. | |
| **Post-Conditions** | An updated list of students is uploaded to the database so that relevant users that need the information can use it. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0500 | Update funding status | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | Allows the funder to update funding statuses of students that have applied for funding. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | The funder clicks the update funding status. | |
| **Post-Conditions** | Funding statuses for applications of each student is updated in the database for students to view. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0600 | Employee | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | The people that will be responsible for doing different activities for the funder. This will include maintaining funds, providing funding statuses for students, authorized personnel updating the funders' information. | |
| **Pre-Conditions** | The funder user must be logged in. | |
| **Triggers** | Funder needs to select the employee button in the interface. | |
| **Post-Conditions** | A list of employees will be kept in the database that can provide certain functions for a funder. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0700 | Maintain funds | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | A funder can maintain student funds for a particular academic year. This means authorized personnel can provide the funds for a student | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | Funder selects the maintain funds button in the interface | |
| **Post-Conditions** | The funds for each student in the database is maintain for that particular academic year. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0800 | View applications | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | The viewing of students that applied for funding according to the students’ faculties. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | View application button is selected by the funder. | |
| **Post-Conditions** | A list of all the students that applies is shown on the interface. | |
| **Basic Flow of Events** |  | |

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | **Use Case Name** | |
| C0900 | View list of potential bursars | |
| **Primary Business Actors** | | **Other participating Actors** |
| Funder | |  |
| **Description** | Funders can view a list of students that have been accepted in different institutions. This is to confirm that students are planning on studying for a certain academic year. | |
| **Pre-Conditions** | A funder user must be logged in. | |
| **Triggers** | Funder selects the option of potential bursars' button from the interface. | |
| **Post-Conditions** | A compiled list of potential bursars is displayed in the interface. | |
| **Basic Flow of Events** | 1. The system displays a list of students that have been accepted in different institutions. | |

# NON-FUNCTIONAL REQUIREMENTS

## Interface Requirements

**Usability Goals**

* Effectiveness: The website is effective since it provides one platform where all parties can share information in real-time.
* Efficiency: The website is efficient by executing queries relatively fast.
* Safety: All users are required to login to use the system by providing their credentials which are then validated. The systems authenticate all the users and only authorized users are allowed to access specific web pages with functions that are specific for that user. All the data is encrypted and cannot be altered while in transit by third parties. Users are asked to confirm before executing some functions that can result in loss of the user information like delete functionality. The website also uses validation to ensure users provide all the necessary information and minimize errors when executing a query.
* Utility: The system has a good utility since it supports CRUD functions to create, read, update, and delete records from the database.
* Learn ability: The website is easy to learn as it follows the basic bootstrap structure webpages. The user does not need to take a tutorial on how to use our website to be able to navigate their way around the system since the website is very well organized.
* Memorability: The website is easy to memorize since most of the buttons are self-explanatory and have icons to help users get identity the button easily.

**User Experience Goals**

* Helpful: The system is very helpful to the users, and it enables them to access the information they need in real time without having to communicate via emails and phone calls to get updates.
* Informative: The system is an information-based system that is responsible for the exchange of information between the users.
* Satisfying: The system is all about fast communication and that alone can be satisfactory to the user as they don’t have to go through the trouble of having to wait for an email from the other party.

**User Requirements**

* System use: This system can be used by novice or expert users.
* Novice and/or casual users can use the website with or without the help of the tutorial section depending on the user.
* Expert users like administrators of the website have full access to all the functionality of the website for testing purposes.

## Performance Requirements

* Response Time: The website is mobile first, meaning the code of the that will load first the code that is used to render pages for mobile devices. This reduces load time for smartphone as the web browser does not load all the code of which some might not apply to a device of smaller screen size. The website will contain less images of which are compressed and optimized. The web application has less redirect, too many redirects prolong the HTTP request and response process. The website will use caches to store copies of the website on the visitor’s browser. That way the website will not have to reload everything.
* Reliability: The website will have a 95% reliability during the year but when registration time comes it will drop to 50% reliability. To avoid disappointments, the system will have to shut down 25% of the time, that will increase the reliability to 75% regardless of the high volume of request made by at least 10000 users per hour.
* Maintainability: The maintainability of system is 75%, giving the administrator 75% chances to fix the any issue the website might have in 24 hours. That will be enough to allow the website to be fixed during the shutdown time as the web application does not have many operations that it performs meaning that it has less behind-the-scenes code that needs to be debugged.
* Availability: The website will be available 24 hours a day from Monday to Monday. Only during registration time when the website will be in high demand will the users be asked not to access the website from midnight to 6am, that will allow the server to recover and prevent it from crashing making it unavailable during the business hours when most users need to use the web application.

## Security Requirements

* The system has 4 user roles which are student, institution, funder, and admin.
* All users are required to login to use the system by providing their credentials which are then validated.
* The systems authenticate all the users and only authorized users are allowed to access specific web pages with functions that are specific for that user.
* Student user can register and update their personal details. Can view their own funding status and submit documents if needed.
* Institution user can upload the registration data of the student and request funds from the funder. Can view fund status of all students in the system.
* Funder user can upload bursary letters and/or agreement for the student they are funding. Can request students result which can only be provided by the student due to the POPI Act.
* Admin user is responsible for registering both the institution and funder users. Has access to all the functionality on the website but cannot access confidential documents like bursary agreements, identity documents and academic transcripts.
* All the data is encrypted and cannot be altered while in transit by third parties.

## Operational Requirements

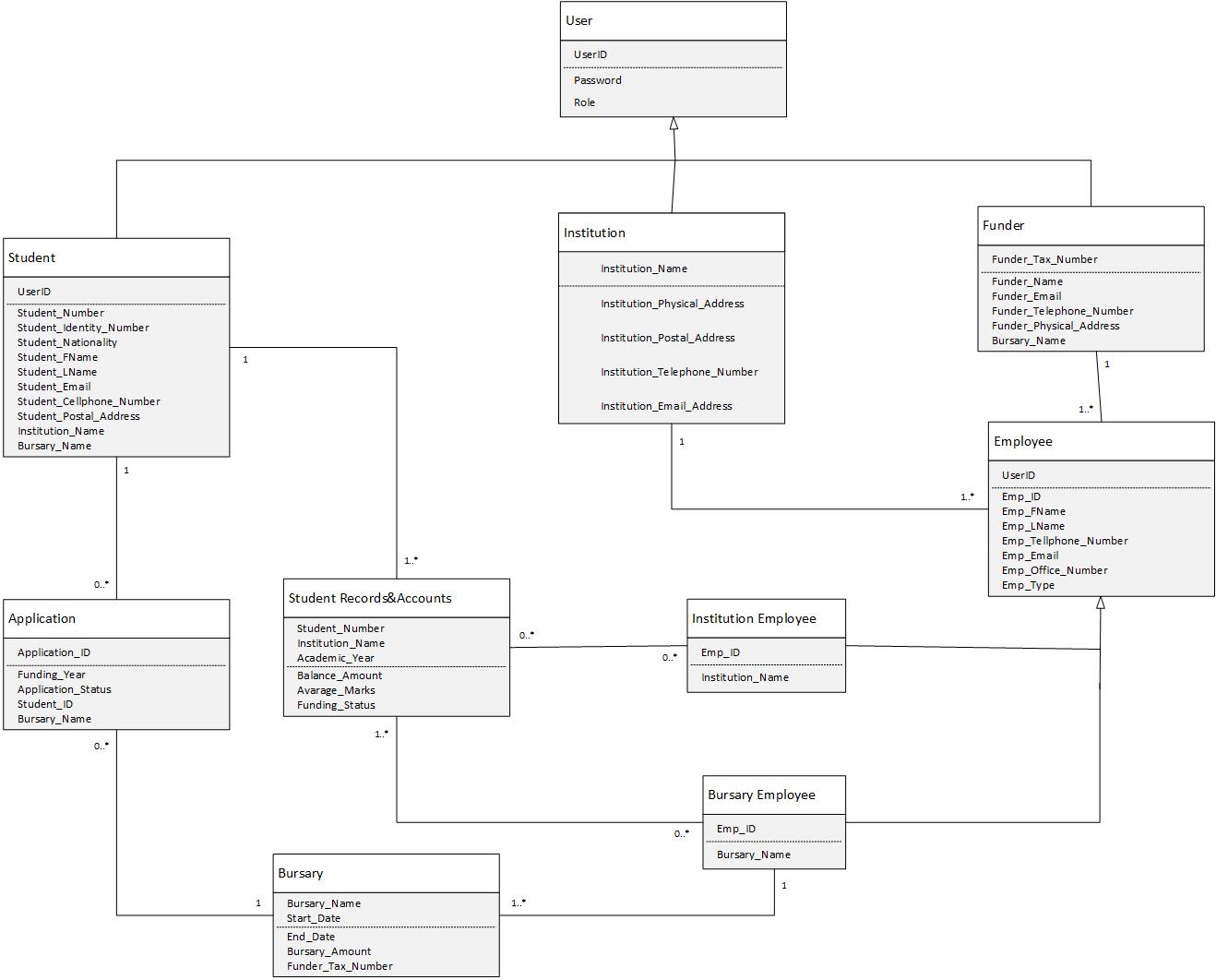
* The system is a web application. Web pages can work on any device, only need a web browser to be rendered. This web site will be a mobile first application (responsive design).

# DATA REQUIREMENTS

## Preliminary List of data and attributes

|  |  |
| --- | --- |
| **Entity** | **Entity Attributes** |
| User | UserID  Password  Role |
| Institution | Institution\_Name  Institution\_Physical\_Address  Institution\_Postal\_Address  Institution\_Telephone\_Number  Institution\_Email\_Address |
| Funder | Funder\_Tax\_Number  Funder\_Name  Funder\_Email  Funder\_Telephone\_Number  Funder\_Physical\_Address  Bursary\_Name |
| Student | UserID  Student\_Number  Student\_Identity\_Number  Student\_Nationality  Student\_FName  Student\_LName  Student\_Email  Student\_Cellphone\_Number  Student\_Postal\_Address  Institution\_Name  Bursary\_Name |
| Bursary | Funder\_Tax\_Number  Bursary\_Name  Start\_Date  End\_Date  Bursary\_Amount |
| Employee | UserID  Emp\_ID  Emp\_FName  Emp\_LName  Emp\_Telephone\_Number  Emp\_Email  Emp\_Office\_Number  Emp\_Type |
| Bursary Employee | Emp\_ID  Bursary\_Name |
| Institution Employee | Emp\_ID  Institution\_Name |
| Application | Application\_ID  Funding\_Year  Application\_Status  Student\_ID  Bursary\_Name |
| Student Records&Accounts | Student\_Number  Institution\_Name  Academic\_Year  Balance\_Amount  Average\_Marks  Funding\_Status |

## Domain Class Diagram



# Preliminary Schedule of Responsibilities

|  |  |  |
| --- | --- | --- |
| **Use Case ID & Name** | **Team Member Responsible** | |
| **Student Number** | **Name** |
| A0100: Register Student  A0200: Maintain Student details  A0300: Apply for funds  A0400: Track funding  A0500: View financial statement  A0600: View fund request | 219121923  221517979 | Bekikaya Mda  Simthembile Shezi |
| B0100: Register Institution  B0200: Maintain Institution details  B0300: Provide student results  B0400: Request student fund  B0500: Upload financial statement  B0600: Employee  B0700: View list of students funded | 219121923 | Bekikaya Mda |
| C0100: Register Funder  C0200: Maintain Funder details  C0300: Update fund request  C0400: Upload student list  C0500: Update funding status  C0600: Employee  C0700: Maintain funds  C0800: View applications  C0900: View list of potential bursars | 221517979 | Simthembile Shezi |

# Researching the Requirements

## 10 User Interview Questions

1. What is your past experience dealing with financial aid offices?
2. How often do you visit the financial aid office per year?
3. During those visits how many times did you get the assistance you need?
4. How fast does financial aid respond to emails on a regular day?
5. How many days does it take for a query to be solved via email?
6. What is the issue you usually run into when looking for funding?
7. Do you think bursary applications can be easier than they are now?
8. Have you ever had a hard time communicating financial matters with relevant personnel?
9. As the funder, what information would you like to get from prospective bursars?
10. As the institution, is there a process that you would like to simplify when dealing with student funding?