

**Team Name:** Team 22

**Project Name:** Admin Dashboard

Hannah Champury, Josh Scally, Michael Oflazian, Mateo Calderon

**VM:** csc415-team22.hpc.tcnj.edu

**Path:** ArmInArm-Spr24/src

**Github:**

**Liaison:** Josh Scally

CSC 415-02

Iteration IIb

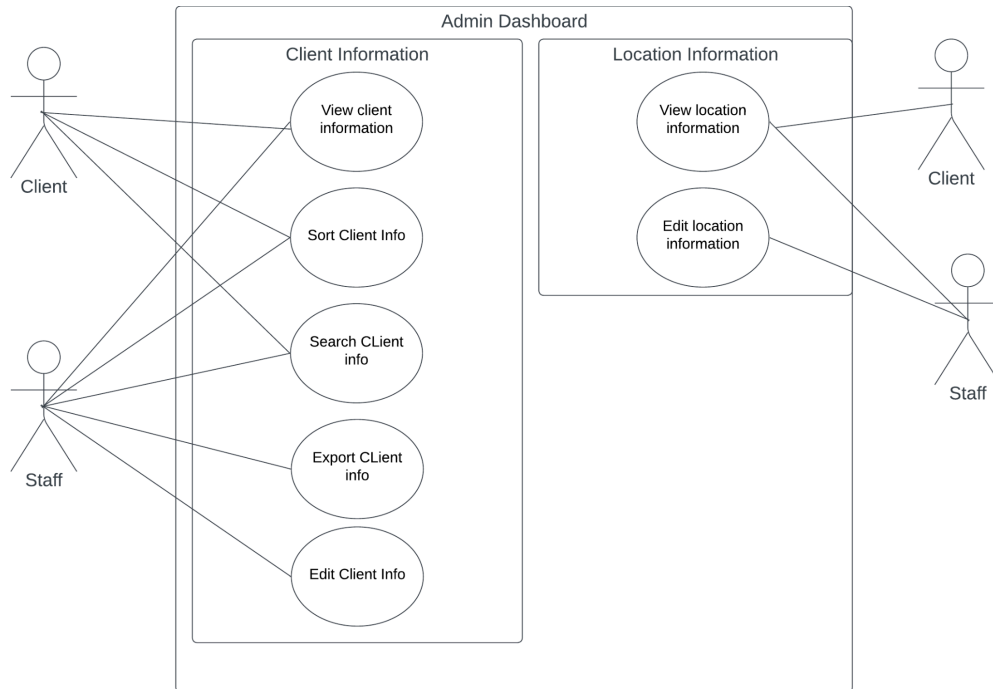
March 21, 2024

## **Project Proposal & Specifications Document**

- **Problem:** The problem we are solving pertains to the overall structure and accessibility of information regarding the client list and the Arm In Arm locations themselves. As it stands, there needs to be an efficient and reliable way for the Arm In Arm staff to access and update client and location information.
- **Objective:** We propose to create functionality that will allow Arm In Arm staff to update client credentials and change information regarding a location's opening/closing times and addresses if necessary. Our objectives include making an easily navigable webpage that displays all desired content in an organized manner.
- **Description:** The end goal of our application will incorporate improvements to user and admin pages. In the admin dashboard, users will be able to access client and pantry location information. The admin will be able to view and and edit client and pantry location information. They will be able to use different sorting functions to view clients based on specific criteria and search for specific data. On the client side, users can only view information and cannot edit information.
- **Stakeholders:** The primary stakeholders include the Arm In Arm staff/volunteers, the clients who will be creating accounts and utilizing our website, and us as the developers.
- **Importance of continued contact with company:** It is important to keep in contact with the company/organization so that we understand what their requirements are. It is not uncommon for requirements to change abruptly, especially if the organization thinks of new ideas that they wish to have implemented. Of course, part of keeping in touch with the company/organization is letting them know what is or isn't feasible given their budget and in our case, our timeframe. Additionally, it is important to inform the company/organization about the development process. Software engineers should be updating their clients on their progress to let them know that work is being done. Without communicating in this aspect, their clients might not be sure whether or not their money is being put to good use.
- Our intention is to consistently communicate with Arm In Arm staff every time we update the project significantly. This would let them see and track our progress, allowing them to feel like they are part of the development process.

- **Social Issues:** The social issue we are addressing is hunger, specifically in Mercer County. Throughout the state (not only Mercer County) families are struggling to provide for themselves and their families due to many factors. The volunteers and staff at Arm In Arm are driven to provide food and care to those who are in need of help. Our job is to aid Arm In Arm so that they can provide for individuals who are suffering from hunger.
- **Ethical Issues:** The majority of the ethical issues surround data collection and presentation. There is sensitive information within our databases, including addresses, names, and financial status. It is imperative that we keep this data secure so that no one who is unauthorized to view certain information can access it. If not, it places the clients of Arm In Arm at risk. Another ethical issue could arise from not providing a way to translate our pages into multiple languages. Doing this would ensure that we are not catering to the needs of just one specific demographic of people. Allowing everyone regardless of preferred language to easily access our web pages will ensure that we are implementing ethical practices when developing our software.
- **Economic Issues:** Economic challenges are perhaps the most influential factor contributing to hunger in Mercer County, and possibly in all of New Jersey. Families are not able to afford healthy food or food at all, so they resort to eating quick meals like fast food or not at all. With many families, there is a degree of stress that comes with trying to pay bills and taxes while also paying for food. Arm In Arm helps alleviate some of this stress by providing food for those who are struggling financially. In developing software to help Arm In Arm, we are indirectly helping families who may be struggling to put food on the table.
- **Security & Privacy Concerns:** Due to the nature of the information stored in the databases (sensitive information including names, addresses, and financial information) it is important to limit access to authorized users only. This may be accomplished by including checks such as ensuring that the user is signed in and has appropriate permissions. Furthermore, we need to ensure that these values are not able to be modified without the explicit command of a staff member to ensure that the values are not fraudulent.

- **Use Case Diagram**



- **Prototype 1:** Create basic UI and link it to database
- **Prototype 2:** Add more sophisticated features to the UI such as filtering results and searching for results, and displaying results
- **Prototype 3:** Address any bugs encountered in the program and ensure that it is robust before committing as final project
- **Final Project:** Make UI visibly more pleasing and easier to use
- **Future Implementation:** Limit number of results per page and allow to navigate multiple pages, Customize columns that are visible, Make working design for multiple devices
- **Use Case Description**

1. **Use case:** View Client Information

**Primary actor:** Staff/Client

**Goal in context:** Provide Arm-in-Arm staff a means to view client data within their database

**Preconditions:** Login to system as user

**Trigger:** User navigates to client page

**Scenario:**

1. Open website
2. Login in
3. Navigates to the admin page
4. Client info will be displayed via a table
5. Logout at anytime

**Exceptions:**

1. Staff member can't access webpage due to network issues

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Whenever user needs to view client information

**Channel to actor:** Admin page of the Arm-in-Arm website

**Secondary actors:** Staff, Support technicians

**Channels to secondary actors:** Admin page of the Arm-in-Arm website

**Open issues:**

1. How to provide admin/staff access to users
2. Provide efficient algorithm to sort/filter requested data

**2. Use case: View Location Information**

**Primary actor:** Staff/Client

**Goal in context:** Provide Arm-in-Arm staff a means to view location information

**Preconditions:** Login as a user

**Trigger:** User navigates to location page

**Scenario:** The user enters their login information and navigates to the admin page. From there, the user finds the location page. There, there will be a display of all of the necessary information regarding locations

**Exceptions:**

1. The user can not access the page due to network connectivity issues

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Frequently; whenever a user needs to view location addresses/operation times

**Channel to actor:** Location page of the Arm-in-Arm website

**Secondary actors:** Staff, Support technicians

**Channels to secondary actors:** Admin page of the Arm-in-Arm website

**Open issues:**

1. Ensuring changes made to location page are able to be viewed
2. Performance when handling large amounts of data

**3. Use case: Edit Client Information**

**Primary actor:** Staff

**Goal in context:** Provide Arm-in-Arm staff a means to modify client information

**Preconditions:** Login as a user

**Trigger:** Staff member navigates to the admin page

**Scenario:** Staff member navigates to the admin page. From the admin page, there will be several options to modify/delete via buttons adjacent to the table.

**Exceptions:**

1. Staff member accidentally makes change to database entry

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Frequent for admin to edit client info

**Channel to actor:** Admin page of the Arm-in-Arm website

**Secondary actors:** Staff, Support technicians

**Channels to secondary actors:** Admin page of the Arm-in-Arm website

**Open issues:**

1. Performance when handling large amounts of data
2. Security for sensitive data & unauthorized access

**4. Use case:** Edit Location Information

**Primary actor:** Staff

**Goal in context:** Provide Arm-in-Arm staff a means to view and modify location information such as days/times of operation and address

**Preconditions:** Login as an admin

**Trigger:** Staff member navigates to the admin page

**Scenario:** Staff member navigates to the admin page. From the admin page, there will be several options to modify/delete via buttons adjacent to the table.

**Exceptions:**

2. Staff member accidentally makes change to database entry

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Whenever an Arm In Arm location changes hours or addresses

**Channel to actor:** Admin page of the Arm-in-Arm website

**Secondary actors:** Staff, Support technicians

**Channels to secondary actors:** Admin page of the Arm-in-Arm website

**Open issues:**

1. Performance when handling large amounts of data
2. Security for sensitive data & unauthorized access

**5. Use case:** Sort Client Information

**Primary actor:** Staff

**Goal in context:** Provide Arm-in-Arm staff a way to customize sorting based on different data of clients

**Preconditions:** Log in as an admin

**Trigger:** Staff member navigates to the admin page

**Scenario:** Staff member navigates to the admin page. From the admin page, there will be several options to sort results based on a drop down list of criteria to sort by.

**Exceptions:**

3. Admin only has ability to sort by certain characteristics

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Whenever an Arm In Arm client requires information to be updated

**Channel to actor:** Admin page of the Arm-in-Arm website

**Secondary actors:** N/A

**Channels to secondary actors:** N/A

**Open issues:**

3. How to provide admin/staff access to users
4. Provide efficient algorithm to sort/filter requested data

**6. Use case:** Search Client Information

**Primary actor:** Staff/Client

**Goal in context:** Provide Arm-in-Arm staff and clients a way to search for specific client information

**Preconditions:** Log in as an admin

**Trigger:** Staff member navigates to the admin page, user navigates to the account page

**Scenario:** Staff member navigates to the admin page. From the admin page, there will be a search box which may have several ways to filter the search. After selecting the filters, the staff member presses the “Search” button. User may navigate to the account page and view their information

**Exceptions:**

1. Admin applies no filters when searching

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Whenever an Arm In Arm staff member desires to search for client info or whenever a client visits the account page

**Channel to actor:** Admin page of the Arm-in-Arm website

**Secondary actors:** N/A

**Channels to secondary actors:** N/A

**Open issues:**

5. How to limit access to viewing certain client info (client can view their own info, while staff can view all clients info)

**7. Use case:** Export Client Information

**Primary actor:** Staff

**Goal in context:** Provide Arm-in-Arm staff to export data in multiple formats (pdf, csv, potential others)

**Preconditions:** Log in as an admin

**Trigger:** Staff member navigates to the admin page

**Scenario:** Staff member navigates to the admin page. From the admin page, there will be export options above the sorted data.

**Exceptions:**

1. There is no exportable data

**Priority:** High; must be implemented Spring 2024 Semester

**When available:** End of semester; Final Project Submission

**Frequency of use:** Whenever an Arm In Arm client requires information to downloaded in a specific format

**Channel to actor:** Admin page of the Arm-in-Arm website

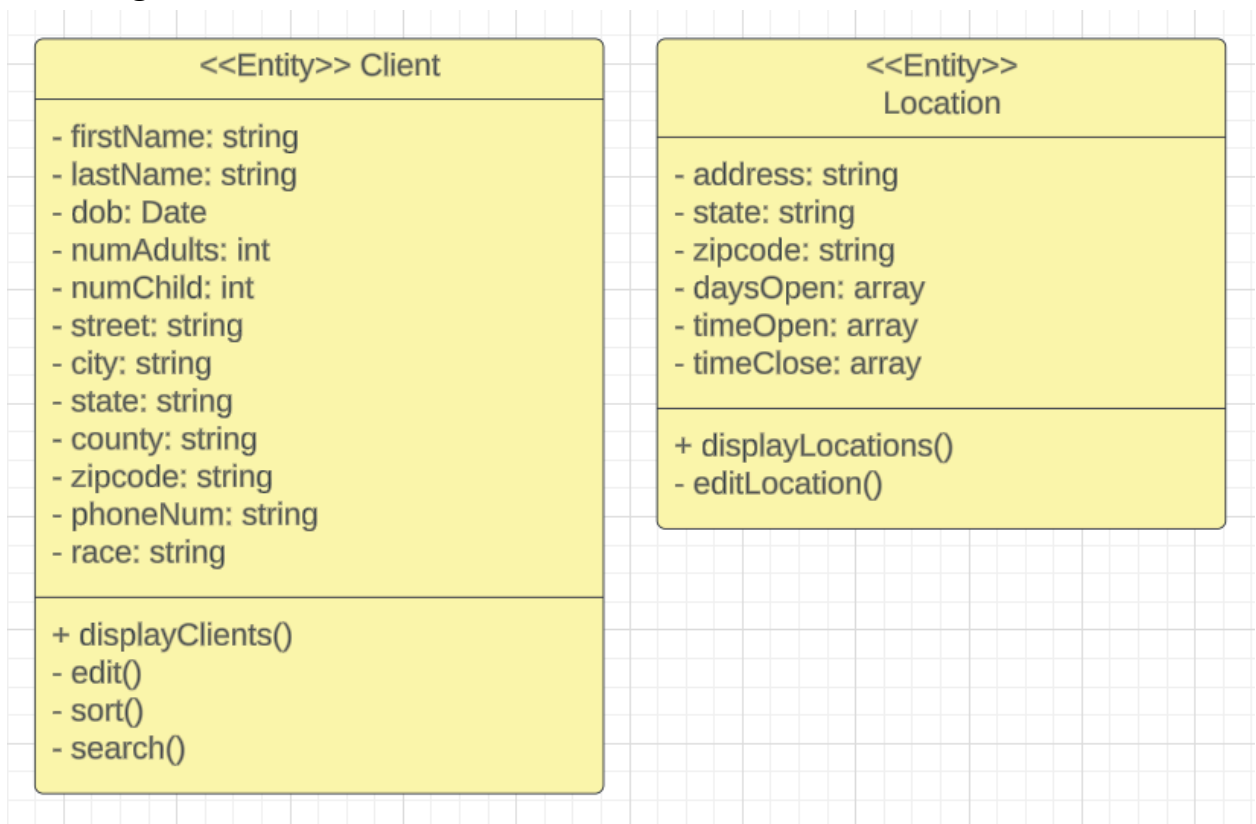
**Secondary actors:** Staff / technicians

**Channels to secondary actors:** N/A

**Open issues:**

1. How to provide admin/staff access to users
2. Properly implement an export feature.

- **Class Diagram**



- **Software Engineering Content to Learn:** It is important that we learn how to use Postgres, which will be the primary database used within the program's backend.

Furthermore, it is important to learn rails since that will be the primary web development framework that we will base our frontend on.

- **Timeline:**

3/25: Begin work on Prototype 1

4/1: Finalize Prototype 1, begin work on Prototype 2

4/8: Finalize Prototype 2, begin work on Prototype 3

4/15: Finalize Prototype 3, begin work on Final Implementation

4/22: Continue implementation for Final Implementation, including bug fixes

4/29: Final submission