Project Catawba Area Agency On Aging Database (CAAAD)

Catawba Area Agency on Aging

Date of client feedback: N/A

Date of client approval: N/A

Winthrop University CSCI475 Team # 2

Justin Carmona, Jordan Douglas, AJ Ellis, Preston Fisher, Walker Immel, Christian Morris, Donnie Sullivan

Computer Science & Quantitative Methods Department
College of Business Administration
WINTHROP UNIVERSITY

Table of Contents

1. Introduction

1.1 Purpose, Scope, Prototype

This document serves as the written description for Project Catawba Area Agency on Aging Database (CAAAD) system developed by Team 2 for Catawba Area Agency on Aging (CAAA). This document shall contain definitions of the functionalities and design of the system to clarify the expectations between the user and developer. To ensure that the final delivered system is satisfactory, its design is based upon the requirements listed herein.

The proposed system aims to serve as an efficient solution for CAAA's need for a centralized database. The system will possess similar functionalities to filtering and sorting as the current solution (Microsoft Excel) does alongside as well as specialized functionalities as requested by CAAA. These functionalities are enumerated in Section 2.2, and define functions that are to be implemented within the system.

More specifically, the system will be composed of a database that will contain CAAA's client demographics in the form of records and user-generated reports, which will be in the form of a CSV file created as a result of performing a filtered search. This database and all of its contained records will be within the system boundary. The user and further modification of the generated reports, such as creating a graphical representation from the CSV file, will fall outside of the boundary. This boundary will remain consistent throughout the development of the system so as to maintain the scope of development and ensure that the system will be delivered in a timely manner.

The prototype of the entire user interface and one working requirement detailed in Sections 4-5 is an example of what the system will look like when fulfilling the requirements. This prototype serves as a demonstration of how the user will interact with the system and will be used to gauge how successful the proposed design will be. The prototype is representative of the current state of the requirement, but is not representative of the final system. The feedback received based upon this prototype will be used to measure the success of the design so far, and further used to influence the design going forward to deliver the best possible system for CAAA.

1.2 Design Goals

The design goals focused on by Team 2 in the development of the system are as follows:

1.2.1 Reliability

The development of the system will promote an emphasis on the system being reliable. That is, the system will operate as intended with minimal failures. To achieve this, testing will begin early and often as the system is developed and will consist of both unit and integration testing.

1.2.2 Minimum Number of Errors

The development of the system will promote a minimum number of errors in the requirements of the system. By ensuring the requirements of the system are correct, the possibility of faults and failures is minimized as long as the development follows these requirements. To achieve this, communication will be maintained between the developers and customer so that any changes needing to be made to the requirements can be done promptly and completely so that the design can continue to adhere to the requirements.

1.2.3 Good Documentation

The development of the system will promote good documentation for both the design and functionalities of the system. This will ensure that if needed, CAAA has the ability to train new users on how to use the system. Furthermore, since Team 2 will be handing over ownership of the system once completed, any future development will have the necessary documentation to modify or expand the system. To achieve this, any source code will be commented to the best of the developers abilities, and a document containing the functionalities of the system alongside examples of these functionalities will be provided upon the delivery of the system.

1.3 References

Two members of Team 2 have had previous experience working with systems similar to that proposed in this document. Their accounts are documented below. Along with these first-hand experiences, Team 2 has had the most experience using the C languages and as such will be developing the system using C# and SQL. Both of these languages are well documented online and will be referenced by Team 2 throughout the development of the system. The documentation of the system will contain references to these online documentations as needed.

Furthermore, the development and documentation of the system requirements is explained more in-depth in the deliverables already provided to the customer. These deliverables form the basis of the system design, and were drawn from during the creation of this document.

1.3.1 Donald Sullivan, 09/02/2022

The laboratory information management system from Accelerated Technology Laboratories is an Access database used in a similar function as is requested from The Catawba Area Agency On Aging. The software allows the user to import data from excel as well as manual entry, modification, and deletion of data. The software also allows users to generate custom reports based off of the entered parameters. However, this software is aimed toward laboratory samples in a laboratory setting. With knowledge of this system, and by expanding on previously known software, a new application can be created to meet the requirements of the software requested by The Catawba Area Agency On Aging.

1.3.2 Justin Carmona, 09/09/2022

The Control management Inc Access database is very similar to what the client is asking for; this Database works by using the 5 digit tag placed on every piece of equipment owned by the company as the primary identifier for the equipment. The database allows the user to enter specific data for that machine or piece of equipment such as the laptop manufacturer, who its assigned to, and what office the machine is located in, etc in a form and upload that info to the database. In this database the user can also add and remove these machines based on the tag number. The database also allows the user to filter the database based on the tags of the machines.

2. System Requirements Specification

2.1 User Characteristics

Upon discussion with CAAA, it was clarified that the general user of the system will possess the following attributes: the users will have varying levels of technological skill, and the users will have familiarity with Microsoft Excel. These characteristics will be used to determine the overall design of the system, and will be combined with the succeeding requirements to develop an interface that is best suited for the general user.

2.2 Functional Requirements

2.2.1 The system will contain a database of clients and their associated parameters.

2.2.1.1 Input: N/A

2.2.1.2 Output: N/A

2.2.2 The system will allow the user to import records to the database from Excel files.

2.2.2.1 Input: Client records contained in Excel files. (See Section 5.4)

2.2.2.2 Output: N/A

2.2.3 The system will allow manual input of client records into the database by the user.

2.2.3.1 Input: User input, client parameters that make up a client record.

2.2.3.2 Output: N/A

2.2.4 The system must allow modification of existing client parameters within the database by the user.

2.2.4.1 Input: User input, modifications to existing client parameters.

2.2.4.2 Output: modified record.

2.2.5 The system must allow deletion of previously entered client records from the database.

2.2.5.1 Input: User will identify the record to be deleted

2.2.5.2 Output: N/A

2.2.6 The system will prompt the user to resolve duplicate client records.

2.2.6.1 Input: A client record which already exists in the database.

2.2.6.2 Output: Duplicate record prompt.

2.2.7 The system must be able to filter the database by parameters.

2.2.7.1 Input: User input, filtering client parameters.

2.2.7.2 Output: Filtered database

2.2.8 The system will allow the user to be able to generate reports.

2.2.8.1 Input: client parameters to filter for in the database

2.2.8.2 Output: Generated reports based on user filtered client parameters.

2.3 Non-Functional Requirements

2.3.1 The system must be accessible through the organization's intranet system.

2.3.2 The system will only be accessible by the organization's staff members.

2.3.3 The system will emphasize ease of use with a simple interface as tested and verified by the customer.

2.3.4 The system will take at most 10 seconds to return all applicable results when filtering by client parameters.

2.3.5 The system must have the storage capacity to accommodate all existing client data.

2.3.6 The system will be able to scale for at least double the initial amount of client records given enough storage is provided.

3. Access Control and Security Decisions

3.1 Security Decisions

For the security of our database, we will require the user to login using their work-associated email address and a password of their choice. For safety, a new user can only be added and edited by an administrator. The user information will be kept in an SQL database and encrypted using a one-way hashing algorithm.

3.2 Access Control Matrix

The access control matrix below enumerates the access to system functionalities by user types. Since the system will not be used by the public or the clients whose data will make up the database, these rows have access to no system functions. This lack of access is indicated by “---” for clarity.

	Access
Public	---
Client	---

User	addRecord()
	updateRecord()
	searchRecord()

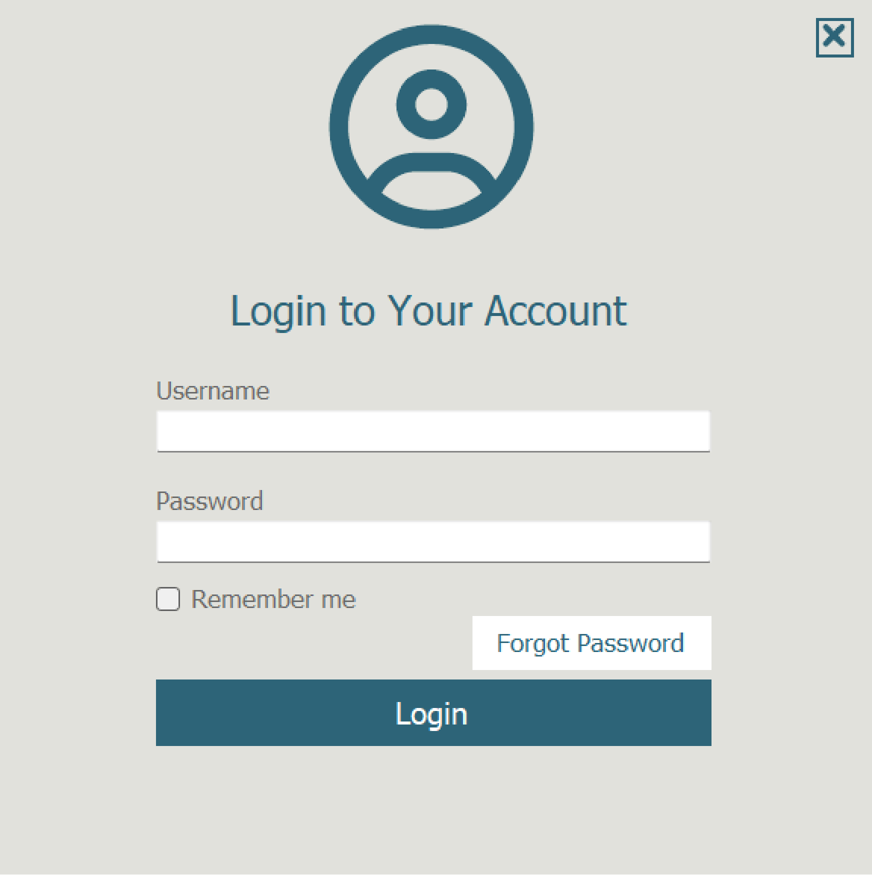
Administrator	addRecord()
	updateRecord()
	deleteRecord()
	searchRecord()
	generateReport()
	createUser()
	editUser()

4. Design Decisions

4.1 User Interface

4.1.1 Login Page

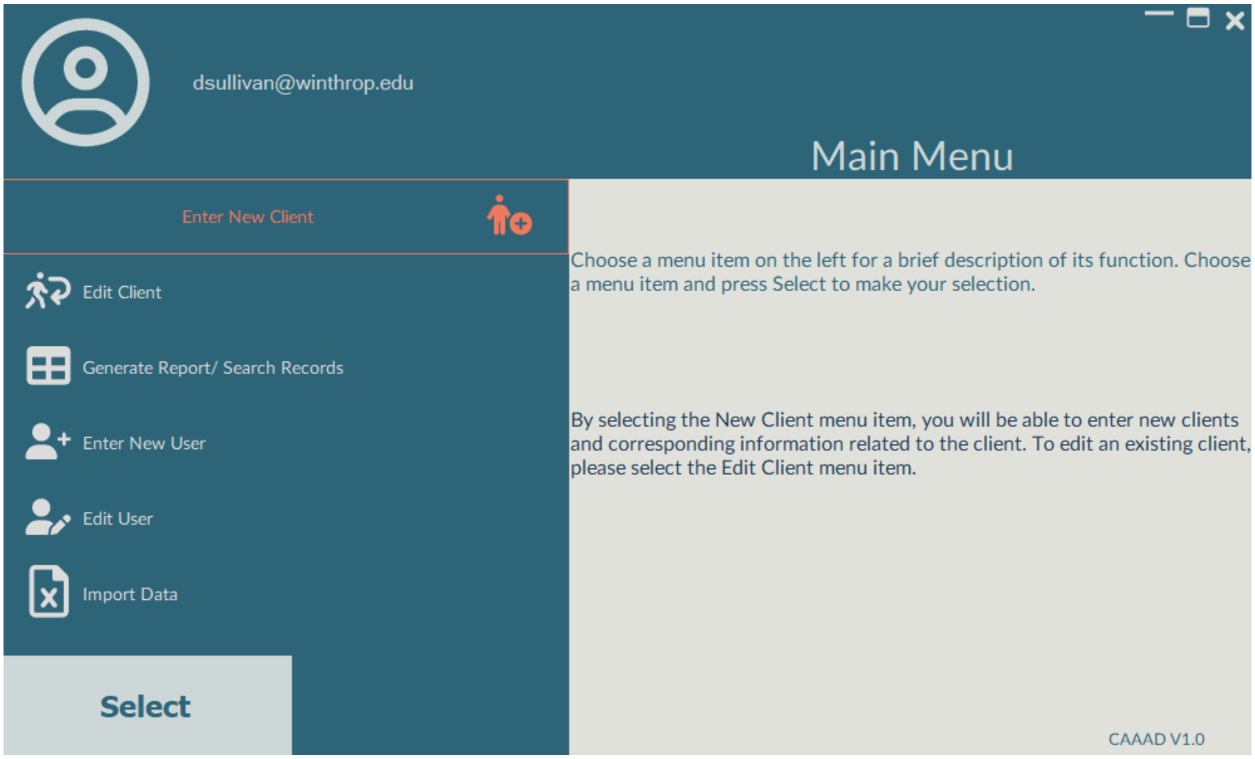
The login page will be the first page accessed by the user upon opening the software. The user or administrator will then enter their login information, consisting of their individual work-associated email address and their chosen password. Following the authentication of the provided credentials, they will be directed to the main menu page.



4.1.2 Main Menu

The main menu is where the user or administrator will make a selection on what action they would like to perform. The selection items will change depending on permissions of the current user. For example, a standard user account will not include an option to create a user as that action is reserved for an administrative account.

As shown below, an administrator account has the options to edit and enter new users into the system, as well as the ability to use the system as a user account (create/edit client, generate report/search records, import data). On the right hand side of the menu, a brief description of the function selected (indicated by the orange box around the function name) is shown so that the user or administrator can preview and decide whether it is the function they are looking for.



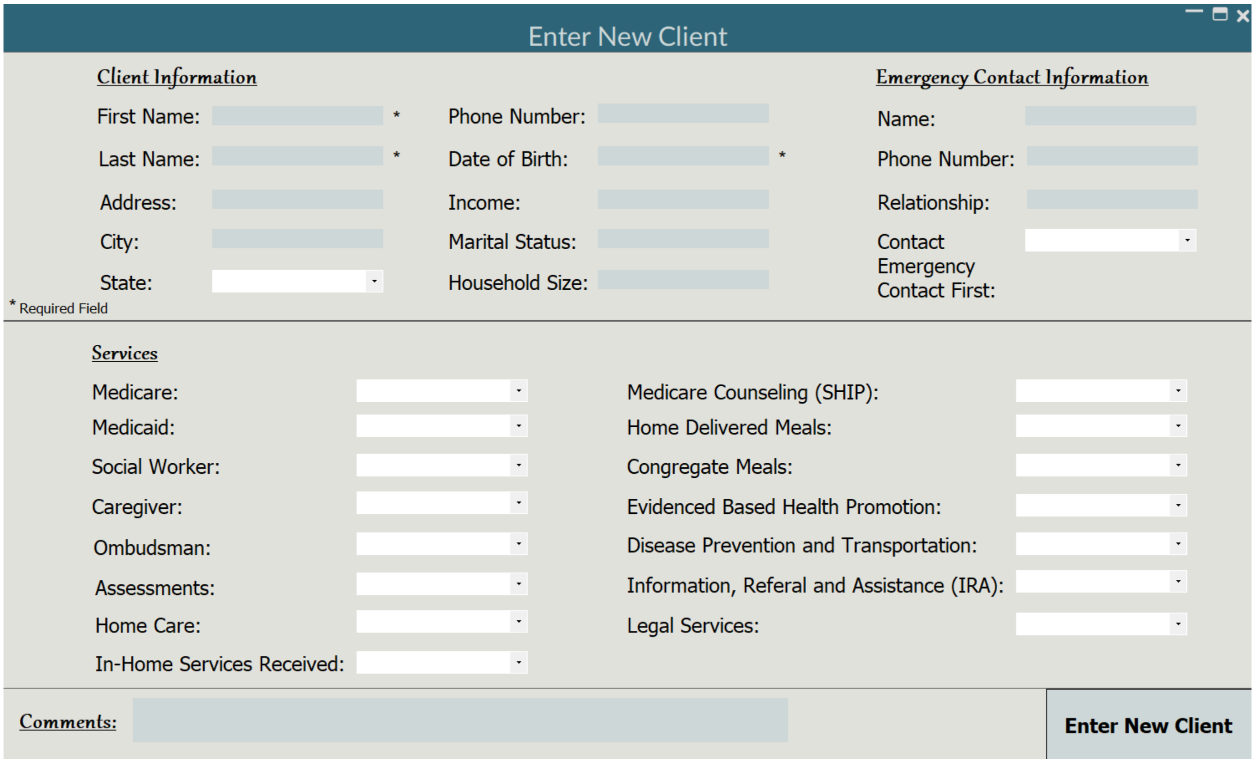
Choose a menu item on the left for a brief description of its function. Choose a menu item and press Select to make your selection.

By selecting the New Client menu item, you will be able to enter new clients and corresponding information related to the client. To edit an existing client, please select the Edit Client menu item.

4.1.3 Create Client

The create client page will allow the user/administrator to create a new client and enter the client's information. The parameters that can be provided are enumerated and split into three categories: Client Information, Emergency Contact Information, and Services. These three sections were implemented to help group information by importance and to allow the user to quickly glance and review the information as needed.

There will be required fields that the user must input about the client before an insertion can be made; these fields will include but are not limited to first name, last name, and date of birth. Required fields are denoted by an “*” along with a subtext on the user interface. A notification will appear if the user attempts to insert a new client without having completed the mandatory fields. After these requirements are met, the user will be able to click the insert button, at which point an entry will be inserted into the database.



Enter New Client

Client Information

First Name: *

Last Name: *

Address:

City:

State:

Phone Number:

Date of Birth: *

Income:

Marital Status:

Household Size:

Emergency Contact Information

Name:

Phone Number:

Relationship:

Contact Emergency Contact First:

* Required Field

Services

Medicare:

Medicaid:

Social Worker:

Caregiver:

Ombudsman:

Assessments:

Home Care:

In-Home Services Received:

Medicare Counseling (SHIP):

Home Delivered Meals:

Congregate Meals:

Evidenced Based Health Promotion:

Disease Prevention and Transportation:

Information, Referal and Assistance (IRA):

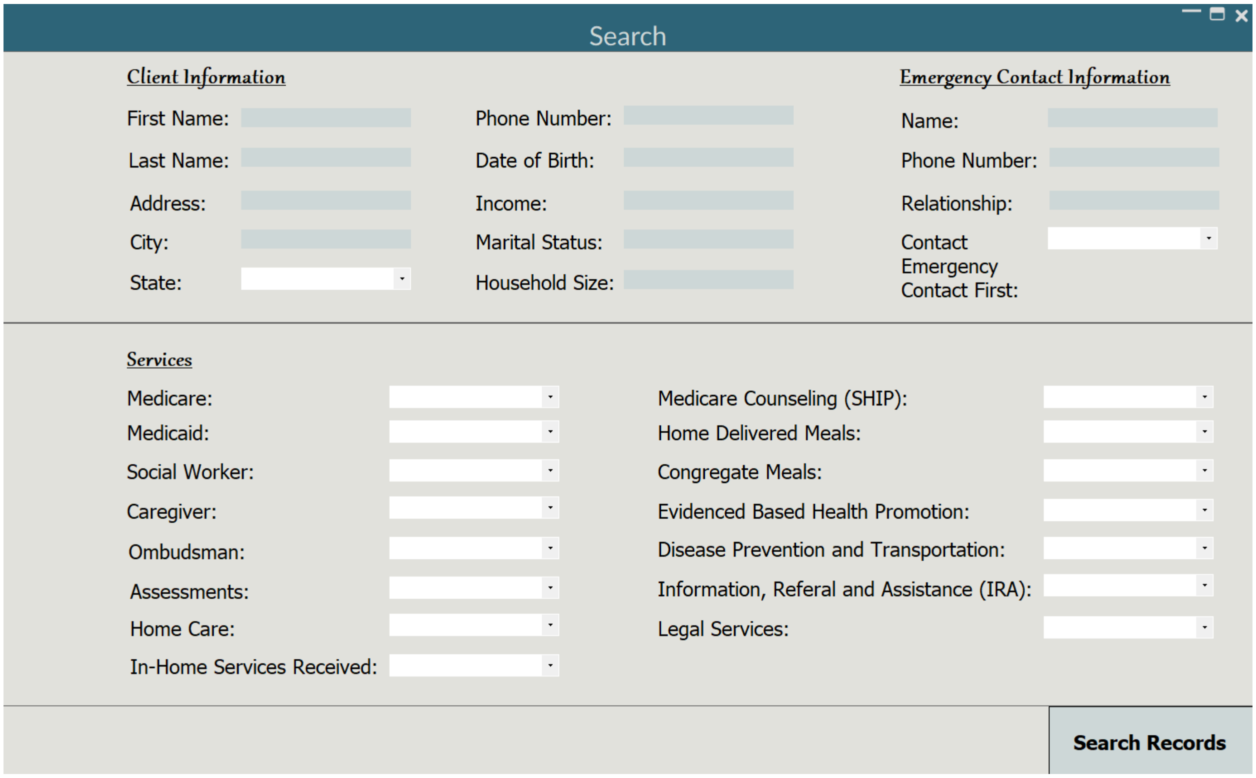
Legal Services:

Comments:

4.1.4 Generate Report/Search Records

The Search Records function will use a similar user interface as that seen in the previous section. Again, the parameters are sectionalized for the convenience of the user, however there are no required fields. By designating search keys in the parameter inputs for search, the user specifies what parameters must match in order for a record to be included in the search results.

This effectively works as a logical AND, and the implementation will be done through iterative search for each parameter that is specified. In the case that no parameters are given, the search results will simply display the entire database, sorted by last name and then first name.



Search

Client Information

First Name:

Last Name:

Address:

City:

State:

Phone Number:

Date of Birth:

Income:

Marital Status:

Household Size:

Emergency Contact Information

Name:

Phone Number:

Relationship:

Contact Emergency Contact First:

Services

Medicare:

Medicaid:

Social Worker:

Caregiver:

Ombudsman:

Assessments:

Home Care:

In-Home Services Received:

Medicare Counseling (SHIP):

Home Delivered Meals:

Congregate Meals:

Evidenced Based Health Promotion:

Disease Prevention and Transportation:

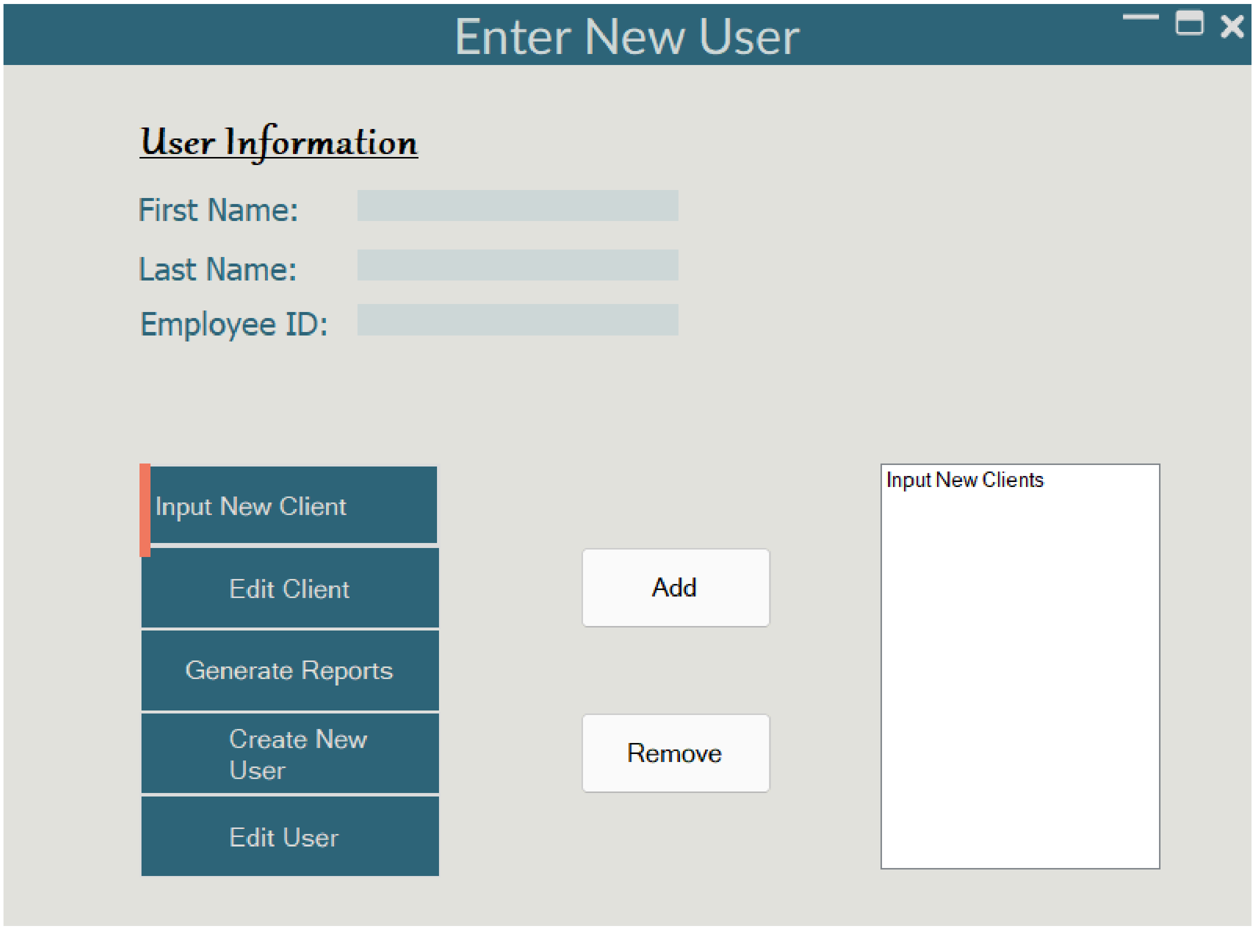
Information, Referal and Assistance (IRA):

Legal Services:

4.1.5 Create New User

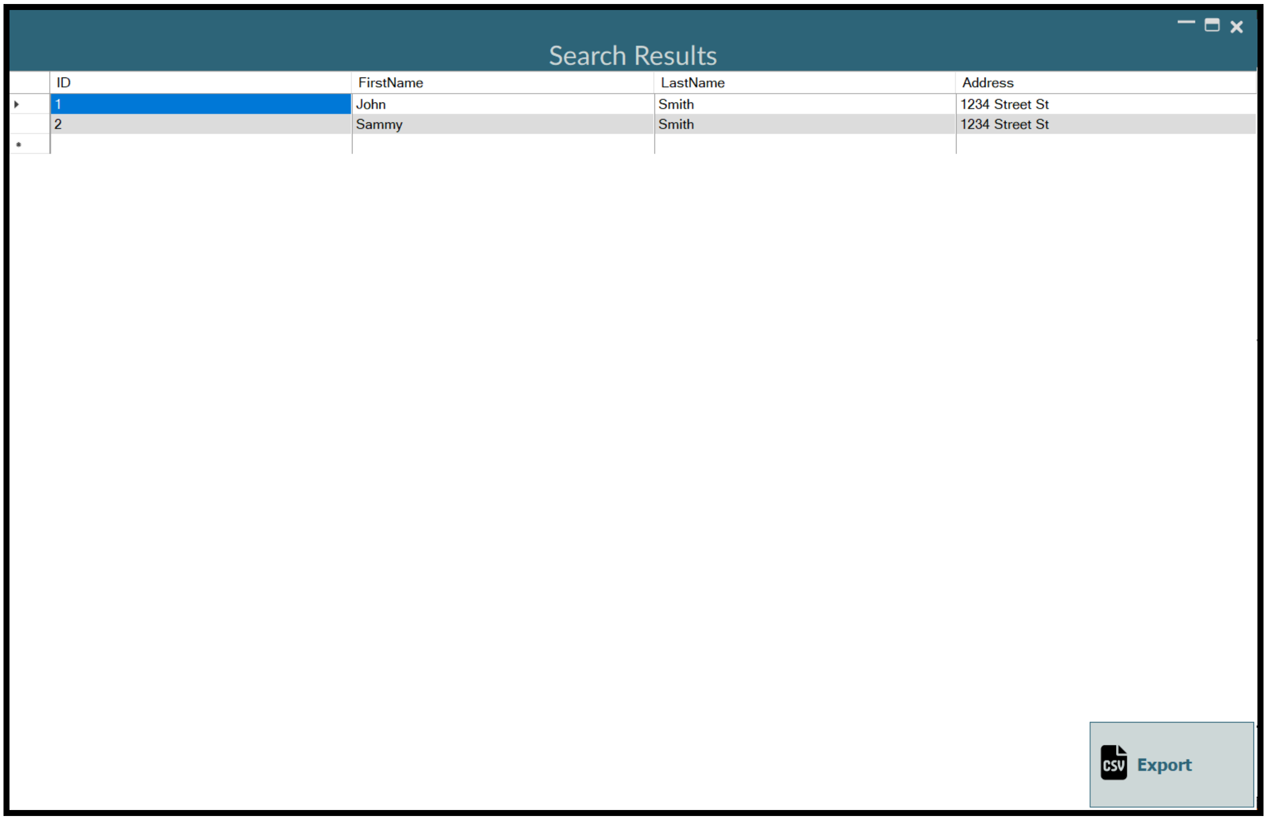
The interface for creating a new user is shown below. As previously mentioned in Section 4.1.2, this function can only be accessed by an administrator account initially. The interface features the ability to add the user's first and last name, as well as their associated employee ID/email address.

Furthermore, the privileges of the account being entered can be edited to allow for as restrictive or open access as desired. This gives the administrator flexibility in choosing who is allowed to access certain parts of the system. For instance, if the administrator chooses to, they can allow another account to edit users, effectively creating another administrator account. The decision to allow the administrator to modify permissions will help keep the system maintainable if the size of the database increases. For more information about the original administrator account and permissions, see the access control matrix in Section 3.2.



4.2 Data Presentation

Data will be presented to the user in a database format. When the user requests information from the database, the system will query the database and return it to the results page. The user will be able to export the queried information to a CSV file for further processing of the data. For result amounts exceeding the window height, a scroll bar will be used to allow the user to view more results.



ID	FirstName	LastName	Address
1	John	Smith	1234 Street St
2	Sammy	Smith	1234 Street St

4.3 Exception Handling

The system will focus on value and type checking as often as possible when taking inputs. An example of this implementation is the required parameters that are discussed in section 4.1.3 along with the data types used in the C# language.

For example, a client's date of birth will be gathered as three different integer values. If the user does not provide any values for the date of birth, an error message will display informing the user as such. If the user attempts to input a string of characters, a float, or any other input that is not an integer, an error message explaining the expected type input will be displayed and the user will be prompted to re-enter the parameter. The error message format will be similar between all error messages, but the contents of the message will vary depending on the issue detected.

Alongside Team 2's goal of good documentation discussed in Section 1.2.3, good documentation of errors will also be implemented so that the system is as reliable as possible. Error messages are not yet implemented, but will be a large focus of later testing stages.

4.4 Language and Rules

As mentioned in both the previous section and Section 1.3, the system will be developed using the C# and SQL languages. Since both of these languages are strongly typed, it is critical that type checking be implemented for every input. Inputs from the user will be gathered in C# and checked for type and format (such as in the case of dates of birth and the DATE type in SQL) before being used to query the database with SQL.

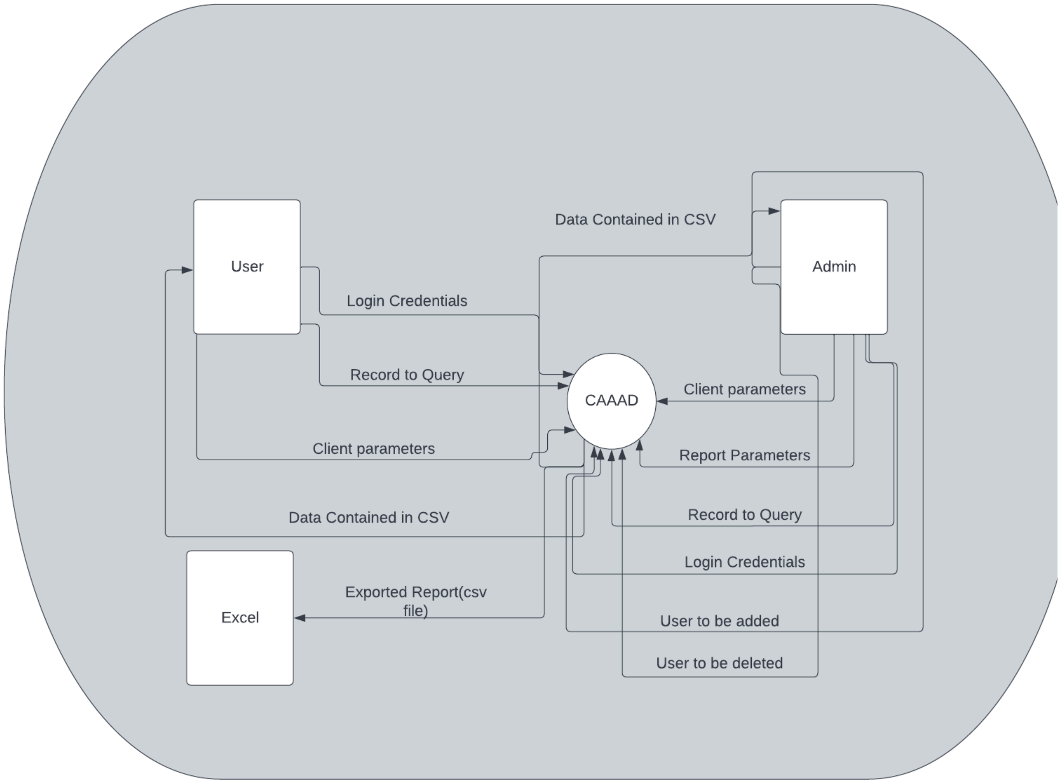
Team 2's current goal is to handle all type checking in this manner to avoid any redundancies or repetition in the programming that could negatively affect the system and lead to faults or failures. Alongside their emphasis on type checking, Team 2 will ensure that error messages are appropriately targeted for the characteristic user defined in Section 2.1. That is to say, that the output of the system will be given in terms that are explicit and unambiguous to the characteristic user.

5. Architectural/Detailed Design

5.1 System Overview

The system will contain functions that fulfill the functional requirements listed in Section 2.2. Elements that interact with the system and these functions are illustrated below with their appropriate interfaces and actions labeled.

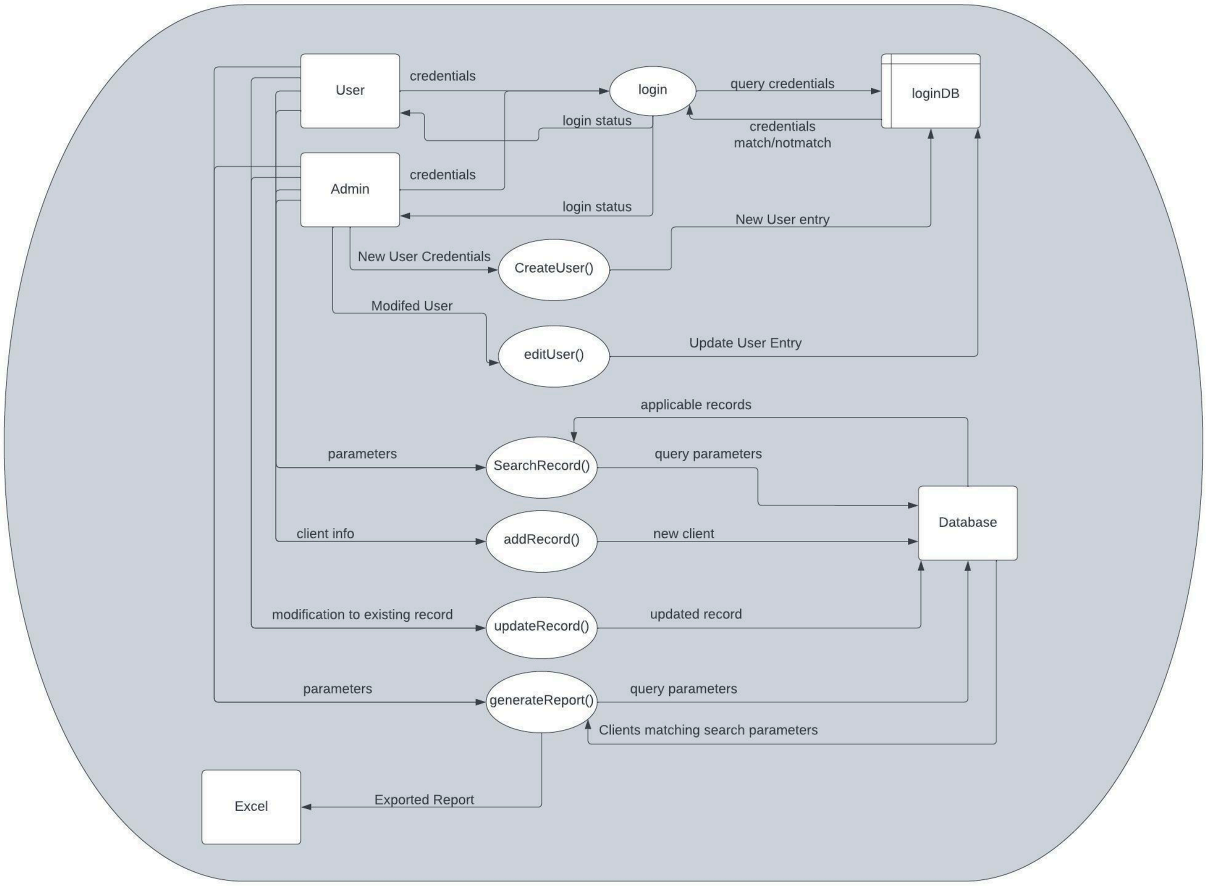
Specifically, the system will interface with the database, .csv files, the user and administrator accounts:



5.2 Subsystem Design

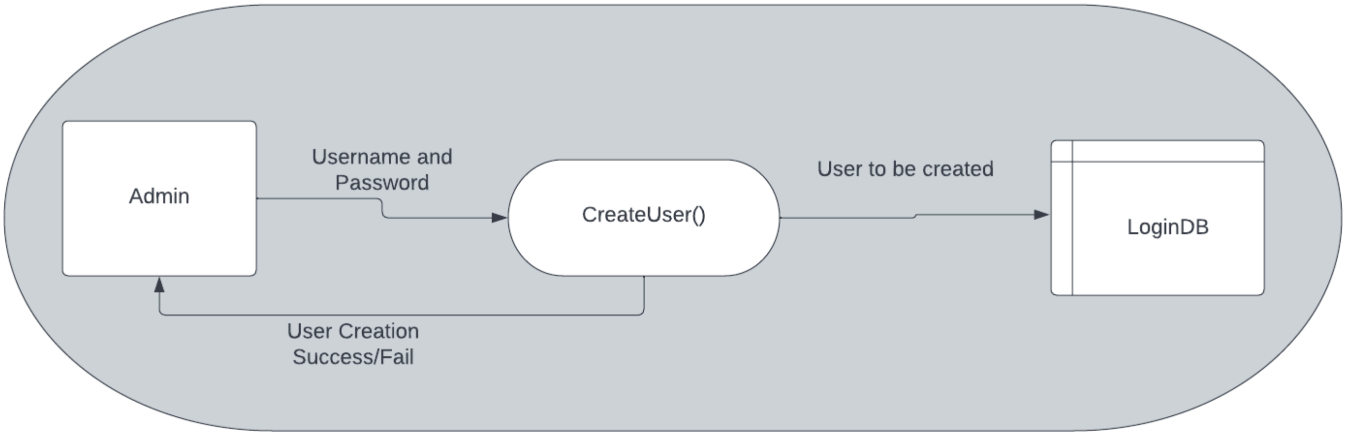
The subsystems involved can be broken down into the primary actions performed through the system. The access to these subsystems is listed in Section 3.2 as well as illustrated in the diagram below.

Specifically, the system contains the login, createUser, editUser, searchRecord, addRecord, updateRecord, and generateReport subsystems:



5.2.1 Diagram for CreateUser Subsystem

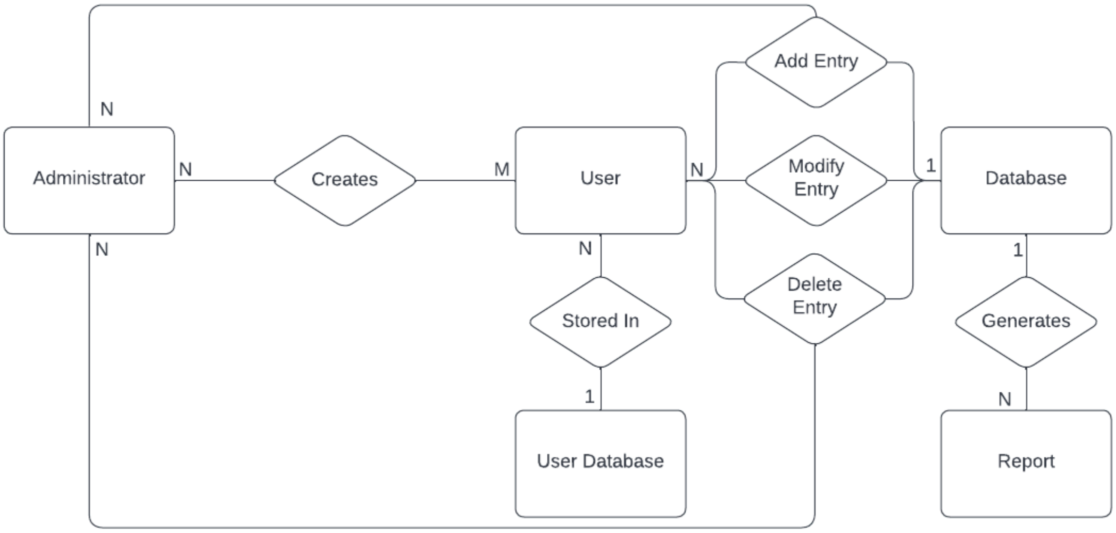
The following diagram is an abstraction of one subsystem. This general abstraction can be applied to any of the subsystems and shows how the above graph can be interpreted. Usernames will consist of the employees firstname and last initial while passwords will be made up of a combination of letters and numbers and at least one special character.



5.3 Persistent Data Management

Since the database will be developed using SQL, Team 2 will be relying on the methods available for this language to store and retrieve items in its expected format. The setup of the necessary file structures will be handled by Team 2 upon delivery of the system. This setup will either be a physical setup or an installer that is prepared prior to deployment.

The general structure of the system's data and interaction of types is shown below:



5.4 Client Parameter Data Types Matrix

Field Name	**Field Description**	**Validation**	**Type/Length**	**Required**
Last Name	Client's Last Name	Yes	VARCHAR, ()	Yes
First Name	Client's First Name	Yes	VARCHAR, ()	Yes
Address	Client's Address	Yes	VARCHAR, ()	No
City	Client's City	Yes	VARCHAR, ()	No
State	Client's State	No	ENUM, (All 50 states listed by their two letter identifier)	No
Zip Code	Client's Zip Code	Yes	CHAR, (5)	No
Phone Number	Client's Phone Number which can be used to contact the client	Yes	CHAR, (10)	No
Date of Birth	Client's Date of Birth	Yes	DATE	Yes
Income	Client's Income	Yes	MONEY	No
Marital Status	Client's Marital Status	No	ENUM, (Married, Single, Divorced, Widowed)	No
Size of Household	Client's Size of Household	Yes	Int, (2)	No
Medicare	Client's Medicare	Yes	VARCHAR, ()	No
Medicaid	Client's Medicaid	Yes	VARCHAR, ()	No
Social Worker	Client's Social Worker	Yes	VARCHAR, ()	No
In-Home Services Received	Client's In-Home Services Received	No	BOOL, (True or False)	No
Emergency Contact Name	Name of person to contact in case of a client emergency	Yes	VARCHAR, ()	No
Emergency Contact Phone Number	Phone number of person to contact in case of a client emergency	Yes	CHAR, (10)	No
Emergency Contact’s Relationship to Client	Client's Emergency Contact’s Relationship to Client	Yes	VARCHAR, ()	No

Field Name	**Field Description**	**Validation**	**Type/Length**	**Required**
Last Name	Client's Last Name	Yes	VARCHAR, ()	Yes
Comment	Client's Comment	No	VARCHAR, ()	No
Information, Referral and Assistance (IR&A)	Is the client receiving Information, Referral and Assistance (IR&A) services?	No	BOOL, (True or False)	No
Medicare Counseling (SHIP)	Is the client receiving Medicare Counseling (SHIP) services?	No	BOOL, (True or False)	No
Caregiver	Is the client receiving Caregiver services?	No	BOOL, (True or False)	No
Ombudsman	Is the client receiving Ombudsman services?	No	BOOL, (True or False)	No
Assessments	Is the client receiving Assessments services?	No	BOOL, (True or False)	No
Home Delivered Meals	Is the client receiving Home Delivered Meals services?	No	BOOL, (True or False)	No
Congregate Meals	Is the client receiving Congregate Meals services?	No	BOOL, (True or False)	No
Home Care	Is the client receiving Home Care services?	No	BOOL, (True or False)	No
Evidenced Based Health Promotion/Disease Prevention and Transportation	Is the client receiving Evidenced Based Health Promotion/Disease Prevention and Transportation services?	No	BOOL, (True or False)	No
Legal Service	Is the client receiving Legal Service services?	No	BOOL, (True or False)	No

6. Requirements Traceability Matrix

The requirements traceability matrix on the following page will serve as a guide throughout the development of the CAAAD system and serve as the primary method of measuring the progress of development. The specific matrix below represents the current status of the system, and when development of the system is complete, all statuses will be “Complete” - representing the finished state of the system. Each of the requirements’ reference IDs refer to their respective section number in this document.

Project Catawba Area Agency on Aging Database						Team 2		
Requirements Information and Status								
Req. ID	Req. Desc.	Ref. ID	Date Estab.	Design pg#	Validation Method	Status	Status Date	Prod. Date
1	Database	2.2.1	Oct.22	18	Inspection	Initial	Nov.22	TBD
2	Import Records	2.2.2	Oct.22	NYI	Testing	Initial	Nov.22	TBD
3	Input Records	2.2.3	Oct.22	10	Testing	Initial	Nov.22	TBD
4	Modify Records	2.2.4	Oct.22	NYI	Testing	Initial	Nov.22	TBD
5	Delete Records	2.2.5	Oct.22	NYI	Testing	Initial	Nov.22	TBD
6	Resolve Duplicate	2.2.6	Oct.22	NYI	Testing	Initial	Nov.22	TBD
7	Filter Database	2.2.7	Oct.22	10	Testing	Initial	Nov.22	TBD
8	Generate Reports	2.2.8	Oct.22	NYI	Testing	Initial	Nov.22	TBD
9	Intranet	2.3.1	Oct.22	NYI	Inspection	Initial	Nov.22	TBD
10	Access Control	2.3.2	Oct.22	8	Inspection	Initial	Nov.22	TBD
11	Ease of Use	2.3.3	Oct.22	NYI	Inspection	Initial	Nov.22	TBD
12	Query Time	2.3.4	Oct.22	NYI	Testing	Initial	Nov.22	TBD
13	Storage	2.3.5	Oct.22	NYI	Inspection	Initial	Nov.22	TBD
14	Scalable	2.3.6	Oct.22	NYI	Inspection	Initial	Nov.22	TBD

*NYI - Not yet implemented

7. Glossary of Terms

The definitions of commonly used terms in this document can be found below. Only the first instance of a term is underlined to avoid cluttering.

Client: The individual person whose information is being used in the database.

Client parameter: See glossary entry for "parameter".

Client record: See glossary entry for "record".

CSV file: a text file in which the data fields are separated by commas

Developer: Team 2: Justin Carmona, Jordan Douglas, AJ Ellis, Preston Fisher, Walker Immel, Christian Morris, Donnie Sullivan

Duplicate record: A record in the database that has the same [Name] and [Address] as another record.

Excel Files: An Excel file (.xls, .xlsx, etc) containing client parameter data.

Intranet: A local or private computer network.

NYI: Not Yet Implemented - feature not demonstrated in the current prototype.

Parameter: An individual field associated with a client representing one of the following data types:

1) Last Name
2) First Name
3) Address
4) City
5) State
6) ZIP Code
7) Phone Number
8) Date of Birth
9) Income
10) Marital Status
11) Size of Household
12) Medicare
13) Medicaid
14) Social Worker
15) In-Home Services Received
16) Emergency Contact Name
17) Emergency Contact Phone Number
18) Emergency Contact's Relationship to Client
19) Comment
20) Information, Referral and Assistance (IR&A)
21) Medicare Counseling (SHIP)
22) Caregiver
23) Ombudsman
24) Assessments
25) Home Delivered Meals
26) Congregate Meals
27) Home Care
28) Evidenced Based Health Promotion/Disease Prevention and Transportation
29) Legal Service

Record: A collection of parameters that is associated with a client in the database.

Report: Output of filtered client data by specific parameter(s) as a CSV file and presented in a readable manner in our program UI.

User: Catawba Area Agency On Aging staff member.