

Group #4  
Christopher Juncker  
Justin Greever  
Samantha Zeigler  
Tori Anderson  
Naya Mairena  
Ian Guy  
Dan Jang

# Term Project: *ChocAn*

## Design Document

### Table of Contents

<b>Introduction</b>	<b>2</b>
Purpose and Scope	3
Target Audience	3
Terms and Definitions	4
<b>Design Considerations</b>	<b>6</b>
Constraints and Dependencies	6
Scheduling	6
Member Error	7
Provider Error	7
Methodology	7
<b>System Overview</b>	<b>9</b>
<b>System Architecture</b>	<b>12</b>
User Interface	13
Simulated Terminal	13
Authentication & Authorization	13
Login	13
Role Checking	14

Form and Menu Manager	15
Data Validation	15
Information Retrieval & Modification	16
Patient Info Manager	16
Provider Info Manager	17
Data Import & Export	18
Report Generation	18
Transaction Management	19
Patient Database	20
Provider Database	20
Services Database	20
<b>Detailed System Design</b>	<b>21</b>
User Interface	21
Authentication & Authorization	21
Information Retrieval & Modification	22
Transaction Management (Database)	24

# Introduction

This document describes the architecture and design of the Chocoholics Anonymous' data processing software. This software will be used by ChocAn to manage data of the members/providers in several methods. Going through the limitations of the system and what needs to be considered when implementing. What methodology was used and why the team chose it. Using diagrams to show high level abstractions of the overview of the system. Abstraction used to understand the architecture of the software. The architecture is separated out into different subsystems and components. The final section is a detailed description of how the system is designed. It includes details such as previously shown diagrams, pseudo code, and helper functions implemented in the system.

## Purpose and Scope

The purpose of this document is to provide a detailed description of the ChocoAn system. The level of detail being that it allows for development to proceed knowing how the software is going to be built and expectations. There are different models and diagrams used to start conversation with the stakeholders. The abstraction makes it easy to understand when discussing the system by producing a model of the system that incorporates the many components and their relationships between them.

## Target Audience

The target audience for this document is the stakeholders of the ChocAn system. This includes the management team and people using the software such as a ChocAn provider. Discussing the architecture of the system and why the chosen model is the best one for the system's purpose. The members of ChocAn aren't in the direct conversation involving the architecture or design of the system. The members

would be able to make their own opinions of the system known later on in development. This later stage is possibly when maintenance and updates will be implemented to the system to improve it. Getting user feedback to see what members and providers think of the software product. The management will be able to discuss their thoughts on the system in this design phase. If there is anything they dislike or else needed in the system, this is the stage to confirm and nail in the fine details of it before implementing.

## Terms and Definitions

<b>Software Product (Software)</b>	The Chocoholics Anonymous data processing software which is currently under development.
<b>Group 4</b>	The organization which has been awarded the contract to develop the Software Product (us).
<b>Chocoholics Anonymous (ChocAn)</b>	An organization dedicated to helping people addicted to chocolate in all its glorious forms.
<b>ChocAn Data Center (CDC)</b>	The data center which contains member and provider information.
<b>Member</b>	A person who pays a monthly fee to ChocAn and is entitled to unlimited consultations and treatments with providers.
<b>Provider</b>	A health care professional who provides services to members.
<b>Provider Number</b>	A nine-digit number which uniquely identifies a provider.
<b>Member Number</b>	A nine-digit number which uniquely identifies a member.
<b>Member Card</b>	A plastic card given to members containing their name member number.
<b>Terminal</b>	A computer terminal operated by a provider which scans member cards and which serves as an interface to communicate with the Software Product.
<b>Manager Terminal</b>	A terminal with elevated privileges that can request

	individual reports at any time.
<b>Provider Directory</b>	An alphabetically ordered list of all service names, and their corresponding codes and fees.
<b>Member Directory</b>	A list of all ChocAn members, both active and suspended.
<b>Validated Member</b>	A member whose number has been verified by the CDC.
<b>Suspended Member</b>	A member whose number has been marked as suspended by the CDC. Usually due to not paying on time, can be rectified by contacting a Care Administrator.
<b>Service / Session</b>	A health care service that is provided to a member. These services/sessions are offered by a provider.
<b>Service Code</b>	A six-digit number uniquely identifies a service offered by a provider.
<b>Fee</b>	The amount that ChocAn pays a provider for a service.
<b>Membership Fee</b>	The monthly amount that a member pays to ChocAn.
<b>Main Accounting Procedure (MAP)</b>	A weekly procedure that reads in all services provided during the week, and creates reports.
<b>Member Report</b>	A weekly report was provided to each member who has consulted with ChocAn that week.
<b>Provider Report</b>	A weekly report is sent to each provider containing a list of the services they provided that week and a summary.
<b>Electronic Funds Transfer (EFT) Report</b>	A record of electronic funds transfer data which is written to disk for use by banking computers later.
<b>Summary Report</b>	A weekly report is provided to the manager of accounts payable listing all providers that need to be paid.
<b>Interactive Mode</b>	A software model that allows members to be added, deleted, and updated in the CDC.
<b>Milestone</b>	A point in the software production schedule where we can assess the project progress.
<b>Deliverable</b>	A point in the software products that can be delivered to the customer before the final product.

# Design Considerations

The purpose of design considerations is to keep track of possible issues, concerns, and details that may affect the software design process before it is finalized. The operational environment, development methods, and architectural design are the main things to consider when developing this design document. More specific examples of things that should be considered are performance, reliability, maintainability, security, and safety of the software. It is important to highlight specific constraints and dependencies in this design document. Being unaware of these can negatively impact the software if not addressed before finalizing the design.

## Constraints and Dependencies

Constraints can bring limitations to the software. They need to be addressed in this design document so that we are aware of the issues they may bring up. Dependencies are tasks in the software design that are dependent on each other and their relationship must be accounted for.

## Scheduling

Time is the main constraint to the software design because of the need for a schedule for delivering the milestones of this software development. This ChocAn data processing software must be delivered by December 3rd, 2021. Due to the limitation of time, the software must at least reach minimum requirements and be functional.

Within the software, a schedule must be incorporated to function with the third-party software. Every Friday at midnight, the software must prepare weekly reports for the ChocAn Data Center. Each evening at 9 PM, Acme accounting services update relevant ChocAn Data Center memberships. These tasks are dependent on a set schedule that must be incorporated into the software, otherwise, there will be constraints with the third-party software.

## **Member Error**

A constraint that could occur is the member forgetting their membership ID card. This will not allow for the provider to be able to access this member's information. This is something to be considered when implementing the software. We should allow for the providers to be able to search for the member based on their full name and check a form of ID to validate them. The interface is dependent on the member providing their ID card.

## **Provider Error**

Providers may cause an error when entering data about a member causing the constraint of false data. It is important to implement the software to double-check the data entered by the provider before accepting the data. There may be a chance that the provider accidentally enters the wrong information or the member gives the provider the wrong information. Having a check before finalizing the data entered is important to have more accurate data. Proper report generation is dependent on the accurate information of the member.

## **Methodology**

The software methodology that will be used is the waterfall model. The waterfall model is a plan-driven process where everything is planned and scheduled

before starting the software development. The first thing that was planned was the requirements document that was finalized on October, 25th 2021. The requirements document established the goals, definitions, specifications, and milestones for the ChocAn data processing software. The next step in the plan is the software design document. This current document is the design document for the ChocAn data processing software that will be delivered by November 8th, 2021. The design document is important for the process because it organizes the specifics for the ChocAn software and prepares the team for implementation. The design and implementation will assist in the creation of a test plan. The test plan must be finalized by November 15th, 2021. The test plan will promote the proper implementation of the software that will avoid user and developer errors. Using different methods of testing will discover bugs in the code and allow for improvements to the developing software. The final step in the model will be the operation of the ChocAn software and enhancing it as needed, which must be finalized by December 3rd, 2021. The waterfall model is perfect for this project because each step has a scheduled delivery date and there are no active requirement changes that will affect the process.

Architectural patterns are specific descriptions of system organization that has been inspired by other software systems. The one that will be used for this project is a layered architecture. Layered architecture pattern allows for the separation of the system into layers. This helps organize each system functionality into its layer. Each layer relies on the one below it. For the ChocAn data processing software, the layers will be User Interface, Authentication & Authorization, Information Retrieval & Modification, and Transaction Management (Database). The User Interface layer is for the functionality between the user and the system. In our ChocAn software, the target user is the providers and ChocAn management while the target system is the data processing software. The Authentication & Authorization layer is the portion of the software that confirms the user's identity



and level. The Information Retrieval & Modification layer manages the main functions of the software, such as providing the provider with data and manipulating reports. The Transaction Management (Database) layer is the portion of the software in which all of the program data is stored. The layered architecture is beneficial for our software system because of the organization of each layer. We can categorize each major component into its layer while each layer utilizes the layer below it. The Interface allows for access to the Authentication & Authorization layer which is essential for security before reaching the Information Retrieval & Modification layer. Then the Information Retrieval & Modification layer accesses the Database layer where all the data is stored. These different layers add security and maintainability to the data.

## **System Overview**

The ChocAn Data Processing Software is designed to perform numerous tasks to ensure the manipulation, storage, and retrieval of data is efficient, accurate, and secure. This software is designed to be used by ChocAn providers and managers to document services they provide and to whom, as well as the associated cost, fees, and fee due dates. This information is also collected and used to provide managers with reports that may be provided to their accountants, as specified in the requirements. Depending on who is logged into the software (provider or manager), this permits access to different amounts of information. Providers are allowed to access their data, the data for which members they have provided services, and the service database of service codes. Managers have access to all data that is entered into the software, not limited to providers data, members data, service code directory, and various reports used in the financial management of the ChocAn services. Various input forms are submitted by each user that is run through the various databases to display the requested data. There is an interactive mode that is

accessible by both providers and managers that allow data to be modified for both members' and providers' databases. Four database tables store the list of all providers' data, the list of all members' data, the list of what services were provided to which member by whom, and the service code directory. All four of these can be queried by providers and managers, and the four tables are used in the reporting functionality to display the information in a way the managers can use for financial purposes.

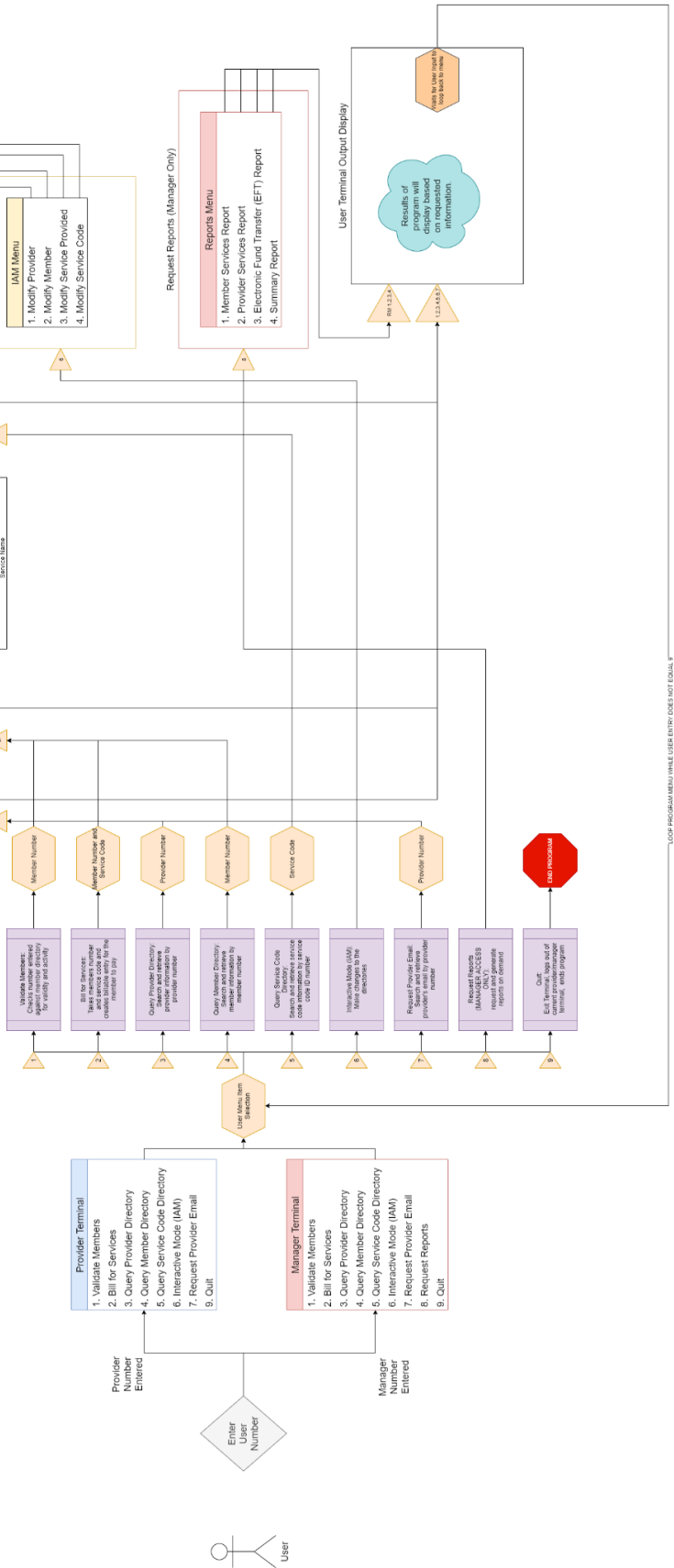
*(Diagram of Basic Flow of Data on Following Page)*

Provider Table Layout	
Description	Data Size & Type
Provider Name	25 Characters
Provider Number	9 Digits
Provider Street Address	25 Characters
Provider City	14 Characters
Provider State	2 Letters
Provider Zip Code	5 Digits
Total Member Consultation Number	3 Digits
Total Weekly Fee	<\$99,999.99

Services Provided Table Layout	
Description	Data Size & Type
Member Reference ID	9 digits
Provider Reference ID	9 Digits
Date of Service	MM-DD-YYYY
Service Reference ID	6 Digits

Member Table Layout	
Description	Data Size & Type
Member Name	25 Characters
Member Number	9 Digits
Member Street Address	25 Characters
Member City	14 Characters
Member State	2 Letters
Member Zip Code	5 Digits

Service Code Table Layout	
Description	Data Size & Type
Service Code Reference ID	6
Service Name	9 Digits



LOOP PROGRAM MENU WHILE USER ENTRY DOES NOT EQUAL 8

# System Architecture

The System Architecture of the ChocAn Data Processing Software is designed based on the non-functional requirements of the software. The architecture of the system is made up of four main layers:

1. User Interface
2. Authentication & Authorization
3. Information Retrieval & Modification
4. Transaction Management (Database)

The User Interface layer is the layer in which the user (an Operator or Provider) interacts with the system. The Authentication & Authorization layer is the section of the application which confirms the user's identity and permission level. The Information Retrieval & Modification layer comprises the main functions of the Software, responding to Operator queries and updating Provider and Member information. Lastly, the Transaction Management (Database) layer is the portion of the software in which all of the program data is stored. An in-depth discussion of each layer and its major subcomponents can be found below.

## **User Interface**

The Group 4 software team's contract specifies that the user interface for the ChocAn software product will be designed and implemented by another group. Until such a time as the user interface section of the software is completed, the Data Processing Software will be tested via terminal simulation.

## **Simulated Terminal**

The Data Processing software will be tested by the creation of a simulated terminal. The simulated terminal will receive data via keyboard input and will display data by printing it to the screen. It is intended that at the time of completion of the final product that the Provider's and Manager's terminals will be separate pieces of hardware. However, it has been agreed that the same screen simulation will be used for both Provider's terminals and Manager's terminals at this time.

## **Authentication & Authorization**

The ChocAn data processing software will not be made available for patients or the general public. Instead, it will be solely used by authorized Providers and Managers. The sections below outline the protections provided by the system to assure that it is only accessed by the intended users.

### **Login**

Healthcare Providers will log in to the terminal by entering their provider number. A provider number is a nine-digit number that uniquely identifies an authorized provider. If the software receives an invalid provider number then access to the system will not be granted. The software will display the login screen again to

allow the provider to perform additional login attempts. At this time it is not required that the provider provides a password or any additional identifying information besides his/her provider number.

## **Role Checking**

After a successful login, the Data Processing Software will perform a role checking routine. The purpose of this routine is to determine the role of the Provider. At the present, the software is required to support two separate roles. The first role is the primary role, which is of a Provider. A provider has the following permissions:

- Validate Members
- Bill for Services
- Query Provider Directory
- Request Provider Directory Email
- Modify Member Records (Interactive Mode)
- Modify Provider Records (Interactive Mode)
- Query Service Code Directory

The second role is that of a Manager. A provider who logs in with a provider number that has been designated as holding a Manager role will obtain all of the permissions granted to a Provider plus the following additional permissions:

- Generate Reports

## **Form and Menu Manager**

The Form and Menu Manager will keep track of the various forms and menus which need to be filled out or displayed to the software user. The software will need to generate the following menus:

- Provider Activity Selection Menu
- Manager Activity Selection Menu
- Report Generation Selection Menu

The software will need to generate and process data returning from the following forms:

- Login Form
- Provider Directory Query Form
- Patient Validation Form
- Patient Billing Form

When in Interactive Mode, the software will also require forms for the following activities:

- New User Creation
- User Modification
- User Deletion
- New Provider Creation
- Provider Modification
- Provider Deletion

## **Data Validation**

To ensure the safety and security of the software system, all data which is entered into the system will be validated. The following tests will be performed on a case-by-case basis as necessary to ensure that provided data is both safe and correctly formatted:

- Check that data is numeric only.
- Check that data is alphanumeric or conforms to a list of approved characters.

- Check that data is of a sufficient length.
- Check that data does not exceed a maximum length.

## Information Retrieval & Modification

### Patient Info Manager

The Patient Info Manager will keep track of the patient data necessary to provide services to patients and to fill reports with necessary patient data. The Patient Info Manager will manage the following information for each patient:

Description	Data Size/Type
Member Name	25 characters
Member Number	9 digits
Member Street Address	25 characters
Member City	14 characters
Member State	2 letters
Member ZIP Code	5 digits

Additionally, the Patient Info Manager shall keep track of the following data for each service that is provided to a patient:

Description	Data Size/Type
Date of Service	MM-DD-YYYY
Provider Name	25 characters
Service Name	20 characters



## Provider Info Manager

The Provider Info Manager will keep track of the provider data necessary to query the Provider Directory and to populate reports with the necessary provider data as specified in the reports section (below). The Provider Info Manager will keep track of the following information on a per provider basis:

Description	Data Size/Type
Provider Name	25 characters
Provider Number	9 digits
Provider Street Address	25 characters
Provider City	14 characters
Provider State	2 letters
Provider ZIP Code	5 digits
Total Member Consultation Number	3 digits
Total Weekly Fee	up to \$99,999.99

Additionally, the Provider Info Manager shall keep track of the following data for each service that a Provider has offered to a Patient:

Description	Data Size/Type
Date of Service	MM-DD-YYYY
Date and time data were received by the computer	MM-DD-YY HH:MM:SS
Member Name	25 characters
Member Number	9 digits

Service Code	6 digits
Fee	up to \$999.99

## **Data Import & Export**

Per Group 4's contract, all data import at this time will be performed via keyboard input on a simulated terminal. Patient and Provider data will be input individually and sequentially. The process by which ChocAn's current provider and member lists will be imported into the system has not been discussed at this time and currently falls outside of the scope of Group 4's contract.

Data will be exported by this software in two main forms. First, data will be exported to the simulated terminal when an Operator is logged in and interacting with the software. Second, data will be exported in batches in the form of reports which are generated by the software. At present, it has been agreed that generated reports will simply be stored on a disk. In the future, the report files will also be sent as email attachments. This is again outside of the scope of Group 4's contract.

## **Report Generation**

The ChocAn Data Processing Software shall generate a variety of reports. These reports will be generated on a set schedule. These reports can also be generated on demand by the request of an Operator with Manager privileges. The software will be capable of generating the following reports:

1. **Member Services Report.** This report is sent weekly to every member who received services that week. The member services report contains all information relevant to that member. Please refer to the Patient Info Manager

section above for a complete list of the data which will be provided in this report.

2. **Provider Services Report.** This report is sent weekly to every provider who has provided services to patients that week. The provider services report contains all information relevant to that provider. Refer to the Provider Info Manager section above for a full list of the data which will be provided in this report.
3. **Electronic Funds Transfer (EFT) Report.** This report is saved to disk and contains the amount owed by ChocAn to each provider for their services that week.
4. **Summary Report.** This report is provided to the accounts payable manager. The summary report contains:
  - Every provider is to be paid that week.
  - The number of consultations each provider provided.
  - The total number of providers who provided services.
  - The total number of consultations.
  - The total overall fee across all providers.

## **Transaction Management**

All data transactions will be stored in one of three databases which are operated by the ChocAn Data Management Software.

### **Patient Database**

The Patient Database will store all patient data used by the Patient Info Manager, whose activity as outlined above. The Patient Database will contain an entry for each patient and will be indexed by Member Number.

## **Provider Database**

The Provider Database will store all provider data used by the Provider Info Manager, whose activity is also outlined in the sections above. The data in the Provider Database will be used to populate reports and to generate responses to queries to the Provider Directory. The Provider Database will contain an entry for each provider and will be indexed by Provider Number.

## **Services Database**

The Services database will store data for all services provided by ChocAn. The Services Database will have an entry for each service that is provided by a Provider to a Member. The service database will be searchable by Provider Number or by Patient Number. This is because both the Patient Info Manager and the Provider Info Manager require information about services given to function correctly.

# **Detailed System Design**

## **User Interface**

The user interface will be provided to the user in a text-only console window. Using Python 3, we can ensure the software will run on virtually all hardware systems at the user's facility, as well as make updates and/or bug fixes to the

software easy to redeploy. Proper setup and deployment of the terminal devices will be managed by the users or a third party.

## Authentication & Authorization

Authentication & Authorization will be handled by taking user input (User ID) and checking if that User ID is in the authentication database and if the User ID is valid (active). Once a User ID has been validated, a flag will be set inside the user management module that will allow said User ID access to the system that is allowed within their permissions. Permissions will be added to the User ID field in the authentication database to determine if the user has read/write permissions, what data they can access and/or modify (User/Patient/Provider/Services).

```
BEGIN
procedure authuser
  IF UserID is valid AND UserID has R OR RW flag
    Set UserIsValid flag to authenticate UserID
    RETURN Success

  ELSE IF UserID is NOT valid OR UserID does NOT have R/RW flag
    Display error "Invalid User ID or No Access."
    Increment UserLoginFail flag + 1
    Restart procedure authuser

  ELSE
    IF UserLoginFail equals 3
      Display error "Login attempts failed. Please contact support for help."
      RETURN Fail
END
```

From here, the users' terminal will present them with a prompt to select Provider activities, Manager activities, and Report generation. They will be described in the next section.

## Information Retrieval & Modification

After the User ID is authenticated, the user will be presented with a menu of options based on their access level (READ/RW) as well as their security level (USER/MANAGER/SUPPORT).

```

BEGIN
  IF UserID is Authenticated
    Read UserID SecurityLevel

  IF SecurityLevel equals USER
    Display USER menu

  ELSE IF SecurityLevel equals MANAGER
    Display MANGER menu

  ELSE IF SecurityLevel equals SUPPORT
    Display SUPPORT menu

  ELSE
    Display error "No Security Level Access."
    RETURN Fail

  Return Success
END

```

When the user has USER access, they can access functions to add/delete/update a member, add/delete/update member billing, query provider services, and print specific reports. This is done by passing in either a MEMBER ID, NAME, TELEPHONE NUM. The calling function passes the user-supplied input to the database functions. The SQLite DB is then queried for the specific information from a specific table (patients/members/providers/services). The information requested is then parsed by the database functions so only the requested information is handed back to the calling function.

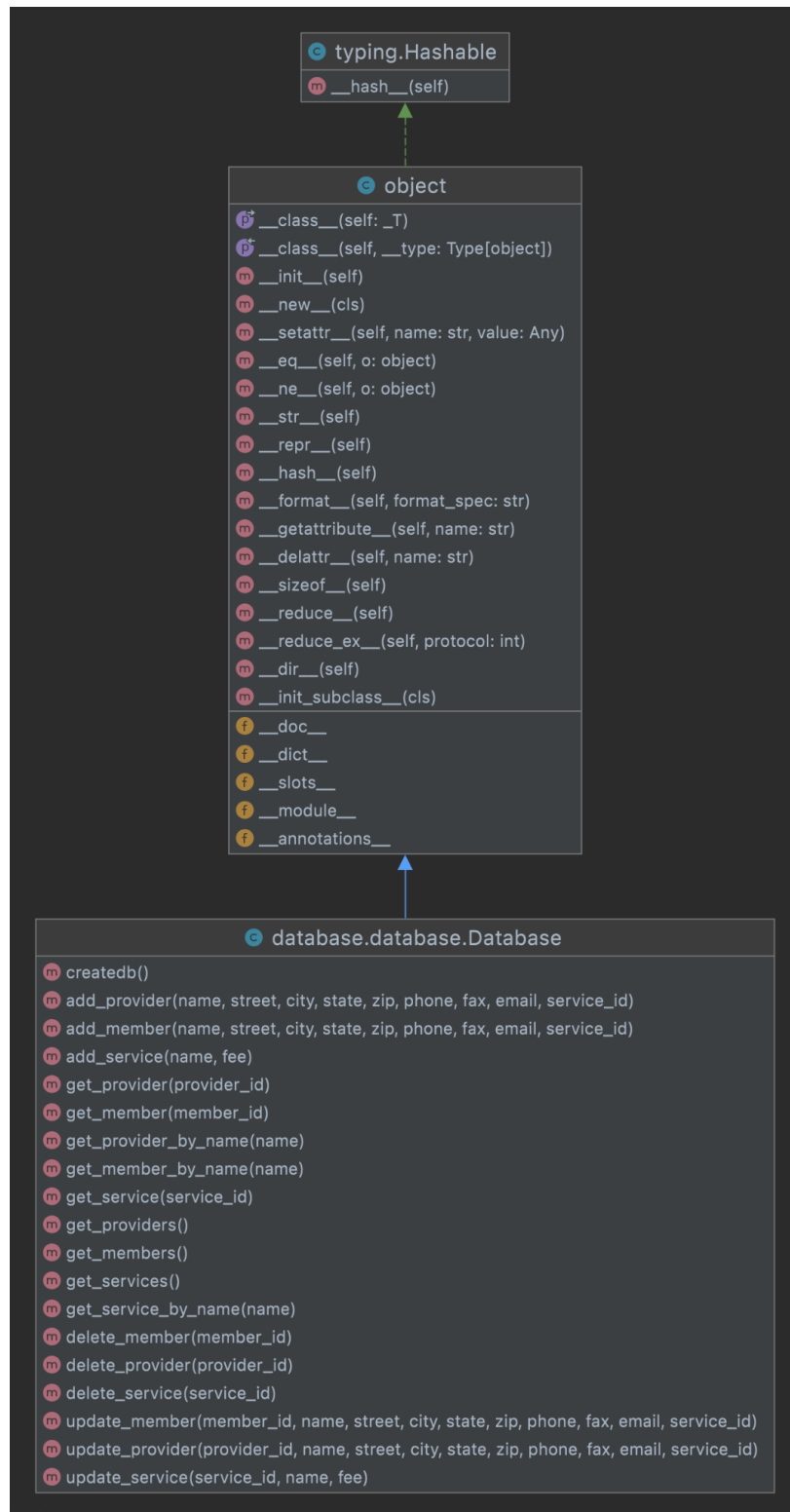
When the user has MANAGER access, they can add/delete/update members, services, billing information, providers, and generate detailed reports. When the Manager menu is selected, it will verify that the specific user has the permissions in the database for the MANAGER role, if not, it returns fail. If they have the permission, a local variable is set within the database functions that will stay until the user logs out of the system, which will then zero out the USERTYPE variable, requiring an authorization check to happen before accessing specific functions.

The SUPPORT role is responsible for report generation. Because this specific role has access to much more data than the previous roles, it requires both an

authorized MANAGER user to be logged in, but also a REPORTGEN ID to access all information required for report generation. This is an extra layer of security to ensure that permissions are partitioned properly on the system to prevent improper access. The database functions will check the MANAGER user to see if they have been flagged in the USERTYPE variable. If the flag is set, check the REPORTGEN ID passed in by the user. If it is valid, set another local variable MGRRPT (manager report) so the user doesn't need to authenticate for each report while they are logged into the REPORTS menu. Once they log out of that menu, zero out both MGRRPT and USERTYPE variables and require reauthorization to access menus on the terminal.

Database information modification works by passing in the information for the service/member/provider/billing that you want to edit. Once it's found, the user will be able to view the current information on the terminal and decide what to update by passing in the value to change. The database will update the fields and print the new information to the terminal for the user to confirm correctness.

# Transaction Management (Database)



UML Diagram of the Database functions and class in Python3



The database will be an SQLite FLAT Database (stored to a file). Access to the database is restricted to only the Database functions within the database class. This partitions the database from the user terminal. The first function called when the system is initialized at the clients' location is the `createdb()` function. This will check if the database currently exists on file, and if so, ensure the proper tables and structures are created. If no file exists, it creates the file, then creates the tables and structures.

At this point, the database will be entirely blank. It will be the user's responsibility to enter in any needed information about providers, services, members, billing, etc. via the terminal at their location. Entering the required information will be done via the menu systems. A user will select their desired action, and it will prompt the user to input the needed information via their terminal, pass it to the database functions, enter the information in the database, then return the information stored for the user to confirm correctness.

The database creates the PRIMARY KEY for each entry by hashing a specific value and using the hash as the PRIMARY KEY. The Member ID will not require hashing to be stored. Hashing will only occur for the NAME field of the Service and Provider, which will then become the Service ID and Provider ID for each entry respectively.

Deletion of data within the database will involve the information about the specific service, provider, member, etc. to be flagged as INACTIVE for a set period determined by the stakeholders, users, and other third parties. This requirement is due to data retention and allows the customer to retain information after deletion for reporting purposes. From there, either the set period passed and the data is fully removed from the database, or the MANAGER will be able to manually delete the

information via a menu option. The database functions remove the entire row associated with the removal request instead of zeroing out the fields and leaving the row structure intact. This is to both preserve data integrity and reduce file size for larger providers.

```
SELECT id  
CASE WHEN to_delete = TRUE THEN 1 ELSE 0 END AS del_flag  
FROM table  
ORDER BY id
```

*Flag row for deletion so information is not accessible but not removed from the database yet*

```
DELETE FROM table  
WHERE search_condition;
```

*Deletion of the entire row from Database table after X days or when MANAGER manually performs the action*