# 3 Functional Requirements

This section outlines and defines the softwares functionality and behavior. Each requirement is a functionality that the software must provide, and contains a description of that functionalities behavior in the system.

## 3.1 Terminal User-Interface (UI)

Terminal based software that provides 3 different UI's that correspond with the user-type. A Manager terminal, Provider terminal, and Member terminal, all provide different accessible features.

### 3.1.1 Login

Upon software execution, the user is prompted to login using their User-Number. The terminal accessed corresponds to the User-Number, i.e. Provider numbers correspond to the Provider terminal, etc… If valid number is provided, "VALIDATED" appears, and the user is logged in to the appropriate terminal. If invalid number is provided, "INVALID NUMBER" appears, and if a Member is suspended, "MEMBER SUSPENDED" is displayed.

### 3.1.2 Terminals

Once logged in, the user is given access to features that correspond with their user-type. Terminal is navigated via option-select, i.e. entering characters that correspond with actions displayed.

## 3.2 User Objects

User objects contain data that correspond with every user-type, and are at the top of the user hierarchy. Contains name, and user-ID number data members.

### 3.2.1 Provider Object

Contains provider type, either Dietitian, Internists, or Exercise Specialist, and hold list of services offered (Pointer to service list). Has street address, city, state, and zip data members.

### 3.2.2 Member Object

Contains member street address, city, and zip-code. Tracks account balance.

### 3.2.3 Manager Object

Has access to account reports, holds summary report for current week.

```
                        User Object

                    String   name;
                    Uint     id;


        Provider              Member              Manager

   String    type;       String    street;    report*   current;
   String    street;     String    city;
   String    city;       String    state;
   String    state;      uint      zip;
   uint      zip;        Int       balance;
   servList* list;
```
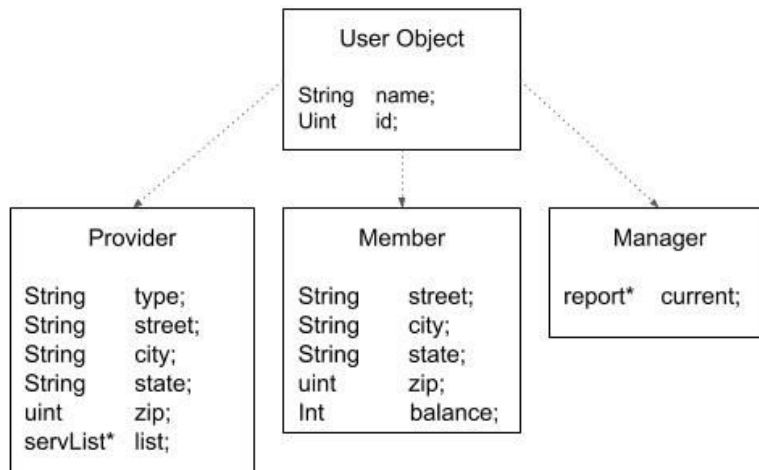
Figure 3.2.1 (a) User-Object hierarchy

# 3.3 Service List

A list of service objects. Each provider type has a unique service list that corresponds to the services offered by that provider. Alphabetically sorted by service name.

### 3.3.1 Service Object

Each service object contains data members that correspond with the service code, date, name, and fee (Up to $999.99).

# 3.4 User Storage

Each user type is stored in a file that contains the credentials for each user of that type. When logging in, the first 2 digits of the login code provided, corresponds with the user-type, and tells the system which file to search in. All members stored in member file, providers in provider file, and manager(s) in manager file. Each file is sorted alphabetically by user name.

# 3.5 Billing

When a member is billed, a provider is prompted to enter the members number; when "VALIDATED" appears, the provider is prompted to enter the date in the format "MM-DD-YYYY". The provider then enters the code that corresponds with the service provided to the member, which then searches the service list of the provider, for a match. When a match is found, the provider is then prompted to verify if the service found is the correct service. If an incorrect code is entered, and no match is found, an error is displayed, and the provider is prompted to enter another code.

### 3.5.1 Writing Record to Disk

Once the information in **3.5** is filled out, a record of the transaction is then recorded to the disk, containing the following data:
- Current date and time (MM-DD-YYYY HH:MM:SS).
- Date service was provided (MM-DD-YYYY).
- Provider number (9 digits).
- Member number (9 digits).
- Service code (6 digits).
- Comments (100 characters) (optional).

The system then searches for the verification code entered, and displays the corresponding fee on the provider terminal.

### 3.5.2 Billing ChocAn

At the end of the week, the records for that provider are totalled up, and sent to ChocAn to be paid. Thus, each provider keeps a list of records.

### 3.5.3 Member Service Report

At the end of each week, if a member has received services that week, they are sent a report of all the services they received, order by the service date. The report contains the following data:
- Member name (25 char limit).
- Member number (9 digits).
- Member street address (25 char limit).
- Member city (14 char limit).
- Member state (2 char limit).
- Member zip (5 digits).
- For each service, the following is required:
    - Date service was provided (MM-DD-YYYY).
    - Provider name (9 digits).
    - Service name (20 char limit).