**ChocAn Software User Manual**

**Special Note about the .jar file:** The .jar **must** be located in the main fall2020team18 directory instead of in the release folder for it to function. This is because of how the file paths are set up for the persistent databases.

When the program is first started, the user is prompted with three options for terminals and an option to quit. The terminals are as follows:

**Operator Terminal:** This terminal allows the users to manage one of two persistent databases, which are a list of providers and a list of members. The terminal allows for the addition of new members/provider, the editing of their information, and the deletion of their entries in the database. All management is done by user input.

**Provider Terminal**: This terminal allows the user to request the provider directory, which is a list of possible services along with their respective codes and fees, verify whether or not a member is in the database, and if they are suspended or not, and bill ChocAn. The bill ChocAn option allows the provider to add an entry into the service record. The verification requires the provider to enter a member number to be verified, and the billing process requires user input such as the date and the service provided.

**Manager Terminal:** There are two main options in the manager terminal. The first is labeled as “Produce Report”, which, when selected, allows the user to choose one of three report types. They are the manager, member, and provider reports, as well as the option to make all of them at once. The second main option other than “Produce Reports” is the option to say that the current time is Friday at midnight, which simulates the weekly report production that should occur on Friday at midnight as described in the software requirements. This option clears the weekly service record.

The user has the option to log out of any terminal to log into another terminal. The user may completely exit the program from the main terminal selection screen, and any changes made to any records or databases will remain. The only information that must be unique between members and providers is the number. No two members can have the same number and no two providers can have the same number. If the user attempts to add a new member with a number already in use, the program will inform the user that the number is already in use and will return to the member management screen. If the user tries to edit a member to have a number already in use, the same thing happens. These scenarios also occur if done with a provider. If a member or provider is deleted from the database while they still have an entry associated with them in the service record, the report will list the provider/member as having been deleted or modified in lieu of an actual report.

|  |  |  |
| --- | --- | --- |
| Student Names | Roles | Percentages |
| Bre Layton | VerifyMember class, ManagerTerminal class, Junit Test: ProviderTest | 17% |
| Jake Frances | Login class, OperatorTerminal class, ManageAccounts class, MemberTest Junit test | 17% |
| Henry Schenck | Class Diagram revision, Sequence Diagram revision, Databases(member/provider/directory/servicerecord), RequestingProviderDirectory class, User Manual | 25% |
| Thomas Davidson | Administrative, Account Class, Provider Class, Member Class, Timer Class, Report Creator Class, MemberDatabase JUnit test, Service JUnit Test, Sequence diagram revision, Project Debugging/SQA, ANT script, reorganized repository | 24% |
| Danny Braun | ProviderTerminal class, BillChocAn class, Service class, ProviderDatabase Junit Test, Sequence Diagram revision, Activity Diagram revision | 17% |