Health Digital Assistant Documentation

This is a document that contains information about the program as a whole that shouldn’t be included in any particular class because it is program logic as a whole. This document will be vague when it comes to details of the specific implementation of an area of code as that is covered in class documentation.

Writing Flow

The flow of the program is built on the idea that the GUI and any ‘above’ code such as the AI which will drive the user through the interview process, never has a direct connection to the database. The GUI will interact with a temporary class which matches up to a database table. Because there are usually many rows in any table I have created a class called collections. Each implementation of the collection class has a holder class object, which has an active database connection but no data, and an array list of the classes which have data but no database connection. For the GUI to write to the database it passes the object it wishes to write to the history object which then fills the holder object with data then using the active connection writes to the database then clears the data from the holder class, and re-initiates the array list with the new data.

Individual Class Layers

-Temp

This is the main class of any implementation, it has all the definition variables, private methods to read, write, and update, and get and set methods. It is intended that only objects that match up directly with a database table will be put in the temp class package. Each class in the temp package is an abstract class to be inherited by the display package.

-Display

This package contains two distinct implementations. Frist it will override the temp class and provide override set methods. These set methods will provide validation to the class, at first matching based off of individual regular expressions for each type of data.

The second use of the display package will be classes that match up with database views for reports and other similar functions providing get methods. Each class in the display package is an abstract class to be inherited by the interface package.

-Interface