**Design**

***See attached UML diagram Social\_Network\_UML***

**Classes:**

Profile – Parent of Dependent and Independent classes.

Connection – Instantiates Connection objects to store connections in profile objects.

ProfileDatabase – Parent of Driver class and engine of the program, stores the profiles, queries and manipulates them by calling methods from Profile class.

Driver – Generates menus and calls methods from super class ProfileDatabase.

MiniNet – Initiates social network program.

**Mechanics:**

The profile class is used to instantiate the profile as either Dependent or Independent profiles depending on age. I realised a bit too late that this design lacks foresight as it is not adaptable to profiles aging over time. The Dependent or Independent objects are stored in an ArrayList of profiles in the ProfileDatabase class. The Profile class stores all the attributes of a Profile: name, age, status, image, date of birth and an ArrayList of Connection objects. There is another Array of Strings that stores profiles children or parents.

The children of Profile class Dependent and Independent consist of constructors to handle dependency constraints and a single method for setting connections dependent on the age of the profile.

The methods in Profile class consist of the usual getters and setters as well as methods to add and print connections in the connections ArrayList. There is an abstract method for setting connections used to ensure dependent constraints are enforced on Dependent profiles when connecting to a profile.

The ProfileDatabase stores the ArrayList of Dependent and Independent profile objects. Methods are called by the child class Driver to perform methods from the Profile class on the Profile objects.

The method printProfList() is used to print the names of the profiles alongside a number for user input. I had a lot of trouble trying to get the numbers and names to display on the same line and eventually gave up on it. I would be interested in your advice on how to do this as hours of googling still couldn’t help me with this.

The Driver class prints menus and uses switch statements to utilise numeric keyboard input. The getInput() class handles errors from incorrect entry and is utilised throughout Driver class. The input is used to call methods on the ProfileDatabase class for adding, removing, updating and querying profile information.

*I learnt a lot doing this project and if I had time to start again would have used a different approach. Although my design definitely isn’t perfect it works for what was requested. Initially I attempted to utilise as many different oo concepts as possible and had a lot of difficulty before deciding to strip it back and keep it simple. Therefore I haven’t utilised things like interfaces or composition or nested classes and haven’t taken as much advantage of things like polymorphism as I would have liked.*