Assignment 1

Yunfei Zhou ID: s3598797 Manan Tyagi ID: s367421

Design for miniNet

Initial information for logging in the demo: Name: admin Password: password

Github: rmit-s3598797-yunfei-zhou/assignment1 (Collaborator : rmit-s3674211-manan-tyagi)

1. How the design will be able to store the profile information.

   Make Profile() method in Network class is used to add profile information of each individual in the network. The network is maintained in the form of a two dimensional string array called 'net' that stores name, age and status of each individual. A separate 2-dimensional string array called friends is created for storing the friends of each individual in the network. Both the arrays are updated to store the information stored in the database to create the initial network as soon as the program is executed. Now in order to further expand the network add user option can be selected from the menu class. This will keep on adding the details of the new users to the existing network. In this design it is mandatory that a dependent can only be added provided that his/her parents already exist in the network. So, if a dependent is added to the network the program automatically asks for the names of the parents who are already present in the existing network. And as soon as the names of the parents are given they are added to the friend list of the dependent (i.e. the update is made in the friends array).

   Initial information

Name	Age	Status	Friends
subhneet	32	business	Trapti, aarav, mana, sam
trapti	30	housewife	Subhneet, aarav, manan, anu
aarav	5	kindergarten	Subhneet, trapti (parents)
manan	25	student	Trapti, jacky, subhneet
sam	33	trainer	subhneet
anu	31	housewife	trapti
jacky	24	student	manan
yoyo	44	unemployed	

2. How your class hierarchy will facilitate the network management.

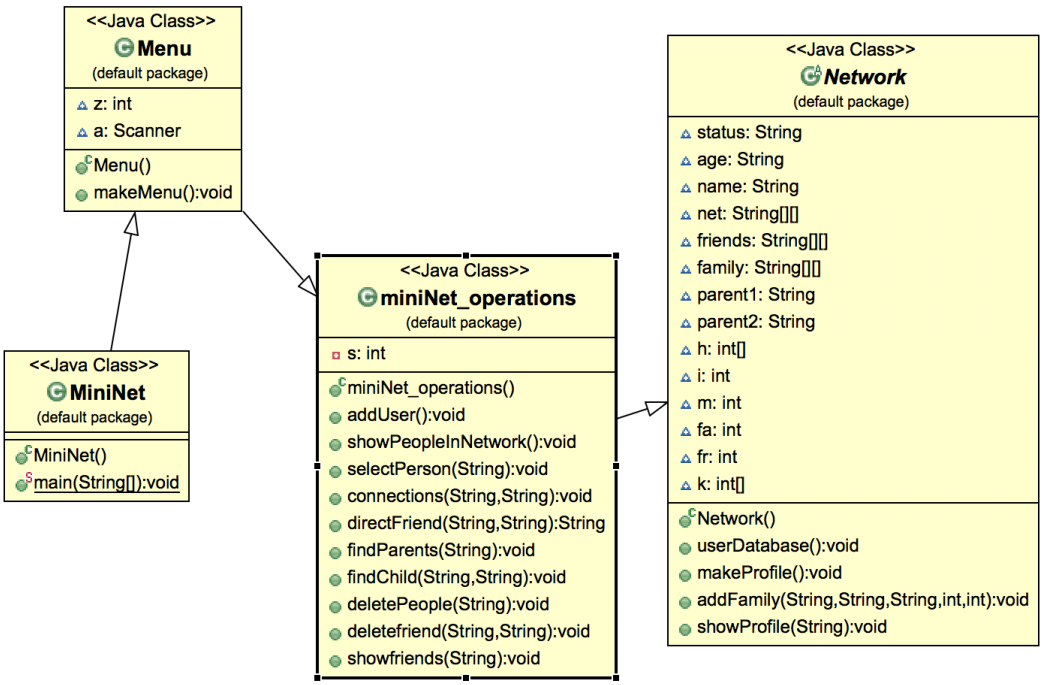
   There are four classes in total. The base class is Network from which mininet_operations is derived. Then menu is derived from miniNet _ operations and finally miniNet is derived from menu. Now the object created in main function of miniNet class is of type menu so that the methods of all the succeeding base classes can be accessed through it. Menu class contains a method called make_ menu that is accessed directly through the object in main method of miniNet class. This is how the user would be prompted for filling in the desired choice based on the activity that is needed to be performed on the network such as adding a new user, deleting an existing user, updating the profile of existing user etc. After selecting a choice, the control transfers to the corresponding method in the mininet_ operations that would be responsible for performing that particular activity. Now since the object is of type menu it can access all the methods of mininet_operations and network

class. This is how the network is managed in the proposed design.

3. The process by which the program can maintain the networks and find connections more efficiently.

   The network is maintained through nested inheritance and menu class. The make-menu option is directly accessed in main method of menu class that prompts the user to fill in the desired choice based on which the desired activity is performed. The methods corresponding to various choices are executed as soon as the choice is entered. All these methods are defined in the mininet_operations class. A function called addUser() is used to add a new user to the existing network. Connections method is used to make connection between to given members of the existing network. findParents() is used to find the parents of the given dependent in the network. findChild() is used to find the child/ children of the parents in the network and at last deletePeople() is used to delete the given member of the network. So, as it can be readily observed that findParents() and findChild() functions are used to find the family connections in the network and a function called direct friend is used to differentiate between immediate friends and mutual friends of all users in the network

4. Class diagram



5. Contribution 50:50
6. Reflection

   Yunfei Zhou

   That is the first time for me to make a code work as a group. It is a good experience. As we can know, there are lots of complex project we need to face in the future and it must be finished by a group. I learned that communication and design are better to be finished

before coding when work as a group. Because we need somethings which are made by another teammate when we code. Team work are more efficient.

Manan Tyagi

Before commencement of this assignment, I have never been fully devoted towards coding as a team member. So, initiating the assignment was quite hard for me. The most challenging part was dividing the work load equally among both the team members. But, all these events provided me with a loads of knowledge like-

1. how to plan before starting any project.
2. How to manage all the milestones
3. How to communicate with the team member when any issue arises
4. Using github
5. Debugging
6. Importance of team work

and many more.

I also think this assignment will help me a lot with my professional life.