Advanced Programming Assignment-1 report
JiaQi Tang s3598284

Developing Plan

For this mini network I will create the following lists to store the profile information:

- Name lists
- Age lists
- Status lists
- Friends lists
- Parents lists
- Children lists
- Image lists

I will initial with 9 people in the system, every person will have name and age. Some of them will have status at the beginning, because status is optional in this scanio, so it is not necessary for everyone to have it. But there will be 3 methods to display, set and remove the status for the people. I will also give some people an array of friends at the beginning, the friends array will be defined at friends class, because each person can have many friends, so in the Friends List, each element is an array. So as Parents List and Children List. There will also be some methods to display, add and remove friends, but there will be some limitations, because as the assignment description mentioned that age matters when making friends in this mini network. Some of the methods will be able to show the people's parents or children. Furthermore, I will have two profile pictures for people to choose(one is cat the other one is dog), but not compulsory. In the Image List, I only save the number of the picture which the people picked, in the driver class, I will have "if else" to decide which picture to display on this person's profile

The initial profiles of the people in the system will be set as table-1:

Name	Age	Image	Status	Friends	Parents	Children
Alice	28	1(cat)	Blonde hair	Don,Eddie	null	Gary
Bob	29	2(dog)	Big muscles	Eddie	null	Gary
Cathy	25	1(cat)	Vegetarian	null	null	Helen
Don	33	2(dog)	Love cats	Alice	null	Helen
Eddie	43	1(cat)	Super dog people	Alice,Bob	null	Ivan
Fiona	38	2(dog)	Play guitar	null	null	Ivan
Gary	6	1(cat)	Play piano	Helen	Alice,Bob	null
Helen	4	2(dog)	Like candy	Gary	Cathy,Don	null
Ivan	1	1(cat)	Born in Japan	null	Eddie,Fiona	null
….	….		….	….	….	….

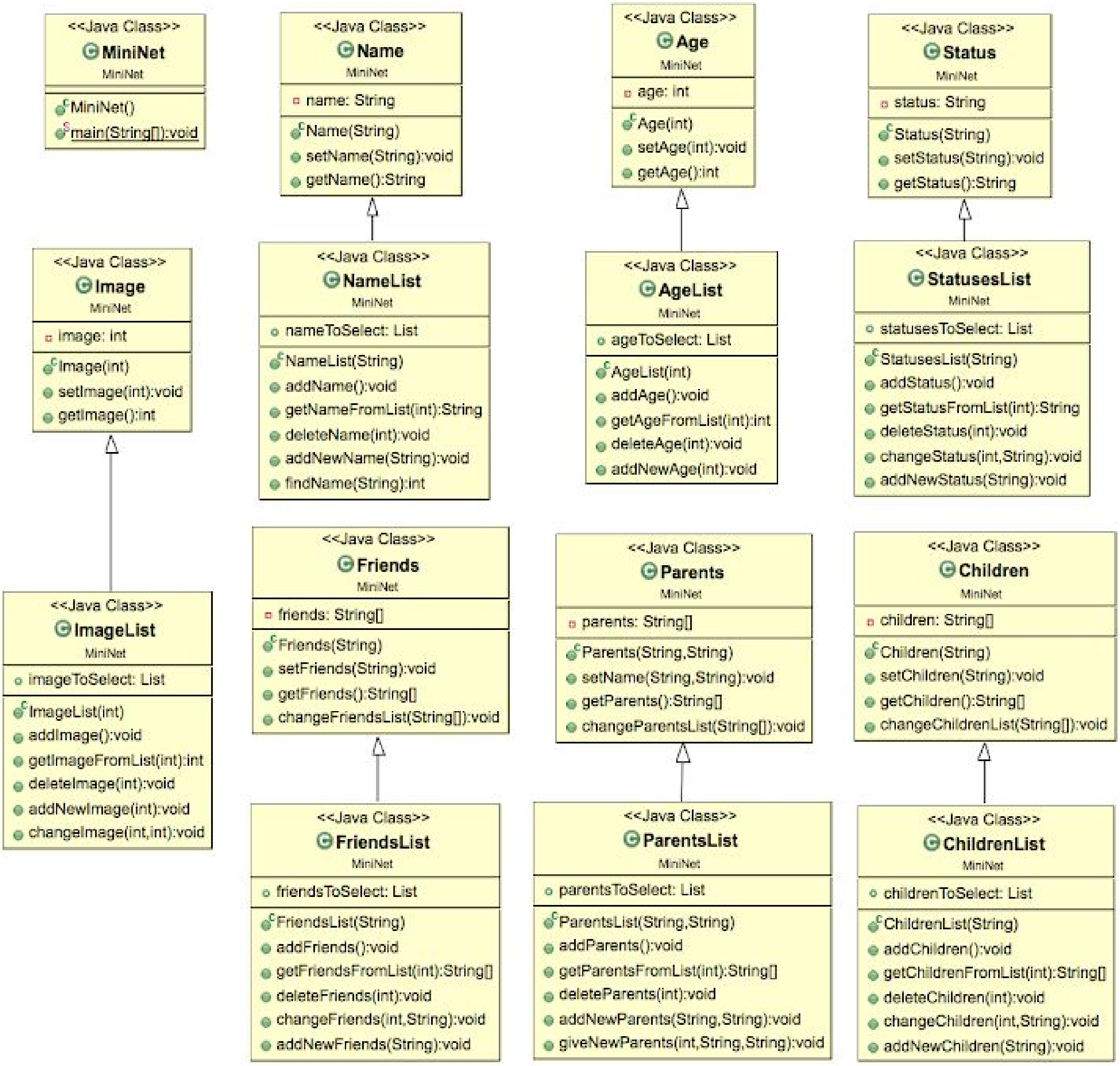
table-1

Structure Design

I will have the following classes in my program:

- **MiniNet class**,which is for the user to operate the system. It can use every public methods from other classes by creating objects and calling their methods.
- **Name class**,define the variables of Name and methods will be used in NameList class
- **NameList class**,the subclass of Name class.It can create a list to store names for every person in the system,also define some important public methods about “name” for the driver class
- **Age class**,define the variables of Age and methods will be used in AgeList class
- **AgeList class**,the subclass of Age class.It can create a list to store the Age for every person in the system,also define some important public methods about “age” for the driver class
- **Image class**,define the variables of picture number, and methods will be used in ImageList class
- **ImageList class**,the subclass of Image class.It can create a list to store the picture number for every person in the system,also define some important public methods about “image id” for the driver class
- **Status class**,define the variables of statuses,and methods will be used in StatusesList class
- **StatusesList class**,the subclass of Status class.It can create a list to store statuses for every person in the system,also define some important public methods about “status” for the driver class
- **Friends class**,define the array of Friends and methods will be used in FriendsList class
- **FridendsList class**,the subclass of Friends class.It can create a list to store the friends array for every person in the system,also define some important public methods about “friends” for the driver class
- **Parents class**,define the array of Parents and methods will be used in ParentsList class
- **ParentsList class**,the subclass of Parents class.It can create a list to store the parents array for every person in the system,also define some important public methods about “parents” for the driver class
- **Children class**,define the array of Children and methods will be used in ChildrenList class
- **ChildrenList class**,the subclass of Children class.It can create a list to store the children array for every person in the system,also define some important public methods about “children” for the driver class

UML Diagram



picture-1

(The picture-1 is generated by eclipse)

Maintaining Description

As the picture-1 showing, we can clearly understand that the driver class “MiniNet” uses the public methods from the “xxxList” classes to implement the whole system and all the “xxxList” classes are subclasses of their super classes. The three things what “MiniNet” do is :

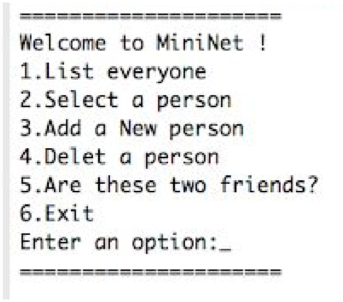
- initialling all the datas in the lists
- creating objects and call their methods
- using logic rules to organize those methods

So when I found logical errors, I will fix “MiniNet” class, because all the logics are there in “MiniNet” class. When I meeting with some problem about methods, there is nothing to do with “MiniNet” class, it is not the one who define the methods. Methods problems can be easily located because each class only define methods about one element.

Through this design structure I can maintain the whole program efficiently

Operating Explanation

The main operating menu is like picture-2

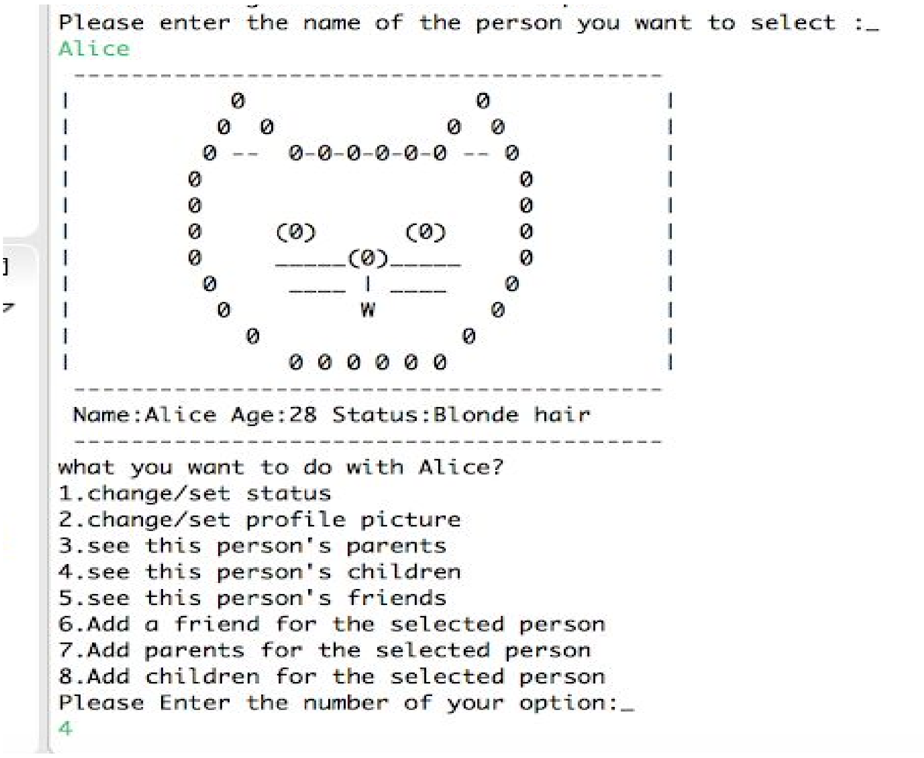


```
=====================
Welcome to MiniNet !
1.List everyone
2.Select a person
3.Add a New person
4.Delet a person
5.Are these two friends?
6.Exit
Enter an option:_
=====================
```

picture-2

The user can control what to do by inputting numbers. Every time when a function is done, the system will jump back to the main menu. Only when the user input “6” at main menu,the system will stop running and waiting for commanding.

All the person related functions are in the sub-menu after selecting a person by searching the name, as picture 3

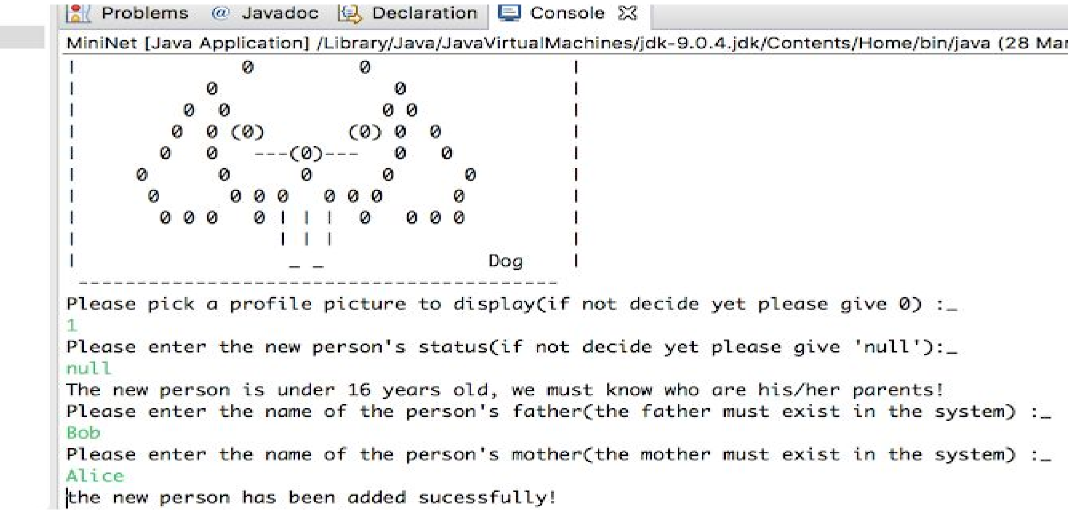


```
Please enter the name of the person you want to select :_
Alice
 -----------------------------------------
|           0                0            |
|          0  0            0  0           |
|         0 --  0-0-0-0-0-0 -- 0          |
|        0                      0         |
|        0                      0         |
|        0    (0)      (0)      0         |
|        0    _____(0)_____     0         |
|         0    ____  | ____    0          |
|          0         W        0           |
|            0              0             |
|               0 0 0 0 0 0               |
 -----------------------------------------
 Name:Alice Age:28 Status:Blonde hair
 -----------------------------------------
what you want to do with Alice?
1.change/set status
2.change/set profile picture
3.see this person's parents
4.see this person's children
5.see this person's friends
6.Add a friend for the selected person
7.Add parents for the selected person
8.Add children for the selected person
Please Enter the number of your option:_
4
```

picture-3

Everytime when the user update the data, we can save the change. When the menu turns to the main menu again, we can see whether the changes make sense by many ways.

As picture-4 shows ,when the user choose to add a new person in the system, the system will ask a series of questions for the user to input, if the new person’s age is under 16, will also need to input the person’s parents.



```
MiniNet [Java Application] /Library/Java/JavaVirtualMachines/jdk-9.0.4.jdk/Contents/Home/bin/java (28 Mar
|              0         0                 |
|           0               0              |
|         0  0             0 0             |
|        0  0 (0)       (0) 0  0           |
|       0   0   ---(0)---   0   0          |
|     0      0      0      0      0        |
|      0      0 0 0   0 0 0      0         |
|       0 0 0   0 | | |  0   0 0 0         |
|                 | | |                    |
|                  _ _              Dog    |
 -----------------------------------------
Please pick a profile picture to display(if not decide yet please give 0) :_
1
Please enter the new person's status(if not decide yet please give 'null'):_
null
The new person is under 16 years old, we must know who are his/her parents!
Please enter the name of the person's father(the father must exist in the system) :_
Bob
Please enter the name of the person's mother(the mother must exist in the system) :_
Alice
the new person has been added sucessfully!
```

picture-4

Discussion and comments

JiaQi Tang s3598284(contribution 100%):

- Possible issues:

As the assignment 1 description mentioned, in this program, user should be able to delete people from the system, but also , a young kid can not exist in this system with only one parents. I was planning to make my program be able to delete a person not from the system,but also delete him from his friends' friends lists and his children's parents lists. Considered about a young kid must have two parents, I was a little bit confusing.

- Gain and Learn;

Through this assignment I learned how to design and implement a project. OO-programming is very close to the reality, the conception is excellent. I also practiced a lot about coding and fixing errors because I finished this program alone, it is quite a challenge for me as well, after complete this program, I have got deeper understanding about OO-programming