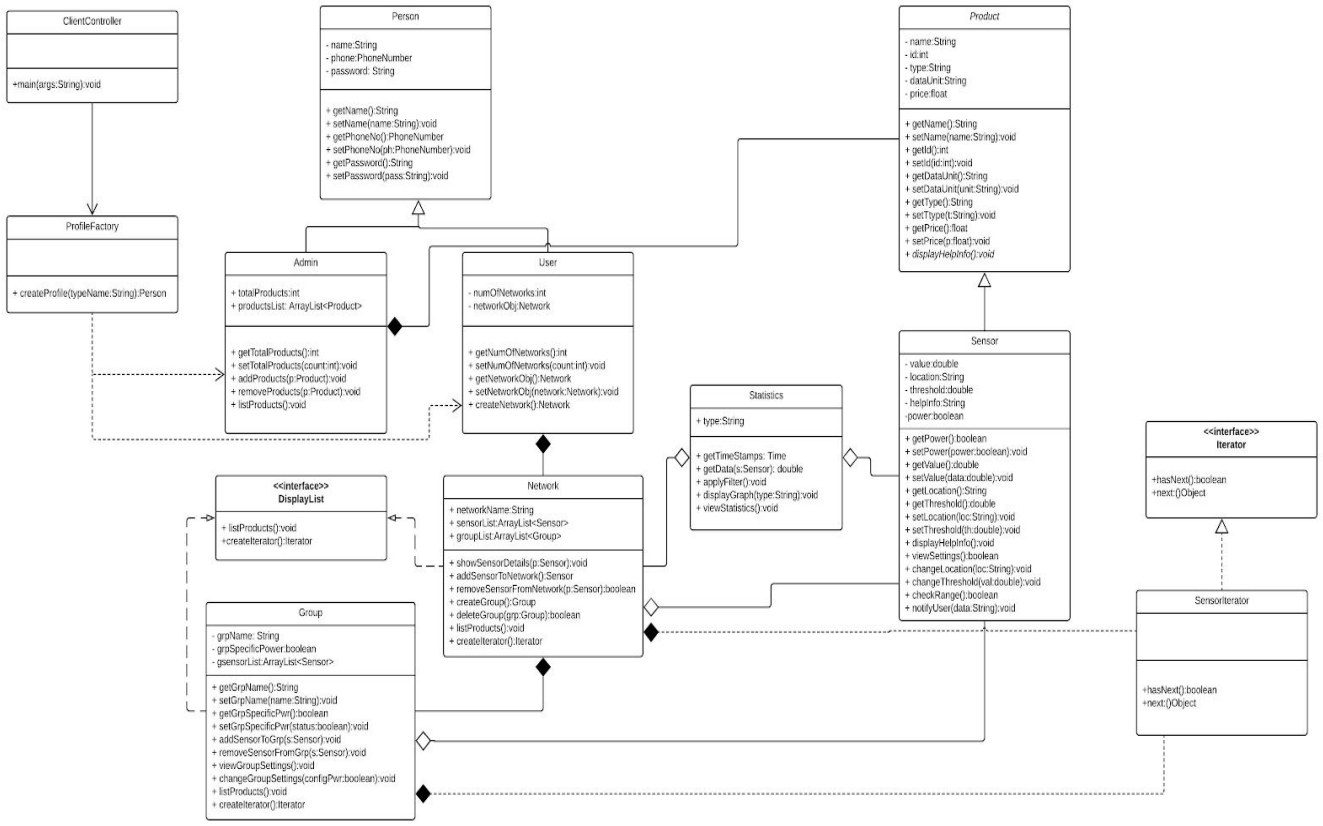
CSCI 5448 - Object Oriented Analysis and Design
Project Part 4 - Progress Report 2
#31 Smart Home Management System

Team:
1. Raj Kumar Subramaniam
2. Saritha Senguttuvan
3. Savitha Senguttuvan

Title:
Smart Home Management System

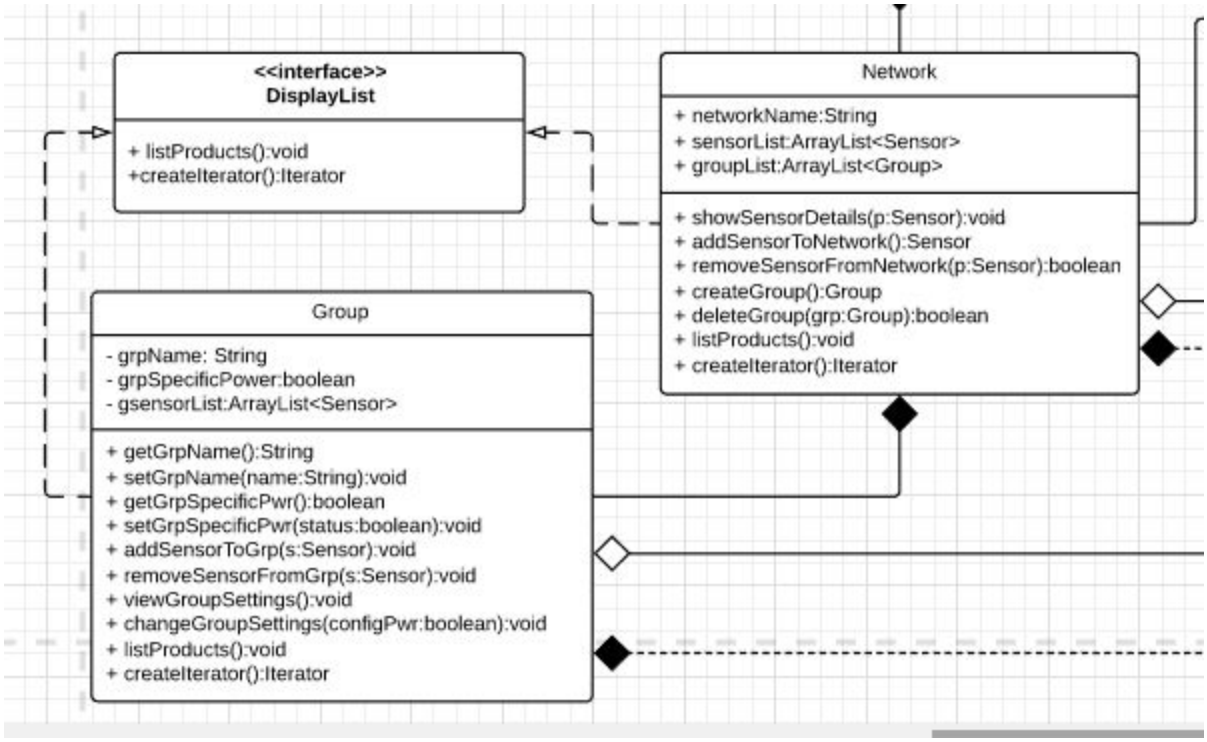
Description:
Smart Home Management System is a desktop app, which helps to monitor and update notifications from the smart sensors at Customer's house. The project is assumed to be for a client company, ABC Smart Home which manufactures smart home products, so the Admin will be the company's employee and the customers will be the one who purchases the products from the company. The app will help the customer and the admin to configure sensors in the home network and update any notifications to the user.

Updated Class Diagram:



We have added the class diagram from the previous design and also added design patterns in the design. The class diagram is updated for new design patterns and the functions needed inside the current classes for the addition of design patterns was also updated.

Completed Class Diagram:



The Network and Group classes are updated. The complete class diagram of the classes are shown above and the relation between those classes are shown in the above completed class diagram.

Summary:

We have been working design patterns and the areas in the code which needed refactoring. We have implemented the Person, Admin and User, Product and Sensor classes. The Network and Group classes was also implemented and the design patterns added are Factory and Iterator patterns. We have also thought of restructuring based on design patterns and added factory design pattern to create profile and also added Iterator design pattern to iterate through the list of sensors in the network and group class.

Breakdown of work:

All of us were working on the separate classes and added the design pattern classes. The implementation is moving forward with frequent meetings and progressing further.

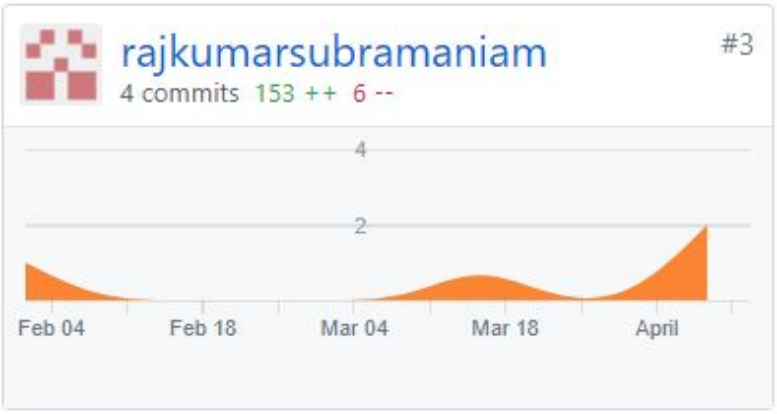
Raj Kumar Subramaniam - was working on implementing the Sensor and Product class and is in the process of adding more functions related to the Sensor and Product classes

Saritha Senguttuvan - was working on adding functionalities related to Person, Admin and User classes. Also planning to make use of Observable pattern in the design of our software, as it is needed to update the sensor values to user when a threshold limit reached by the sensors.

Savitha Senguttuvan - was working on implementing the Network and Group classes to implement the functionalities related to adding sensors to the network and adding new groups of sensors.

Github Graph:

Commits by Raj Kumar Subramaniam



Commits by Saritha Senguttuvan

Commits by Savitha Senguttuvan

Estimated Remaining Effort:

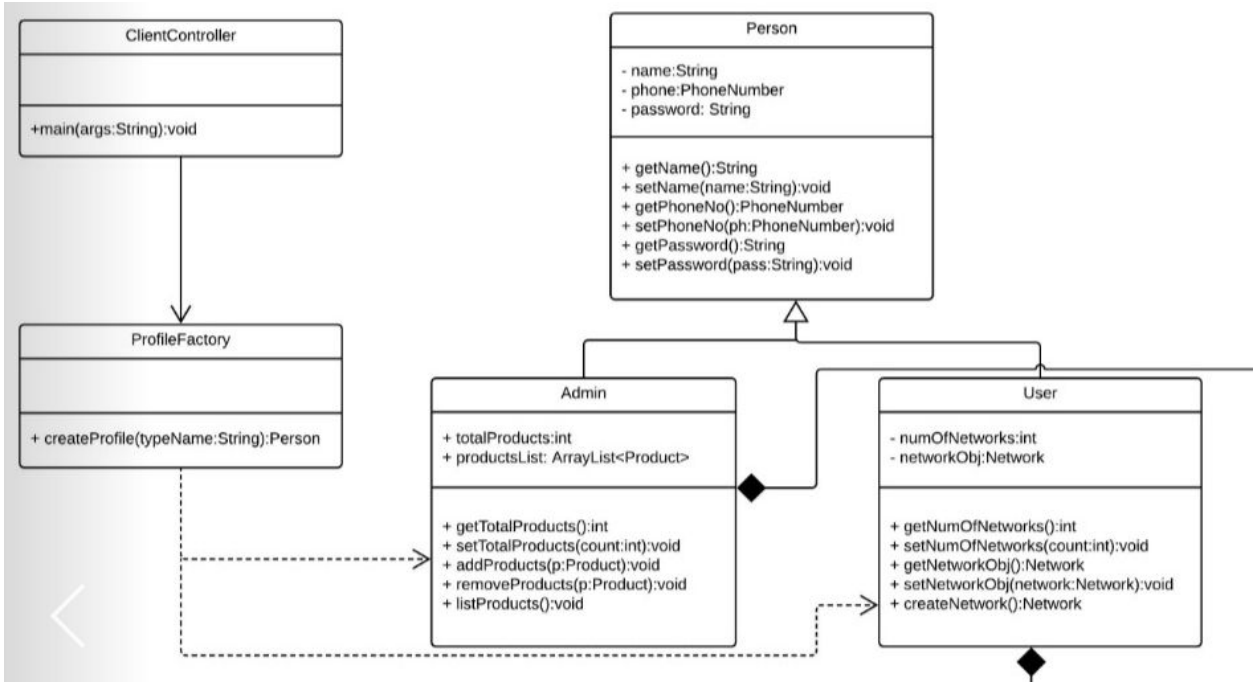
As we have added design patterns to the class diagram, we will implement it in the next few days. We will also be working on improving the source code of all the classes and will be adding more code for the user interface. We have to create mimic data for the sensor values and have to use it in the database.

Design Patterns:

We have used 2 design patterns in our implementation. Factory Design Pattern and Iterator Design Pattern.

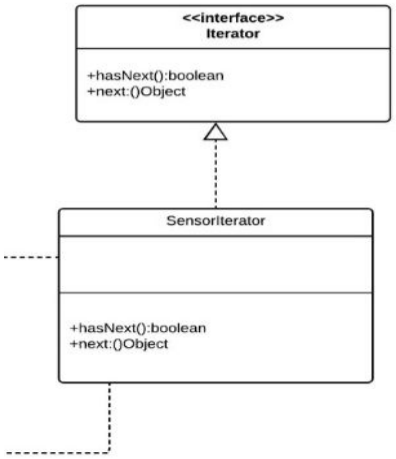
Factory Design Pattern:

This creational design pattern was used to create object without exposing the creation logic. We have created the objects for Admin and User based on the factory design technique. Here is the implemention of Factory design pattern,



Iterator Design Pattern:

The Iterator Design Pattern will be implemented in both Network and Group class to iterate through the items in the respective class. An interface is created with hasNext and next function which is them implemented by SensorIterator class. The pattern can be referred below. It can also be seen in the full class diagram for complete understanding.



Observer Design Pattern:

We are planning to use the observer design pattern so that if the admin adds a product into the Products List, all the users get notified regarding the new addition of a product

Next Iteration:

The next iteration will be the final iteration. So, we will also add any design pattern as we see fit. We have satisfied the basic requirements till now and will be trying to improve the quality of the code. We will be adding the database and more commands to interact with the user. User interface is an area we have to concentrate on and will be adding more details. We will be working on presenting the project and delivering it as a product before the presentation.