GTU Department of Computer Engineering

CSE 222/505 - Spring 2021

Homework 3

Report

Muhammet Fikret ATAR

1801042693

SYSTEM REQUIREMENTS

In this system we have a company which has 4 different branches and sells both in-store and online.Firstly i created users who are admin , , branch employee, customer for company. The duties of these users are as follows;

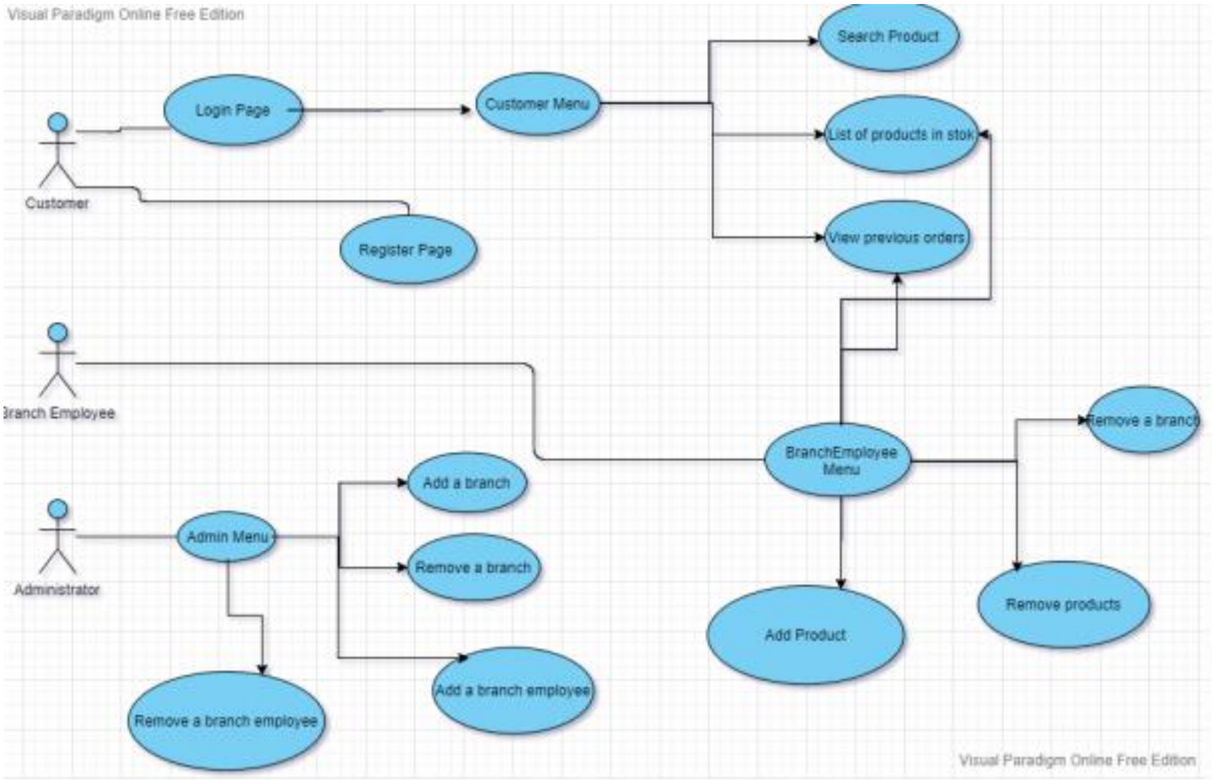
Admin: It can add and remove branches, employees. Also add new product and inquire product in branches. Branches have their own products.

Branch employee:Add product and adds customers to the system

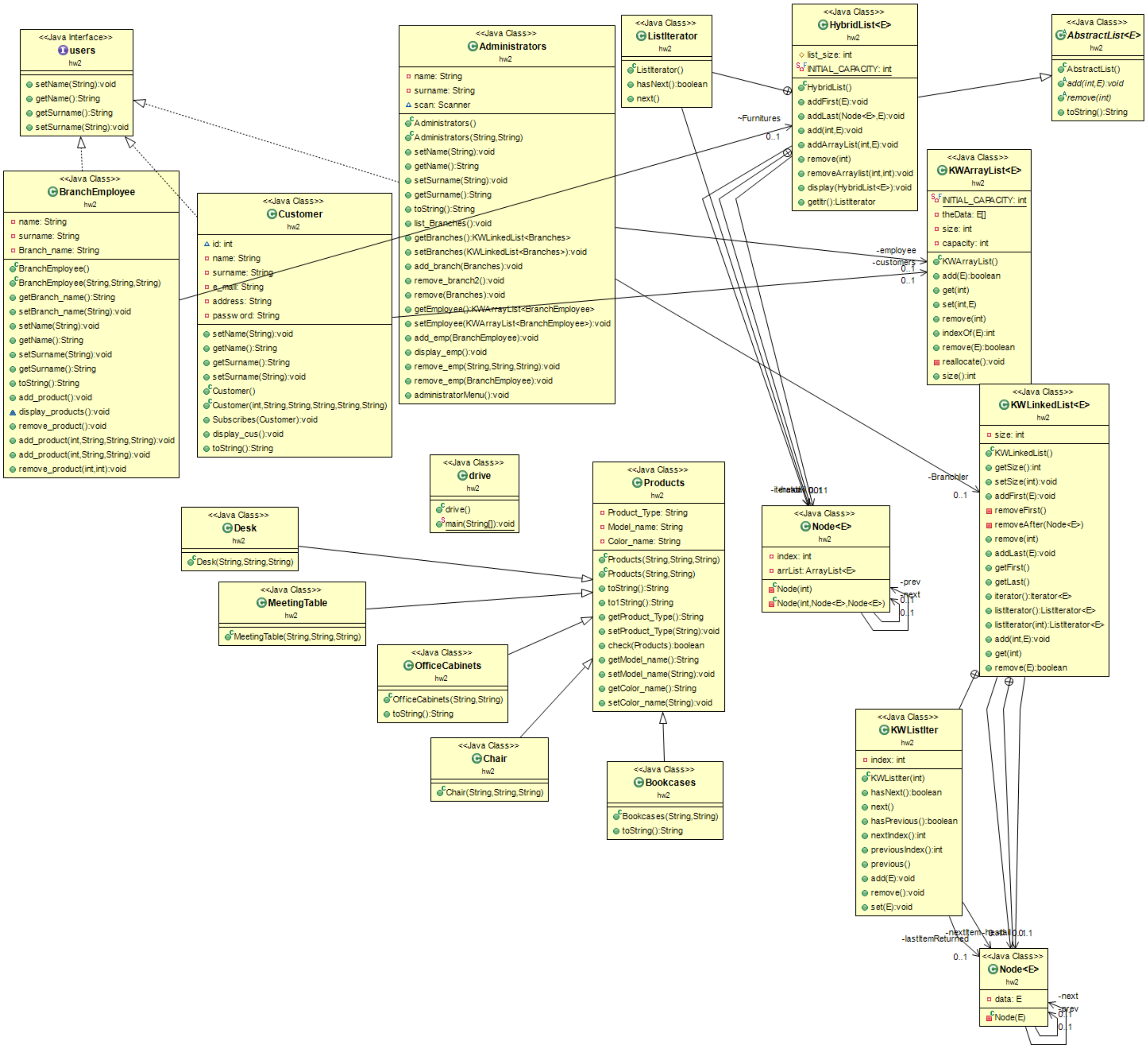
Customer:First It has to be subscribe to system.After then ,customer can purchase product from any branch.So that the same product which is purchesed has to delete from system. Decreasing or ending products are reloaded into the system by branch employees.

I used LinkedList for infos about branches , arraylist for users in the system and İ implement HybridList for products.

Use Case Diagram



CLASS DIAGRAM:

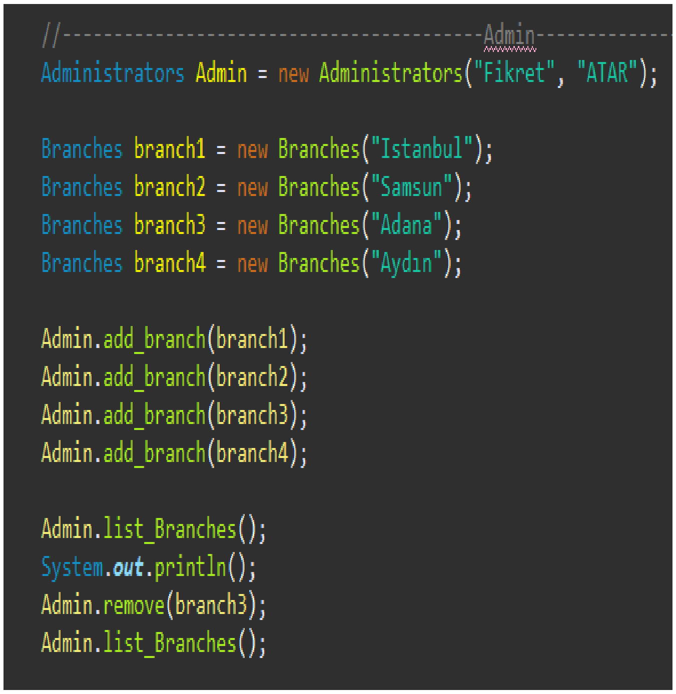


PROBLEM SOLUTION APPROACH

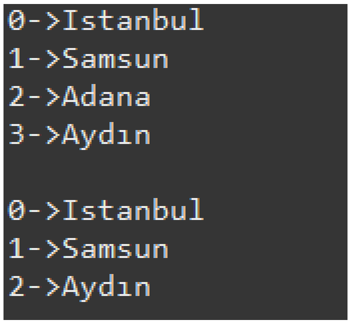
Solution Approach: We can divide the problem into sub problems and solve each of them separately. We can first solve the problem of keeping track of the branches by collecting them in an LinkedList of sort and this LinkedList will be managed by the administrators of the company. Then we can store the employees of each branch inside their respective branches(Array list) so we can keep track of each employee and the branch he belongs to. As for the Products each of them will have some information to keep track of it like customer's names and each will have a unique ID so we don’t mix them up.(HybridList)

Running and Test Cases:

Add Branch;



```
//----------------------------------------Admin------------
Administrators Admin = new Administrators("Fikret", "ATAR");

Branches branch1 = new Branches("Istanbul");
Branches branch2 = new Branches("Samsun");
Branches branch3 = new Branches("Adana");
Branches branch4 = new Branches("Aydın");

Admin.add_branch(branch1);
Admin.add_branch(branch2);
Admin.add_branch(branch3);
Admin.add_branch(branch4);

Admin.list_Branches();
System.out.println();
Admin.remove(branch3);
Admin.list_Branches();
```

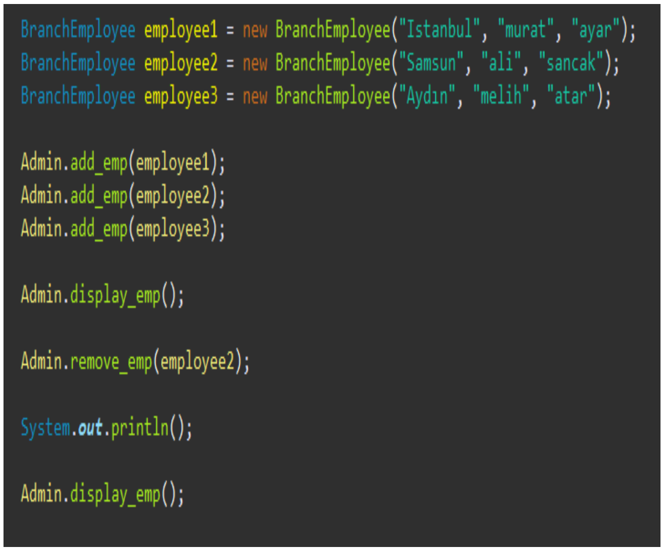


```
0->Istanbul
1->Samsun
2->Adana
3->Aydın

0->Istanbul
1->Samsun
2->Aydın
```

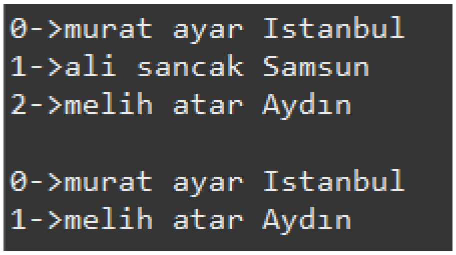
4 different branches have been added to the system by admin.

Branch3 have been removed from system by admin.

Add and remove branch employee;



```
BranchEmployee employee1 = new BranchEmployee("Istanbul", "murat", "ayar");
BranchEmployee employee2 = new BranchEmployee("Samsun", "ali", "sancak");
BranchEmployee employee3 = new BranchEmployee("Aydın", "melih", "atar");

Admin.add_emp(employee1);
Admin.add_emp(employee2);
Admin.add_emp(employee3);

Admin.display_emp();

Admin.remove_emp(employee2);

System.out.println();

Admin.display_emp();
```

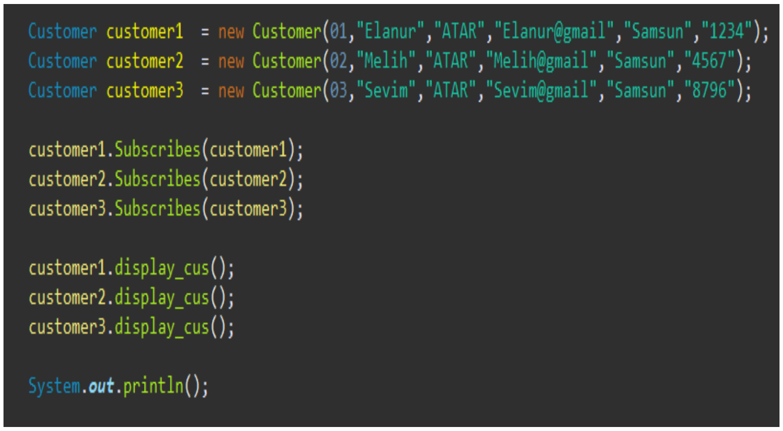


```
0->murat ayar Istanbul
1->ali sancak Samsun
2->melih atar Aydın

0->murat ayar Istanbul
1->melih atar Aydın
```

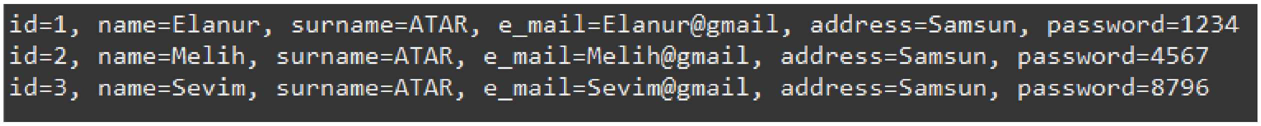
3 different branche employees have been added to the system by admin.

Employee2 has been removed from the system by admin.

Subscription Customer;



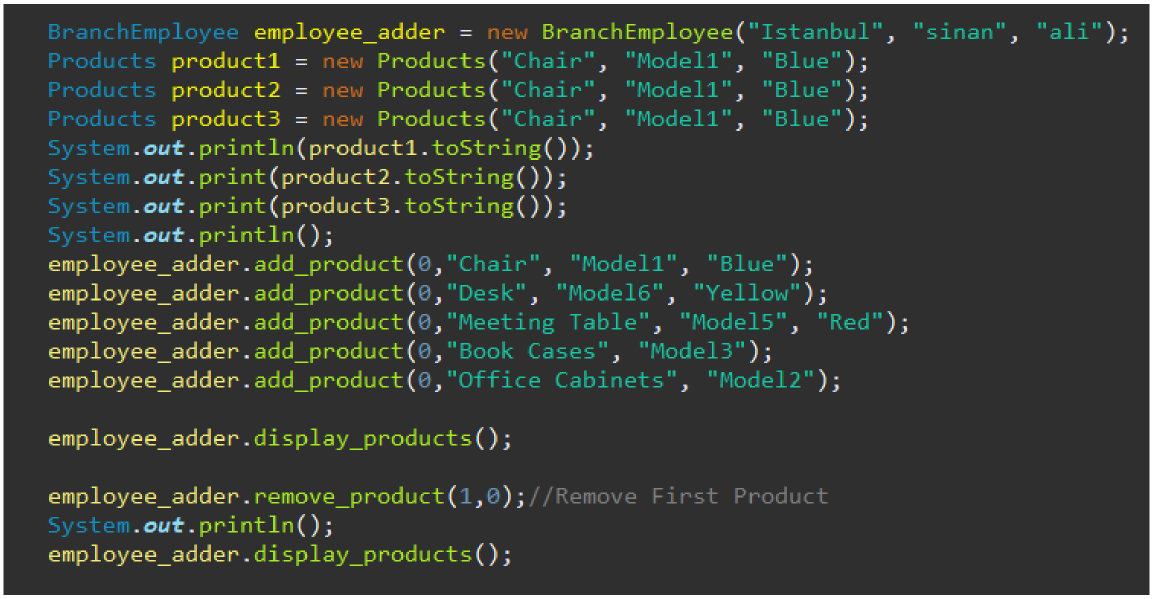
```
Customer customer1  = new Customer(01,"Elanur","ATAR","Elanur@gmail","Samsun","1234");
Customer customer2  = new Customer(02,"Melih","ATAR","Melih@gmail","Samsun","4567");
Customer customer3  = new Customer(03,"Sevim","ATAR","Sevim@gmail","Samsun","8796");

customer1.Subscribes(customer1);
customer2.Subscribes(customer2);
customer3.Subscribes(customer3);

customer1.display_cus();
customer2.display_cus();
customer3.display_cus();

System.out.println();
```



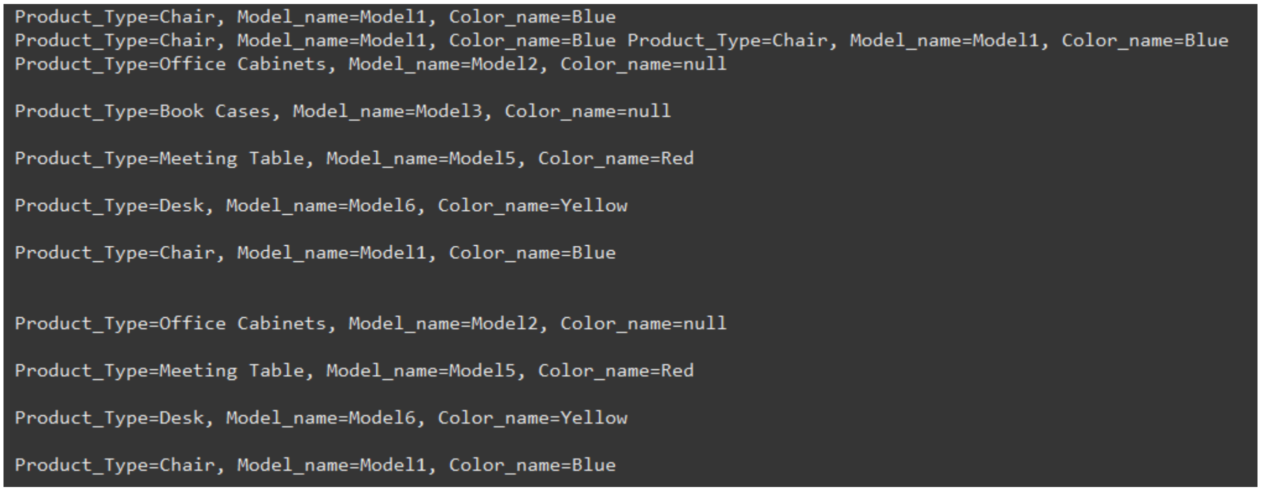
```
id=1, name=Elanur, surname=ATAR, e_mail=Elanur@gmail, address=Samsun, password=1234
id=2, name=Melih, surname=ATAR, e_mail=Melih@gmail, address=Samsun, password=4567
id=3, name=Sevim, surname=ATAR, e_mail=Sevim@gmail, address=Samsun, password=8796
```

Customers has become a member of the system.

Add and Remove Product;



```
BranchEmployee employee_adder = new BranchEmployee("Istanbul", "sinan", "ali");
Products product1 = new Products("Chair", "Model1", "Blue");
Products product2 = new Products("Chair", "Model1", "Blue");
Products product3 = new Products("Chair", "Model1", "Blue");
System.out.println(product1.toString());
System.out.print(product2.toString());
System.out.print(product3.toString());
System.out.println();
employee_adder.add_product(0,"Chair", "Model1", "Blue");
employee_adder.add_product(0,"Desk", "Model6", "Yellow");
employee_adder.add_product(0,"Meeting Table", "Model5", "Red");
employee_adder.add_product(0,"Book Cases", "Model3");
employee_adder.add_product(0,"Office Cabinets", "Model2");

employee_adder.display_products();

employee_adder.remove_product(1,0);//Remove First Product
System.out.println();
employee_adder.display_products();
```



```
Product_Type=Chair, Model_name=Model1, Color_name=Blue
Product_Type=Chair, Model_name=Model1, Color_name=Blue Product_Type=Chair, Model_name=Model1, Color_name=Blue
Product_Type=Office Cabinets, Model_name=Model2, Color_name=null

Product_Type=Book Cases, Model_name=Model3, Color_name=null

Product_Type=Meeting Table, Model_name=Model5, Color_name=Red

Product_Type=Desk, Model_name=Model6, Color_name=Yellow

Product_Type=Chair, Model_name=Model1, Color_name=Blue


Product_Type=Office Cabinets, Model_name=Model2, Color_name=null

Product_Type=Meeting Table, Model_name=Model5, Color_name=Red

Product_Type=Desk, Model_name=Model6, Color_name=Yellow

Product_Type=Chair, Model_name=Model1, Color_name=Blue
```

The products have been added to the system by the branch employee.

The product purchased by the customer has not been removed from stock. This part is not working properly.