

**GTU Department of Computer
Engineering CSE 222/505 - Spring 2020
Homework 01 Report**

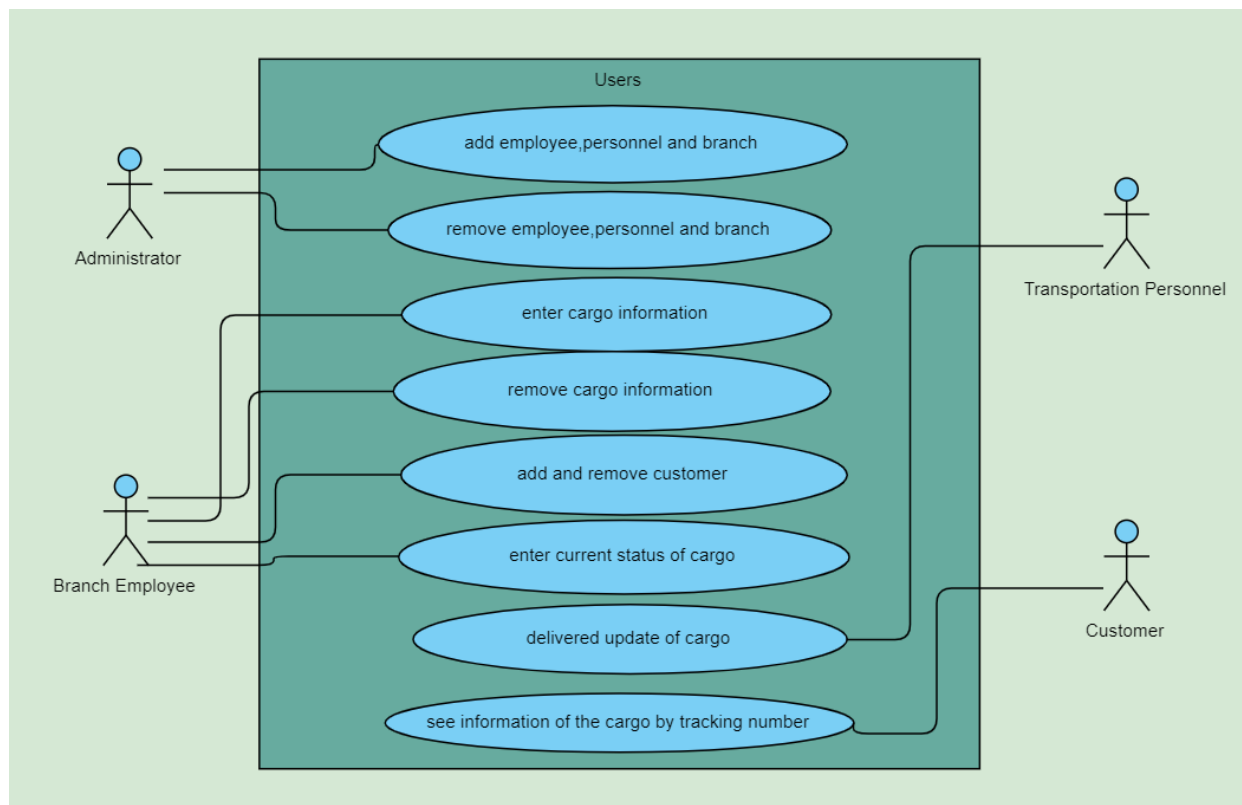
**ESRA ERYILMAZ
171044046**

1. SYSTEM REQUIREMENTS

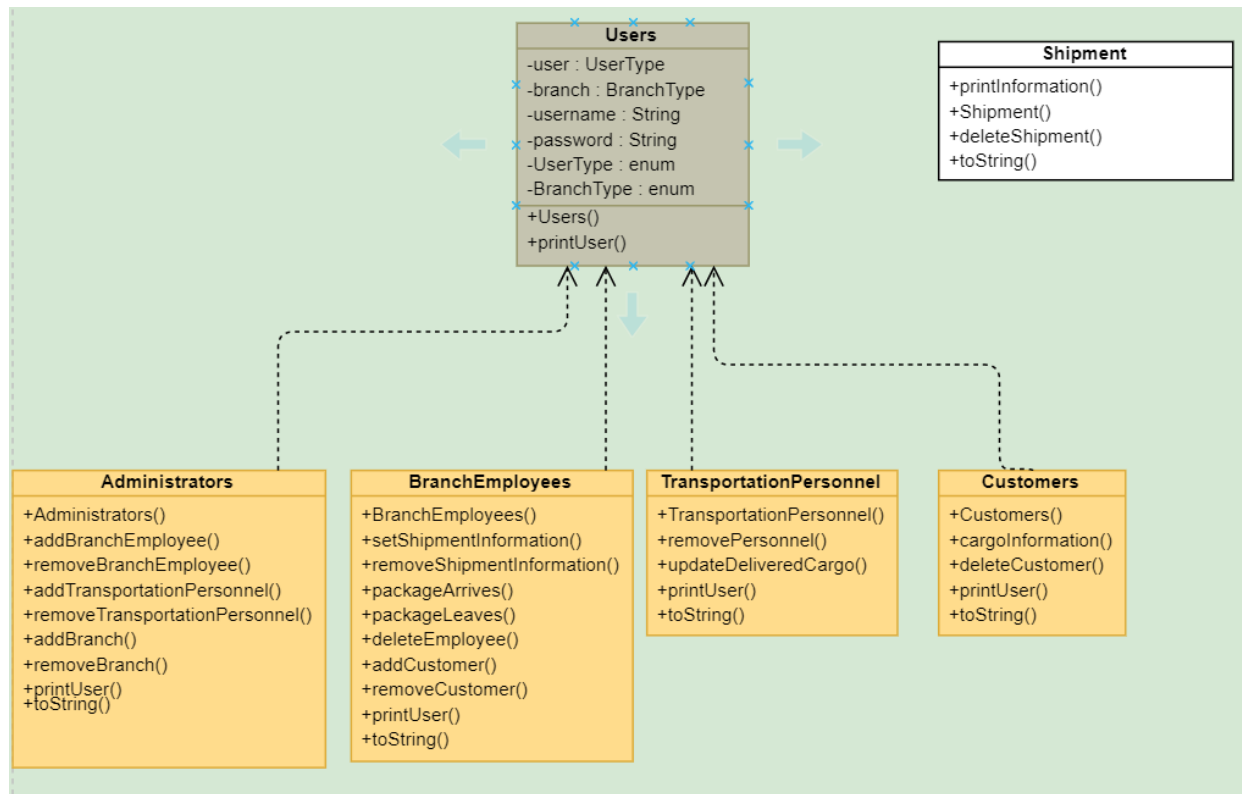
In this homework, we require Users class. In that class I need to keep some user informations. We also require some inherited classes from Users which names are Administrators, BranchEmployees, TransportationPersonnel and Customers classes. I don't need to keep any data in that classes. These classes have some functions for cargo automation system. We need one more classes which name is Shipment. In that class I need to keep all informantions about the shipment.

If we want to operate this system , we need to create array of class objects or just objects than we can manipulate this system.

2. USE CASE DIAGRAMS



3. CLASS DIAGRAMS



4. OTHER DIAGRAMS

5. PROBLEM SOLUTION APPROACH

- *Firstly we need to create users. I create Users class to keep all users information. We need to keep names ,surnames, types of the user and types of branch. Then we need to create some other class for specific tasks.*
- *For the specific tasks I create four new classes which names are Administrators, BranchEmployees, TransportationPersonnel and Customers.*
- *Administrators can manage the system by adding and removing branches, branch employees and transportation personnel. In Administrators class , there are functions for doing these tasks.*
- *BranchEmployees can enter and remove information about the shipments sent from that branch, add and remove customers to the system. When the package arrives or leaves in that branch, its current status is updating branch employee.*
- *TransportationPersonnel makes update about the shipment.*
- *Customers can see information about the shipment if they enter the tracking number.*
- *For the shipment , I create another class to keep all cargo information which name is Shipment.*

6. TEST CASES

Firstly, I create array of objects.

```
Administrators[] adminTest = new Administrators[2];
BranchEmployees[] empTest = new BranchEmployees[2];
TransportationPersonnel[] perTest = new TransportationPersonnel[2];
Customers[] custTest = new Customers[2];
Shipment[] cargo = new Shipment[2];
```

CASE 1: Administrators test

```
//creating two new administrators
System.out.printf("1.->");
adminTest[0] = new Administrators(Users.UserType.ADMINISTRATOR,Users.BranchType.ANKARA,"Esra Eryilmaz","123456");
adminTest[0].printUser();
System.out.printf("2.->");
adminTest[1] = new Administrators(Users.UserType.ADMINISTRATOR,Users.BranchType.IZMIR,"Fatma Kaya","abc123");
adminTest[1].printUser();
```

```
//administrator removes one of the branch employee
adminTest[1].removeBranchEmployee(empTest[1]);
empTest[1].printUser();
```

```
//administrators add two new branch employees
System.out.printf("1.->");
empTest[0] = adminTest[0].addBranchEmployee(Users.UserType.BRANCH_EMPLOYEE,Users.BranchType.KOCAELI,"Ahmet Demir","111111");
empTest[0].printUser();
System.out.printf("2.->");
empTest[1] = adminTest[1].addBranchEmployee(Users.UserType.BRANCH_EMPLOYEE,Users.BranchType.ISTANBUL,"Mehmet Sahin","000000");
empTest[1].printUser();
```

```
//administrators add two new transportation personnel
System.out.printf("1.->");
perTest[0] = adminTest[0].addTransportationPersonnel(Users.UserType.TRANSPORTATION_PERSONNEL,Users.BranchType.AMASYA,"Ayse Yildirim","135790");
perTest[0].printUser();
System.out.printf("2.->");
perTest[1] = adminTest[1].addTransportationPersonnel(Users.UserType.TRANSPORTATION_PERSONNEL,Users.BranchType.ISTANBUL,"Mustafa Kara","010101");
perTest[0].printUser();
```

```
//administrator removes one of the transportation personnel
adminTest[1].removeTransportationPersonnel(perTest[1]);
perTest[1].printUser();
```

CASE 2: BranchEmployees test

```
//creating two new customer
System.out.printf("1.->");
custTest[0] = empTest[0].addCustomer(Users.UserType.CUSTOMER, Users.BranchType.ANKARA,"Ali Yilmaz","000111");
custTest[0].printUser();
System.out.printf("2.->");
custTest[1] = empTest[1].addCustomer(Users.UserType.CUSTOMER, Users.BranchType.KOCAELI,"Emine Aslan","abcdef");
custTest[1].printUser();
```

```
//branch employees add two new cargo with their informations
System.out.printf("1.->");
cargo[0] = empTest[1].setShipmentInformation("45322567","Ali Yilmaz","Meryem Yildiz", Users.BranchType.ANKARA);
cargo[0].printInformation(cargo[0].getTrackingNumber());
System.out.printf("2.->");
cargo[1] = empTest[0].setShipmentInformation("54327924","Emine Aslan", "Ismail Kilic", Users.BranchType.KOCAELI);
cargo[1].printInformation(cargo[1].getTrackingNumber());
```

```
//branch employee removes one of the cargo
empTest[0].removeShipmentInformation(cargo[1]);
cargo[1].printInformation(cargo[1].getTrackingNumber());
```

```
//branch employee updates when package arrives at that branch
empTest[0].packageArrives(cargo[1]);
```

```
//branch employee updates when package leaves at that branch
empTest[0].packageLeaves(cargo[1],Users.BranchType.AMASYA);
```

CASE 3: TransportationPersonnel test

```
//transportation personnel makes the update when cargo is delivered
perTest[0].updateDeliveredCargo(cargo[0]);
cargo[0].printInformation(cargo[0].getTrackingNumber());
```

CASE 4: Customers test

```
//customers can see the cargo information if they enter the tracking number
custTest[0].cargoInformation(cargo[0].getTrackingNumber(),cargo[0]);
```

7. RUNNING AND RESULTS

RUN CASE 1:

```
TEST STARTING...

-----
ADMINISTRATORS CLASS TEST STARTING...

1.->Administrator [user=ADMINISTRATOR, branch=ANKARA, username=Esra Eryilmaz, password=123456]
2.->Administrator [user=ADMINISTRATOR, branch=IZMIR, username=Fatma Kaya, password=abc123]

1.->BranchEmployee [user=BRANCH_EMPLOYEE, branch=KOCAELI, username=Ahmet Demir, password=111111]
2.->BranchEmployee [user=BRANCH_EMPLOYEE, branch=ISTANBUL, username=Mehmet Sahin, password=000000]

BranchEmployee [user=null, branch=null, username=null, password=null]

1.->TransportationPersonnel [user=TRANSPORTATION_PERSONNEL, branch=AMASYA, username=Ayşe Yildirim, password=135790]
2.->TransportationPersonnel [user=TRANSPORTATION_PERSONNEL, branch=AMASYA, username=Ayşe Yildirim, password=135790]

TransportationPersonnel [user=null, branch=null, username=null, password=null]

-----
```

RUN CASE 2:

```
-----
BRANCHEMPLOYEES CLASS TEST STARTING...

1.->Customer [user=CUSTOMER, branch=ANKARA, username=Ali Yilmaz, password=000111]
2.->Customer [user=CUSTOMER, branch=KOCAELI, username=Emine Aslan, password=abcdef]

1.->Shipment [trackingNumber=45322567, senderName=Ali Yilmaz, receiverName=Meryem Yildiz, deliveredBranch=ANKARAcurrentStatusnull]
2.->Shipment [trackingNumber=54327924, senderName=Emine Aslan, receiverName=Ismail Kilic, deliveredBranch=KOCAELIcurrentStatusKOCAELI]

Shipment [trackingNumber=null, senderName=null, receiverName=null, deliveredBranch=nullcurrentStatusnull]

-----
```

RUN CASE 3:

```
-----  
TRANSPORTATIONPERSONNEL CLASS TEST STARTING...  
Shipment [trackingNumber=45322567, senderName=Ali Yilmaz, receiverName=Meryem Yildiz, deliveredBranch=ANKARACurrentStatusANKARA]  
-----
```

RUN CASE 4:

```
-----  
CUSTOMERS CLASS TEST STARTING...  
Shipment [trackingNumber=45322567, senderName=Ali Yilmaz, receiverName=Meryem Yildiz, deliveredBranch=ANKARACurrentStatusANKARA]  
-----
```