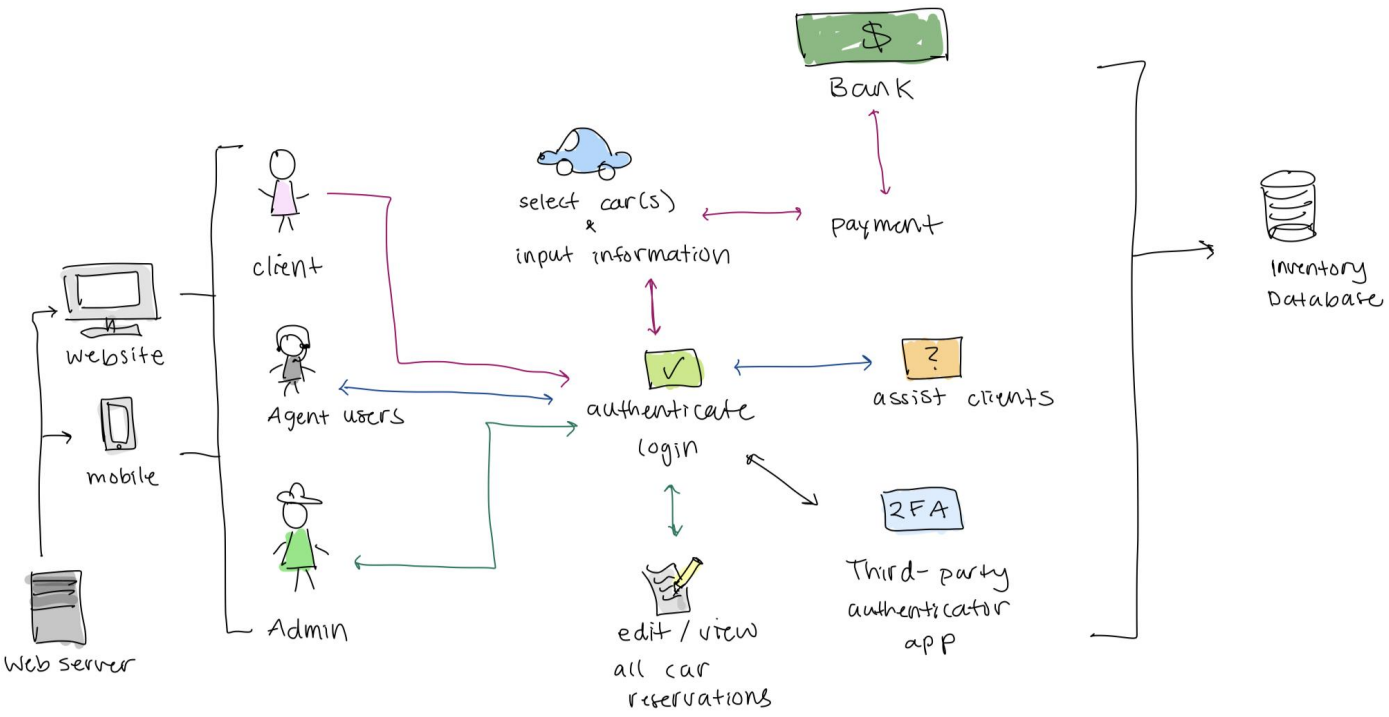
Car Rental System – Test Plan

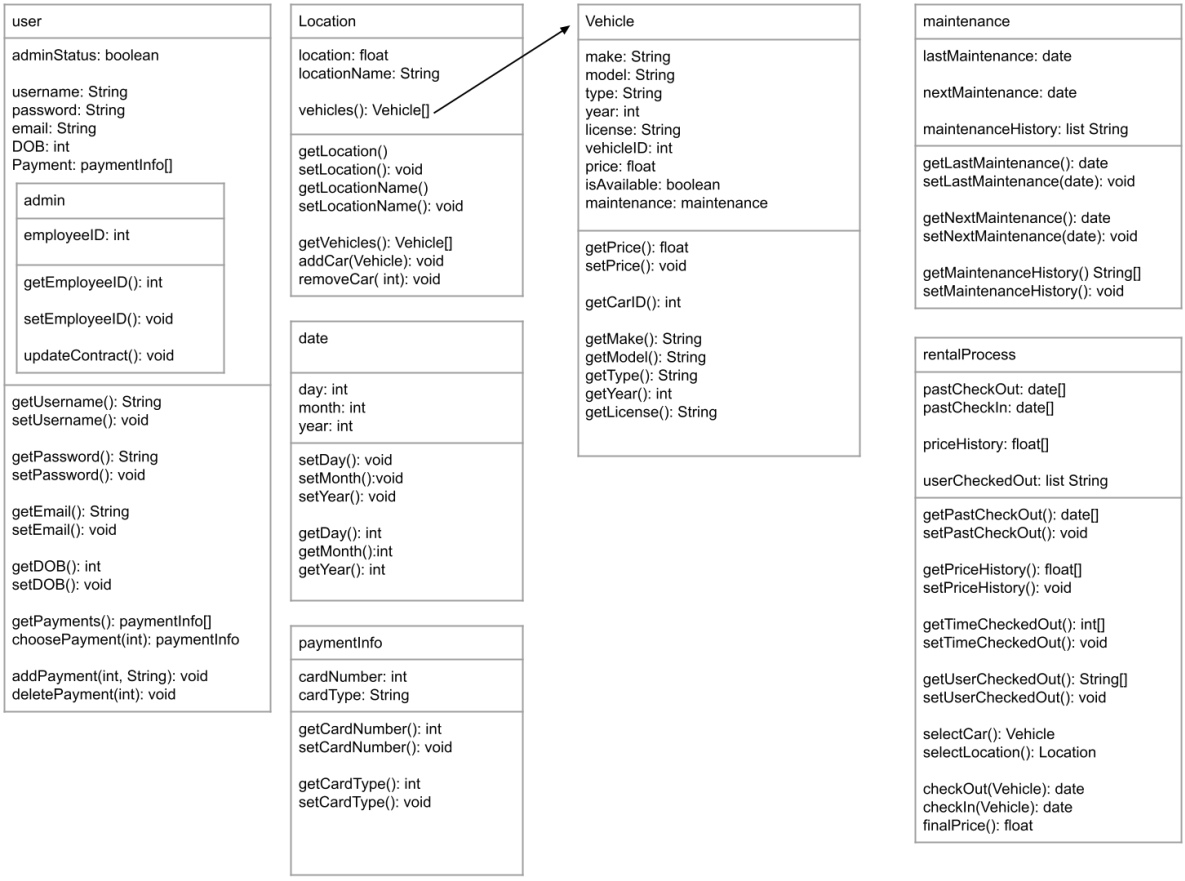
By: Winnie Thong, Khoi Tran, Chloe Kershner

CS250 Group 5

Software Architecture System Diagram:



UML Diagram:



Test Plan 1:

<u>Unit:</u>

Set an employee ID to test number. Check if getEmployeeID returns correct test number. Return true if getEmployeeID works and return false otherwise.

```
employeeID X
x.setEmployeeID(num)
If (X.getEmployeeID == num)
     Return true
```

```
Else
     Return false
```

Integration:

User adds a car. Verify that the car is added and displays the correct details. Use Vehicle Y as an input for X.addCar(). Check if Vehicle Y exists within X's vehicle list, if it does return true and break, if not, return false

```
Location X
Vehicle Y
X.addCar(Y)
Bool exists = false
for (Vehicles : X.getVehicles)
     if(Vehicles == Y)
          return true
          Break
```

System:

User rents car and makes payment. Check if the car that user rented is still marked as available in the system. Return true if the car is not available anymore, and return false if the car is still available even after it's booked.

```
customer X
location Y
Vehicle Z
X.selectCar(testCar)
X.makePayment
if( testCar.isAvailable != true) return true
else return false
```

Test Plan 2:

Unit:

Check if getModel works. Create test car and set model to " Honda Civic". Check if it returns "Honda Civic" when we call getModel.

```
car C
C.setModel("Honda Civic")
if( C.getModel == "Honda Civic") return true
else return false
```

Integration:

Log in as a user, and create a new payment type. Check if all payment information is correct. We will use a random card number and type, and make sure that using the get functions return the given card number and type. We will fail if the returned values dont match the given values

User X
X.addPayment(1234567890, visa);

userPayments[] = new paymentInfo[]
userPayments = X.getPayments()

If userPayments[1].getCardNumber != 1234567890
    Return false

Else If userPayments[1].getCardType != "visa"
    Return false

else return true

System:

Check for a vehicle at a certain location. Already existing will be a Vehicle X and a location Y. Input will be the ID number of the vehicle, if it returns the correct car, we pass the test. If not, we fail the test

vehicle X
location Y
inputID = ID

```
Vehicle[] list = new Vehicle[]

list = X.getVehicle()

for(Vehicle : list)
        if(Vehicle.getID() = ID)
                Return true
        Else
                Return false
```