

# Design a Car Rental System

To build a Car rental system we would require following end users

1. User who will be booking the car, registering for membership and cancelling membership
2. Manager will be organizing the rental process including maintaining customer records, car availability, approve pickup, dropoff, view statuses of the car and customer payment
3. Admin will be responsible for adding and modifying the car entries, maintenance of cars, sending notifications for overdue vehicles

**Class :** User

**Data :** emailaddress, password, phonenumber, licensenumber, icexpirydate, Licdateofissue, placeofissue, country

**Behaviour :**

```
login(emailaddress) {  
    List<String> emailaddresslist  
    Foreach(string mailaddress in list) {  
        if (emailaddress = mailaddress)  
            status = true;  
        else  
            status = false;  
    }  
}  
  
createAccount() {  
    get.emailaddress(emailaddress)  
    get.password(password)  
    get.phonenumber(phonenumber)  
    get.Licenseno(licensenumber)  
    get.Licexpirydate(licexpirydate) {  
        Date currentdate = new Data();  
        Return currentdate.compareTo(licexpirydate);  
    }  
    get.LicDOI(Licdateofissue);  
}
```

```

        get.PlaceIssued(placeofissue);
        get.country(country);
    }
listCaronRental(searchfilter) {
    if (car.available = 'Y') {
        switch(searchfilter) {
            case "car type" :
                listcar.basedontype();
                break;
            case "customer recommendations" :
                listcar.basedonrating();
                break;
            case "Noofseats" :
                listcar.noofseats();
                break;
            case "nearby" :
                listcar.nearby(location);
                break;
            case "specifications" :
                listcar.byspecifications(make,model);
                break;
        }
    }
}
bookcar() {
    get.dates(fromdate,todate,time);
    get.location(pickup,drop);
    payment();
}
payment(paymentmode) {

```

```

switch (paymentmode) {
    case "cash" ;
        break;
    case "card" :
        get.ccardetails(cardnumber,expirydate,securitycode)
        break;
    case "NetBanking" :
        get.bankingdetails(username,password)
        break;
}
}
CancelMembership() {
    cancel.membership(end membership);
}
cancelride() {
    get.cancelride(confirmation);
}

```

\*\*\*\*\*  
\*\*\*\*\*

**Class : Manager**

**Data :** CustomerName, PhoneNumber, PickupDate, DropoffDate, Cartype, NoofCars, TotalDays, StatusofTrip, Remark, PaymentType, Amount, CustEmailaddress, PaymentDate

**Behaviour :**

```

reservation() {
    display.reservationrecords(CustomerName, PhoneNumber, PickupDate,DropoffDate,
    Cartype, NoofCars, TotalDays, StatusofTrip, Remark);
    update.reservationstatus("Completed", "Cancelled", "OnRental", "Open");
}
quotes() {
    get.caronrent.quotes( Cartype, NoofDays);
}

```

```

        update.quotes(Cartype, NoofDays);
    }
    payments() {
        get.paymentinfo(CustomerName, CustEmailaddress, Amount, Paymenttype,
        PaymentDate) ;
    }
    vehicleInformation() {
        get.vehicleInfo( VehicleType, VehicleModel, VehicleMake, Renter);
        get.vehiclestatus("Available","Under Repair","On Rent") ;
    }
    membershipAcceptanceRequest() {
        get.request(Request)
        if (CustLicenceDetails = valid)
            return approverrequest;
        else
            return rejectrequest;
    }
    rentalInformation() {
        display.renterInformation( CustomerName, Address, PhoneNumber,
        CustEmailaddress);
        display.rentalInformation( PickupLocation, PickupDrop, PickupTime, DropoffDate,
        DropoffTime);
        display.vehicleInformation( Vehicletype, VehicleName, VehicleNumber);
        display.PaymentInformation( NoofDays, Amount, AmountPaid, DueAmount);
    }
    sendNotifications() {
        send.membership.notification( CustEmailaddress, PhoneNumber);
    }

```

```

*****
*****

```

**Class :** Admin

**Data :** VehicleType, Make, Model, LicensePlateNumber, NoofSeats, FuelCapacity, Mileage, vehiclestatus, VehicleInsurance, perday, perweek, permonth, repaircost, RepairStatus

**Behaviour :**

```
addVehicle() {  
    add.vehicledeatils(VehicleType, Make, Model, LicensePlateNumber, NoofSeats,  
FuelCapacity, Mileage, vehiclestatus, VehicleInsurance);  
}  
updateVehicleDetails() {  
    update.vehdetails(update.insurance, vehiclestatus);  
}  
removeVehicle() {  
    remove.vehicle(update.vehiclestatus);  
}  
updateCarlog() {  
    get.carlogs( CustomerName, PhoneNumber, Make, Model, NoofSeats,  
PickupDtae,DropoffDate, NoofDays);  
    get.status("Refueling", "Cleaning", "Damage", "On Rent", "Under Repair",  
"Available");  
}  
rates(){  
    get.rates( VehicleType, HourlyRates, perDay, perweek, permonth,taxes);  
}  
repairOrders() {  
    get.vehicleRepairList( VehicleType, LicensePlateNumber, startdate, scheduleddate,  
totalcost, RepairStatus);  
    update.vehicleonRepairStatus("In-Progress", "Open", "Completed","Pending");  
}  
sendNotifications() {  
    send.reservationInformation( CustEmailaddress, PhoneNumber);  
    send.overdueVehicle.notification(CustEmailaddress, PhoneNumber);  
}
```