EU- Car Rental

System Specification

Group 8

Jash Pithadia

Saurabh Singh

Rahul Deore

Peerada Looareesuwan

Table of Contents

2. UML Static diagram

package Model[Model1]

Employee
attributes
-Employee_id : String
-First_name : String
-Last_name : String
-SSN : String
-Phone_Number : Integer
-Email : String
-Start_date : String
-Designation : String
-Branch_id : String
operations
+Perform_task()

Manager
attributes
-Manager_id : String
-Team_count : Integer
-Branch_assigned : String
operations
+Authorization()
+Managing_cars()

Clerk
attributes
-Clerk_Id : String
-certification : String
operations
+rate_car()
+cal_performance()

Mechanic
attributes
-Mechnaic_id : String
-Mech_certi : String
operations
+repair_Car()
+car_chec()

Branch
attributes
-Address : String
-City : String
-State : String
-zip : Integer
-Branch_id : String
-Branch_Manager : String
-Manager_Id : String
-Team_id : String
-Operation_time : Date
-Operation_day : Date
-Branch_rating : Double
-branch_capacity : Int
-Revenue : Long
operations
+allot_car()
+Recieve_car()
+inspect_car()
+cal_penelty()
+unnamed1()
+availability()

Brach Performance
attributes
-car_utilization : String
-No_Of_Rentals : int
-Turnover : Long
-profit : Long
-customer_satisfaction : String

Customers History
attributes
-Customer_Name : String
-Customer_Id : String
-Rating : double
-Last_reserved : Date
-description : String
operations
+Get_details()

Cars
attributes
-model_no : String
-Company_Name : String
-Make_year : Date
-Mileage : Int
-Registration_number : Long
-color : String
-Type : String
-Cost : double
-Gross_Weight : double
-Length : double
operations
+Get_Mileage()
+Get_Location()
+get_Fuel_check()
+get_Last_Serviced()

ownership
attributes
-ownership_id : String
-start_date : Date
-end_date : Date
-branch_id : String
-car_id : String
operations
+get_branch()
+cal_duration()

Maintainance
attributes
-Mantainance_id : String
-Date : Date
-Depot_ID : String
-Damage_description : String
-schedule : String

Service Depot
attributes
-Depot_id : String
-Depot_address : String
-Depot_capacity : Int
-Depot_manager : String
-Operation_hours : Date
operations
+Check_last_servived()
+Check_fuel()
+check_oil()
+check_water()
+check_tire()
+check_windshield()
+check_lights()
+check_exhaust()
+check_wiper()
+check_breaks()

Customer
attributes
-Name : String
-DOB : Date
-Gender : String
-SSN : String
-Licence_No : String
-Address : String
-Email : String
-Phone_No : Long
-Rating : Double
-CreditCard_no : Long
operations
+return_car()
+recieve_car()
+make_complain()
+feedback()
+pay_bill()
+reserve_car()

Reservation
attributes
-Name : String
-Date : Date
-Location : String
-Time : Date
-Branch : String
operations
+cal_Duration()
+is_reserved()

Credit Card
attributes
-Name : String
-CVV : Integer
-Valid_From : Date
-valid_through : Date
-card_number : Long
operations
+Make_pament()
+cal_final_balance()

Car Group
attributes
-carGroup_id : String
-capacity : Integer
-Car_type : String
-Price_Range : double
-Car_Count : Integer
operations
+add_cars()
+remove_cars()
+display_availability()

Loyalty
attributes
-Is_loyal_mem : Boolean

Loyalty Incentive Scheme
attributes
-Member_id
-Number_of_reservation
operations
+Reservation_count()

Rentals
attributes
-Rental_period : Date
-Rental_time : Date
-Rental_category : String
-Rental_rate : Double
-Rental_location : String
operations
+is_valid_rental()

Relationships (labels):
Ask, Records, Checks, ownership, Rent, records, Reserve, Pays through, Maintainance, Have, Books a, Loyalty, Available for, checks
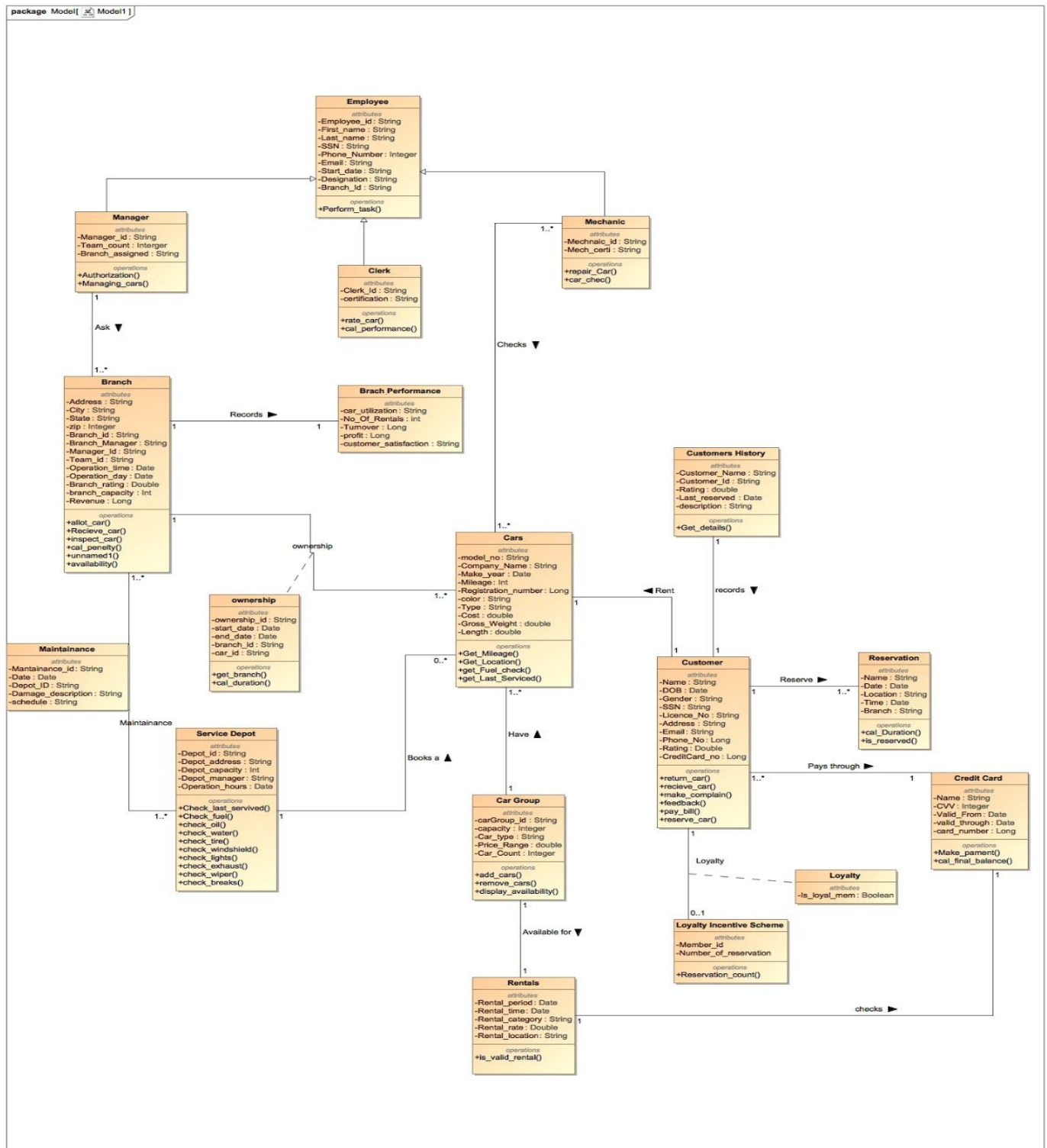
Figure 1:UML diagram

Following is the brief description of **UML Entities**:

1) **Cars** : Cars servers as one of the major entity in the UML model of EU Rental , cars are assigned to the reservation and each car belong to a particular group. Cars rented from one branch can be returned to the other branch.

2) **Branch**: Branches serves as an entity in the EU Rental UML model , Each Branch has a manager. A branch can as the other branch for the car if a car of particular type is not available at the prior branch location, additionally each branch is served by a particular Service depot.

3) **Customers**: Customer as an Entity can make several reservations but only one car rented at a time. EU-Rent keeps records of customers, their rentals and bad experiences such as late return, problems with payment and damage to cars.

4) **Customer History**: EU-Rent keeps records of customers, their rentals and bad experiences such as late return, problems with payment and damage to cars.

5) **Reservation**: At the end of each day cars are assigned to reservations for the following day.A customer can have several reservations but only one car rented at a time.

6) **Car Group**: At each branch cars, classified by car group, are available for rental.The rental period and the car group are specified at the time of reservation.

7) **Credit Card Company**: The credit card used to guarantee a rental must belong to one of the authorized drivers and this driver must sign the rental contract. Before releasing the car, a credit reservation equivalent to the estimated rental cost must be made against the guaranteeing credit card.Only the basic rental cost of a free rental can be bought with points. Extras, such as insurance, fuel and taxes must be paid by cash or credit card.

8) **Loyalty Incentive Scheme**: Customers in the loyalty incentive scheme have priority for free upgrades.To join the loyalty incentive scheme, a customer must have made 4 rentals within a year.

9) **Rental**: Each branch has a manager and booking clerks who handle rentals.EU-Rent will also accept immediate ('walk-in') rentals, if cars are available.EU-Rent keeps records of customers, their rentals and bad experiences.Cars needing repairs (other than minor body scratches and dents) must not be used for rentals. A free rental must be booked at least fourteen days before the pick-up date and Free rentals do not earn points.

10) **Employee**: Employee in EU Rental organization are divided into Managers, Clerks and Technicians.

11) **Branch Performance**: Each branch must be set targets for performance — numbers of rentals, utilization of cars, turnover, profit, customer satisfaction, etc. Where performance requirements conflict (e.g., profit vs. customer satisfaction when a customer requests a reduction in charges after an unsatisfactory rental) heuristics must be provided to guide branch staff.Performance data must be captured.

12) **Service Depot**: EU-Rent also has service depots, each serving several branches. Cars may be booked for maintenance at any time provided that the service depot has capacity on the day in question. For simplicity, only one booking per car per day is allowed. A rental or service may cover several days.

13) **Manager** : Each branch has a manager and booking clerks who handle rentals.The branch manager may ask other branches if they have cars they can transfer to him/her.

14) **Clerk** : Each branch has a manager and booking clerks who handle rentals.

15) **Mechanic**: Rented cars must meet local legal requirements for mechanical condition and emissions for each country that may be visited during the rental.

Following is the brief description of **Associative Classes**:

1) **Ownership**: If a car is returned to a branch other than the one that rented it, ownership of the car is assigned to the new branch. When a car is dropped off at a branch other than the pick-up branch, the car's ownership (and, hence, responsibility for it) switches to the drop-off branch when the car is dropped off.When a transfer of a car is arranged between branches, the car's ownership switches to the 'receiving' branch when the car is picked up.

2) **Loyalty**: Customers in the loyalty incentive scheme have priority for free upgrades.To join the loyalty incentive scheme, a customer must have made 4 rentals within a year.

3) **Maintenance**: Cars may be booked for maintenance at any time provided that the service depot has capacity on the day in question.The end date of the rental must be before any scheduled booking of the assigned car for maintenance or transfer.A customer may request a rental extension by phone — the extension should be granted unless the car is scheduled for maintenance.If there is a shortage of cars for rental, routine maintenance may be delayed by up to 10% of the time or distance interval (whichever was the basis for scheduling maintenance) to meet rental demand.

3) Functional and Non-Functional Requirements
Functional Requirements

1) Manage Booking
a customer can initiate a booking
a customer can cancel a booking
a customer can Raise a complain
a customer can rate his ride and give feedback.

2) produce schedule
 A manager prints and records the total ride of each car in a day
A technician tests the car before they leave an outlet

Non functional requirements
1) Operational requirements
The system will be having a app for IOS and Android and can also be accessed through a EU website.
System should either take a request or cancel it completely during the power cut or system break down.
System should automatically identify the old and new customers and give an offer if available
System should have the coordinate and track every car that is rented if they go out of the radar.
System should be able to safeguard the credentials and customer assets at all time.
 System should have the ability to recover in the case of disaster.

2) Performance Requirement
 System should be able to handle multiple requests at a time, with delay no more than 1.5 sec.
The mobile app of EU rental should be bug free and should be enhanced over time with new functionalities.
A car should choose the fastest route to reach the customers , rather going for the shortest routes which might be time taking some times.
The website and server should be live 24 hours a day , 365 days a year.
System will retrieve and print the memo of all the requests served on a particular day less than 3 seconds.

3) Security Requirement
If a request is being canceled more than twice by same user (or mobile)
then only manager will have the right to pass it.
Only a EU technician(with a unique pin) will have access to the internal machinery of these self driven car.
A car will be having all time tracking and a phone line that will directly
connect a customer to an EU representative or 911 in case of calamity.
4. Use Cases

1. **Use Case Name: EU Rental Structure**

    Actor: EU employees and systems

Description: This use case identifies the systems and objects needed for EU rental, what EU Rental consists of and their relationships.
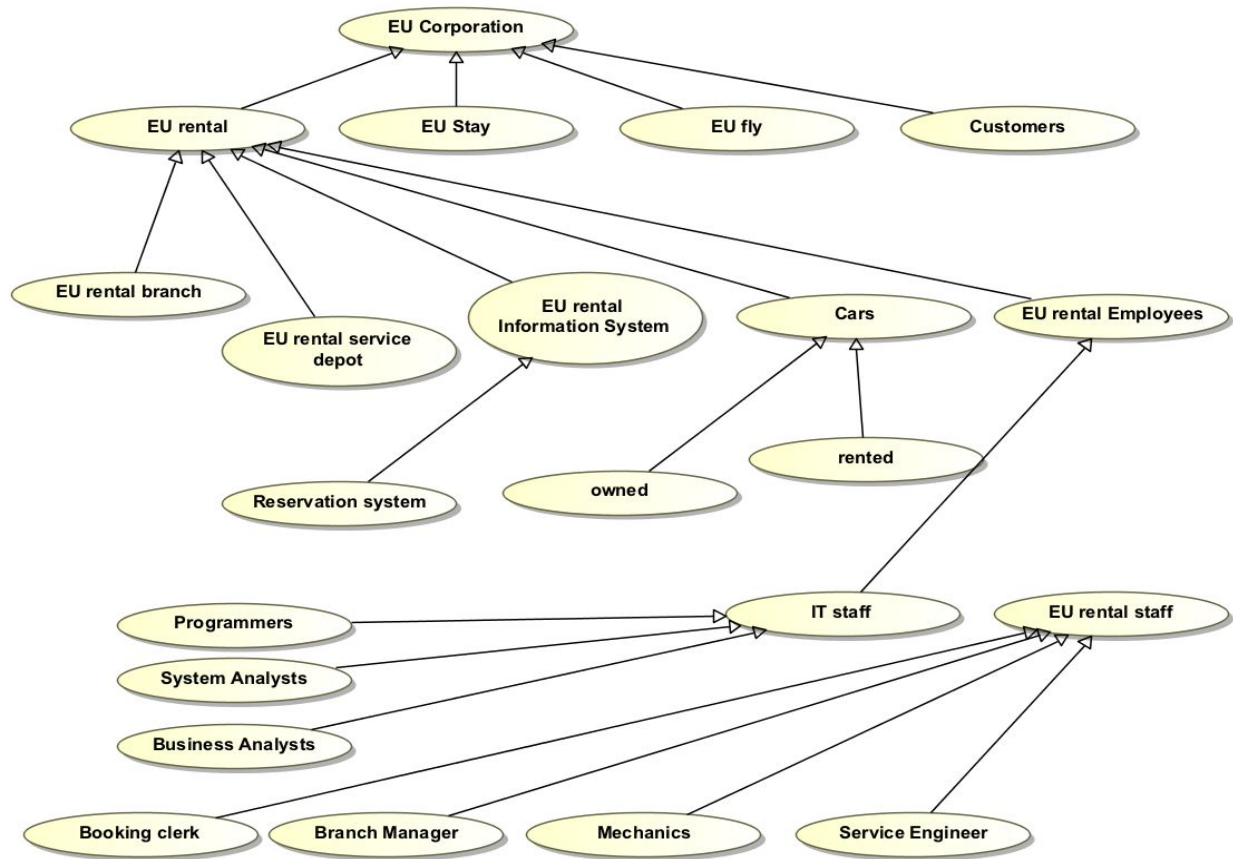
Figure 2: Use case 1 - EU Rental Structure

2. Use Case Name: EU Rental Online Reservation System

Actor: EU Rental customers, reservation system and payment system.

Description: Customers make online reservation using EU-rental online reservation system. Through the system, customers have options to sign up, log in, search car, cancel, and modify reservation. Using the searching feature, customer will be able to see which car model is available, choosing the start and end date and time, pickup and drop off location. After the desired reservation is selected, customer will haves options to make a payment on the reserved car either before or after the car and get the reservation confirmation.
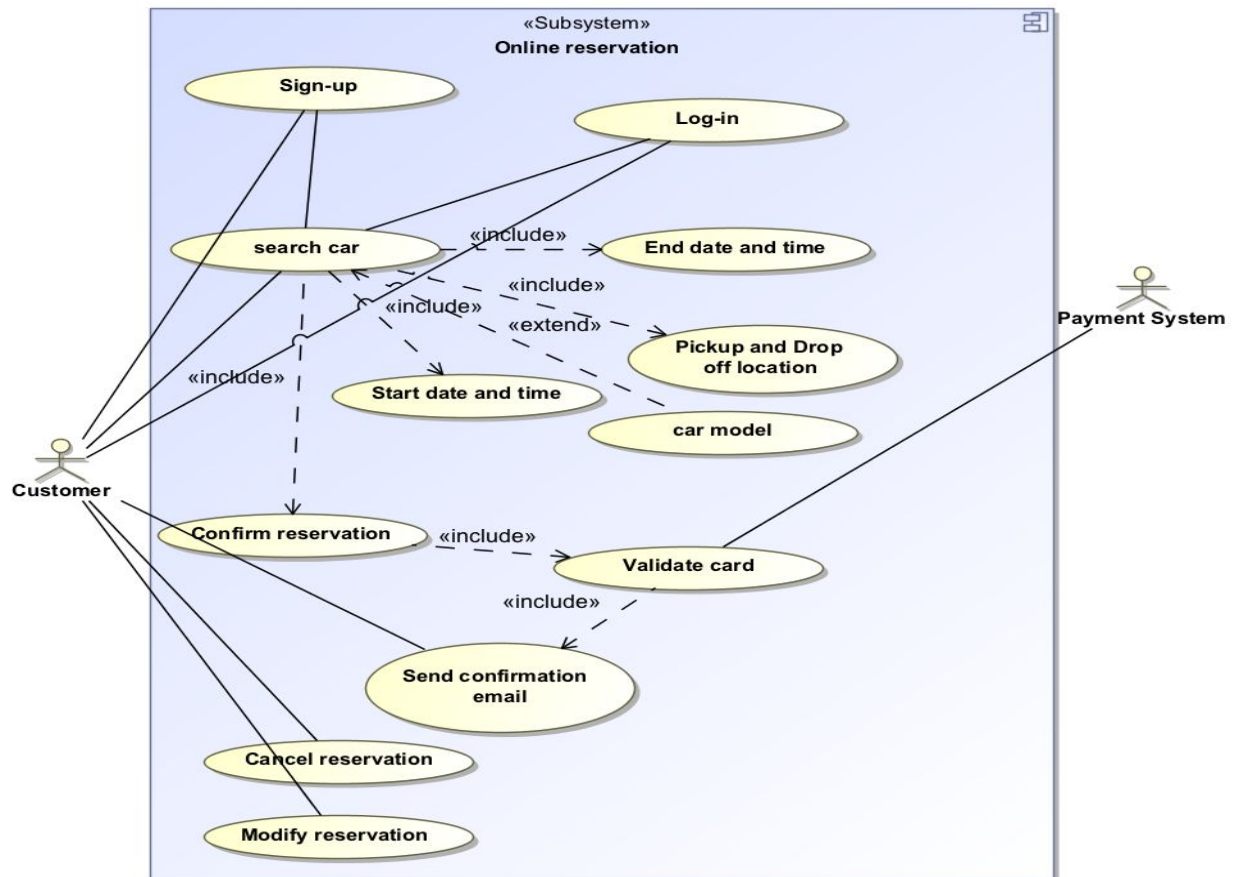
Figure 3: Use case 2 - EU Rental Online Reservation System

3. Use Case Name: Walk in reservation and pick up

Actor: EU Rental customers, reservation system, payment system, Booking clerk, and mechanic.

Description: Apart from using the online reservation system, customer can make reservation at one of EU Rental branch and get the same day pick up if the car is available. Customer approaches the booking clerk at the selected location to make the reservation and payment. Booking clerk will handle the customer request and payment using the online reservation and payment system. Before the car is released to the customer, mechanic need to make sure that the care is working properly.

Figure 4: Use Case 3 - Walk in reservation and pick up

4. Use Case Name: Car drop off

Actor: Customer, booking clerk, mechanic, credit card company, reservation system, payment system, and EU Rental service depot.

Description: After the customer is done using reserved car, the car need to be dropped off at the chosen return location. If the car is dropped off after selected time, the penalty will be applied to the customer. In addition, the car need to be checked by the mechanic to make sure that the car is returned in the same condition (no damage). If so, the customer will also be charged and the car will be send to EU Rental service depot for repair
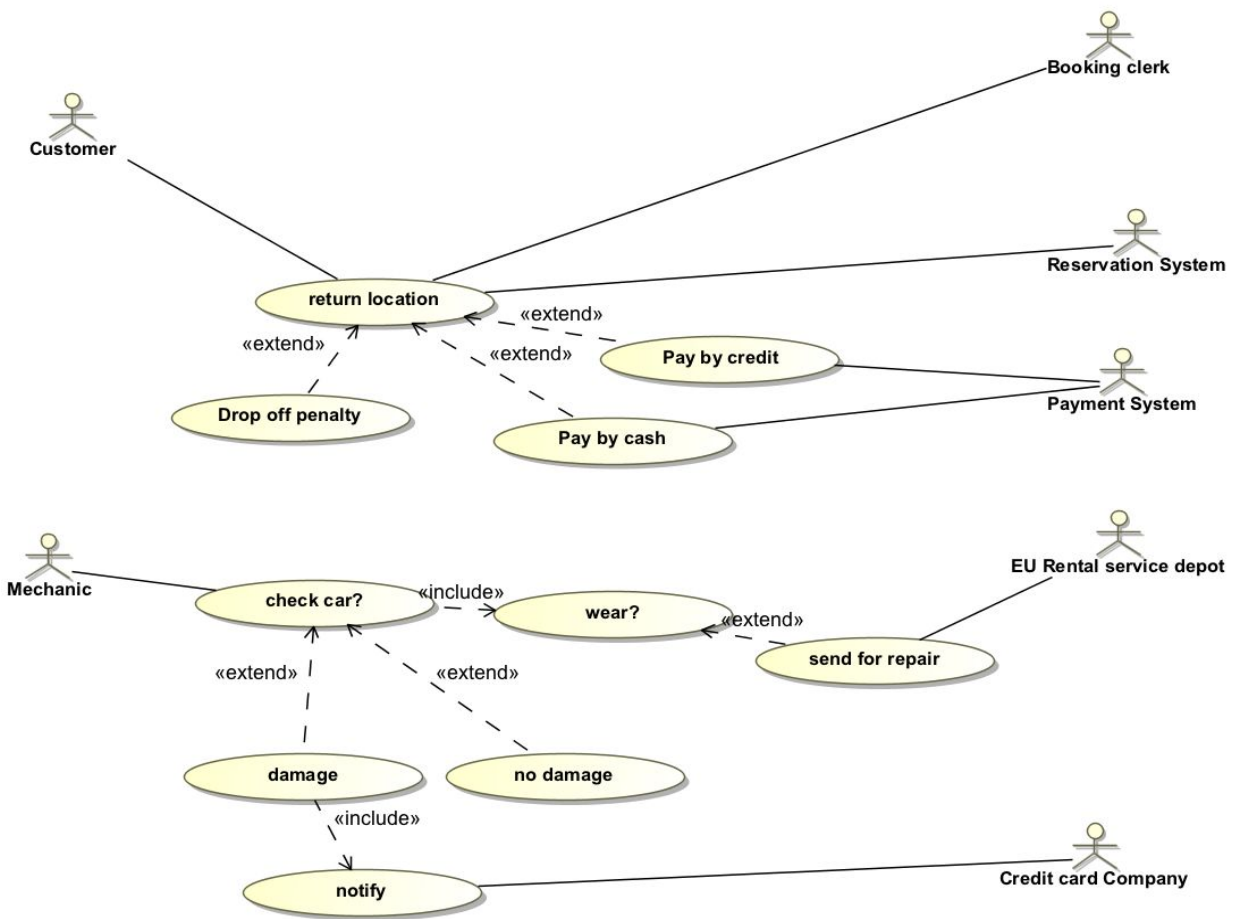
Figure 5: Use Case 4 - Car drop off

5. Use Case Name: Pick up rule

11

Actor: Booking clerk and reservation system
Description: This use case specifies the business pick up rule. Booking clerk monitors the reservation and scheduled time of reserved car. If the reserved car is not picked up within the 90 minutes of scheduled time, the customer will be charged for the fee and that car will be available to release for walk-in customer.
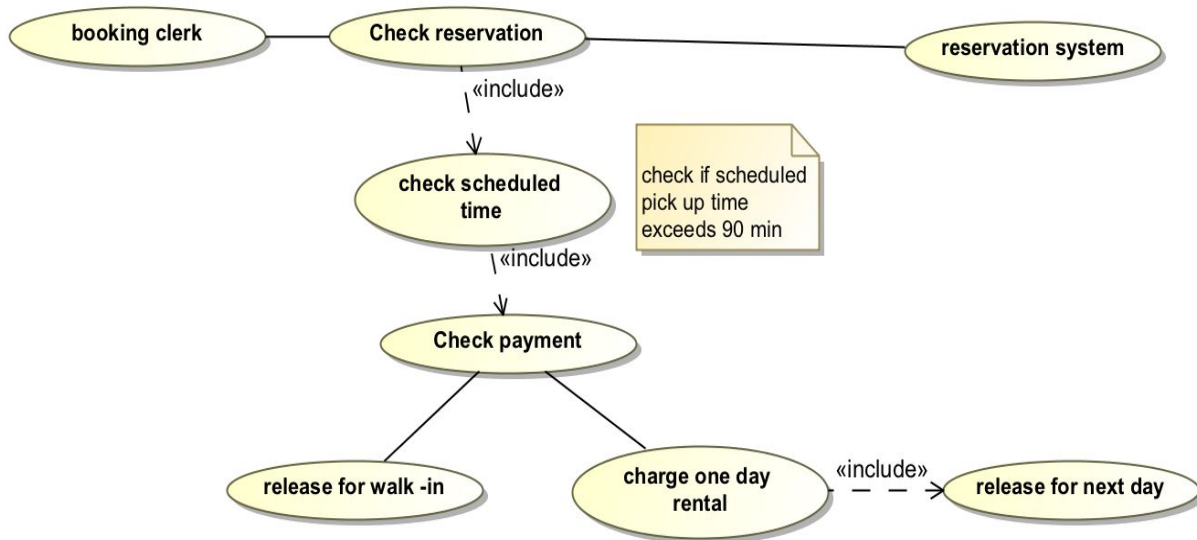
Figure 6: Use Case 5 - Pick up rule

6. Use Case Name: Rental allocation

Actor: Booking clerk, Reservation System, EU rental branch, Competitor branch.
Description:Booking clerk performs the task of car allocation.His duties are to check cars,make a downgrade,make a bumped upgrade.If the car is not available he can call another EU rental branch or he can rent the car from competitor branch.
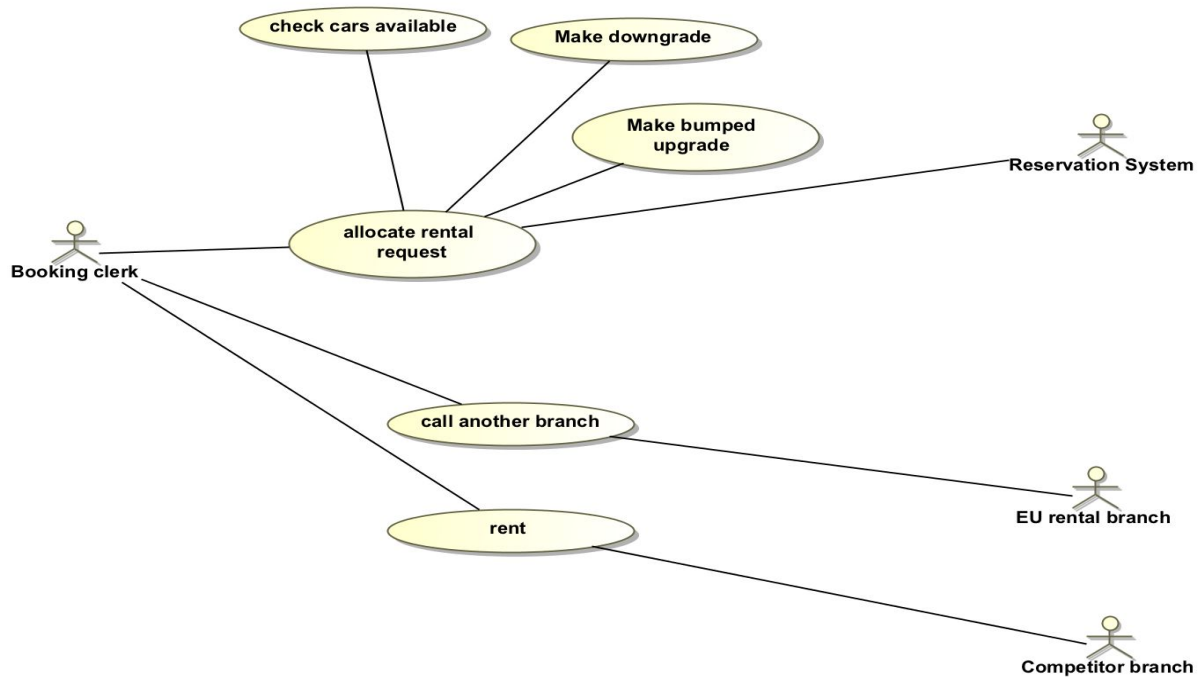
Figure 7: Use Case 6 - Rental allocation

7. Use case name: Compute statistics

Actor: Booking clerk, Branch Manager, EU Rental Information System,
Description: Branch manager is responsible for the profits and loss.Branch Manager uses EU
Rental Information System to compute performance statistics.Branch manager is also
responsible for control actions such as change of resources,change responsibility and
operational guidance.
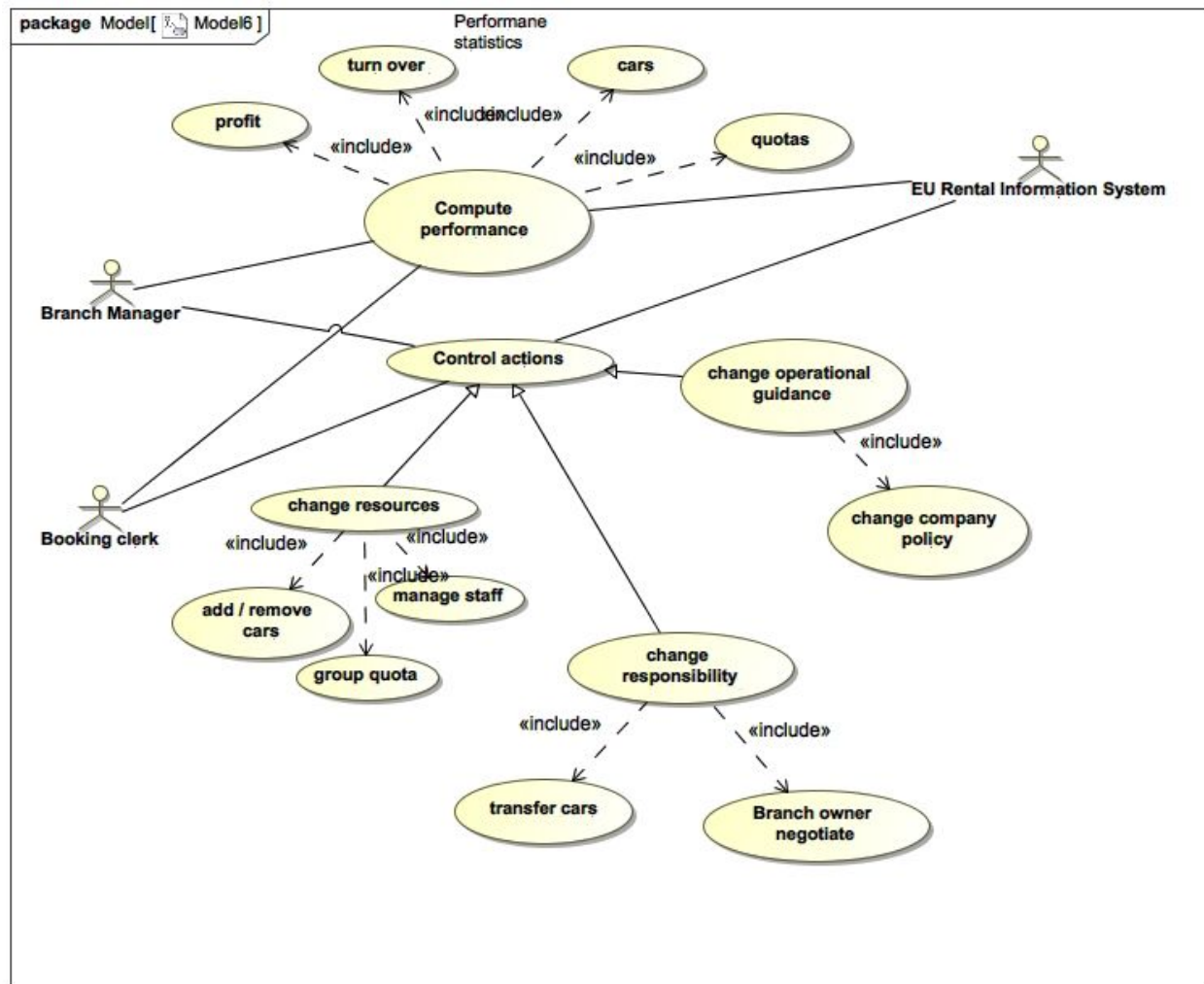
Figure 8: Use Case 7 - Compute statistics

8. Use case Name: Autonomous car rental

Actor: Booking clerk, EU Rental information system, customer, autonomous car, and EU rental mobile/ web application

Description: Customer can look for autonomous car to rent using the EU Rental information System or Eu rental mobile/web application. Customer will be able to select the pick up address, date, and time. Once the reservation is confirmed, the booking clerk will initiate the rental for autonomous car to pick up the customer by feed destination address, date and time into the EU Rental Information System. Then, the autonomous car will be assigned to specific customer.
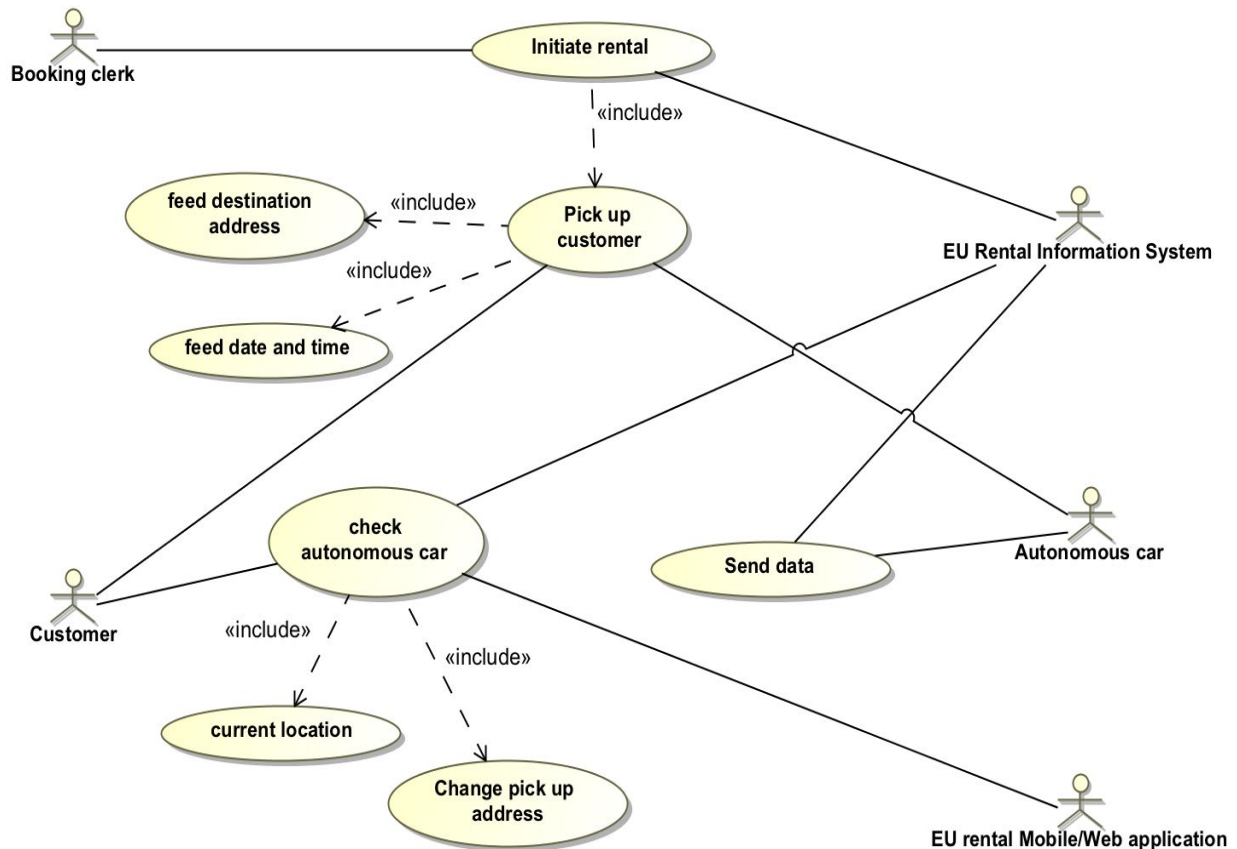
Figure 9: Use Case 8 - Autonomous car rental

5. Dynamic Interaction Models

We will discuss the following two concepts for better understanding of Dynamic Interaction Models:

1) **Communications and Sequence diagram**
2) **DFD**

Communications and Sequence diagram

1) **Communications and Sequence diagram - Return car**
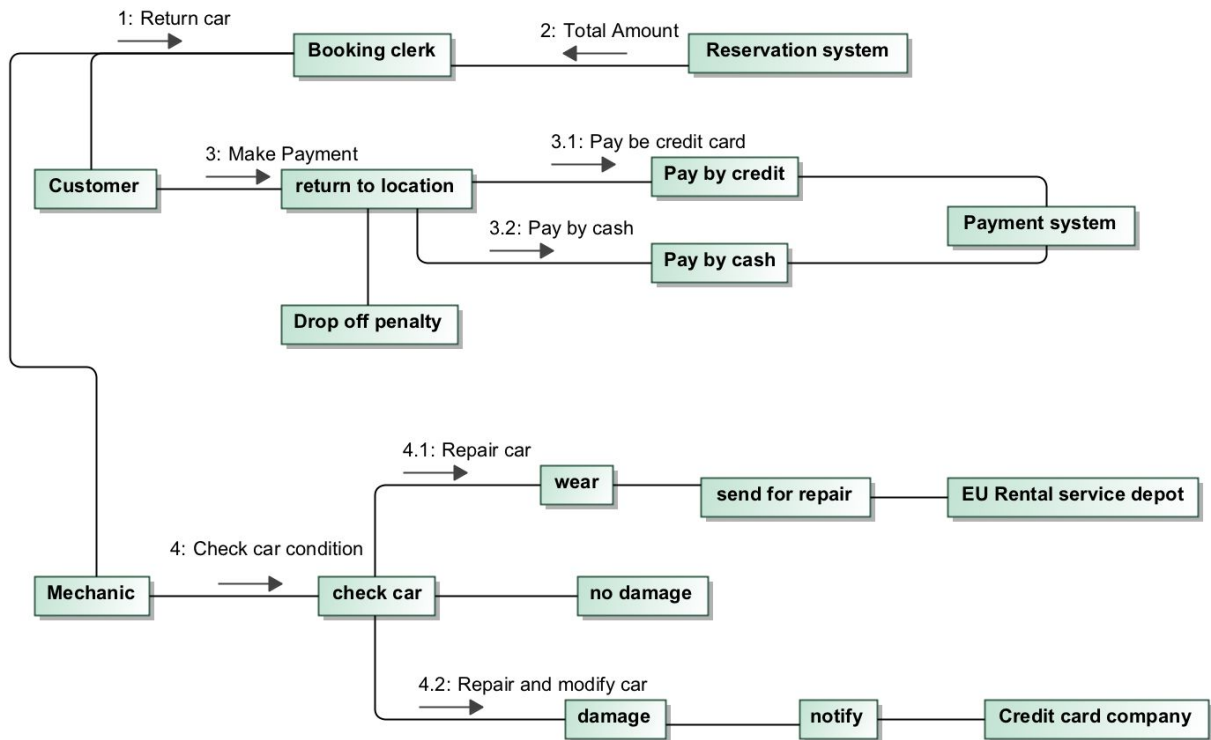
Figure 10
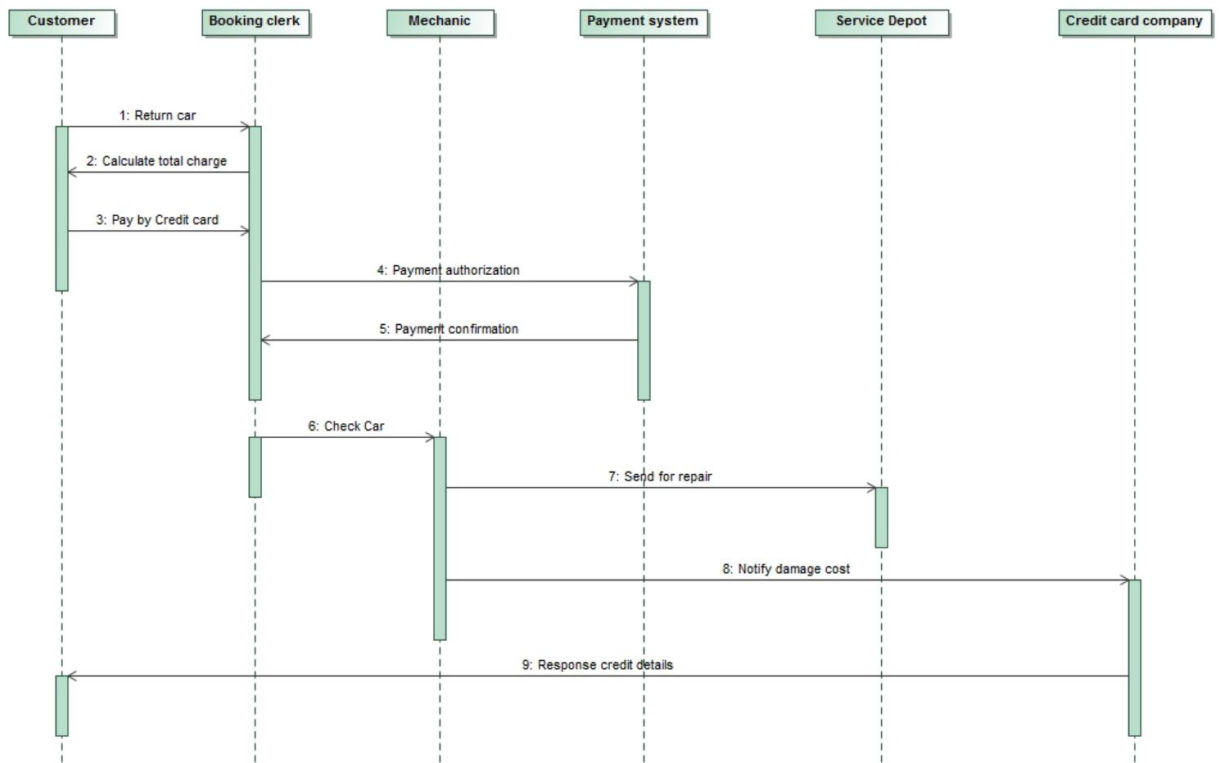
Figure 11

2) Communications and Sequence diagram - Booking
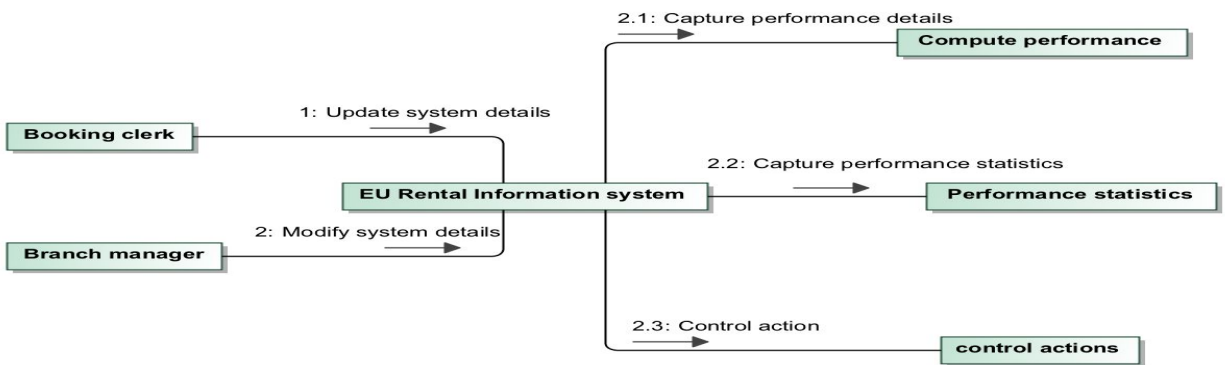
Figure 12
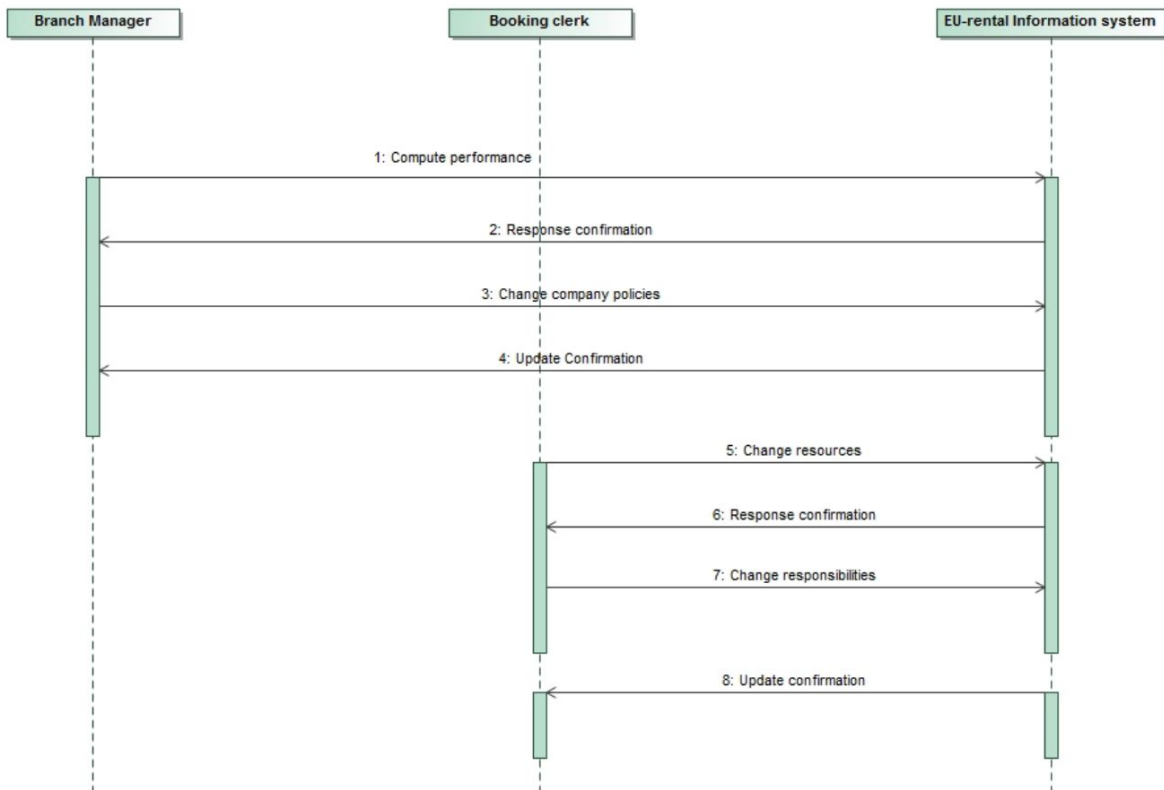
Figure 13

3) Communications and Sequence diagram - Autonomous car rental
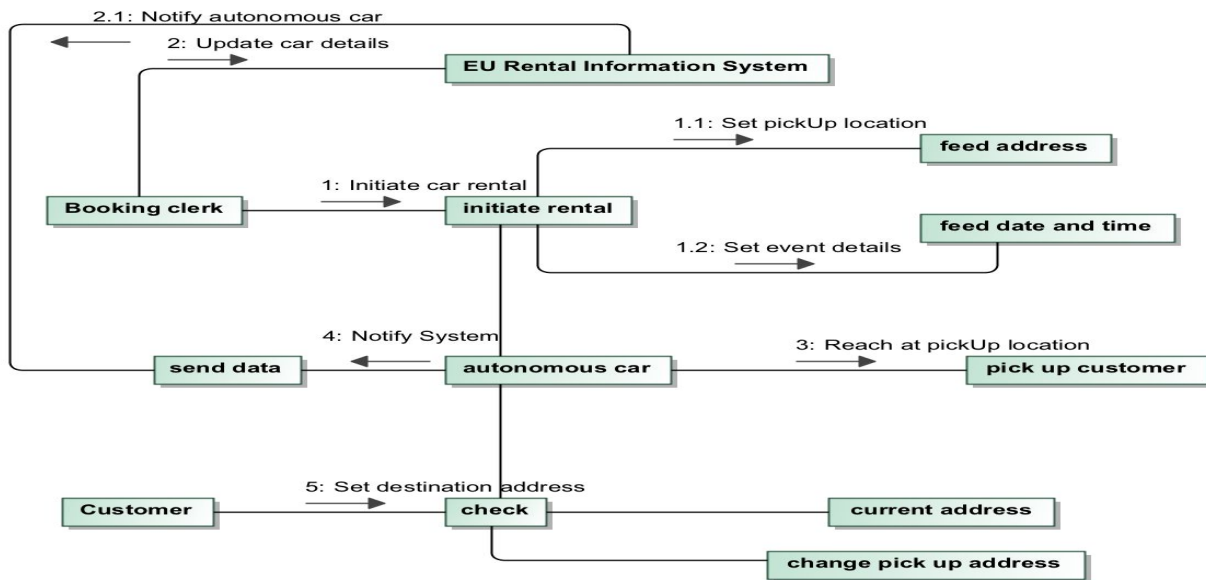
Figure 14
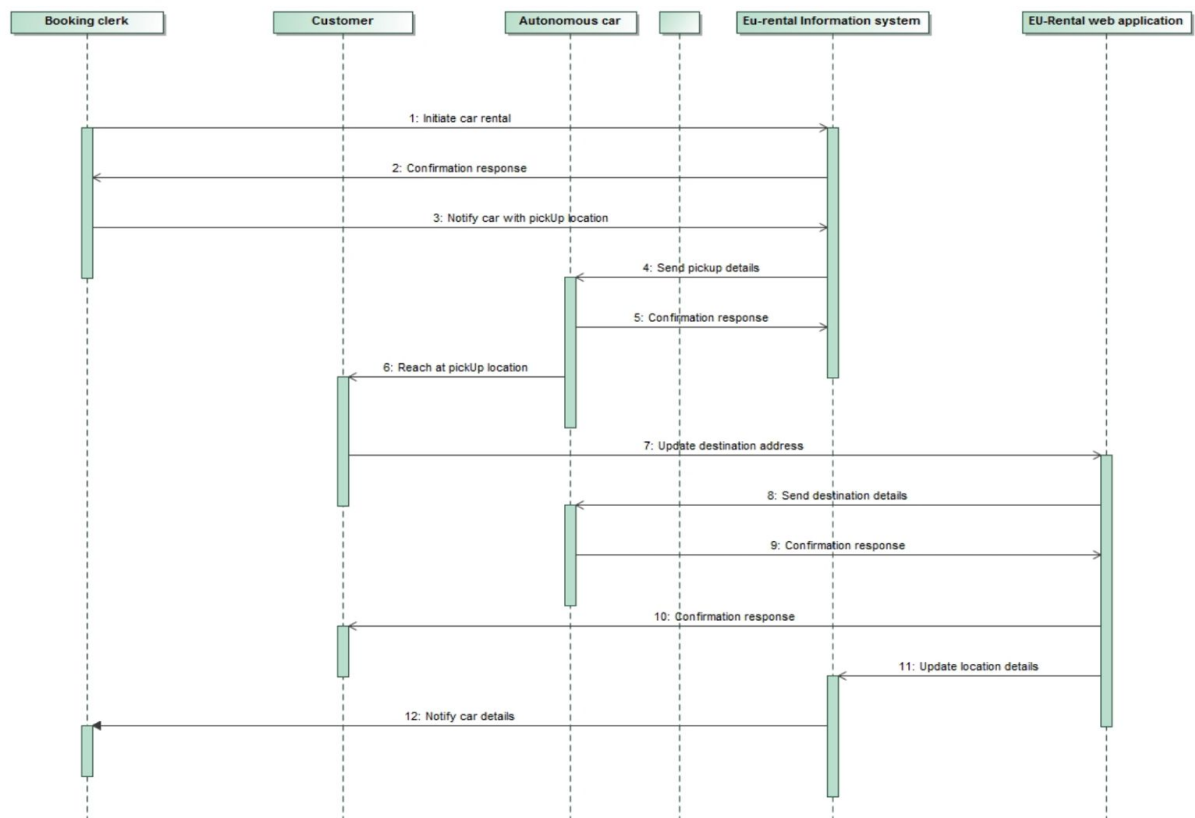
Figure 15

4) Communications and Sequence diagram - Online rental
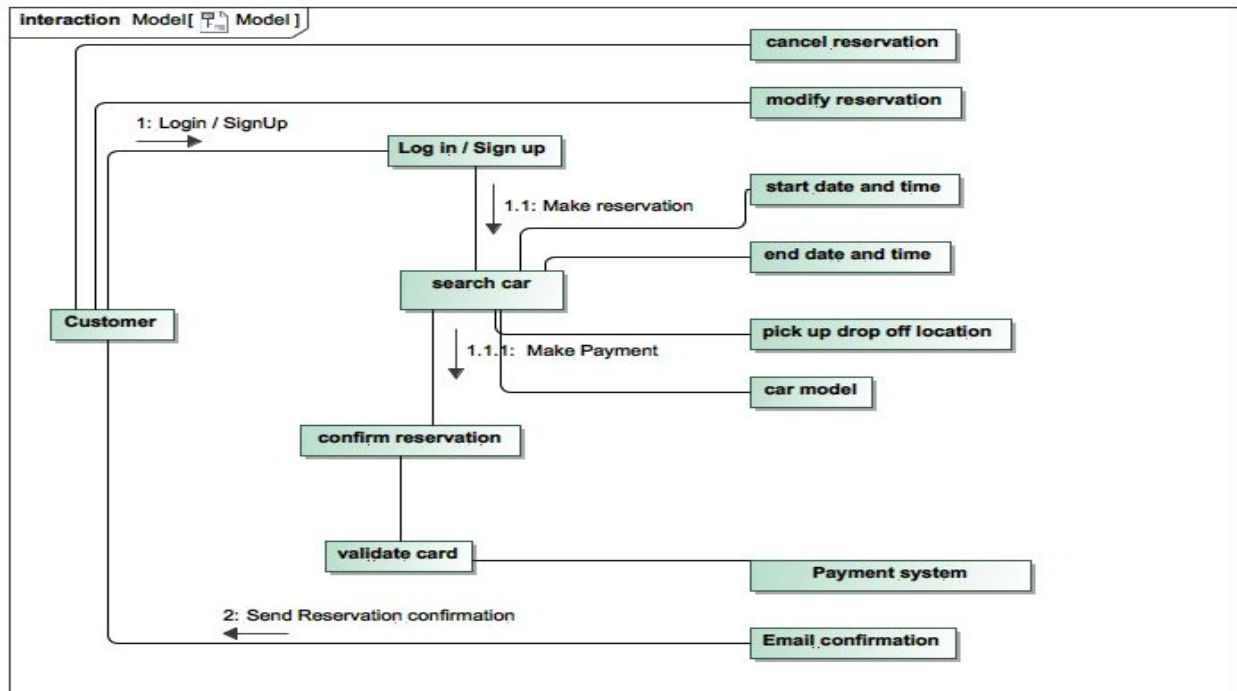
Figure 16
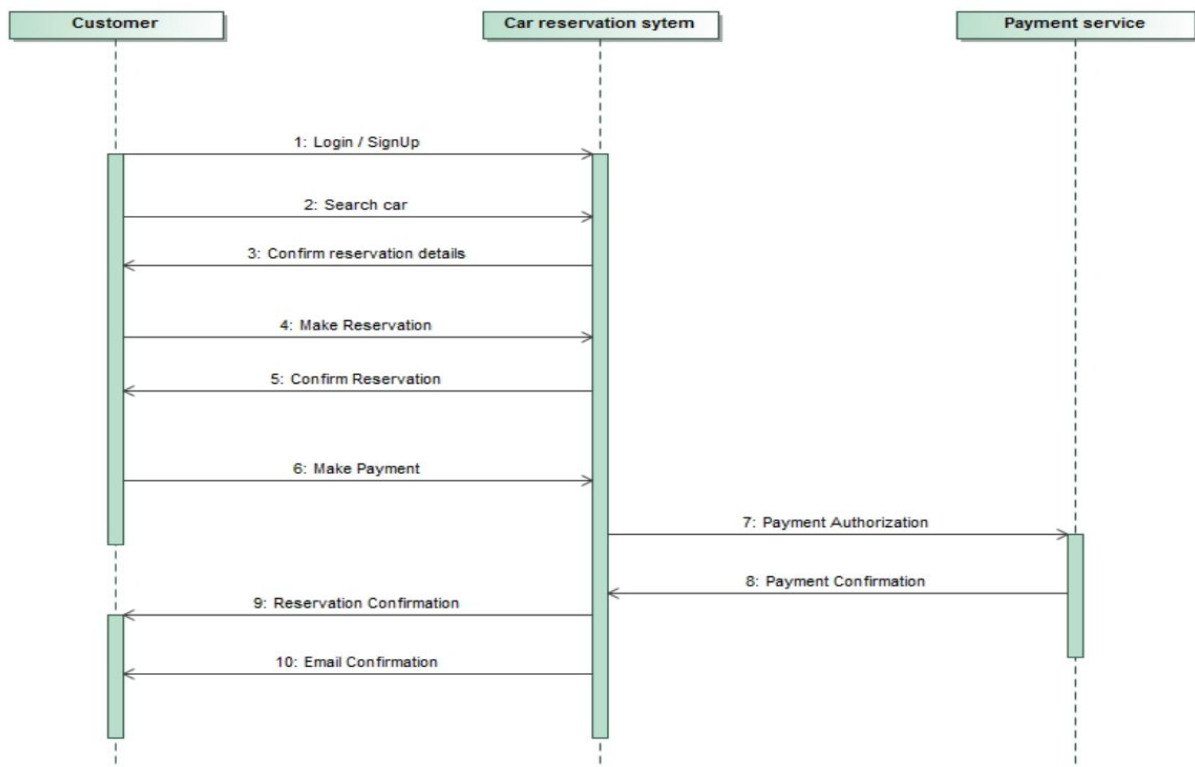
Figure 17

5) Communications and Sequence diagram - Business rule
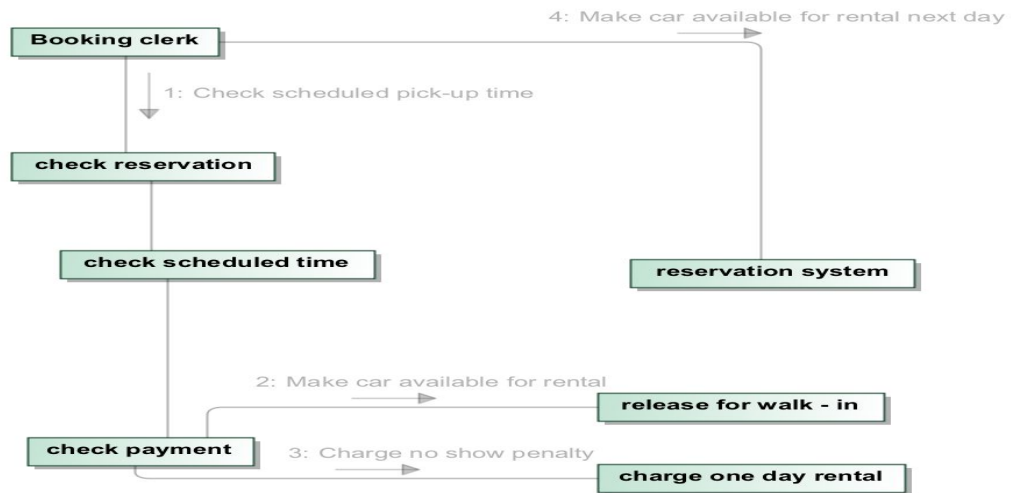
Figure 18
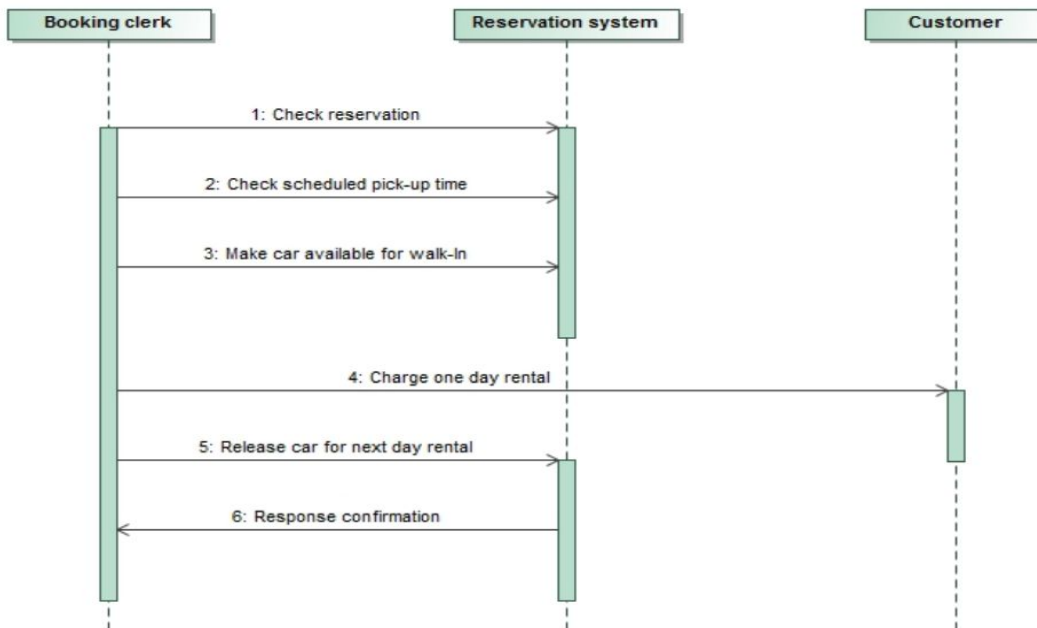
Figure 19

6) Communications and Sequence diagram - Allocate rental
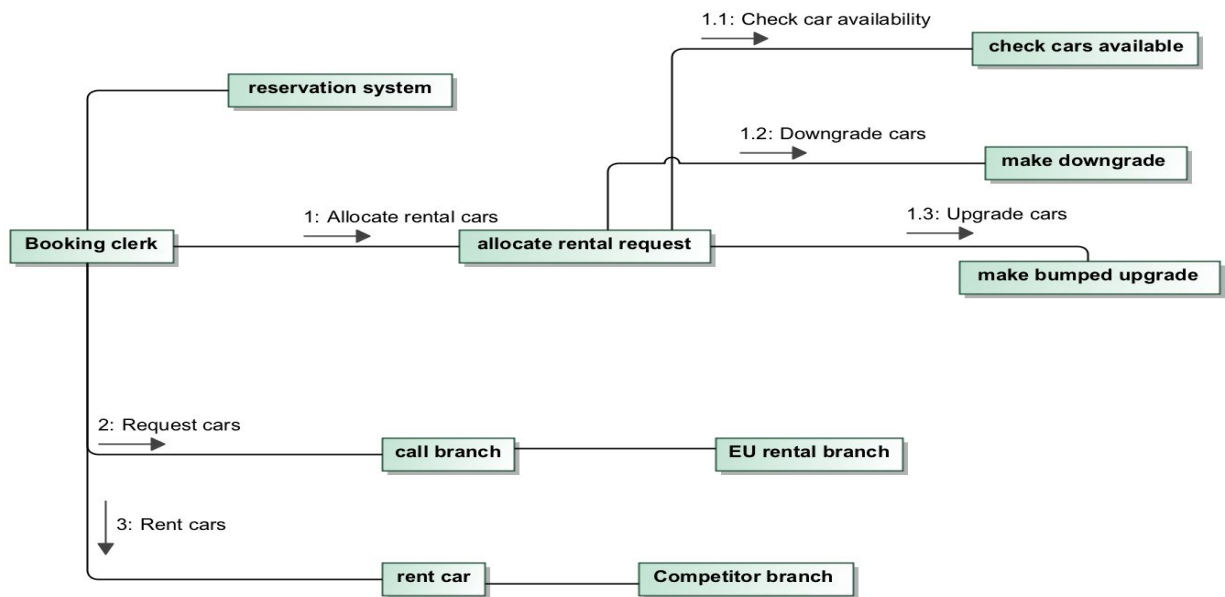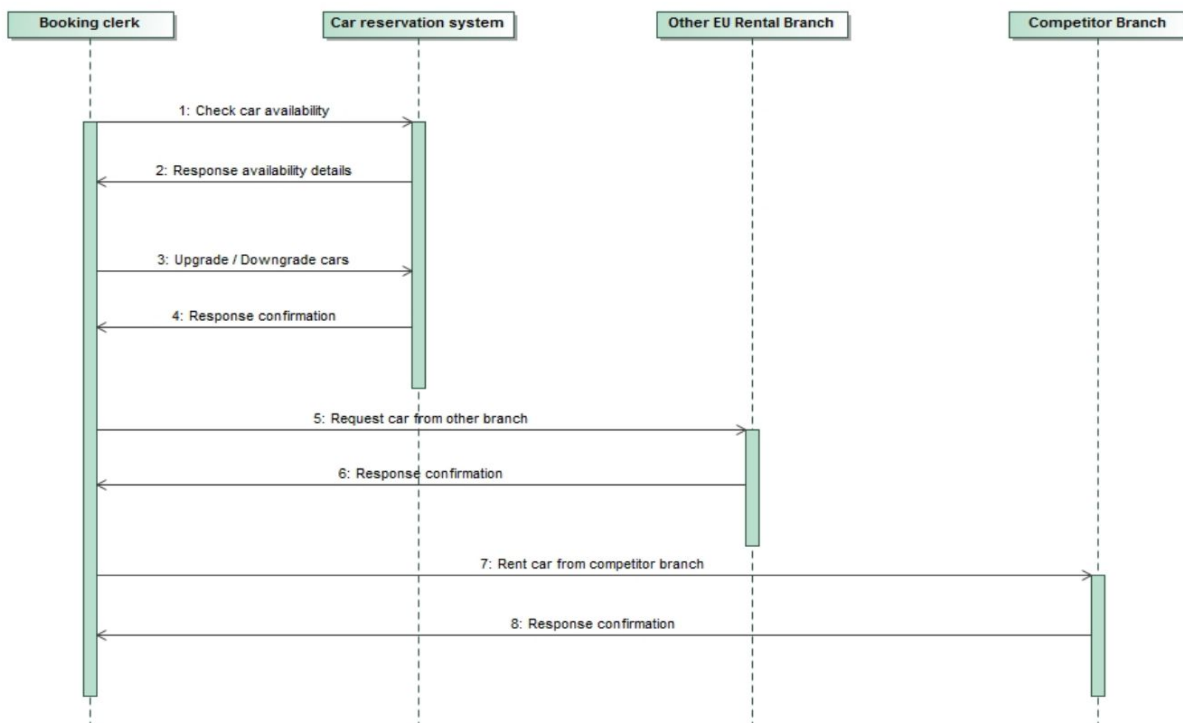

Figure 20

Figure 21

2) DFD (Data Flow Diagrams)

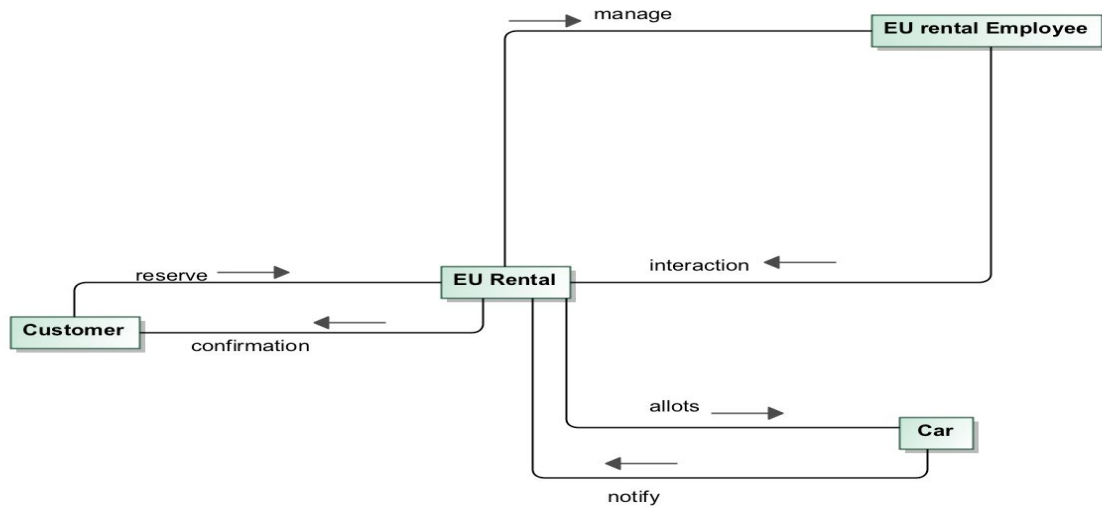Following are the three Data Flow Diagrams for EU rental system proposal.
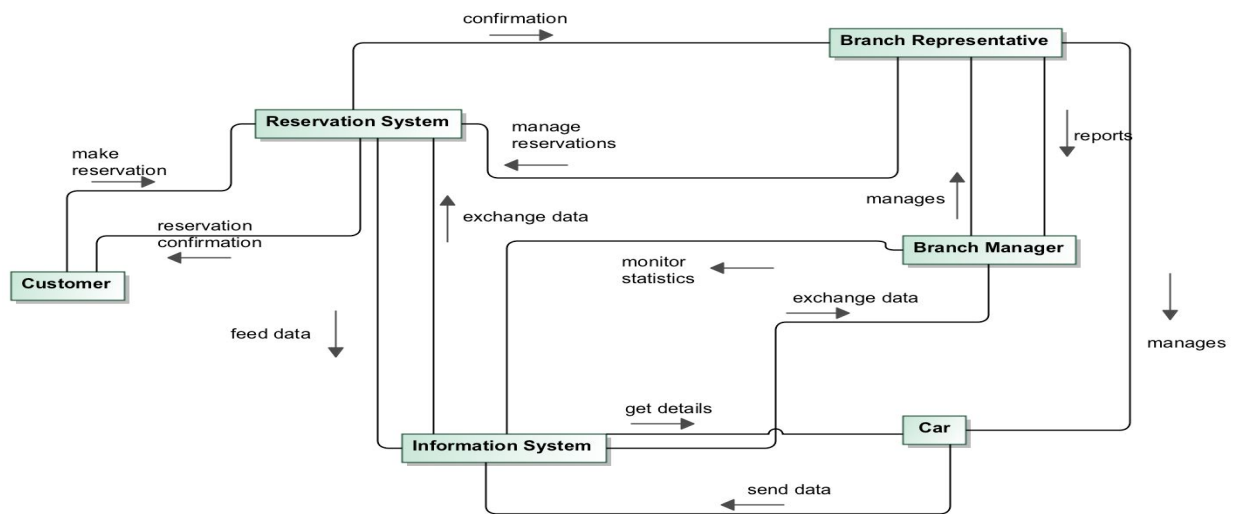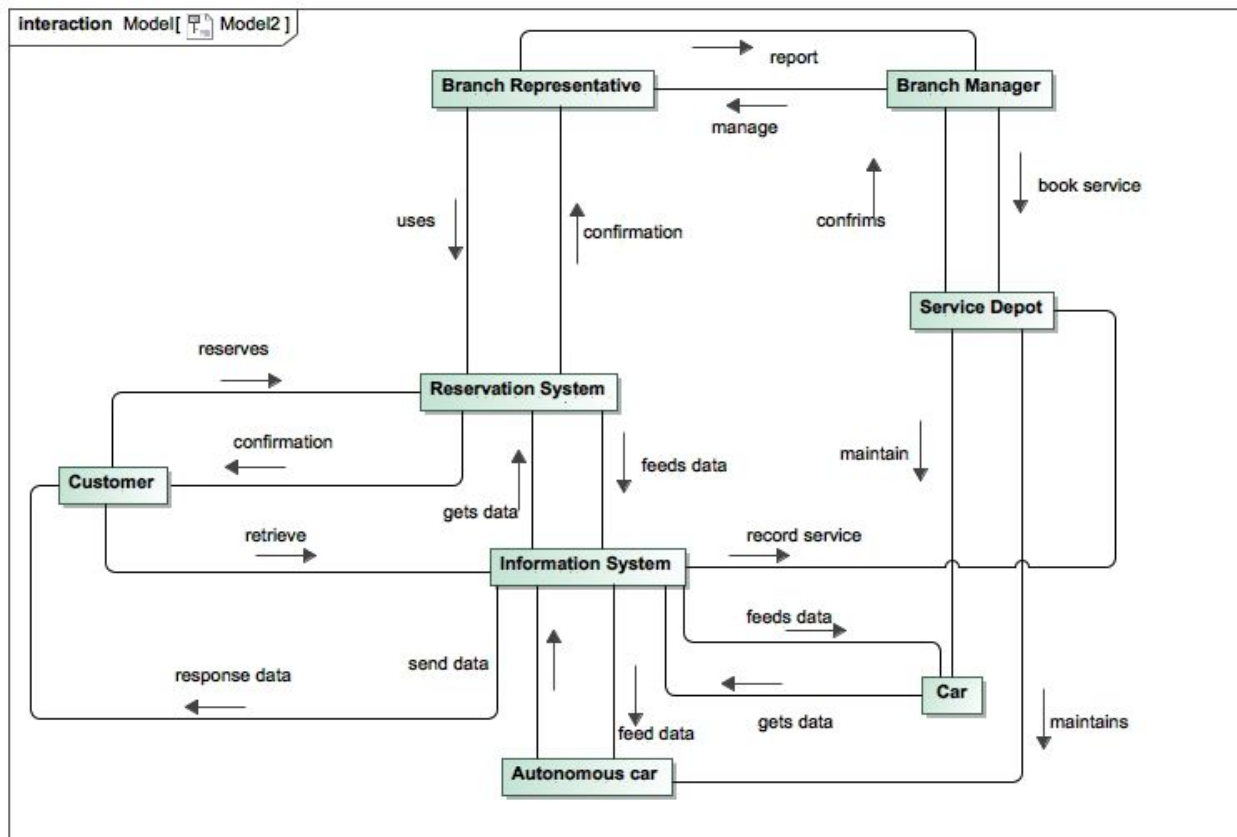

Figure 22: DFD 0

Figure 23: DFD 1

Figure 24: DFD 2

6. **Recommended System Acquisition Strategy**

1) The proposal for EU Rental is to add the new feature, which is an autonomous driving to expand its business and reach more people.
2) To do this, the new system needs to be developed. EU Rental already has an existing system for its current businesses, which include: Car rental, hotel, and airline.
3) The existing system has previously been custom made, as should the new system.EU Rental has its own experienced IT department who developed   and monitoring the existing system.
4) With custom development, it will allow the consistently between the existing and the new systems as it will be developed by the same development team who understand the existing system and how the company works.
5) However, there is no need to develop the new system from scratch. The existing system can be modified and developed into the new one as the features will most likely be the same but only with more features.
6) That is why packaged development will be best suited in this situation.This will save time, produce a unique solution, and get the exact system that company needed.
7) It will also save costs as there is no need to buy the system elsewhere or pay for any other resources.

EU Rental Acquisition Strategy Selection: Packaged Development

	Custom	Packaged	Outsourcing
Business need	The business need is unique	The business is common	The business need is not core to the business
In-house experience	In house functional and technical experience exists	In house functional experience exists	In house functional and technical experience does not exists
Project skills	There is a desire to build in-house skills	The skills are not strategic	The decision to outsource is a strategic decision
Project Management	The project has a highly skilled project management team	The project has a project manager who can coordinate vendor's effort	The project has a highly skilled project manager at the level of organization that matches the scope of the outsourcing deal
Time Frame	The time frame is flexible	The time frame is short	The time frame is short or flexible

Figure 25 Refer to Figure 7.4 Dennis book

7. **Recommended Architecture Design**

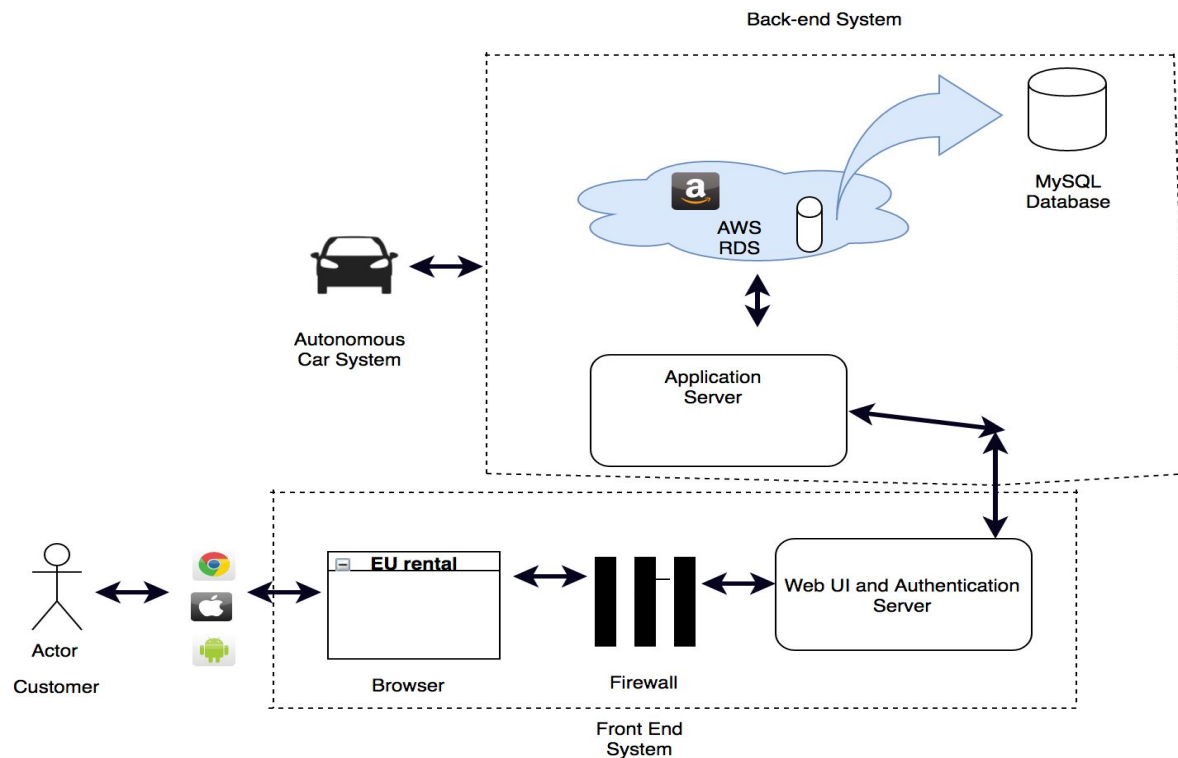Figure 26: Architecture Design

Description:
1) We are proposing to add mobile application and revamp the existing UI of Web.The Customer will interact with the system for reservation using Web or mobile app.
2) The Web or mobile app will try to authenticate the user using Web Authentication Server but before that it has to pass the Firewall.
3) The user will input the credentials for the application and then will be granted access for booking.
4) The Web UI Server will fetch the data for the Web User so that the user can interact with the web application.
5) Tasks performed by user such as booking, these functions will be provided by Application Server.
6) Back-end consists of Application Server and AWS RDS, Front-end consists of Browser,Firewall,Web UI server and Authentication Server.
7) Application Server will fetch data and perform required transactions while interacting with AWS RDS.
8) Large amount of data needs to be captured from autonomous car, this can be done by Back-end.

8. **Hardware and Software Specification**

	Standard client	Standard Web Server	Standard Application Server	Standard Database Server
Operating System	· Windows 7/8/10 · Mac/Linux	· Windows	· Windows	· AWS RDS Instance (Memory optimization)
Special Software	· Eclipse · MagicDraw · MySQL Workbench · Chrome/Firefox	· Apache Tomcat	· Java/J2EE · JBoss	· MySQL
Hardware	· 500-GB disk drive · Intel® Core(TM) i5-5200 CPU @ 2.20GHz	· 500-GB disk drive · Dual- Core Xenon	· 500-GB disk drive · Six-core Xeon	· db.r3.8x.large · 244 GB · 10 Gbps Network Performance
Network	· Always-on Broad-band preferred · Dial-up at 256 Kbps, Possible with some performance loss	· Dual 100 Mbps Ethernet	· Dual 100 Mbps Ethernet	· Dual 100 Mbps Ethernet

Figure 27

EU-Rental will have a its own software system which will be used by all its users including customers, employees, etc. The web application for EU-Rental System will be developed using Java / J2EE framework. J2EE framework (JSP, Servlets) will be used for developing front end of the software system. We are proposing to use Relational Database Service from AWS to develop back-end of the software system. We are proposing to use AWS RDS MySQL instance (db.r3.8x.large) which is having storage capacity 244 GB and 10 Gbps of network performance. Structured Query Language (SQL) will be used for database development using MySQL database engine. Apache tomcat will be used as standard web server which will serve HTTP requests. JBoss server will be used as web application server is a full stack Java EE server and it has much higher memory storage. Eclipse IDE and MySQL workbench will be used for front-end and back-end development respectively.

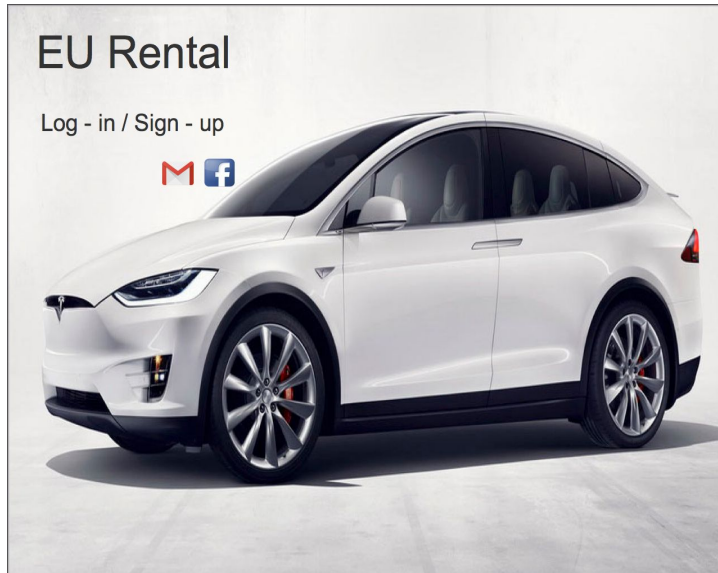9. **User Interface Design and Evaluation**
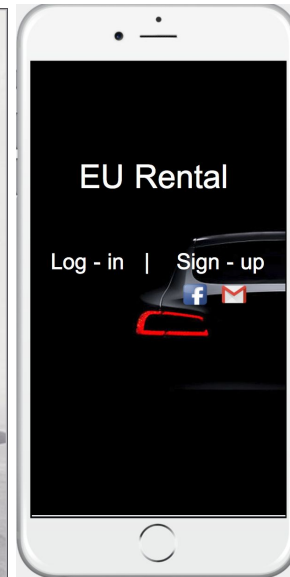
Figure 28: Web UI   Figure 29: Mobile UI

Evaluation:
1. We are proposing a revamped UI for existing EU Rental website.
2. We are proposing mobile application in android and iOS so that customer can book rental cars from mobile phones.
3. We have used heuristic evaluation technique for EU rental UI evaluation.We have focused on the following for heuristic evaluation: visibility of system status,user control and system,aesthetics and minimal design and error prevention.

10. Program Design

We will discuss the program design g using the following three:

1. **Human/machine boundaries**
2. **Structure chart diagram**
3. **Program specification**
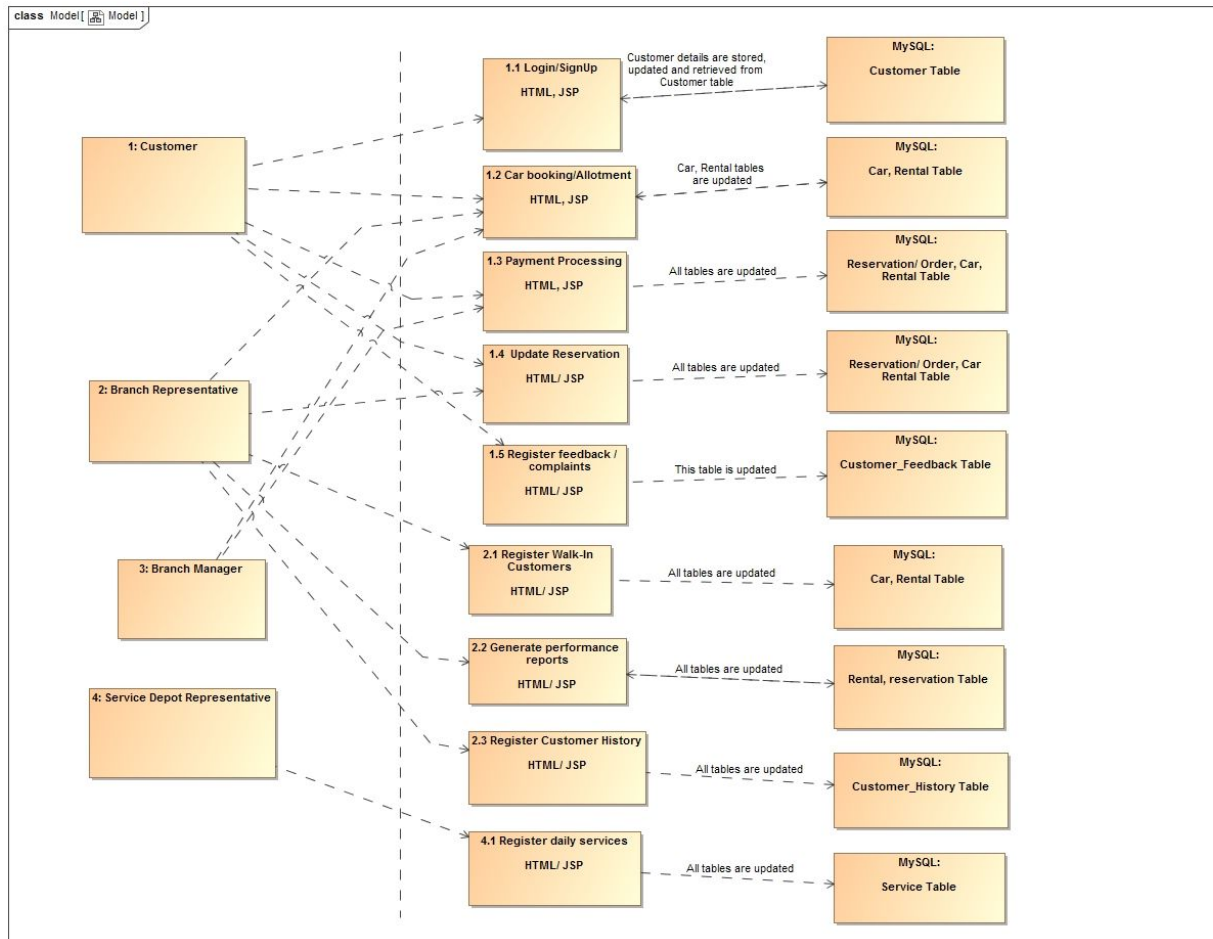
1. Human / machine boundaries:

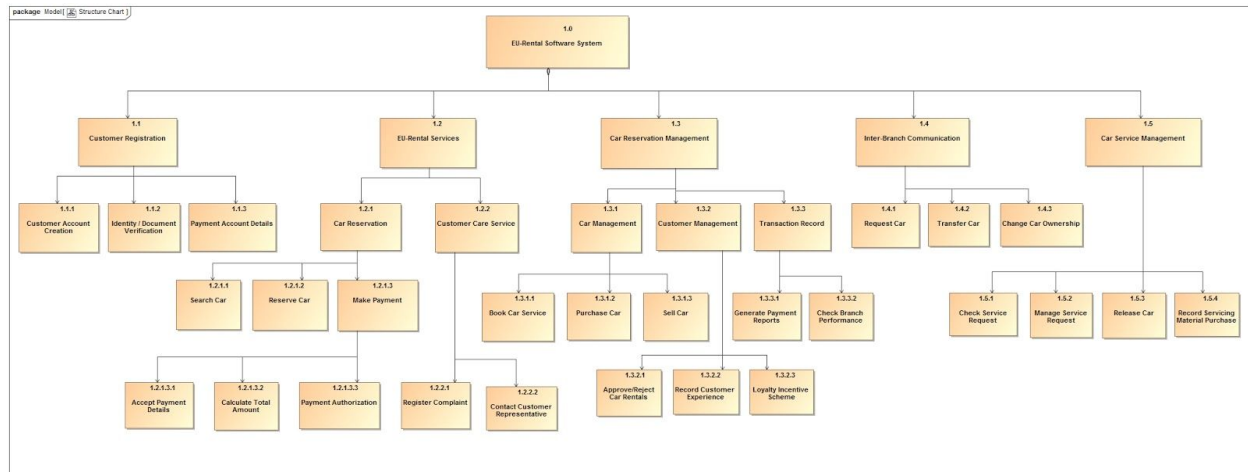Figure 30: Human/machine boundaries

2. Structure Chart Diagram:

Figure 31: Structure Chart Diagram

3. Program specification

1) **Program Specification 1.3.1.3 for Selling Cars**

Module _____

Name: Search_cars_to_be_sold

Purpose: Display old cars eligible for sell using car-group code input by the branch employee

Programmer: Rahul Deore

Date due: June 18, 2017

   HTML/JSP   JavaScript SQL

Events _____

Search by car-group push button is clicked

Input Name:	Type:	Provided by:	Notes:
Car-group code	Integer	Program 1.3.1	

Output Name:	Type:	Used by:	Notes:
Car Registration number	Number	Program 1.3.1	

Pseudocode _____

(Search_cars_to_be_sold)

sell_car = false

If (current_date – purchase_date >= 1 OR car_mileage >= 40000)

  sell_car = true

End If Return

Business rule: All cars with mileage greater than or equal to 40,000 kilometres OR purchase_date 1 year old are returned to user

2) Program Specification 1.3.2.2 for Late Returned Cars

Module _____

Name: Calculate_late_return_penalty

Purpose: Compute and return the penalty amount for late return of the car

Programmer: Rahul Deore

Date due: June 20, 2017

§ HTML/JSP   JavaScript   SQL

Events _____

Enter customer_id and car return time.

Submit inputs by calculate button is clicked

Input Name:	Type:	Provided by:	Notes:
Customer_id	Integer	Program 1.3.2	
Return_time	DateTime	Program 1.3.2	

Output Name:	Type:	Used by:	Notes:
Penalty_amount	Double	Program 1.3.2	

Pseudocode _____

(Calculate_late_return_penalty)

penalty_amount = 0.00

If (delay_hours > 0 AND delay_hours <= 6)

    penalty_amount = delay_hours * charge

End If

If (delay_hours > 6)

penalty_amount = 24 * charge

End If

Return

Other _____

Business rule: Calculate late return penalty amount based on total delay time. For more than 6 hours delay whole day rent is charged as penalty.

3) Program Specification 1.3.2.3 for Loyalty Incentive Scheme

Module _____

Name: Loyalty_incentive_eligibility

Purpose: Checks whether customer is eligible for joining loyalty incentive scheme

Programmer: Rahul Deore

Date due: June 22, 2017

     § HTML/JSP   JavaScript   SQL

Events _____

Enter customer_id

Submit input by check button is clicked

Input Name:	Type:	Provided by:	Notes:
Customer_id	Integer	Program 1.3.2	

Output Name:	Type:	Used by:	Notes:
Eligibility	Logical	Program 1.3.2	

No_of_rentals	Number	Program 1.3.2	

Pseudocode _____

(Loyalty_incentive_eligibility)

eligible = false

If (No_of_rentals > = 4)

      eligible = true

End If

Return

Other _____

Business rule: Returns the eligibility of customer for joining loyalty incentive scheme, eligible if customer has made 4 paid rentals within a year.

11. Data Storage Design

Data	Type	Use	Suggested Format
Customer Information	Simple (Customer Information)	Transaction	Relational
Car Information	Simple (All the information about cars like Model, Last service,Purchase Date)	Transaction	Relational
Rental Information	Simple (Information about the car and the customer mapped to it.)	Transaction	Relational
Lease Information	Simple(System will contain the information about the car lease and its expiration)	Transaction	Relational
Promotio Information	Simple and Complex(will contain shot videos and advertising about new offers.)	Transaction	
Transaction Logs	Complex(Will contain the information about the money transaction between customer and EU)	Transaction	Log Files
Global Positioning System logs	Complex(System will track the car at all time and will store the information in log files)	Transaction	Log Files
Wifi Connectivity Logs	Simple(System will keep the record of internet connectivity of car)	Transaction	Log Files
Temporary	Simple and	Transaction	Transaction Files

Information	complex(In case of rollback of transaction , to cancel order or to put an order on hold) we will keep the data temporarily.		

Figure 32

Special point considered to increase the data access speed:-
1) It gives you access to the capabilities of MySQL, Oracle, SQL Server, PostgreSQL, or Amazon Aurora. Aurora is a MySQL-compatible database designed for scaling database applications without specialized tuning skills being needed
2) Developers familiar with data modeling do not have to be proficient in administrative tasks, such as managing tablespaces and distributing table and index data
3) There is no need to keep up with vendor security and performance patches. RDS also includes backup and recovery services
4) Developers can take advantage of existing commercial licenses. For example, if you already have an Oracle license for an on-premises database, you can use the "bring your own license" option, which will save on hourly charges
5) RDS also manages replication. If developers need high availability in a database, the multi-availability zone option can be used. With that option, a replica database is kept in sync with the primary database in case of failure
6) AWS offers high availability and a hot failover option in another geographic location, so if the master database instance goes down, it will automatically fail to the secondary location.
7) Simple API calls or manual changes with the AWS Management Console are all that is needed to scale a database service up or down as the load dictates. Storage scales up to 3TB and IOPS to up to 30,000 per instance.
8) Database instances can be managed from the AWS console. When you run your own instances, you will manage them using database-specific tools.
9) Overall, I believe Amazon RDS reduces or eliminates many of the database administration tasks that developers face when they run their own database instances. The on demand high scalability is also very attractive. What's your input on RDS? Please share your thoughts in the comments.

12. Other Considerations

1) Testing : The testing of autonomous cars will be monitored and all the incidents and accidents will be recorded on papers and database to help EU autonomous car to enhance and survive the autonomous car market.
2) User Documentation : Down the line, Feedback will be recorded from every customer to scale the productivity and customer satisfaction
3) Performance : EU information system will keep the record of the performance identifying parameters like (on time arrival, Late arrival, car problem , User experience , Feedback from customers and Mileage) to increase the performance of EU rental autonomous cars and to reach a flawless and reliable customer service.
4) Promotions: EU rental Operations will launch time to time promotions(free and discounted) to attract customer, build strong relationship with customer and to familiarise people with autonomous driving.
5) Customer Service : EU rental will incorporate 24 * 7 customer care support which will aim to provide instant resolutions to the problems.
6) Optimizing UI : EU corporation will make moves to get latest technology onboard to make autonomous driving more reliable and safe.
7) Legal : We are proposing to start EU rental's new feature - self-driving rental service in 5 states which are following Michigan,Florida,California,Washington D.C and Tennessee

Figure 33