Project for CSE471- Car Showroom Website

Title of the Project: Car Showroom Website

Group Number: 07

Group Members: 03

Student ID	Student Name
18101140	SOWAT HOSSAIN RAFI
18101142	SHOVON MANDAL
18101047	AFIF BIN JINNAT

Table of Contents

Table of Contents 2

Introduction 3

Motivation 3

System Description 4

Requirement analysis 5

- Functional Requirements: 5
- Non-Functional Requirements: 6
- Other Non-Functional Requirement: 7

Design diagram 8

- Use case Diagram 8
- Sequence Diagram 9
  - Sequence diagram for Update car and Login to System: 10
  - Sequence diagram for Pre-book car: 10
  - Sequence diagram for Purchased car: 12
  - Sequence diagram for Review: 13
- Activity Diagram 13
  - Activity diagram for update car: 14
  - Activity diagram for pre-book car: 16
  - Activity diagram for purchased car: 17
- State Machine Diagram 18
  - State Machine Diagram for Admin: 18
  - State Machine Diagram for Customer: 19
- Data Flow Diagram 20
  - DFD Level 0: 20
  - DFD Level 1: 21

Conclusion 22

References 24

Introduction

In Bangladesh, 4.5K[1] cars are sold every year and the car selling rate of Bangladesh is increasing day by day. However, most of the car showroom has no website. If car showroom has website customer can choose a car by comparing with other cars, can see the review also get a complete idea about the car without going to the car showroom. For making the car business beneficial and easier for a customer in Bangladesh, our project "Car Showroom Website" will help a lot. It will also help to keep track of the current trends of a car in Bangladesh. That will help the importer to match the customers' expectations by analyzing the data from our website.

If any car showroom uses our website for their business, they can reach virtually with the customer with their whole car showroom. The customer gets detailed information, review, rating and so on. That will help the customer to decide which car he/she want. And the showroom owner can sell more car with a less employee.

Motivation

Nowadays car is like a basic need in any city. People are busy with their work and we want to make their time-efficient. For just choosing or analyzing car details many people want instant solution. With the blessed of technology, we want everything at our door. For making car deals easier we are motivated to make a website where people can get every information about a car, also pre-book and purchase a car without going to the showroom. It is like a showroom in one's hand. Just link **click** > **view** > **and purchase**.

But is a matter of sorrow that most of the car showroom has no website. They are providing their services via Facebook, YouTube, Blogs, TVC and Poster. Though they are providing information through 3rd party, they are unable to give details information about the car. So, the customer cannot get real-time information, as well as the customer, has no option to order a car from home at any time.

Also, this is a situation of a pandemic for COVID-19. The people are staying at home. Sometimes for lock down the car showroom are off. So, the business stack. That also impacts our total GDP. To increase the selling of car in off day or any pandemic situation we think our website will also help.

So, the current situation, the current condition of car selling, for increasing the sell as well as to make the car business easier both for the customer and showroom owner we decided to make our project as **“Car Showroom Website.”**

System Description

Project Sponsor:

Sowat Hossain Rafi, Shovon Mandal, Afif Bin Jinnat, Department of Sales and marketing.

Business Need:

This project has been initiated to increase sales by creating the capability of selling cars to the customers, monitoring the sales, also the user needs & opinions which will build a brand value with this website. Also, it will help to increase business by analyzing the car database. It will provide the customer better service rather than an offline showroom.

Business Requirements:

Using the website customers will be able to search for cars & purchase. The specific functionality that the system should have includes the following:

- Search for cars.
- Can get all information about a car.
- Can comment on the cars & share their opinions via multimedia option or with a text comment.
- Can check availability of any car.
- Detailed features of the machine.
- Can pre-book a car that is not available.
- Can place an order.
- Can do the payment.
- Option for registered and guest account.

Business Value:

We expect that this website will increase sales by enabling existing customers to purchase cars & reaching new customers who need detailed information for a car with the best customer support. We expect some increase in cross-selling, as a customer who can share the informative website links with others & introduces a new rapport with new customers by watching the informative videos, opinions (comments) & an option to serve the customers in an emergency. By making a good rapport we can catch the market and attract the customers.

[2]Conservative estimates of tangible & intangible value to the showroom include the following:

- ৳ 7,00,000 in sales from individual orders
- ৳ 5,00,000 in sales for the new emergency service mode.
- Around ৳ 50,000 for ads on the website
- ৳ 10,00,000 for making a new rapport.

Special Issues or Constraints:

- The Marketing Department views this as a strategic system. The online car showroom system will add value to our current business model, and it also will serve as a proof-of-concept for future car dealers and businessman. For example, in the future, car sales may be only will be online and sell products directly over the internet.
- Many of our customers have been requesting this capability, and we need to provide the best service to them.

Requirement analysis

Functional Requirements:

1 **Authentication**:

1.1 Can register with required documents.
1.2 Can log in with OTP.
1.3 Can Log in with user id and password.
1.4 Log in as guest.

2 Car management:

2.1 Admin can update the car to the system database.
2.2 Every visitor can view cars.
2.3 Every visitor can search the car.
2.4 Every visitor can get all details of the car.

3 **Sales System:**

3.1 Add to cart.

3.2 Pre-book car

3.3 Place order

3.4 The system will update the database.

4 **Payment Management:**

4.1 Pay via the website through online API.

4.2 System accepts payment and give it to accounts for verification.

4.3 The system will make an invoice.

4.4 If payment failed the System will cancel the order.

Non-Functional Requirements:

1 **Authentication**:

1.1 Recover account option.

1.2 After 5 wrong attempts account will be blocked for 10 minutes.

1.3 Every OTP validation will be 5 minutes.

2 **Car management:**

2.1 Can search by model, price, colour etc. filter.

2.2 High-quality 3D images and videos for every car will be uploaded.

2.3 Rating will be visible at the top of every car.

3 **Sales System:**

3.1 Customer info will keep secure.

3.2 Pre-book info will be sent immediately to the customer.

3.3 After accounts approve the order will be confirmed.

4 **Payment Management:**

4.1 Multiple Payment options.

4.2 Only accounts can access the payment information.

4.3 For Any duplicate payment system will refund the amount to the customer account.

Other Non-Functional Requirement:

Operational:

1.1 Admin can verify the user and client.

1.2 There will a responsive design for all the devices. (Mobile, Tablet, PC)

1.3 Computable with all browsers.

1.4 The User Interface will be simple and user friendly.

Performance:

1.1 24/7 online support.

1.2 Response time will be less than 4 seconds.

1.3 Synchronized will database in every 7 minutes.

1.4 Can support 1000 simultaneous users.

Security:

1.1 The user’s data should be encrypted.

1.2 Every user login must be unique.

1.3 The system servers should e kept secure from any kind of tampering.

1.4 One user account can be login into 1 device at a time.

1.5 2Fa verification system.

1.6 User can log in with only OTP.

Cultural and Political:

1.1 Company policy says all the IT equipment was purchased from Techland.

1.2 The Car Showroom website will ensure the “Digital Security Act 2018[3]” to ensure the security of the consumers.

1.3 The system will conform to all Data Protection Laws.

Design diagram

Use case Diagram

From the use case diagram, we would be able to know about the actions, actors, entities, relations of the Car Showroom website. In our project, we have 5 actors. They are Customer, Guest, Registered, Admin and Account. Customer is generalized with Guest and Registered. Here, Registered means the Registered Customer. The Customer, Guest, Registered are primary actor and Admin and Account are a secondary actor. There are several use cases related to each of the actors, known as relationship. Here, in ***Figure 1.1*** we have some use case that describes our work in car showroom website. Firstly, the customer registered him/herself in our system with the necessary information. If the customer is registered, he/she can continue with login. But if the actor is a guest, it will access some of the use cases. Guest can do

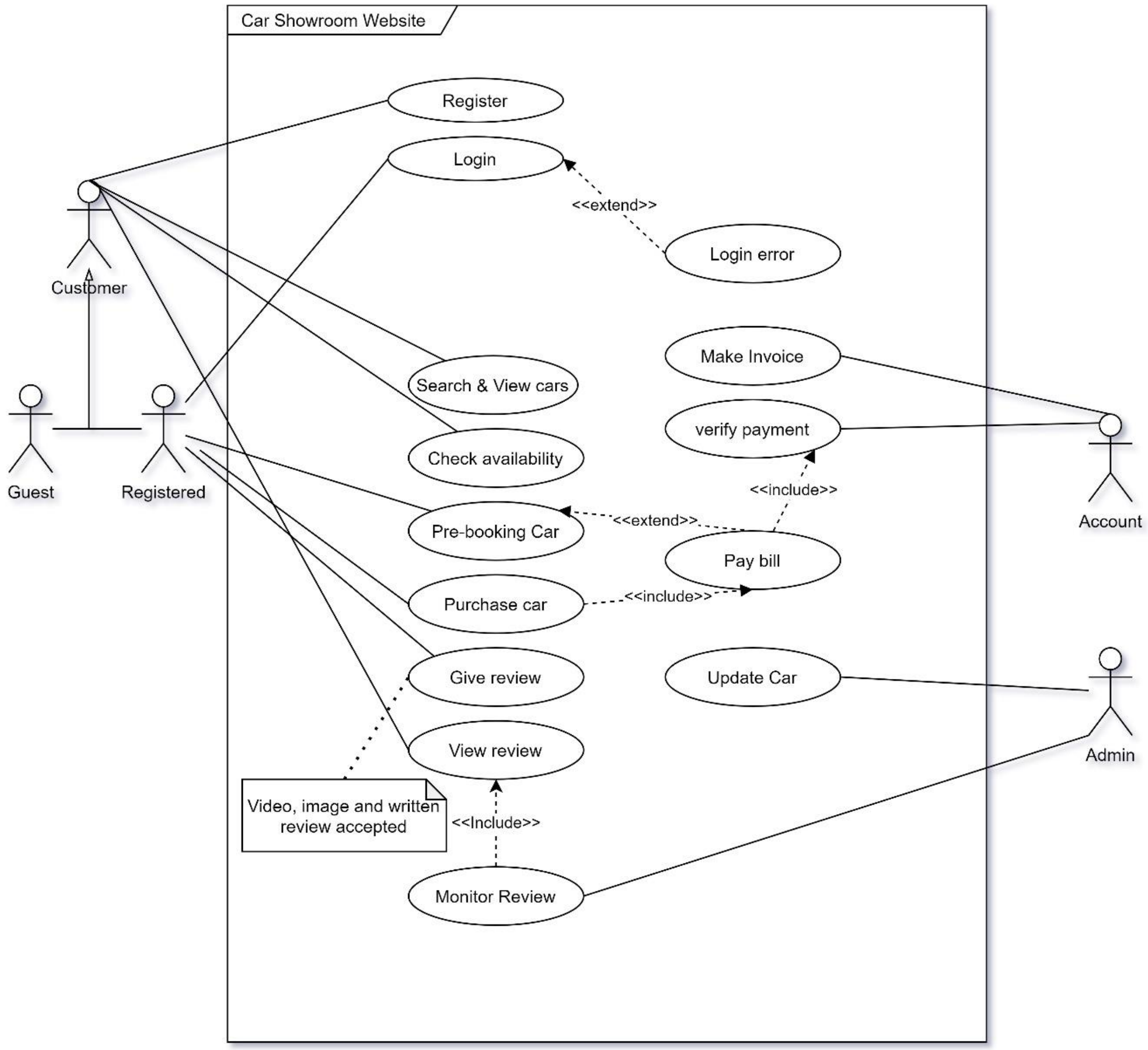

Figure: 1.1

everything that a customer actor does. But the registered customer gets some extra features. Customer actor can search and view the car in the system. Also, customer can check the availability of the car. A customer can act as a review reader of our system. It will also do by both Guest and Registered customers.

If the Customer is Registered, he/she can be an actor who can pre-book car. The registered customer actor can purchase a car. He/ she can give a review.

In ***Figure 1.1*** we can also see some secondary actor. Here the actor admin can update the car in the database. Here is a relation between admin with update car use case.

Here, also an actor name Account, who can verify the payment which is made by the customers. Account actor can also make the invoice.

In F***igure 1.1*** there are some extend and include relation. For example, here is an extended relationship with the login and login error. Because if the login is failed the system will give login failed information to the customer.

If a customer prebooks a car he need not pay the full amount. Or sometimes he/she can prebook without giving any payment to the system. So here is an extended relationship between the pay bill and pre-book.

But for the purchased car the customer must pay. For that reason, here is an include relationship between pay bill and purchased car. Payment verification is a must for any pay bill. So here is an include relationship. Same way, an admin must monitor the review. As a result, we put and include the relation between monitor review and view review.

Sequence Diagram

A sequence diagram of a system illustrates the participation of the actors in the use case. This diagram shows the sequence of messages that pass between objects for a particular use case over time. In our system, we have divided our sequence diagram into four-part for reducing the complexity. They are –

1. Sequence diagram for Update car and Login to System.
2. Sequence diagram for Pre-book car.
3. Sequence diagram for Purchased car.
4. Sequence diagram for Review.

Sequence diagram for Update car and Login to System:

From ***Figure 2.1*** we can see here is 2 actors and 3 active objects. Admin and customers are the actors. There is a login system, system and database as objects. Firstly, the admin updates the car info into a database. Admin cannot access the database directly. It makes a call to the system through `updateCar()`. Then the system will update the database according to the admin's given info. After successfully updating the database give info about *update successful*.

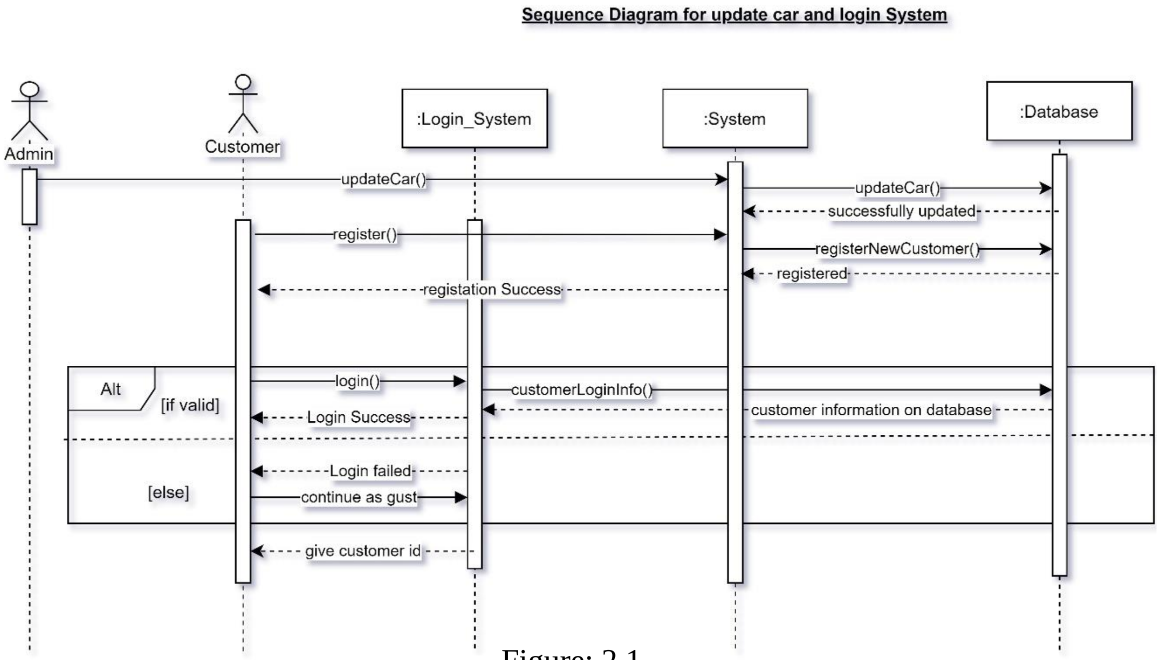

Figure: 2.1

Then in ***Figure 2.1***, we can see a customer request to the system for `register()` then the system request to the database with the customer information through `registerNewCustomer()` , then the database adds the customer info in the database and give info about *registered.* The system shows a message that *registration success.*

Then in ***Figure 2.1*** the login part comes. Customer asks login system for `login()` . The login system checks the `customerLoginInfo()` with the database. If the login successful, the login system shows *login success* otherwise it shows *login failed.* After that, the customer can **continue as guest.** Then the login system gives a *customer id* to the customer.

Sequence diagram for Pre-book car:

From ***Figure 2.2*** we can see that here is the sequence for pre-book a car is displayed. Here are 2 actors (Customer and Accounts) and 2 objects (System and Database) that are active. The customer timeline

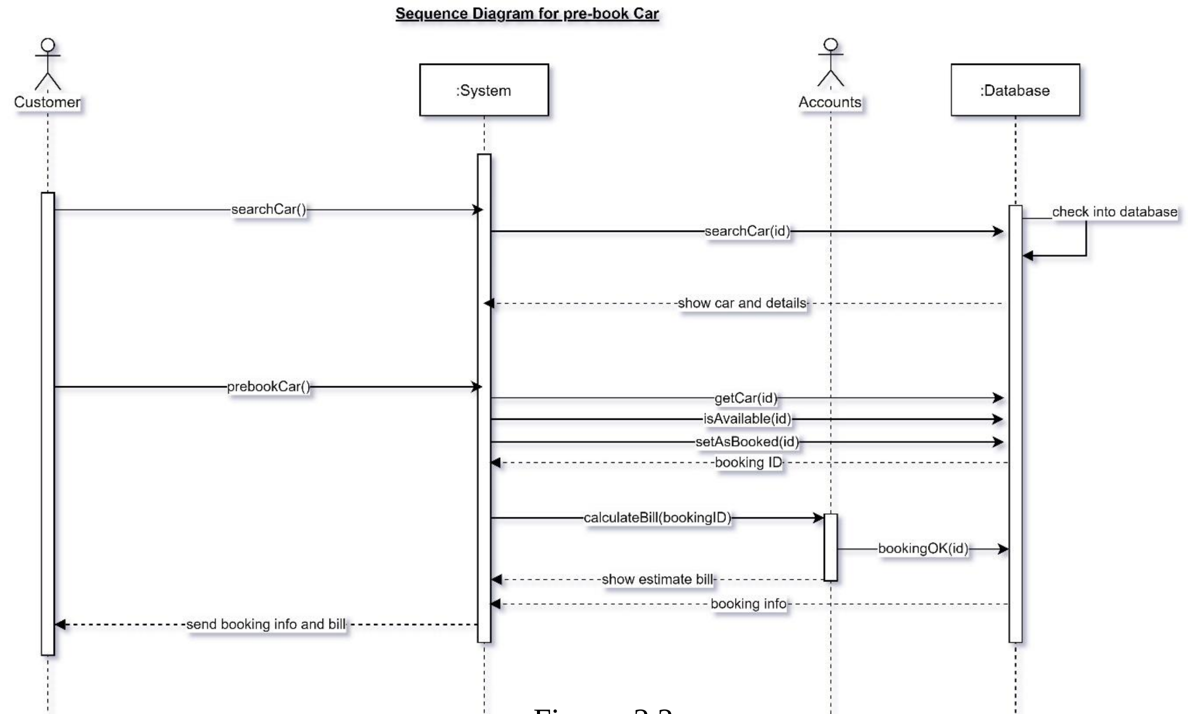

Figure: 2.2

is active during whole the process. The customer first asks the system for searching for a car. The system then calls the database with the car id for searching. In the database, there is a self-loop where the database will check the car according to the id which is given by the system through `searchCar(id)` . If the car is available, the database gives the car details. Then the customer can pre-book a car. Here `prebookCar()` method is called by the customer. But the customer cannot directly call. The `prebookCar()` method is set in a button of the system and customer just click on it. That is why we are showing the method instead of a button.

Then the system request for car and check availability also set the car as booked in a database through the methods `getCar(id)`, `isAvaiable(id)` and `setAsBooked(id)` . Then the system will get the bookingID.

Then system requests the accounts for the bill through `calculateBill(bookingID)` method. After that, the accounts timeline open and it shows the estimated bill to the system at the same time the account set the car as booked in the database with its payment info. And give the booking id to the system. Finally, the system will send the booking info and bill info to the customer.

Sequence diagram for Purchased car:

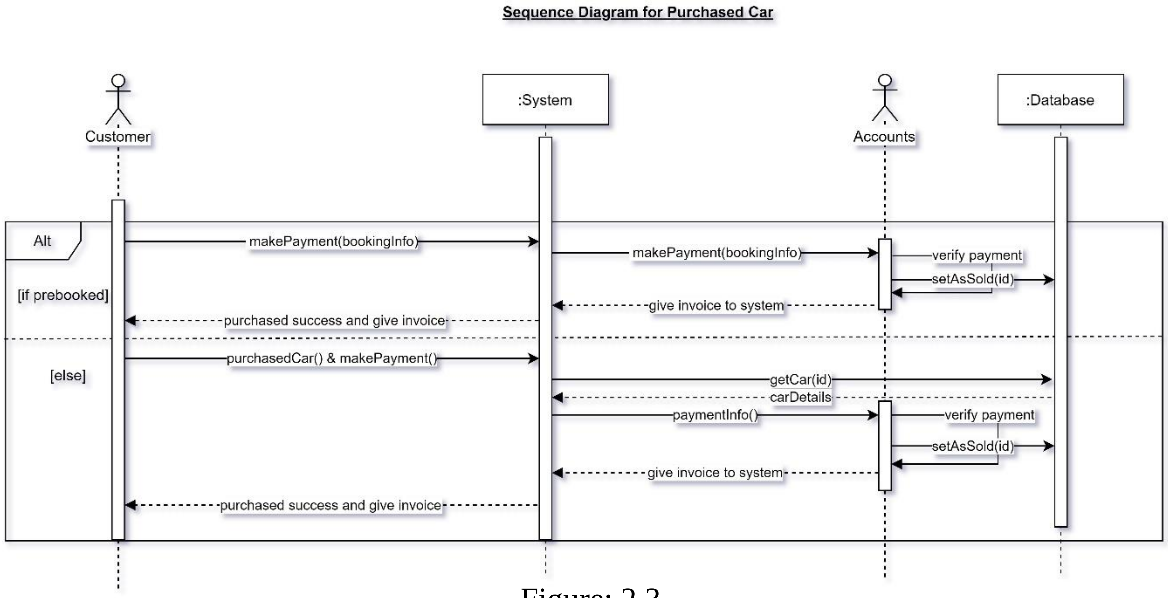

Figure: 2.3

From **Figure 2.3** we can see that here is the sequence for purchasing a car is displayed. Here are 2 actors (Customer and Accounts) and 2 objects (System and Database) that are active. The customer timeline is on in the whole process and the system is also active. But the accounts are active only when the payment has occurred.

In **Figure 2.3** the customer can purchase a car from his/her pre-book or can directly. If the car is prebooked the customer must have booking info. The customer requests for payment with booking info in the system and the system then call the accounts with the booking and payment info. The accounts verify the payment and then set the car sold in a database through `setAsSold(id)` method. After that, the accounts send the invoice to the system. Finally, the system gives an invoice to the customer and shows purchased success. In that sequence, the customer can buy a car from pre-booked.

If the customer does not pre-book any car, he/she can directly purchase by giving the info of payment and purchase (which car he/she want to buy). Then the system gets the car info from the database and the system will send the payment info to the accounts. Accounts verify the payment and set the car as sold in database through `setAsSold(id)` method. After that, the account sends the invoice to the system and as previous, the system sends the info to the customer.

Sequence diagram for Review:

Finally, the customer can give a review. In the review section the 2 actors Admin and customer are active also the object's System and Database is active. In **Figure 2.4** we can see that the customer wants to give a review and give the review to the system then the system gives it to the admin. Then the admin active and verify the review. If verification is passed the review is stored in the database through the system. Methods are called by the system. Where customer line is showing method, it is a button in a system and the button's backend the methods are written. The sequence is the customer does `giveReview()` to the system. The system sends to the admin via `verifyReview()` admin verify it then the system sends it to the database via `addreview()` then the review stored. Customer can view review through the system. System call `getReview(id)` to the database and the database gives it to the system and the system shows it to the customer.

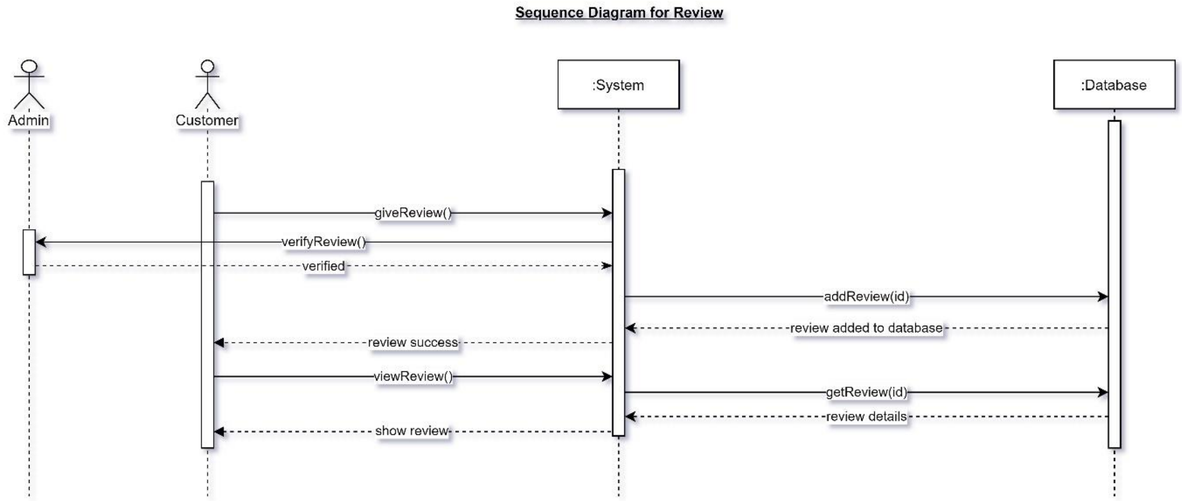

Figure: 2.4

In this sequence "**Our Car Showroom Website"** system is working.

Activity Diagram

The work of an activity diagram is to show sequential and parallel activities in a process. As an activity diagram is one of the behavior diagrams, therefore activity modeling is the sequence and conditions for coordinating lower-level behaviors. These models are commonly known as the control flow and object flow models. To construct an activity diagram, the components we require are actions, nodes, and decisions. The diagram starts with a start node and ends with an end node. The diagram can have multiple start and end nodes. There are more nodes used to construct the diagram. Some of the nodes are merge nodes, fork node, join node etc.

We have divided our activity into 3 parts. They are:

1. Activity diagram for update car.
2. Activity diagram for pre-book car.
3. Activity diagram for a purchased car.

Activity diagram for update car:

In Figure 3.1 we are designing an activity diagram for updating car into database. Based on

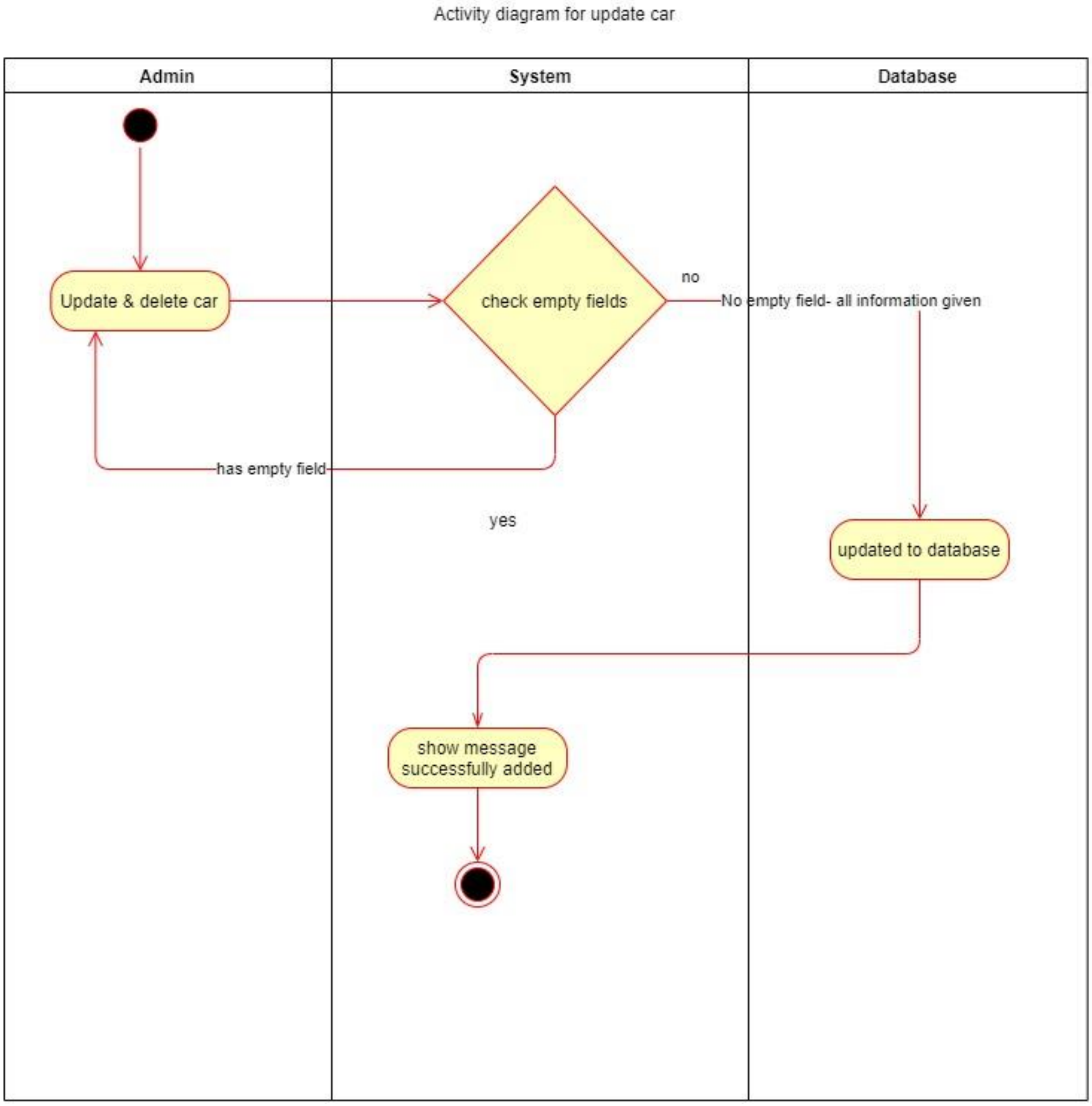
Activity diagram for update car
Admin
System
Database
Update & delete car
check empty fields
no
No empty field- all information given
has empty field
yes
updated to database
show message successfully added

Figure: 3.1

this scenario we have made 3 swim lanes one is for the admin one is for the system and the last one is for the database. The system is started from the admin and ends at the system. First, the admin adds a car or update the car to start the process. Admin will provide all information about the car. This will be checked in the system in a decision node. If the admin gives all the necessary info and there is no empty field it will go to the database and update the database node will update about the car. Then the database will be updated and the system will show the message that the car is successfully added and the process will end. But if there is an empty field the decision node will go back to the admin from the system and ask for the full info. In this way, the activity of updating the car is working.

Activity diagram for pre-book car:

For pre booking car in **Figure 3.2** we have made another activity diagram. Here we have 4 swim lane. They are for Customer, System, Accounts and Database. The activity will start with the customer. The customer will log in to the system then the customer asks to pre-book a car to the system. Then the system will check the availability through a diamond shape decision node. If the car is not available, the system will end the activity. If the car is available, it will go to the database for car details then the accounts make the estimated bill and booking info then the accounts send it to the system. The system shows the details to the customer after that customer get the booking id from the system and the activity will end.

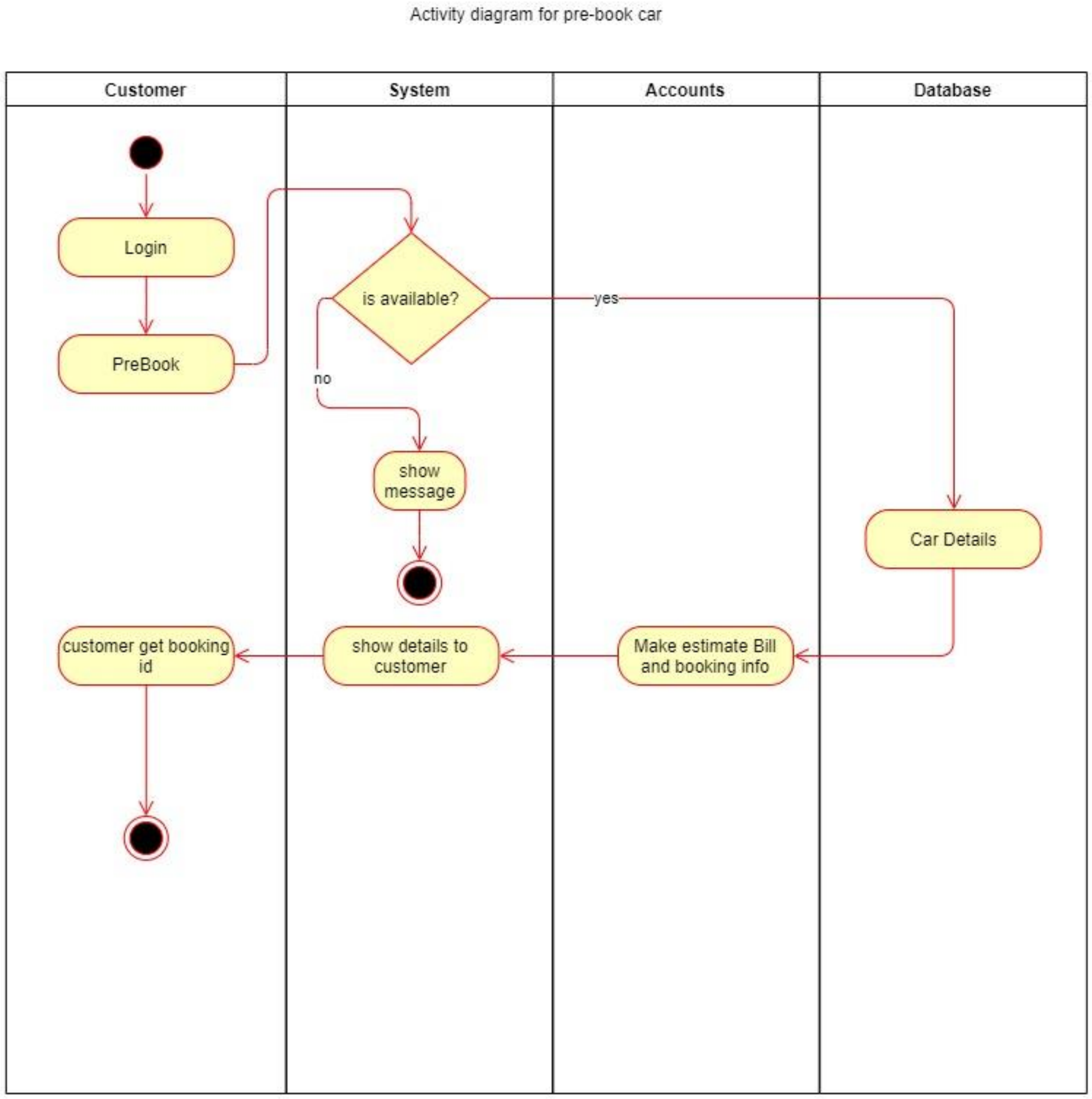

Figure: 3.2

Activity diagram for purchased car:

After all activity, the main activity comes. This this the main activity of the car showroom website. In **Figure 3.3** we can see 4 swim lanes. They are for Customer, System, Accounts and Database. The activity will start when the customer login. After login customer can purchase a car. The purchase car request sends to the system. Then the system checks if the car pre-booked or not. If pre-booked it gets the booking info from the database and matched. Then it goes to the system for order confirmation. For confirming order system ask for payment then the system go to the accounts for payment verification. If the payment is verified the then two works are done parallelly one is in the database which updates the database another one is in the accounts that make the invoice. Then accounts send an invoice to the system and the system

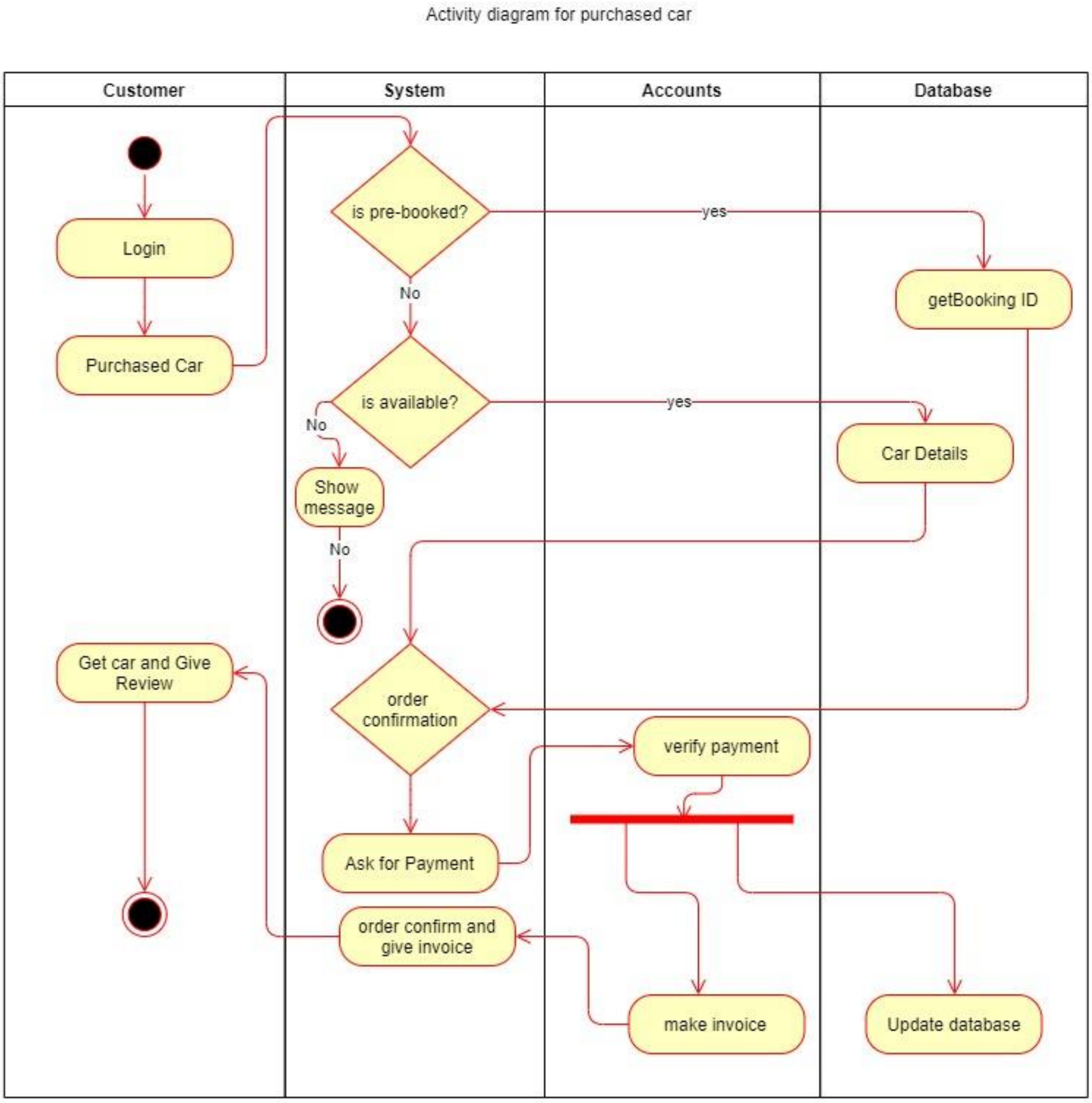

Figure: 3.3

confirmed the order and give an invoice to the customer. Finally, the customer will get the car and give a review then terminate the activity.

If a car is not pre-booked it will check the availability. If not available it shows a message that not available and ends the activity in a system. But if the car is available system get the car information from the database, then confirm the order after that it does the same thing that we have mentioned already.

In this way, the activity of purchasing a car is working.

State Machine Diagram

In-state machine diagram we see the working process of our project. The transition will occur from one state to another with an event. We have designed 2 diagrams for a state machine. One is for admin another one is for a customer.

State Machine Diagram for Admin:

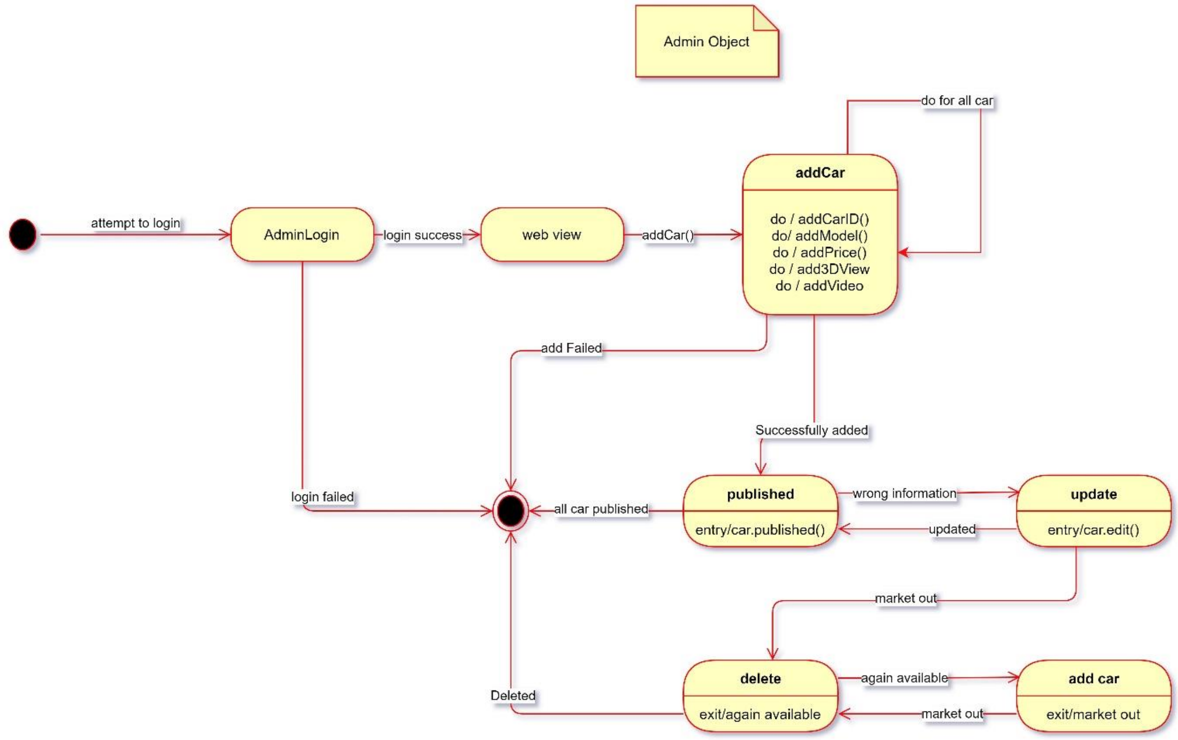

Figure: 4.1

In **Figure 4.1** we see the process start with the admin login state. If login failed the system will end. With the login success event it will goes to the web view. Then the admin can add a car. In the add car state, there is a self-loop that will do for every car. If added failed the system will be terminated. If successful it will go to the published stage. After published if there will be some wrong info it will go to the update stage. After updated it will again go to the published state. If the market out, then it goes to the deleted stage. If again available into a market it will go to the add car stage. After added if again market out it will go to delete. After deleting the system will terminate. If everything is ok after publishing the system will also end at that state.

State Machine Diagram for Customer:

For the customer in **Figure 4.2**, the customer will log in first to start. If login failed it will go to the end state. If login success, then goes to the web view state. From web view with search car event, the state goes to the display car. In-display car to show availability will go to the availability check state. If not available it will go to the not available state. With in-stock, the system will go to the available state. the system will go to the not available again if booked. Then the customer can add to the wish list and go to add to wish list state then the system will end after adding wish list successfully.

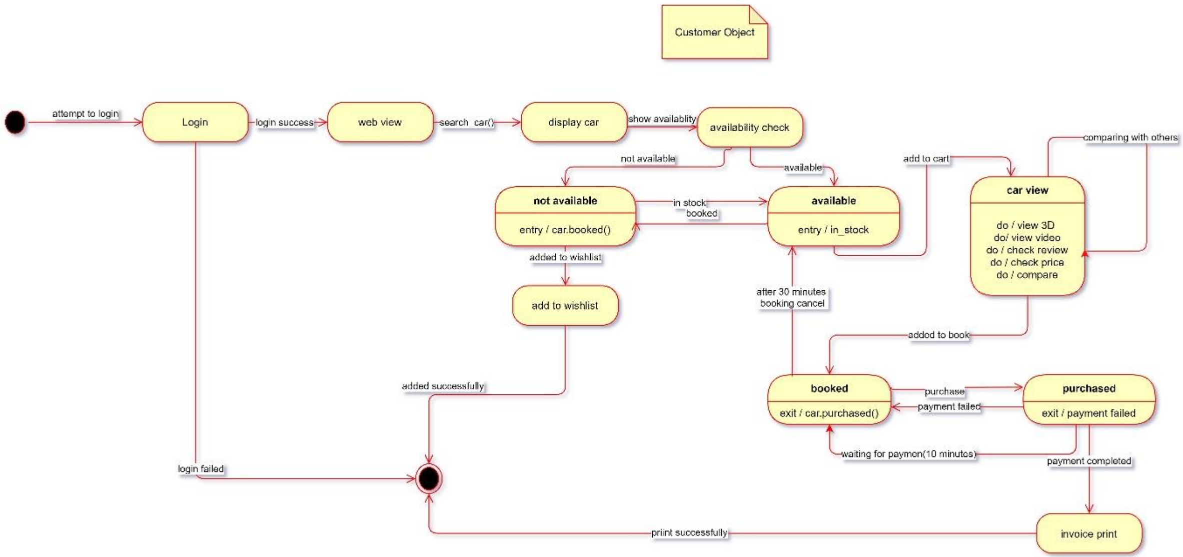

Figure: 4.2

In **Figure 4.2** if the car is available the system will go to the available state. Then by choosing to add to the cart the system will go to the car view state. Here is a self-loop for comparing with other cars. Then added to cart event take the system to the booked state. With the event purchase, it will go to the purchased state. If payment failed again it will go to the booked state. If more than 30 minutes in the booked state, the system will change bring the car to the available state. The purchased state will be waiting for 10 minutes for payment. If payment successful it goes to the print invoice state. After printing the system will go into an end state.

Data Flow Diagram

Data flow diagram means the graphical representation of the movement of data between external entities, processes, and data stores within a system. In short, a data flow diagram shows how data moves through an information system. The components of a data flow diagram showing how data moves through an information system. The components of a data flow diagram are external agents, data stores, process name, data flow name and control flow name. In our data flow diagram, all these components except the control flow name are being used. Besides, a data flow can split or merge, splitting is called diverging and merging is called converging. A data flow diagram can be in several segments: level-0 data flow diagram, level-1 data flow diagram and level-2 data flow diagram. The number of inputs and outputs between all the agents and processes/sub-processes must be balanced. But the number of processes can be extended from lower to higher level data flow diagrams as a process can be divided into several sub-processes in a higher-level data flow diagram. We have constructed a level-0 and level-1 data flow diagram for our **"Car Showroom Website"**.

DFD Level 0:

In our level-0 data flow diagram, there is only a single process covering all the external agents.

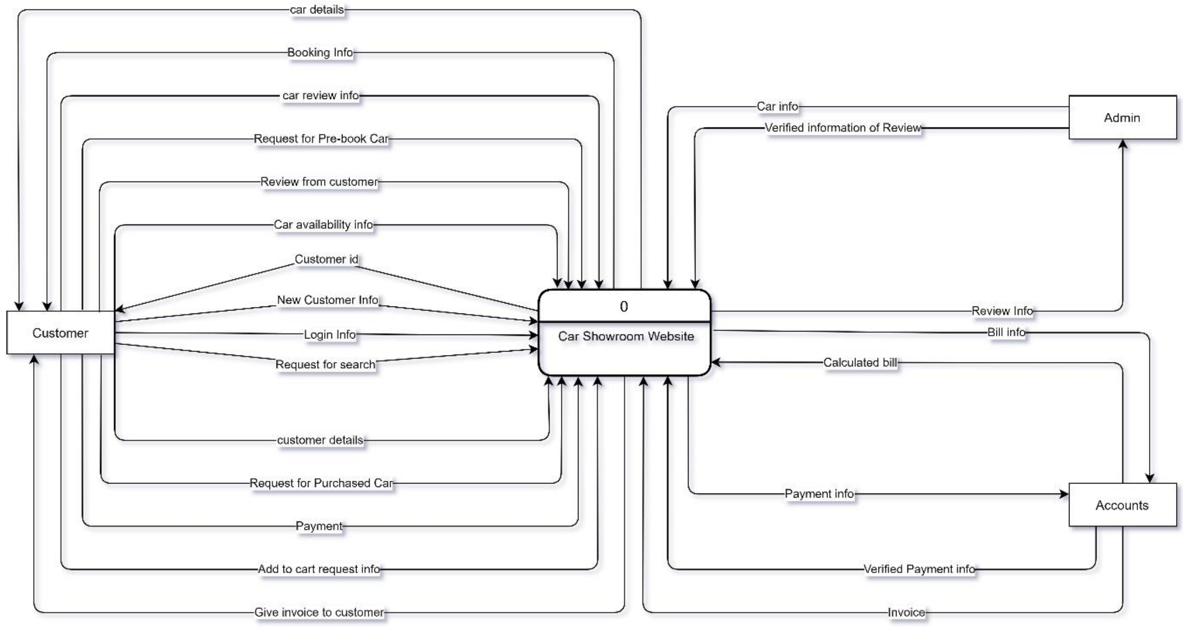

Figure: 5.1

In **Figure 5.1** we are designing a level-0 Data Flow Diagram. Here we have 3 external agents one is the customer one is admin and finally we have the Accounts.

A customer has 4 data flows incoming from the car showroom website and the customer gives 11 data flow outgoing to the car showroom website. Admin has only one data flow from the car showroom website and admin give 2 data flow to the car showroom website. Finally, the accounts take 2 data flows from the car showroom website and it gives 3 data flows to the car showroom website.

DFD Level 1:

In **Figure 5.2** we have the Level-1 diagram. Here the data flow with the external agents are same but here we introduce some database and divide the car showroom website Level 0 DFD into some sub-system. We divide level 0 into 6 sub-components. Firstly, we have the signup

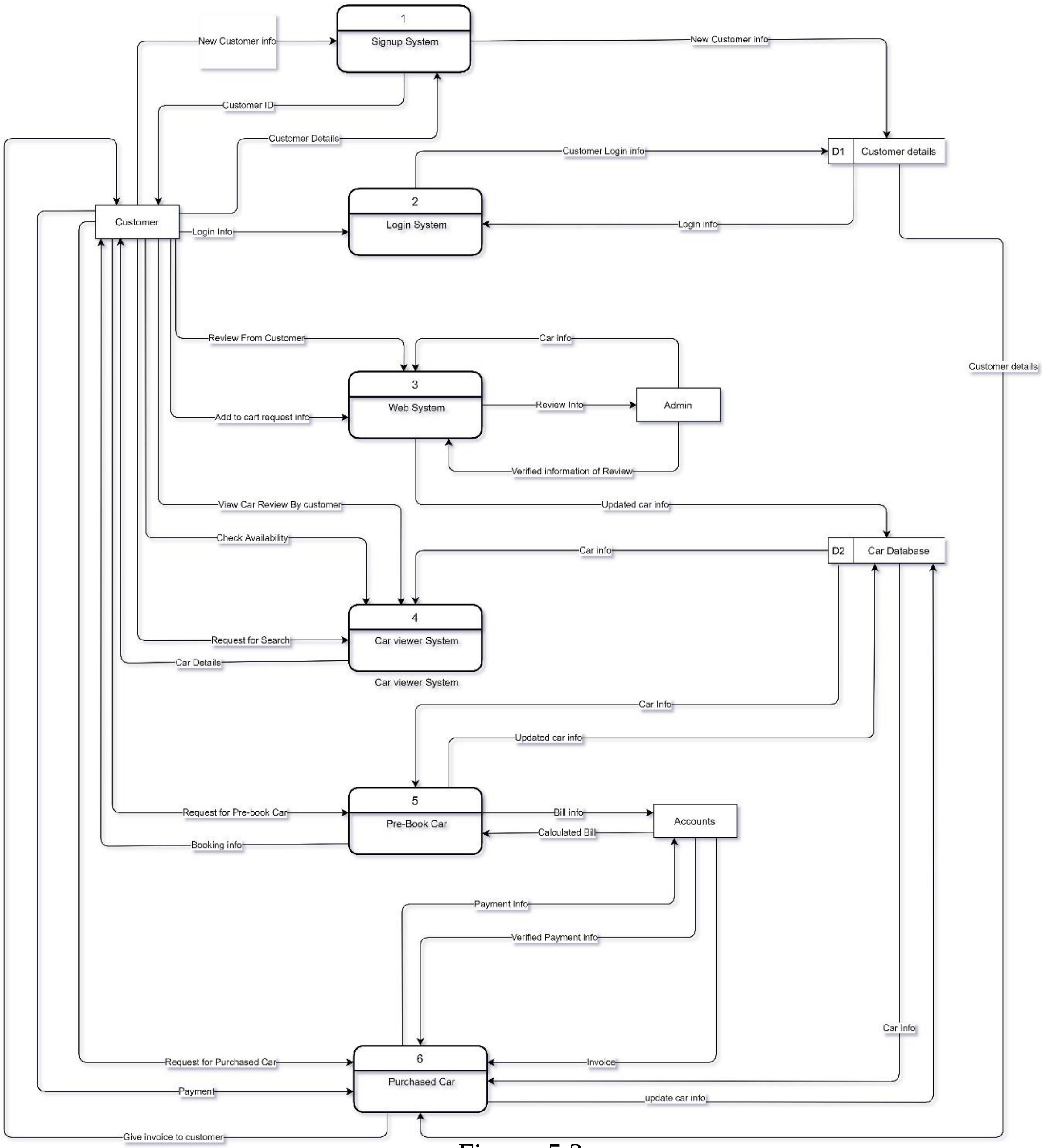

Figure: 5.2

system. Here the new customer data is given, and the signup system will store the data in the customer details database. Then the Signup system gives the data which contains the customer id to the customer. The previous customer can add more details through the signup system.

From **Figure 5.2** we can see there is another subsystem which is the login system. It will take login information from the customer and then with the database info it checks and permit to login. It has data flow with the customer database and customer.

After that, we have the web system it takes to add to cart request information data the review data from the customer and the admin web system take car information data also this system gives admin the review information for verification. The car info added by the admin is stored in the car database from this web system.

There is a subsystem named the car viewer system. This system is transferring data with the car database and customer. Customer search car with some data like car name, model etc. this data is flowing from customer to car viewer system then car viewer system retrieves car information from the car database and sends it to customer.

There is another subsystem which is pre-book car. It takes pre-book information from the customer like pre-payment, customer id etc. then the system retrieves the car information from the car database. Pre-book system then send the billing info to the account and accounts calculate the bill. And send it to prebook car system. The pre-book car subsystem sends the booking info to the customer. Finally, the last subsystem of the level 1 diagram of our car showroom website is the purchased car. It takes information from the customer about car purchased like pre-book or customer id etc. Then the system asks for payment. The purchased car subsystem gives the payment information to the accounts. Accounts verified the payment info and give an invoice to the system. Then the purchased car subsystem retrieves the car information from the car database and customer information from the customer database. After completing every step, the purchased car updates the database with the sold car info and give an invoice to the customer. In this way, on our car showroom website, the data is flowing.

Conclusion

To conclude we want to say that this website is very helpful to both customer and seller. In the database which car is selling most and which one is selling less is stored. So, we can get big data from the database of our website. By analyzing those data set we can get a complete assumption about the current demand for the car. It will be the on-place solution for everything about the car. Customer can get a car easily and seller can sell car anytime. So, the business will increase as a result the total GDP of Bangladesh will also increase. Just link **click > view > and purchased** will be the main goal of our project. We believe in less work more profit. Our car showroom website project can fulfil this motivation. By using this website showroom will be in one hand and people can view and purchase car whenever they want. Also, we hope this project will help to overcome the economical condition of this pandemic situation COVID-19.

References

[1] According to CEIC [2019] URL- https://www.ceicdata.com/en/indicator/bangladesh/motor-vehicles-sales
[2] Talking with some people who are working with e-commerce site.
[3] URL- https://www.cirt.gov.bd/wp-content/uploads/2020/02/Digital-Security-Act-2020.pdf