Table of Contents

Table of figures

Table of Tables

1. Introduction

1.1 Introduction to business organization

For the module "Software Engineering," this is the first coursework assignment. We must create a group of four or five people for the group coursework. For a company called "Allgemein," we are needed to create a vehicle rental system. Throughout the past ten years, Allgemein has dominated the entertainment market. Its revenue comes from work on entertainment-related projects, including those that include creating movies, documentaries, music, games, etc. Recently, nevertheless, this corporation has expressed interest in growing its operations in the transportation sector. To assist their business, they have even purchased a significant number of vehicles such cargo trucks, bulldozers, and cabs.

We must first analyse the needs of the organization to create a system that will work effectively for it. For this, we are required to make Data flow diagrams, structure chart diagrams, entity relationship diagrams, data dictionaries, process specifications, and model specifications.

1.2 Aims and Objectives

The primary goal of this coursework is to use the creation of a vehicle rental system as a means of demonstrating the information we acquired from the module "Software Engineering”. The objectives to achieve the aim of this coursework are mentioned below:

i. To create a project charter for the online vehicle rental system.

ii. To create SRS document

iii. To create DFD diagram, ER diagram, structure chart using the necessary entities and attributes. iv. To create pseudocode for the system

1.3 Tools required

There are several diagrams in this coursework to support and carry out the overall development process. There were tools that helped immensely in creating this work. Some of the tools that were used are:

i. Draw.io

Draw.io is an exclusive tool for creating charts and diagrams. This program gives the option to design a custom layout or use the automated layout feature (Computer Hope, 2022). This tool was used to create Entity Relationship Diagrams, Data Flow diagrams and Structure Charts. This tool had a user-friendly interface which made the working process easy and efficient.

ii. Microsoft word

Microsoft Word is a word processing application that enables the generation of both straightforward and intricate texts. We can have access to both the online version and the ability to download the program with Office 365. With the online version, we may instantly share and work on documents with other people (UA Little Rock, 2022).

2. Project Charter

A project charter is a formal, brief document that establishes the foundation of a project and gives project managers the legal authority to commence work. Before the project is thoroughly scoped out, a project's goals, objectives, and resource requirements are outlined in the project charter document (Gillis, 2022).

Project Charter	
Problem Statements	In context of Nepal, cost of fuel and vehicle is rising day by day. Import tax for vehicles is around 240%. Similarly, road tax is also high in Nepal. Construction Vehicles like bulldozers, cargo trucks are very expensive which might not be affordable for general people. Also, those construction vehicles are used only for construction purpose and not used on daily purpose.
Business Case	Considering the situation of Nepal, moving towards the transport industry can help in diversification of the sources of revenue and even help to get new customers register in the company.
Goal Statement	The goal of this company is to increase its revenue by 40% which sums up to an amount of Rs.50,00,000 by the first 7 months of the deployment of this project.

Timeline	For the completion of development of this system, we estimated a period of 5 months.
Scope	The scope of this project is to develop a transportation application that registers new customers, books and rents vehicles and provides training courses to customers with a user-friendly interface to attract more customers eventually leading to more income. This system, however, does not accept credit card or debit card for payment.
Team Members	Project Manager: Nayasha Shilpakar System Analyst: Bimal Devkota UI/UX: Shahil Thapa Research and development: Sakar Timalsina Backend Developer: Siddhant Thakur

Table 1: Project Charter

3. Software Requirement Specification

Introduction

Purpose:

The purpose of this online Transport Management System is to ease the hectic process of managing transportation and to create a convenient and easy-to-use application for passengers, trying to book a ride.

Scope:

The system is based on a relational database with its transport management and reservation function. We will have a database server always focused on all the of major cities of Nepal as well as thousands of drivers and vehicles ready for use. Above all, we hope to provide a comfortable user experience along with the best pricing available.

We wanted to improve the efficiency of the cab booking process, provide an enhanced customer service and at the same time making a profitable business model and expanding the market reach. Those are main scopes of this project.

Requirement Description

Functional Requirements:

The following list represents the functional requirements that would make the entirety of the software:

Take Membership

- The user should be able to register their personal Information to sign up for a membership.
- The user should be displayed different membership options to choose from such as Annual Subscription or Monthly Subscription and a payment method which the system will verify to complete the registration process.
- The system should store all user data in a database which the user should be able to access at any point in time.
- The system should inform the user via mail or notification in advance if the membership is about to activate.
- The user should be able to cancel his/her membership at any time with ease.

Book a Cab

- The user should be able to enter their pickup and drop-off locations and see the nearby drivers on the map.

- Once a driver confirms the pickup request, the user should get a notification and a call that a driver is approaching them at the requested pickup location.
- The system should calculate the price of the fare and display it to the user along with various payment methods like Cash-Payment and Online Payment.
- Once the ride is over and the user is dropped to the requested location, the user should get an option to rate the ride.
- There should also be an option to cancel the booking if needed.

Track the status of the driver

- The system should be able to track the status of the drivers like they're available for booking or not and display their location to the nearest pick-up request.
- The system must be able to display the driver's previous ratings and reviews to help users better choose the driver of their choice.
- The system must be able to always track the driver's location and provide them with real time traffic updates.

Register Staff and Vehicles

- The System should allow authorized staff members to register newly appointed staff and vehicle Information into the System.
- The system should verify the authentication of the provided details like contact information, vehicle information and insurance information.
- The system should store the staff and vehicle credentials in a database.
- The system should provide a way for staffs to end their contract or membership and remove the staff and vehicle information from the database.

Rent a Vehicle

- If the user Choses to rent a vehicle, the options to rent different vehicles like bulldozer and tractor should be displayed.
- The user should be able to provide the duration of rental.

- The calculated price based on the duration should be displayed to the user along with the terms and conditions for renting the vehicle.
- The user should be able to pay via Online methods or physical method and once the payment is confirmed, a confirmation email should be sent to the user along with the details of the rented vehicle.
- The user should get an option of extending or reducing the rented duration for different prices.

Report Generation

- The admin should be able to generate weekly reports regarding the business models, revenue generated, etc. from the system to help them with decision making or insight generation.
- The admin should be able to generate reports related to customer types like what age group of users make up most customers or what locations have heavy customer traffic, etc.
- The admin should be able to generate reports regarding the conditions of vehicles after the ride, maintenance, and scalability of vehicles.

Rating the Ride experience

- The Users should be able to rate their ride experience on the scale of 1 to 5 and provide feedback to the System regarding the ride.
- The ratings and reviews should then be stored and displayed on each Driver’s profile to help future customers with choosing the best ride for themselves.
- The system should be able to detect positive and negative comments from those feedbacks and generate useful information from them.
- The system should be able to filter those feedbacks according to the company’s policies.
- The users should be able to update or delete their ratings and feedbacks at any time.

Join a Training Course

- The user should be able to choose from a list of available training Courses.
- Various details regarding the courses should be displayed to the user like course description and schedules.
- After selecting a training Course, the duration of the training and other details should be displayed, and the user should be displayed various payment Methods.
- The Course should be activated once the user pays for the service.
- Once activated, the user should get access to the course materials such as videos, labs, assignments, physical meeting notifications and tests.
- The system should be able to track their progress and manipulate the course accordingly to everyone's ease.

Non-Functional Requirements:

Safety Requirements

The recovery method restores a previous copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed-up log, up to the time of failure if there is extensive damage to a wide portion of the database because of catastrophic failure, such as a disk crash.

Security Requirements

The system should be secure in all aspects like if the admins decide to opt to cloud and use its features, an additional copy of important documents and databases should be stored on-premises for a notch better safety and only encrypted data should be allowed to be accessed by the cloud.

SOFTWARE QUALITY ATTRIBUTES

- **AVAILABILITY:** The system should be available on the specified date and specified time to all customers regardless of the external anomalies like a festival, or strike or power shut down.

- **CORRECTNESS:** The system should start from correct start terminal and should reach the correct destination and guide the users along every step of the booking or renting process to make it a smooth experience.

- **MAINTAINABILITY:** The administrators and system in chargers should always keep everything in check and maintain a level of correctness in all its aspects.

- **USABILITY:** The system should be able to satisfy the maximum number of customer needs and provide a comfortable experience for users.

- **SCALABILITY and PORTABILITY:** The system should be scale itself up or down according to the changing needs and if the whole system or part of the system is to be shifted to cloud or any other online platform, it should be portable enough to do so without much hassle.
- **COST:** The cost should be designed in such a way so to not make the customer's feel over-charged and at the same time making profit for the system.

External Interface Requirements

User Interfaces

Front-end software: HTML, CSS, JS

Back-end software: Django, SQL Oracle (Database)

Hardware Interfaces:

We've considered that not all the customers can afford high-tech hardware devices. So, we made our application in a way that it can also be accessed by low-end devices. The Minimum hardware recommendation are:

- 1 GB of RAM

- 1.5 GHz of Processing Speed

The recommended hardware recommendations are:

- 2 GB of RAM
- 2 GHz of Processing Speed

Software Interfaces:

Following are the software used for the Transport management online application.

Software Used	Description
Operating System Software	For making this web application, we've chosen Windows operating system for its best support and user-friendly interface.
Database	To save the User's details, staff details and transaction records we've chosen SQL Oracle database.
Programming Language	To implement the whole project, we've chosen simple yet powerful tools like HTML, CSS, and JavaScript.

Table 2: Software Interfaces

Communication Interfaces:

This project supports almost all known types of web browsers like chrome, Firefox, UC browser, to Opera Mini. We are using simple electronic forms for the booking process, renting vehicle.

Design and Implementation Constraints

These are the requirements or constraints we implement on the development process. These constraints can be set by the organization for which we're making the software or from the developers' perspective. Some of those constraints are:

Time Constraint: The amount of time available for design and implementation can affect the scope and complexity of the system. So, the development process should be carried out with time constraint in mind.

Budget Constraint: The decided budget at the starting of the project shapes the overall development and making of the product. Cutting costs on the unrequired areas and not being able to make a time-consuming fancy product because of specified budget are some of the examples of Budget Constraint.

Regional Constraint: The region for which the project is being designed for example in our case, the transportation system is made by keeping in mind the terrains of Nepal which are mostly made of uneven elevations and slopes and narrower roads all pose a challenge for the drivers and the time in which the service can be delivered, so we estimate time for the service delivery based on those Regional Constraint.

User Constraints: The requirements and needs of the users can greatly influence the design strategy of the application like the app should be high tech and yet simple enough for users to understand it’s use cases and we’ve to program the discount offers and sales offers following the cultural events of most users. Ex- Discount for Diwali bookings.

4. Group Tasks

4.1 Environmental Model Specification

4.1.1 Data Flow Diagrams

Data flow diagrams are used to visually depict the movement of data in a business information system. DFD defines the procedures used in a system to move data from the input to file storage and report creation. There are two types of data flow diagrams: logical and physical. The logical data flow diagram illustrates how data moves through a system to carry out certain business functions. The logical data flow's actual implementation is shown in the physical data flow diagram (Visual Paradigm, n.d.).

4.1.1.1 Context Level Diagram

An internal software system’s interaction with external entities is depicted in a context diagram. It is generally employed to assist organizations in comprehending the extent of a system. They can then determine the best way to build a new system and its specifications or how to enhance an existing system. Context diagrams are high-level diagrams that don't depict the system's intricate workings in detail. Instead, they provide a concise, understandable, and transparent blueprint of the entire system. Arrows, for instance, are used to depict the data flow between the system and each external component. Everyone can comprehend how the system functions, regardless of their level of technical expertise or technological aversion (Miro, n.d.).

In this system, the external entities are customer, bank, driver, and admin.

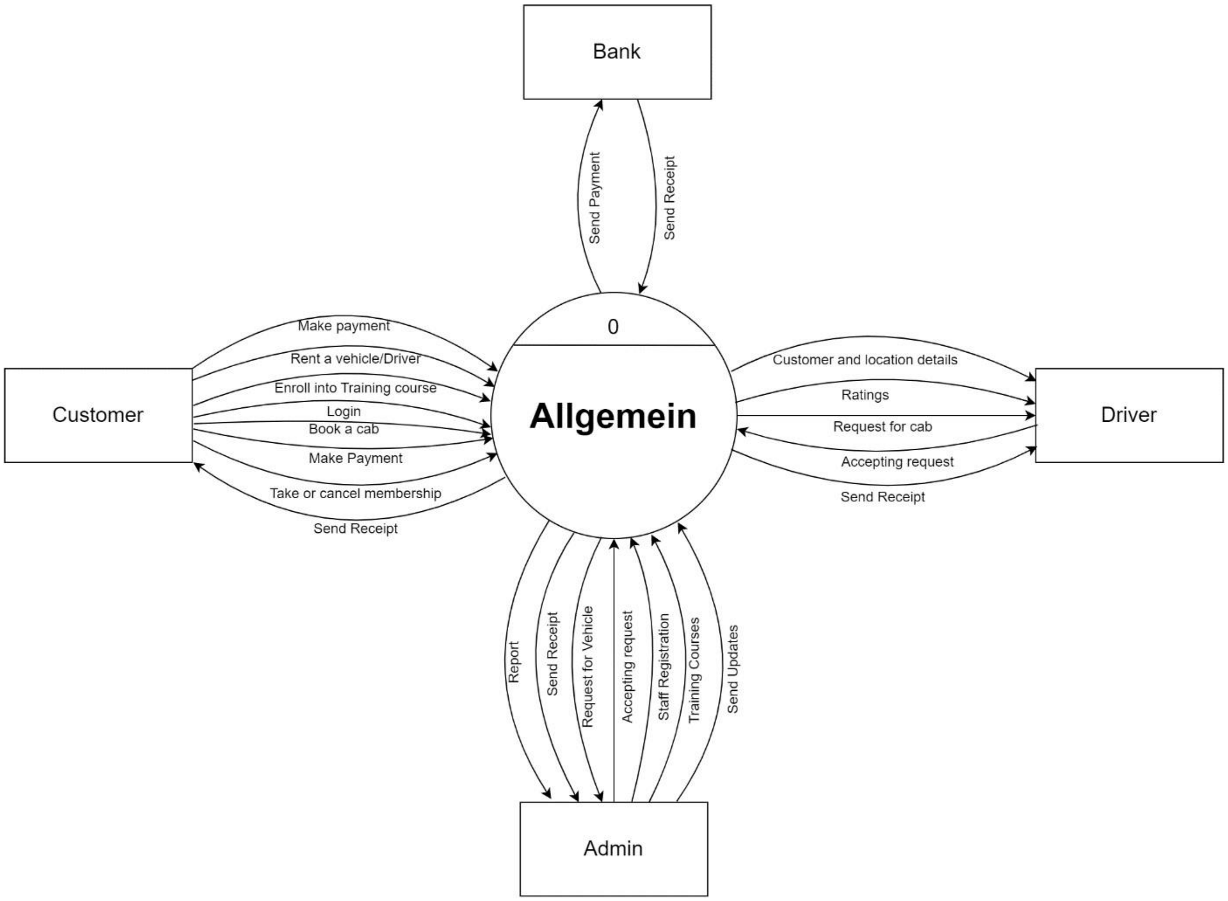


Figure 1: Context level diagram

4.1.1.2 Level 1 DFD

Level 1 DFDs remain to provide a basic overview, but they go into greater detail than a context diagram. The single process node from the context diagram is divided into subprocesses in level 1 DFD. As more processes are added, the diagram will require more data flows and data storage to connect them (Lucid Chart, n.d.).

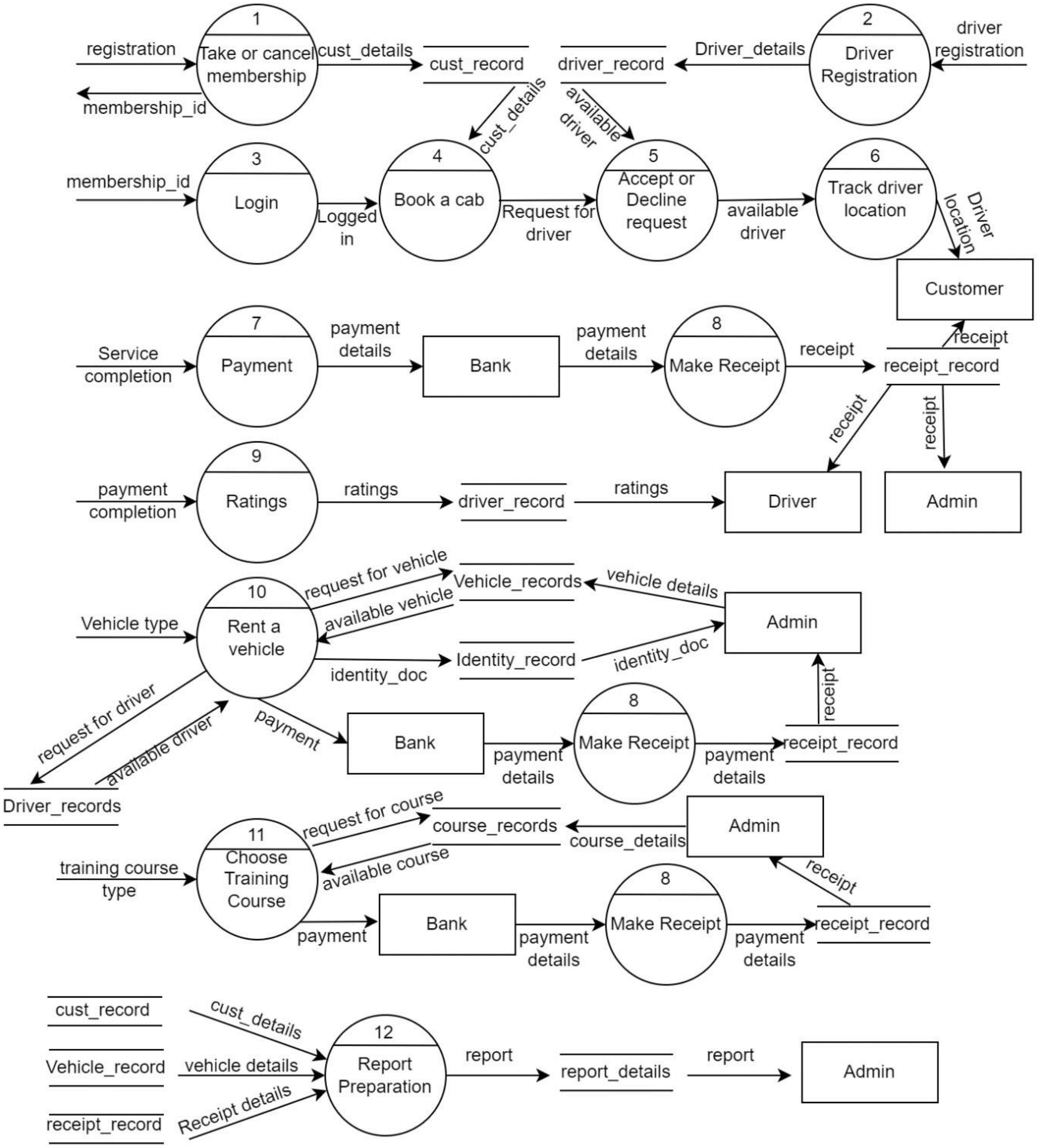


Figure 2: Level 1 DFD

4.1.1.3 Level 2 DFD

In level 2 DFD the processes from level 1 DFD are further divided into sub processes. It is used to plan the details about the system's functionality more specifically.

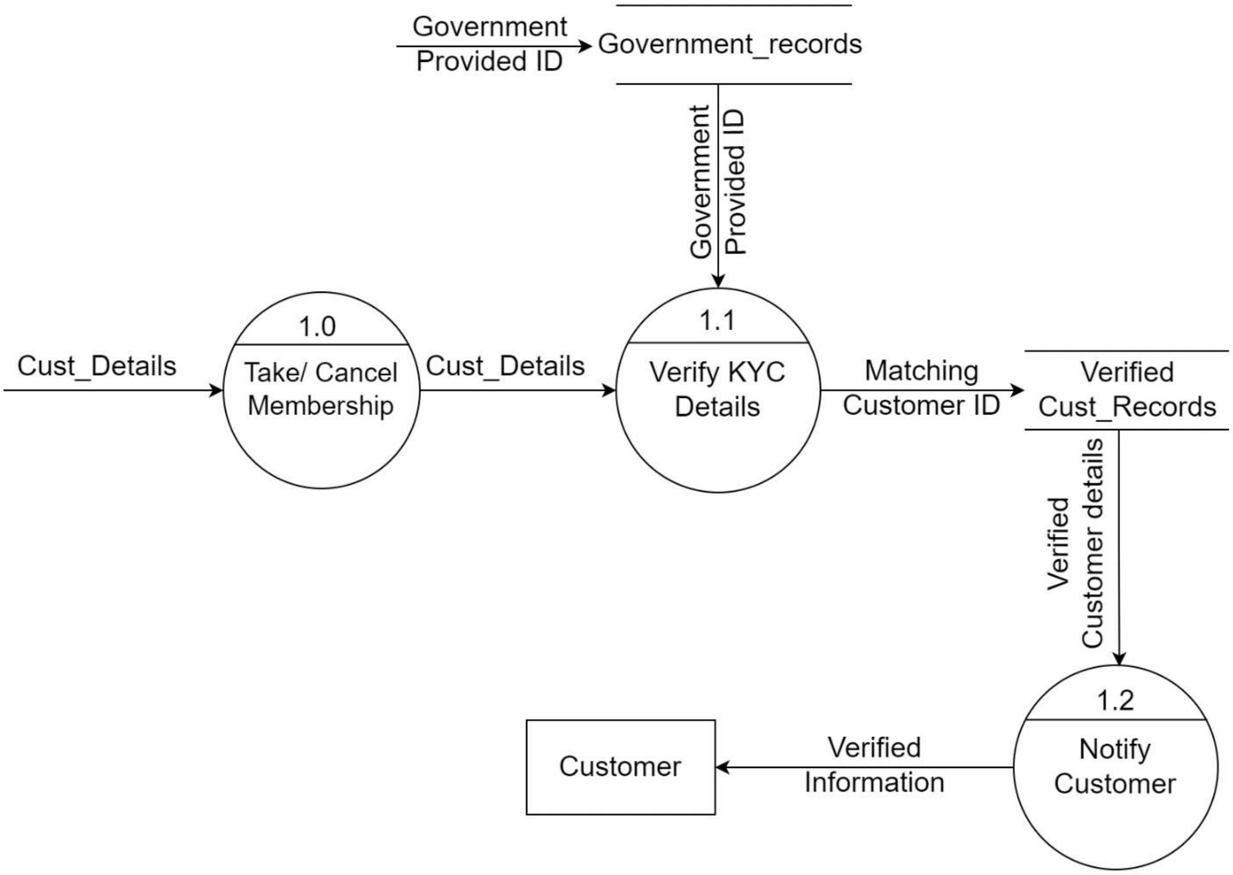


Figure 3: Level 2 DFD of Registration

Here, the customer provides their details for taking a membership which is verified with the help of government's record. Thus, government's database of people's information is used. After the verification process, the details that were matched with the government's record are sent to another database 'Verified Cust_Records'. The information from this database is used to notify the customer when their registration process is completed. To cancel the membership, the customer details are directly checked with the company's datastore 'Verified Cust_Records' and then unregisters that account followed by the process of sending notification to the customers.

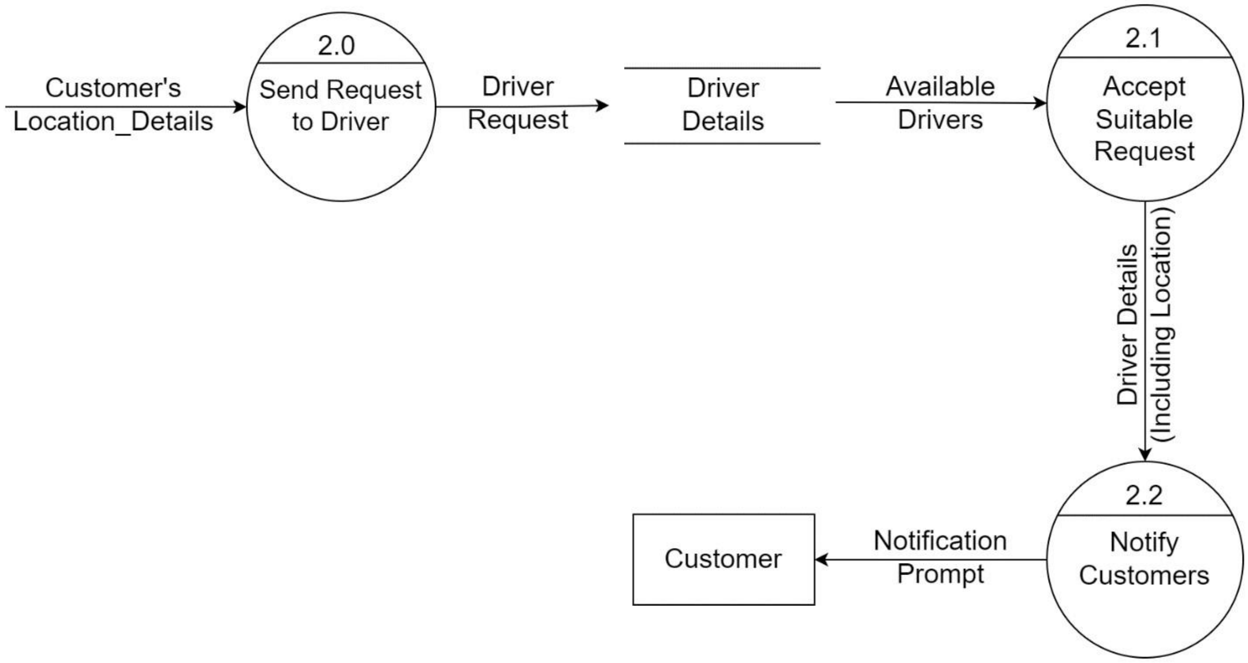


Figure 4: Level 2 DFD of Booking cab

These are the fragmented processes of the main process ‘Book a Cab’. At first, customer’s location and other details are provided to send request to the driver. There is a datastore ‘Driver Details’ that stores the information about the drivers registered in the company. The nearby available drivers accept the request and send notification to the customers along with their location.

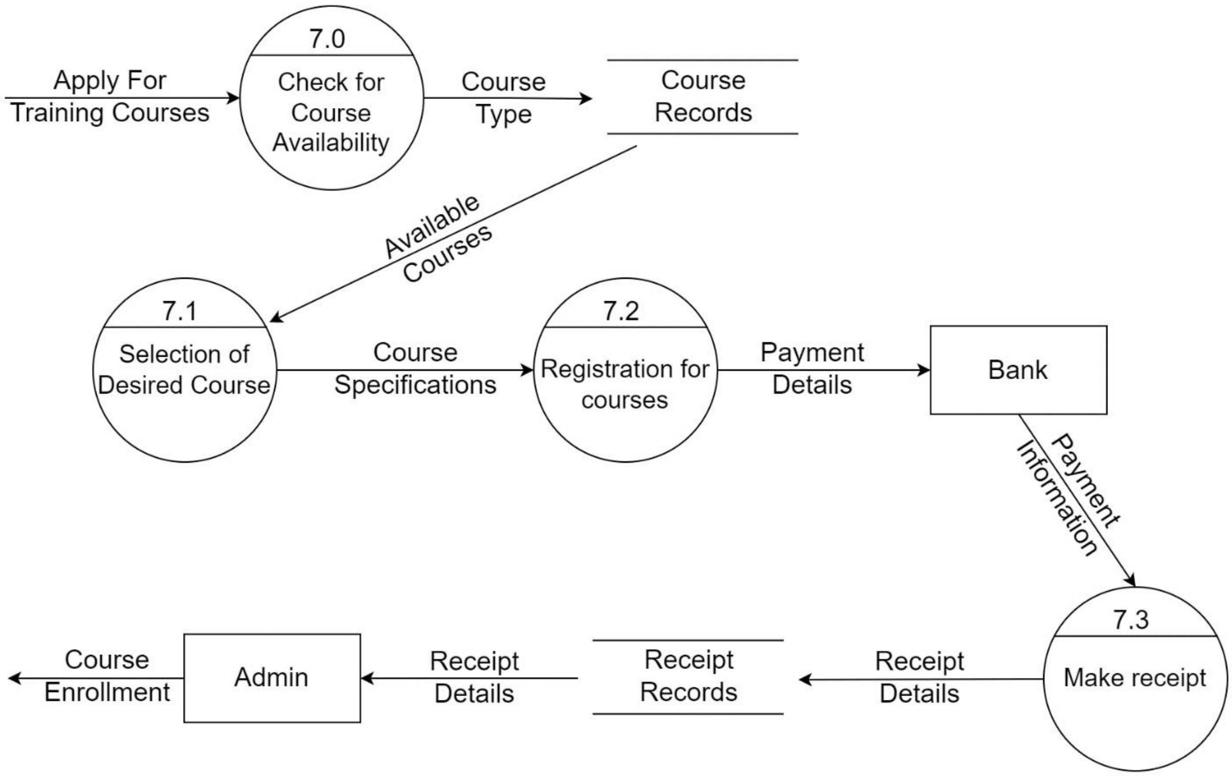


Figure 5: Level 2 DFD of Training course

This is the level 2 DFD for the process ‘Training Course’. This DFD starts with applying for the training course by the customer. Then the system checks if the particular course is available at that moment. There is a datastore ‘Course Records’ that stores all the information about the training courses. The chosen course is checked in this datastore if it is available and allows the customer to select it. After selecting the course, the registration process starts, and the customers are required to make the fee payment. The payment details are sent to bank and the bank makes receipt which is sent to admin. As soon as the admin gets the receipt, they enrol the customers to their respective courses.

4.2 Internal Model Specification for the system

4.2.1 Entity Relationship Diagram (ERD)

An Entity-relationship diagram (ER diagram) depicts the structure of a database using a diagram known as an Entity Relationship Diagram (ER Diagram). An Er diagram is a database architecture or blueprint that may subsequently be implemented as a database.

Entity, attribute, and relationship are the three primary components of the E-R model (Singh, 2012).

i. Entity

An entity is an object or thing in an actual world that can be distinguished from other real-world objects. An entity is represented as a rectangle in an ER diagram. ii. Attribute

Attribute is a piece of information that describes the property of an entity. An attribute is represented as oval in an ER diagram.

iii. Relationship

A relationship is represented by diamond shape in ER diagram, it shows the relationship among entities. There are four types of relationships:

1. One to One
2. One to Many
3. Many to One
4. Many to Many

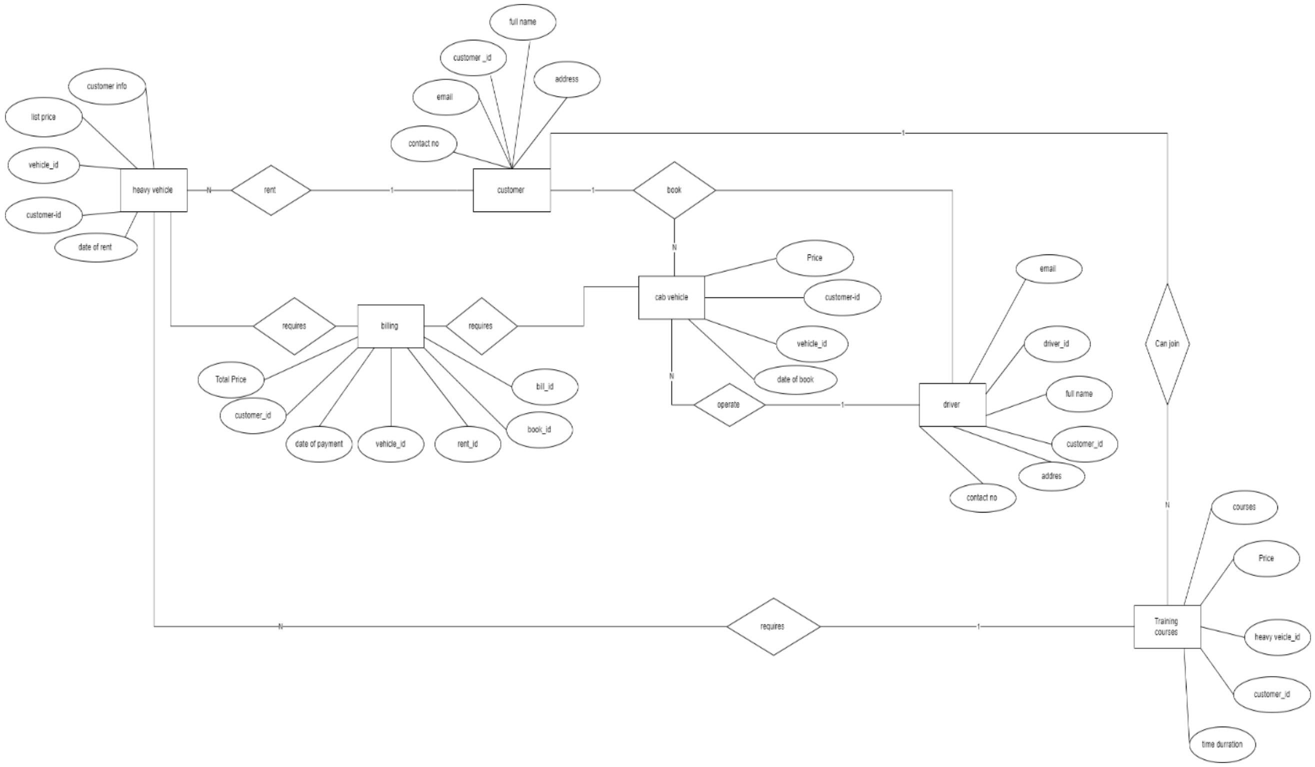


Figure 6: Entity Relation Diagram

4.2.2 Data Dictionary

In a database, information system, or as a component of a research effort, data components are utilized or recorded. A data dictionary is a collection of names, definitions, and properties regarding those data pieces. It gives guidelines on interpretation, acceptable meanings, and representation in addition to describing the meanings and goals of data pieces within the context of a project. Data elements' metadata is also available in a data dictionary. The scope and qualities of data items, as well as the guidelines for their use and application, may be defined with the help of the metadata present in a data dictionary (UC Merced Library, n.d.).

Data Dictionary for Allgemein:

Insert Cust_details into Cust_record = Command

Insert Driver_details into Driver_record = Command

Insert available_vehicle_info into Vehicle_Details = Command

Insert Course_details into Customer_Record = Command

Insert Bank_Receit into receipt_details = Command

Fetch available_vehicles from vehicle_details to rent a vehicle = Command

Insert Course_details to Course_records = Command

Fetch available_courses from courses_record = Command

Driver_Record = Driver_Name + Address + Email + Contact_No

Driver_Name = String

Contact_No = Integer

Available_Driver = {True, False}

Available_Courses = {True, False}

Vehicle_Details = Vehicle_Id + Vehicle_Type + Price

Vehicle_Type = String

Price = String

Receipt_Record = Receipt_No + Total_Price + Date_of_renting

Receipt_No = Integer

Total_Price = Integer

Date_of_renting = Integer

Course_Record = Course_Name+ Price + Duration + Vehicle_ID

Course_Name = String

Price = Integer

Duration = String

Vehicle_Id = Integer

Identity_Record = CitizenShip_No + Date of birth + Age

Identify_Document = {True, False}

Join Training Course = Command

Payment Verification = {True, False}

Rent a Cab = Available_Vehicle + Apply for Vehicle Rent

Book a Vehicle = Cust_Details + Booking_Details

Register Membership = Command

Discount on booking = Command

Rating the Services = Command

Generate Report = Report_List + Report_Info

Cust_Record = Name + Address+Email + Contact_No

Name = String

Address = String

Email = String

Contact_No = Integer

User_Details = Payment_Details + Name + Age + Address + Email

Membership Registered Notification = Membership Notification + Receipt

4.2.3 Process Specification

The method of documenting, interpreting, and clarifying the formulas and logic used to generate output data from process input data is known as process specification (Techopedia, 2012). Process Specification eliminates ambiguity and enables a person or organization to acquire detailed description of the accomplishments including data dictionary and data flow diagrams.

The processes in the DFDs are described as:

Process Number: 1.1

Process Name: Verify KYC Details

Description: This process does KYC verification by comparing the customer details with the government provided details.

Input Data Flow: Customer details, Government provided ID

Output Data Flow: Matching customer ID

Process Logic: The logics for this process are:

- The customers provide customer details.
- Legal documents are taken from government database and is compared with the customer details.
- The customer details that match with the government provided ID are marked as verified and stored in a datastore.

Process Number: 2.0

Process Name: Send Request to Driver

Description: This process sends request to drivers to book a cab.

Input Data Flow: Customer location details

Output Data Flow: Request for a driver

Detailed Logic: The logics for this process are:

- The location of the customer is taken, and a request is sent to a driver.
- The driver request is sent to the datastore that contains drivers' details and the available driver is found which is passed to next sub process (2.1).

Process Number: 2.1

Process Name: Accept suitable request

Description: This process allows the drivers to either accept or decline the request of the customer.

Input Data Flow: Available drivers' details

Output Data Flow: Available driver details including location

Detailed Logic: The logics for this process are:

- The details of the available drivers are taken, and available driver accepts the request.
- The drivers who are not available at the moment shall decline the request.
- The driver details and location are sent to notify the customers

Process Number: 7.2

Process Name: Registration for courses

Description: This process registers the customer in the chosen course.

Input Data Flow: Course Specification

Output Data Flow: Payment details

Detailed Logic: The logics for this process are:

- After the selection of course, the customers are registered in the courses of their choice.
- To register, the customers shall pay the fees.

- The fee structure depends on the type of the course.
- The payment details are sent to the bank which is passed to the next sub process (7.3)

Process Number: 7.3

Process Name: Make receipt

Description: This process makes receipts of the payments and sends them to a datastore.

Input Data Flow: Payment information

Output Data Flow: Receipt details

Detailed Logic: The logics for this process are:

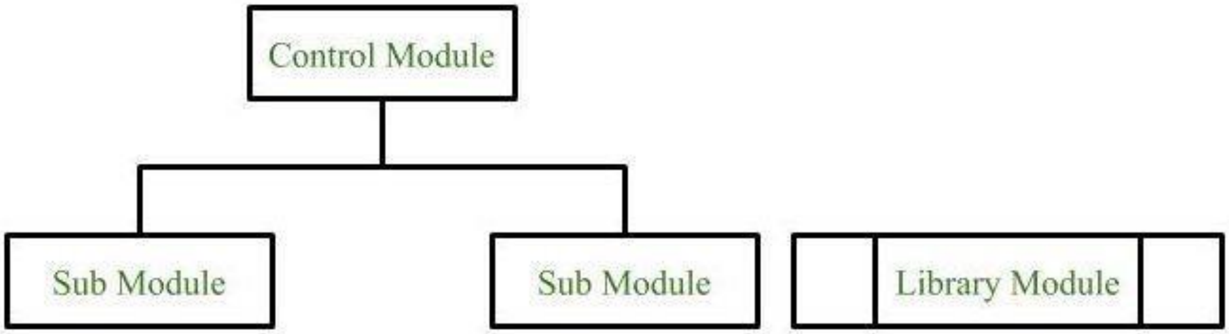
- This process takes the payment details.
- It makes a receipt of the payment made.
- The receipt is then stored in a datastore.

4.3 Design Specification

4.3.1 Structure Chart

In software engineering, a structure chart represents the breakdown of a system into its most controllable components. Module represents the process or task of the system. To organize software modules into a tree, they are employed in structured programming. Each module is represented by a box that includes its name. With the help of arrows, the tree structure illustrates the connections between modules and the data flow between them (Wiki Books, 2020). The module at top level calls the module at low level. When a module calls another, it views the called module as black box, passing required parameters, and receiving results. There are three modules:

- Control Module: A control module branches to more than one sub module.
- Sub Module: It is a part (Child) of another module.
- Library Module: Library Module is reusable and invokable from any module.



Figure

7: Modules in a Structure Chart (Agarwal, 2019)

4.3.2 Structure Chart for the whole system

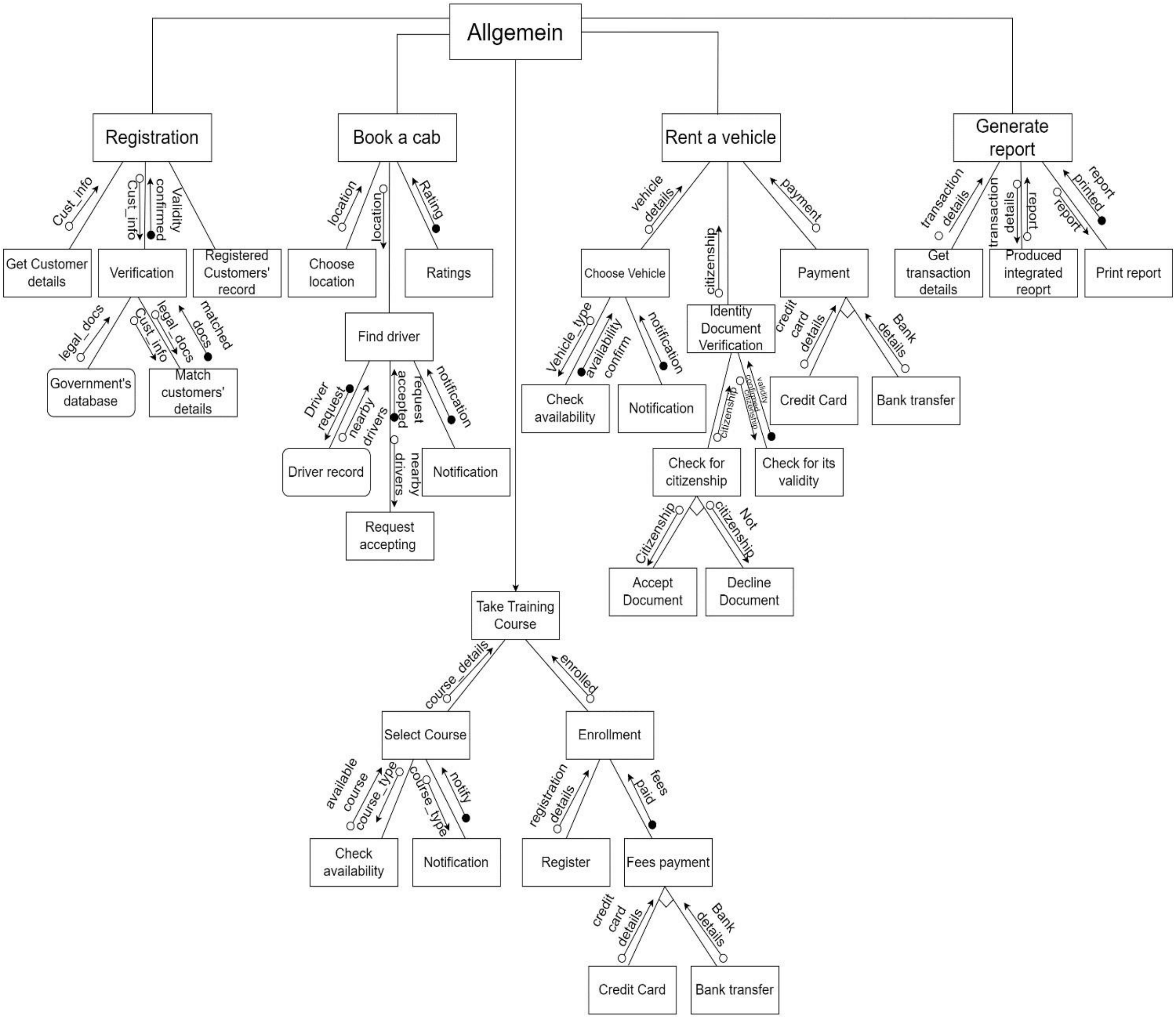


Figure 8: Structure Chart for whole system

4.4 Assignment Diary

4.4.1 Group Member Details and Responsibilities

After the coursework was distributed, all the team members sat together and discussed the scope of the Coursework and made an initial map to dealing with the different aspects of the coursework by dividing the work amongst us. Here’s the assigned tasks for different group members:

Member's Name	Tasks Assigned
Siddhant Thakur	Group Task: Assignment Diary, SRS document, Data Dictionary Individual Task: Take membership
Nayasha Shilpakar	Group Task: Full system DFD, Project Charter, Full system Structure Chart, Process Specification Individual Task: Rent a Vehicle
Sahil Thapa	Group Task: Entity Relationship Diagram Individual Task: Report Preparation
Bimal Devkota	Group Task: Conclusion Individual Task: Book a Cab
Sakar Timsina	Group Task: Overview of the Report Individual Task: Join the Training Course

Table 3: Assignment Diary

4.4.2 Meeting Details

Since, this was a group project, cooperation from all the team members was needed and hence we held several group meetings to better understand the scope of the project and discuss ideas and alternatives among ourselves to come up with a straightforward approach. We consulted our module teachers many times to help us better understand the scope of the coursework and their help was unparalleled.

Date	Time	Discussions

2022/12/20	09:00 AM – 01:00 PM	• Studied the Coursework and divided work among the members.
2022/12/22	12:00 AM – 03:00 PM	• The SRS and Project Charter's attributes and definitions were discussed and a map to doing it was formulated. • Started the Data Flow Diagram and various problems were faced due to the undefined business Scenarios. • Discussed the Individual tasks and divided it among us and set a deadline for the completion of the Individual tasks.
2022/12/25	10:00 AM – 02:30 PM	• • Finally, the SRS and Project Charter was completed and reviewed by all team members and certain points and features were tweaked. The Data Flow Diagram of the System was also completed, and team members were assigned with doing a thorough research on the Structure chart. • Plannings on making the Structure Chart were made and set into motion.

2022/12/26	11:00 AM – 01:00 PM	• Team members were assigned with writing the Data Dictionary and Process Specification. • More time was set for the research part of the Structure Chart. • We used various class slides and online materials to gain insights on the Data Flow in Structure Chart. • Both Data Dictionary and Process Specification was completed Successfully.
2022/12/27	09:30 AM – 01:00 PM	• With respect to the Data Flow Diagram, work on the Structure Chart was Started coherently. • In about three hours, work on Structure Chart was finally Concluded and we were satisfied with the result. • The Team Members had finished the individual tasks to the best of their knowledge, and we were each satisfied with our work.

2022/12/28	08:00 AM – 12:00 PM	• Finally, in the last sitting, we completed the assignment diary, added the individual tasks, and reviewed the entire report to the best of our knowledge. • In the last one hour, we all took one last good look at our report, went over everything briefly, discussed our individual parts and matched our ideas and scope for the projects. After feeling satisfied with the result, we decided to conclude our Coursework.

Table 4: Meeting details

4.4.3 Assumptions

The assumptions made to design the DFDs are listed below:

i. A customer can take as well as cancel his/her membership and the database shall be updated every time.

ii. To verify the customer details, government's database is used.

iii. When a customer places a request for a driver to book a cab, all the available drivers can accept the request and the customer can choose any available driver.

iv. There are different types of training courses available like:

- Heavy vehicle driving training course
- Cab driving training course
- Bike riding training course

v. The fee structures for different training courses are different.

vi. Only online payment is accepted. vii. The payment method can be either be via credit card or via bank transfer.

viii. A receipt is made every time a customer uses any service which is sent to customer, driver as well as admin.

ix. Customers can rate the drivers based on their ride experience.

x. The ratings shall be stored in drivers' database.

xi. All the transactions made in a day is reported by the system to the admin.

5. Individual Tasks

5.1 Register Membership (Siddhant Thakur)

Allgemein allows new users to Register themselves in the system the first time and afterwards logs them in, in every session. The Registration process works by taking user_details from the User, verifying the details via various processes and once verified stores the data in a data store for matching to log them in, in future sessions.

5.1.1 Environmental Model Specification

5.1.1.1 Context Level Diagram

This is the Context Level Diagram for the Membership Registration Process. At this level, the User provides his/her info to the Register Membership process and the process registers the User to the system. The user gets a Notification once their Registration Process is Complete.

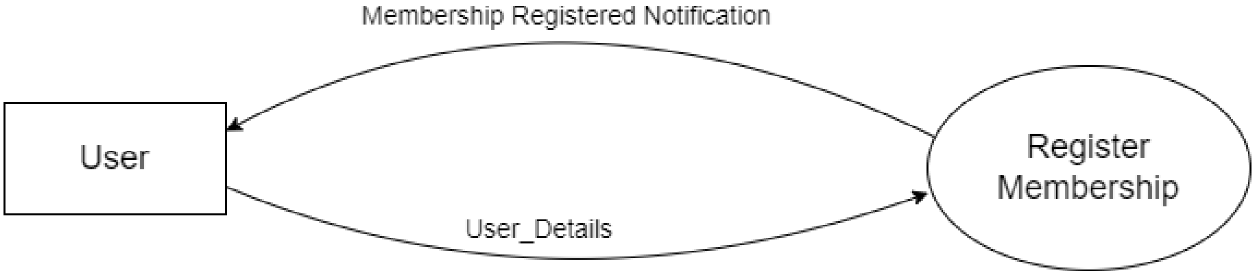


Figure 9: Context Level Diagram for Register Membership

5.1.2 Internal Model Specification

5.1.2.1 Level 1 DFD for Register Membership

In Level 1, we further separate the Verification process from Registration Process. Here, the user's details are verified first and then stored into the data store from where the data is passed to the Registration process and once registered the user gets Notified.

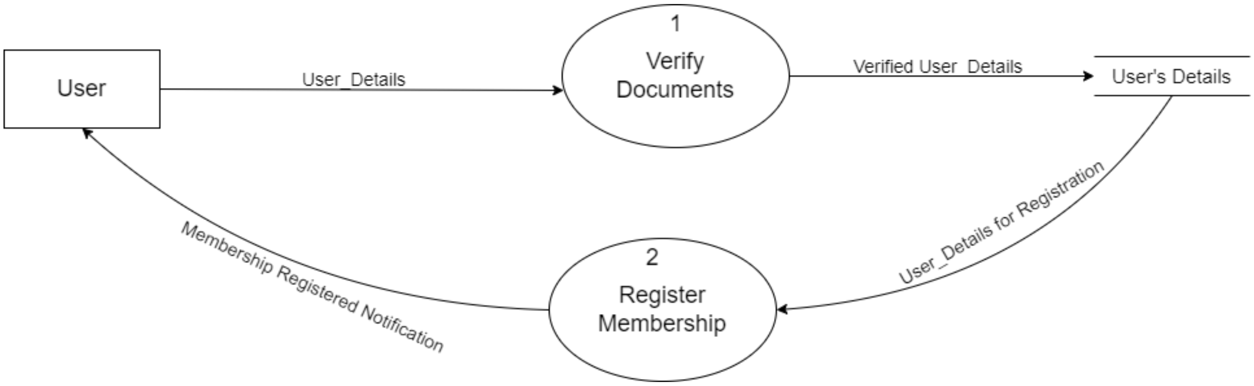


Figure 10: Level 1 DFD for Register Membership

5.2.2.2 Level 2 DFD for Register Membership

In Level 2, Payment_Details and User_Details are separately verified. Bank verifies the Payment_Details and The KYC Verification Process verifies the rest of the user Details, and the verified data is then stored into User_Details and then further goes to Register membership process hereby registering the User in the system.

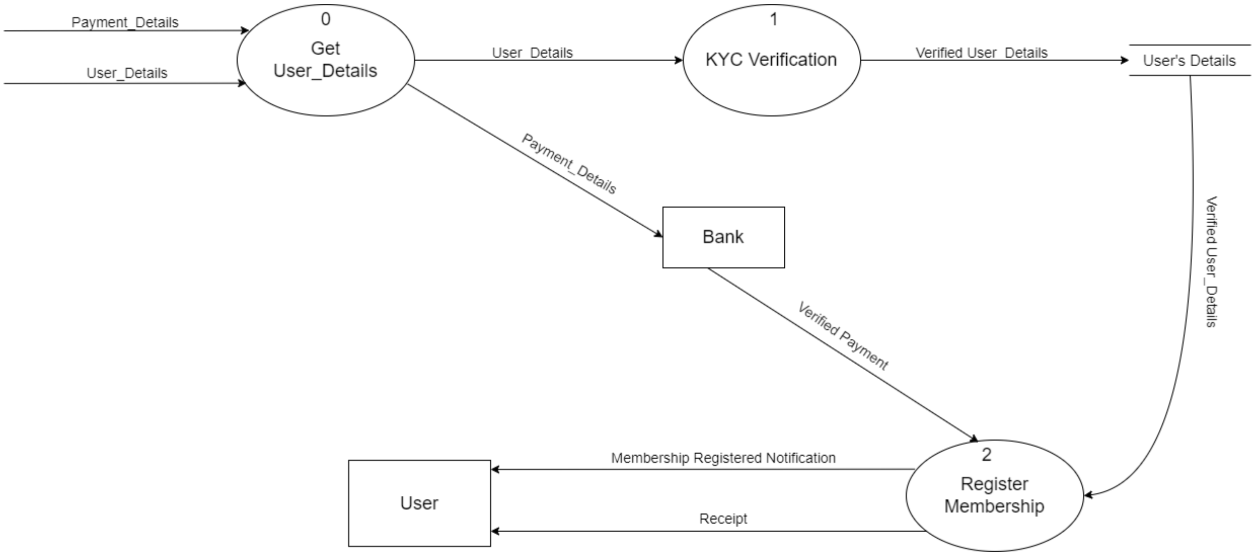


Figure 11: Level 2 DFD for Register Membership

5.1.3 Design Specification

5.1.3.1 Structure Chart

A Structure Chart shows the breakdown of a system or process in its lowest manageable levels. Here, Membership Registration Process is one of the many processes of the Root system. And within the Membership Registration Module, the sub-Module Get User_Details takes input from the user and passes it back to the parent Module which then forwards the input to the Verify User_Details Module where the input details are verified and a verification status and User_Details are then sent back to the Parent Module from where it is redirected to the Display Membership Module which for Verified Details, Notifies the user of successful verification and adds Data into the Data Store and for Unverified Details, notifies users to provide Valid Details.

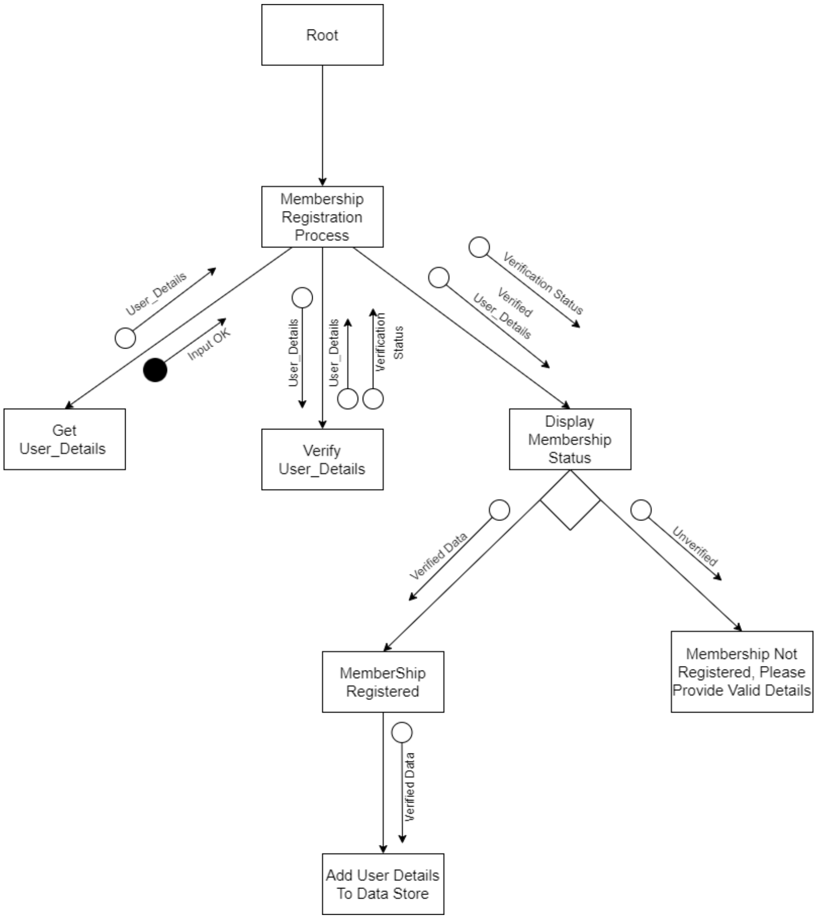


Figure 12: Structure Chart for Register Membership

5.1.3.2 Module Specifications

Module Name	Register Membership
Purpose	It registers a member in the system

Pseudocode	Do
	    Input Full_Name
	    Input Contact_No
	    Input Email
	    Input Address
	  DO
	    IF (User_Details equals Verified)
	      Display (Membership Registered)
	    ELSE
	      Display (Invalid Details)
	  END DO
	END DO

Input Parameters	Full_Name, Contact_No, Email, Address
Output Parameters	Membership Registered Message
Global Variable	Full_Name, Contact_No, Email, Address
Local Variable	None
Call	Registration Process
Called By	User

Table 5: Module Specification for Register Membership

5.2 Rent a Vehicle or specialist drivers (Nayasha Shilpakar)

Allgemein has a significant number of vehicles such cargo trucks, bulldozers, and cabs for the customers to rent. This system for Allgemein has a function ‘Rent a vehicle’ that allows the users to rent the vehicles online. The customers can choose the vehicle of their preference and hire specialist drivers if needed. The customers must provide with their Identity document (Citizenship) to rent the vehicles.

5.2.1 Environmental Model Specification

5.2.1.1 Context Level Diagram

In context level diagram the main process is 'Rent a vehicle'. The external entities are customer, admin, and bank.

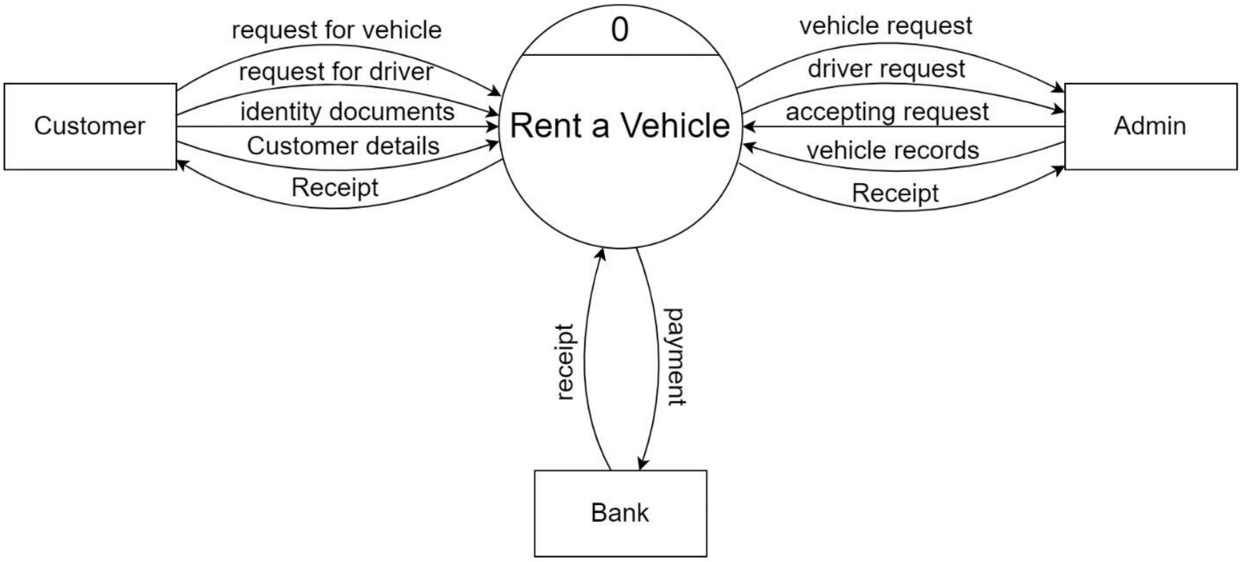


Figure 13: Context level diagram for rent a vehicle

5.2.2 Internal Model Specification

5.2.2.1 Level 1 DFD

The diagram for level 1 DFD of the function 'rent a vehicle' is given below.

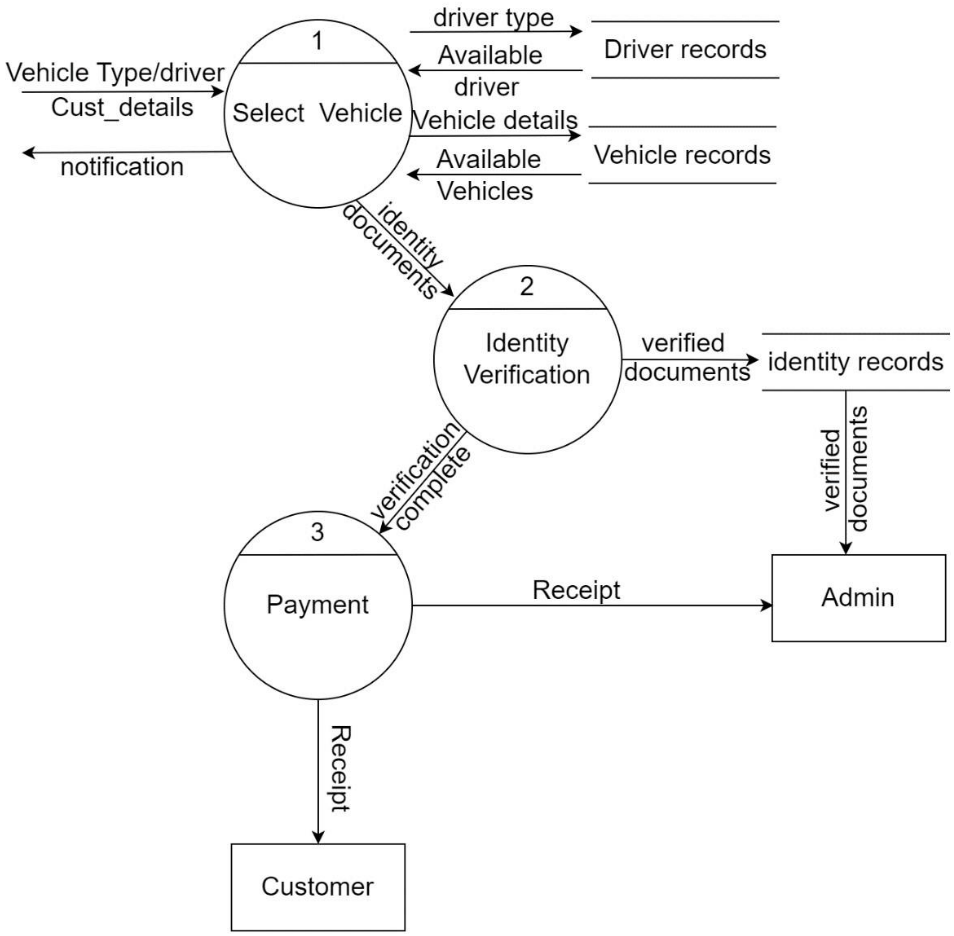
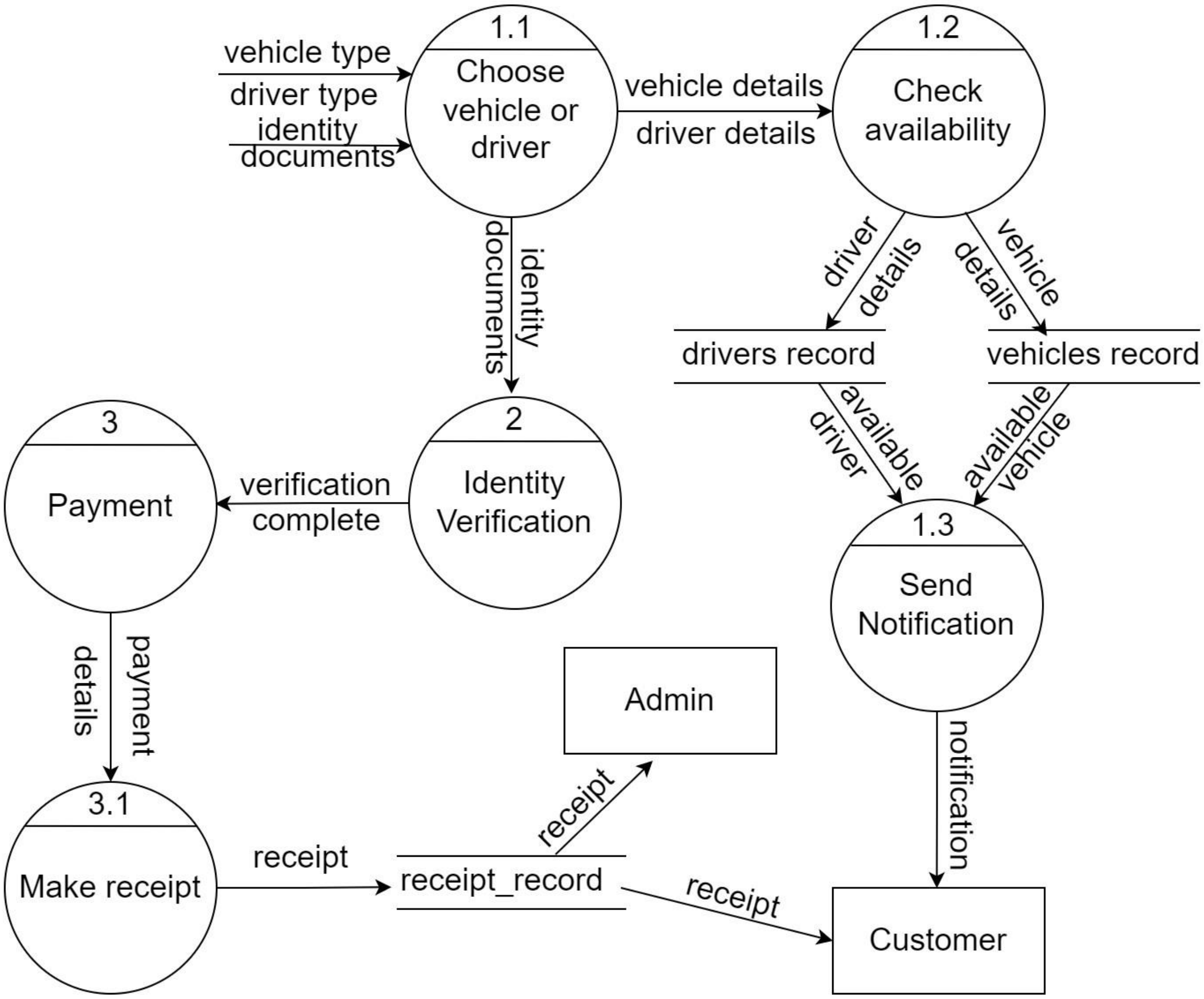


Figure 14: Level 1 DFD for rent a vehicle

Here the customer provides their details and the type of vehicle that they want to rent. Customers can hire specialist drivers as well. The type of vehicle and driver they want is then checked in the databases that contain the details of vehicle and drivers. After selecting the vehicle and driver, the customers need to provide their identity document (Citizenship card). The citizenship is then compared with the government's database and the matched document is stored in a datastore (identity records). After the verification process is completed, the customers make the payment and the receipt is sent to both admin and customer.

5.2.2.2 Level 2 DFD

The diagram for level 2 of the function 'Rent a Vehicle' is given below.



Figure

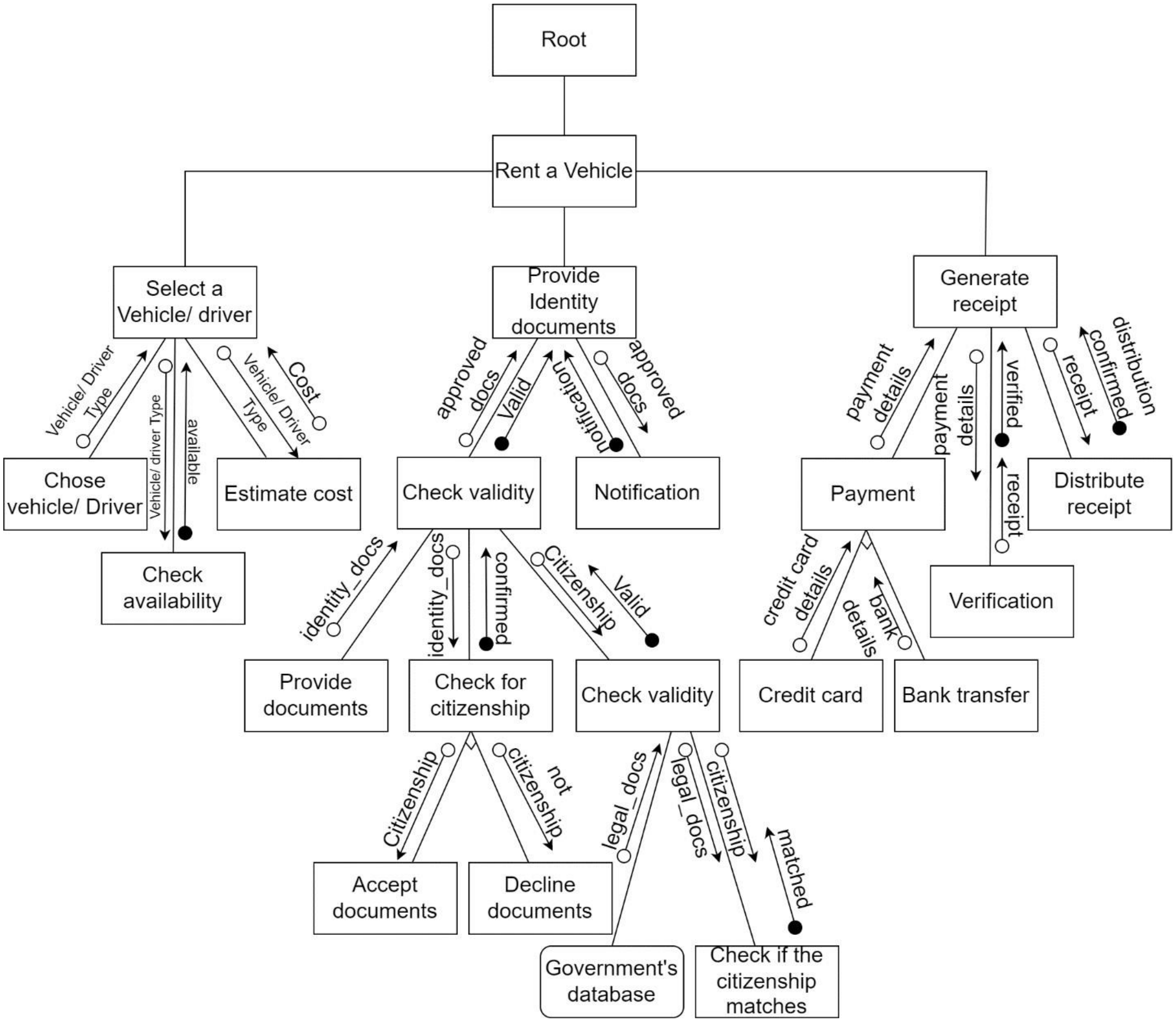
15: Level 2 DFD for rent a vehicle

The processes of level 1 DFD are further divided into sub processes. Here, to choose the vehicle and driver, the type of vehicle and driver are taken as input and their availability are checked. The datastores of drivers record and vehicle records are used to check their availability. When the availability is confirmed, a notification is sent to the customers. The customers are required to provide their identity documents which is verified, and the process for the payment starts. After the customer provides the payment details, a receipt is made which is stored in a datastore and through that datastore the receipts are distributed to admin and customer.

5.2.3 Design Specification

5.2.3.1 Structure Chart

The structure chart for the function ‘Rent a Vehicle’ is given below.



Figure

16: Structure Chart for rent a vehicle

5.2.3.2 Module Specification

Name	Rent a Vehicle
Purpose	To rent Vehicles and hire specialist drivers

Pseudocode	DO
	    INPUT Cust_details, Vehicle/ Driver Type, Identity Document
	DO
	        IF Vehicle_type matches available Vehicle_details
	DO
	            IF Identity document is verified
	                DO
	                    SEND rent request
	                    START payment process
	                    SEND notification
	                    MAKE receipt
	                END DO
	            ELSE
	                DO
	                    SEND error message
	                END DO
	        END DO
	        ELSE
	        DO
	            SEND error message
	        END DO
	    END DO
	END DO

Input parameters	Cust_details, Vehicle/Driver Type, Identity Document
Output parameters	Notification, Receipt
Global Variable	-
Local Variable	Cust_details, Vehicle/Driver Type, Identity Document
Call	Rent Vehicle
Called by	User

Table 6: Module Specification for Rent a Vehicle

5.3 Report preparation system (Sahil Thapa)

Transportation industry software This website has a feature that will create a report for the user based on whether they approve or disapprove of it. The system will verify the user's details after they are provided by the user. The system successfully generates the report after the user information has been verified and uploaded to the database.

5.3.1Context level diagram

This context diagram shows the steps involved in creating a report. The method of "report creation" is utilized by the system to issue reports based on user approval or disapproval at this context level. The customer can see the results of their request.

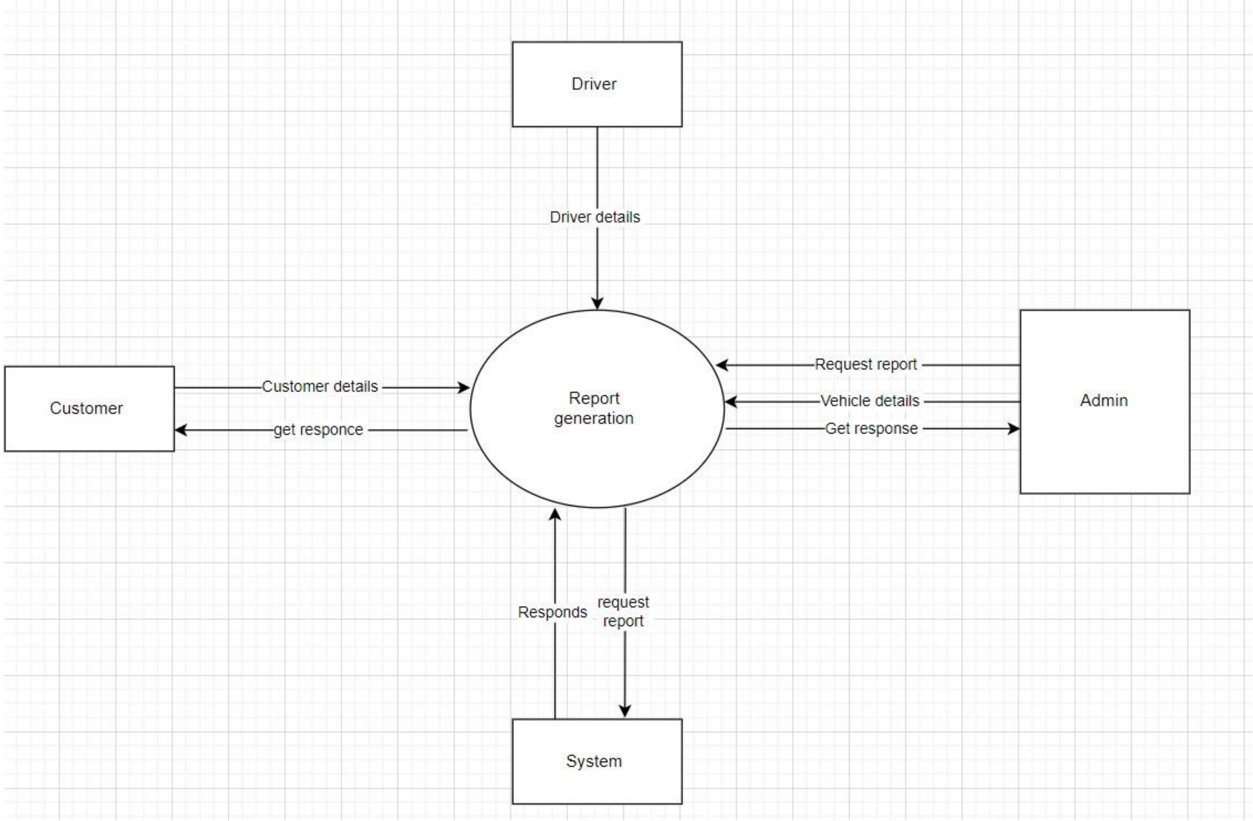


Figure 17: Context Diagram for report preparation

5.3.2 Environmental Model Specification

5.3.2.1 Level 1 DFD

This is the level 1 DFD diagram describing the process of preparing a report. The system request details of customers, drivers, vehicle and verifies. Once the given data are valid, the valid data are stored in a data store, and went further for report generation which is later provided to admin.

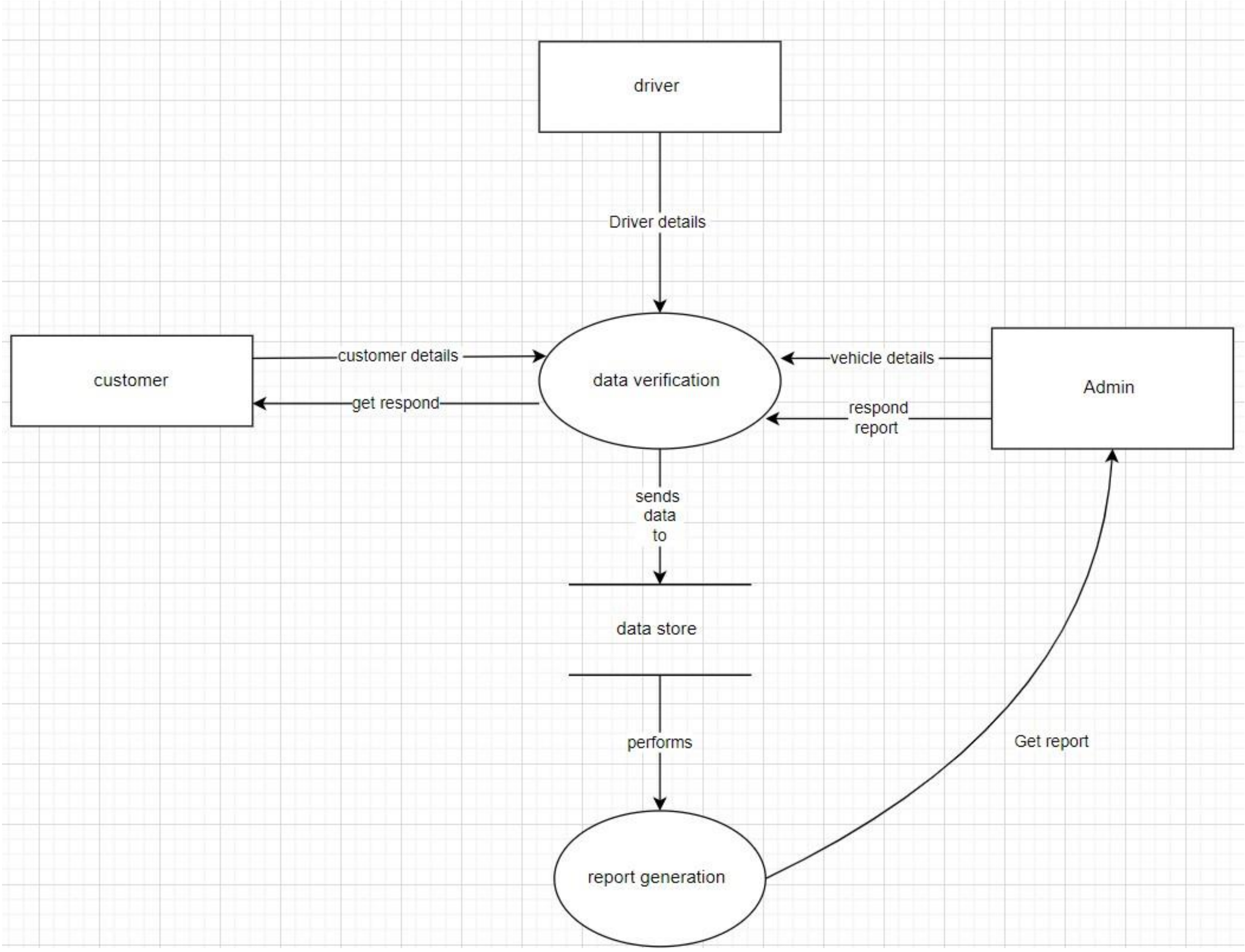


Figure 18: Level 1 DFD for report preparation

5.3.2.2 Level 2 DFD

The most detailed portion of level one in this DFD graphic is at this level. After the data verification procedure, the confirmed data are sent to a separate data storage. The system then generates a report and sends it for further review. At that point, the gathered reports are divided into authorized and unapproved categories before being delivered to the admin.

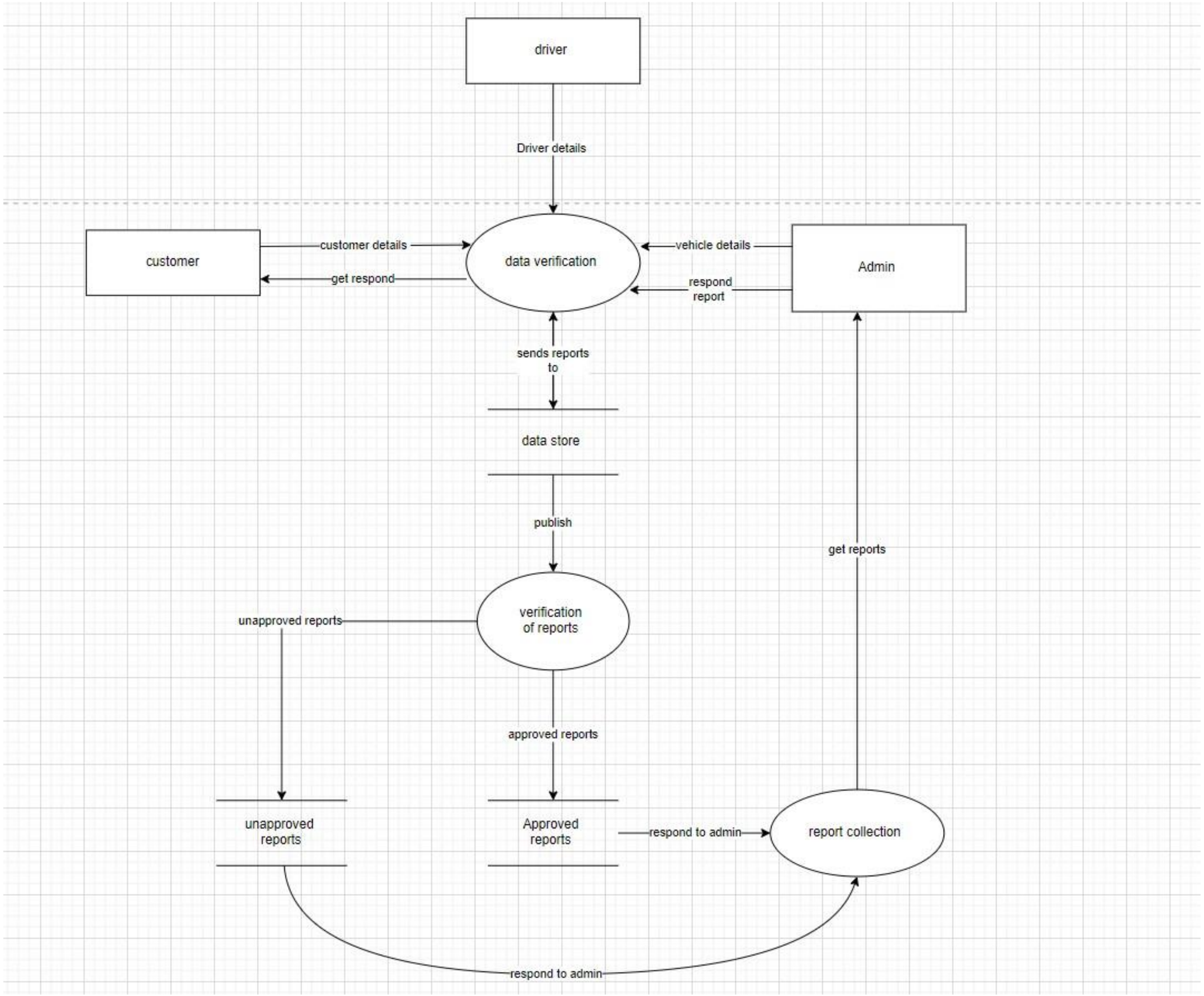


Figure 19: Level 2 DFD for report preparation

5.3.3 Design Specification

5.3.3.1 Structure Chart

The general structure of the report preparation is depicted in this diagram. Procedures including gathering customer details, vehicle details, driver details, and generating reports status are depicted in this diagram.

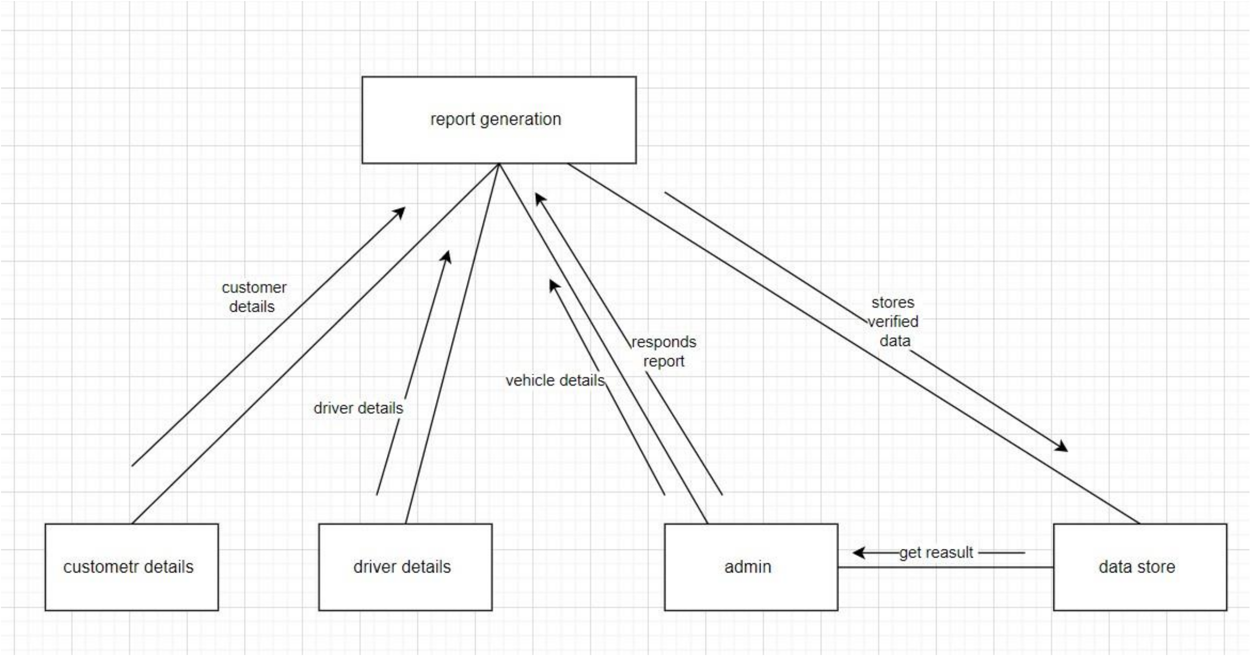


Figure 20: Structure Chart for report preparation

5.3.3.2 Module Specification

Name	Report preparation
Purpose	To generate a report

Pseudocode	DO INPUT customers details INPUT vehicle details INPUT driver details DO IF (verified (customers details, vehicle details, driver details)) END DO STORE Details to Data store GENERATE Report, distribute END DO
Input parameters	customers details, vehicle details, driver details
Output parameters	Report
Global Variable	None
Local Variable	customers details, vehicle details, driver details
Call	Report Generation
Called by	Admin

Table 7: Module Specification for report preparation

5.4 Book a Cab

Transportation industry software This website has a feature that will allow customer to book a cab. The system will verify the customer's details after they are provided by the user. The system successfully books a cab for the customer.

5.4.1 Context level diagram

This context diagram shows the steps involved in booking a cab. The method of "booking a cab" is utilized by the system to issue reports based on user approval or disapproval for cab booking process. The customer can see the results of their request.

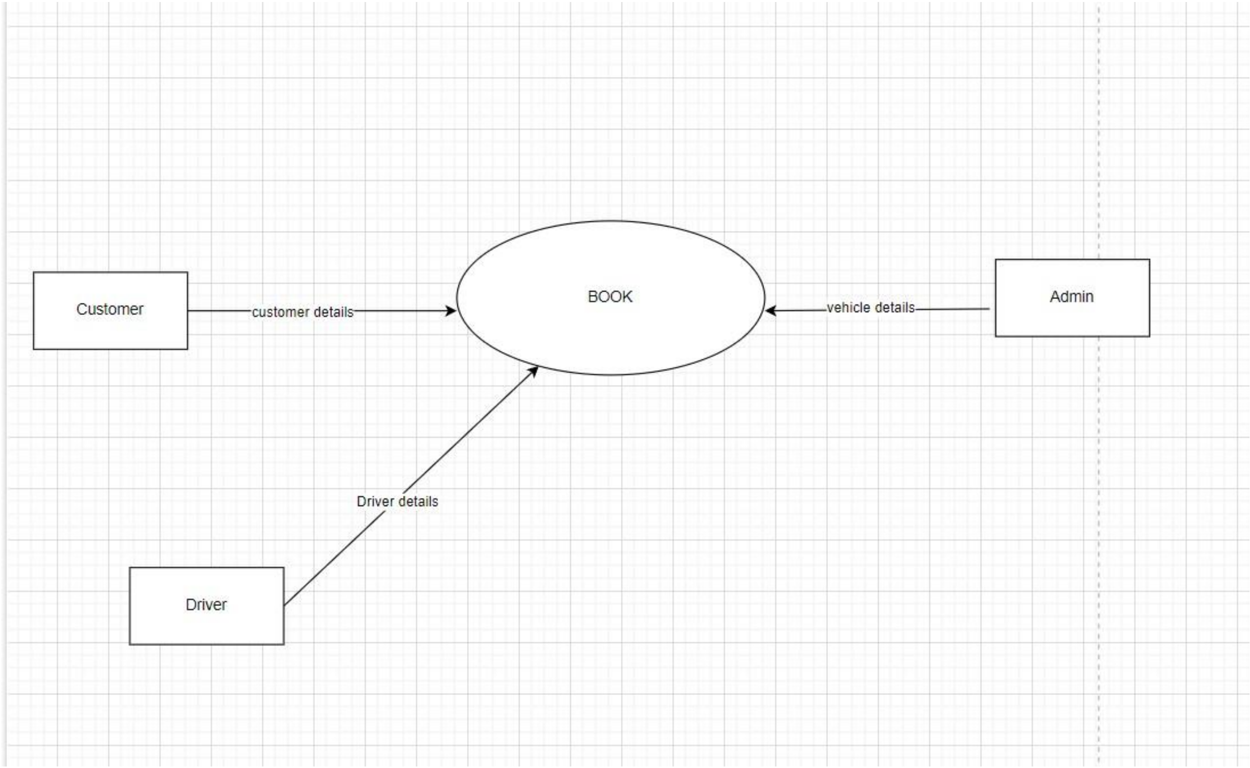


Figure 21: Context Diagram for book a cab

Level 1 DFD

This is the level 1 DFD diagram describing the process of booking a cab. The system request details of customers, drivers, vehicle and verifies . Once the given data are valid, the valid data are stored in a data store , and went further for which is later used for booking a cab.

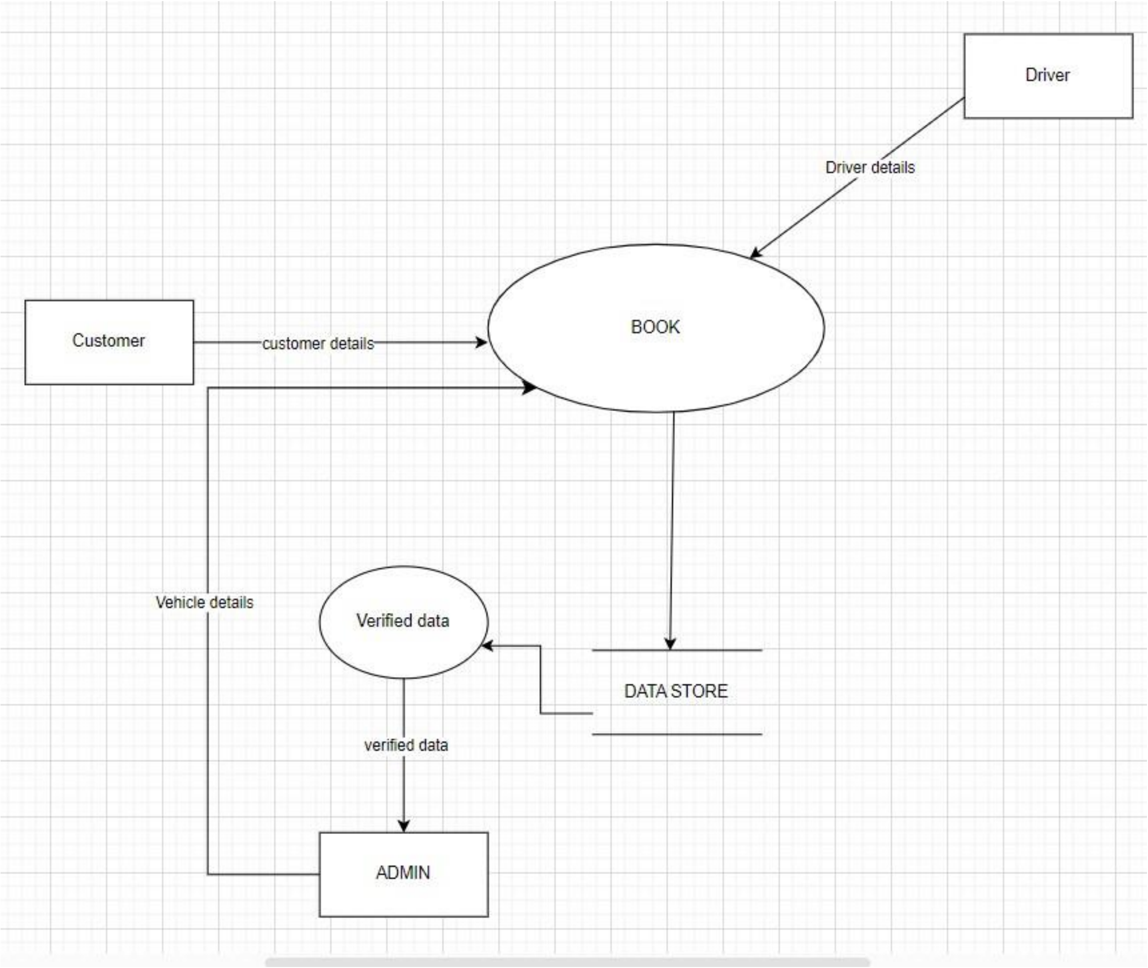


Figure 22: Level 1 DFD for book a cab

Level 2 DFD

The most detailed portion of level one in this DFD graphic is at this level. After the booking process data are sent to a data storage. The system then books a cab and sends the results to the customers.

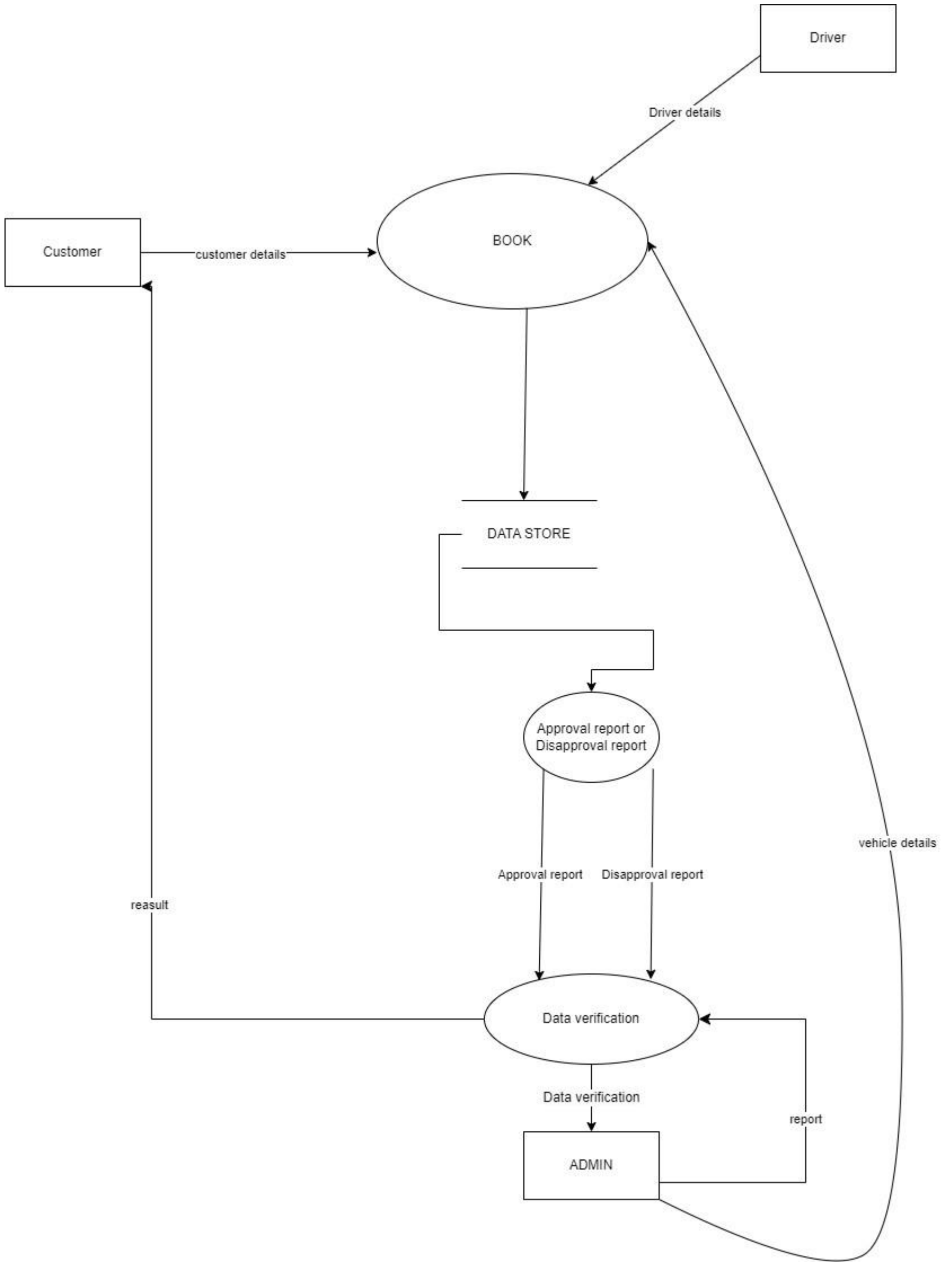


Figure 23: Level 2 DFD for book a cab

Structure Chart

The general structure of the booking cab process is depicted in this diagram. Procedures including gathering customer details, vehicle details, driver details, and carrying out booking procedure status are depicted in this diagram.

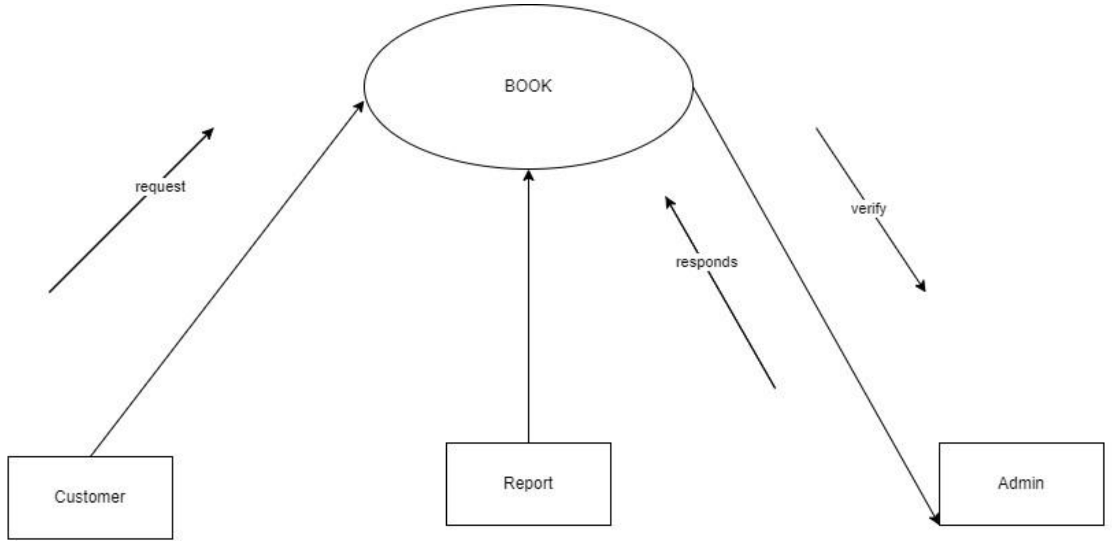


Figure 24: Structure Chart for book a cab

Name	Book a cab
Purpose	To book a cab

Pseudocode	DO INPUT customers details INPUT vehicle details INPUT driver details DO IF (verified data) END DO PROVIDE Details to Admin GENERATE result , distribute END DO
Input parameters	customers details, vehicle details, driver details
Output parameters	Book a cab
Global Variable	None
Local Variable	customers details, vehicle details, driver details
Call	Book a cab
Called by	Customer

Table 8: Module Specification for book a cab

5.5 Join Training Course (Bimal Devkota)

This course provides the various training course such as Bulldozar , Bus , Jeep , Truck which can be accessible by the user.

Level: 0

this stage shows the steps involved in selecting a training courses. The method of selecting a courseis utilized by the system to issue the details for training section.

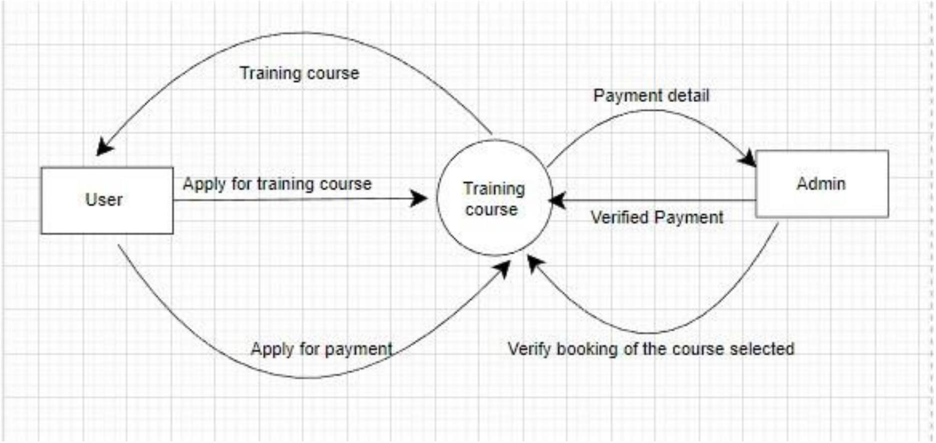


Figure 25: Context Diagram for join training course

Level1:

This level 1 dfd describing the process for training purpose the system request details of the customers , courses , payment and provides a results to the customer

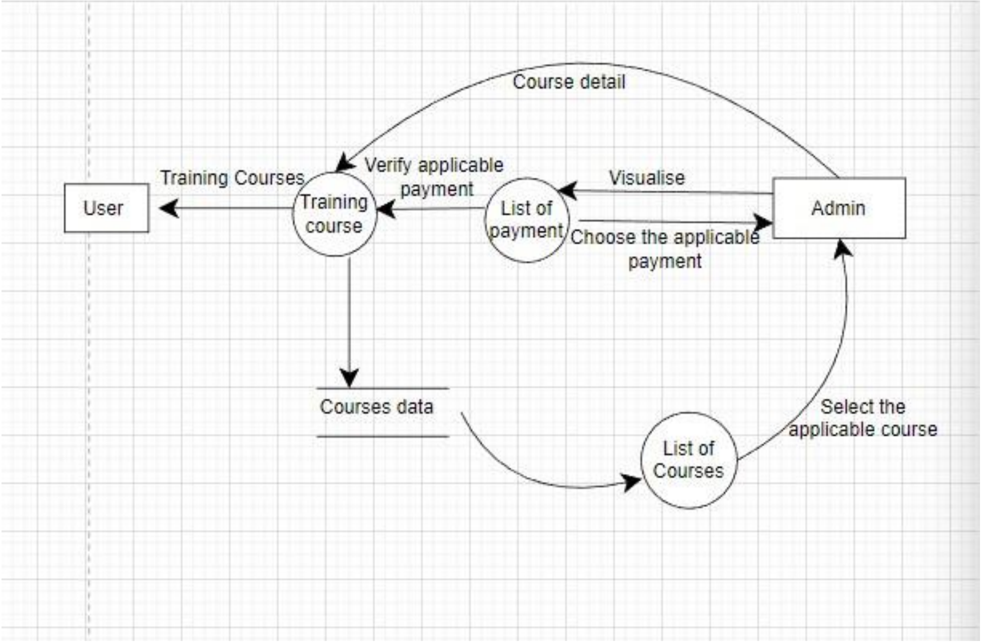


Figure 26: Level 1 DFD for training course

Level:2

In this level 2 of dfd . the customer has to choose the list of courses and payment methods and after harnessing the user’s data it is stored in a user data storage and further sends for verification to the admin and later the results will be provided to the users

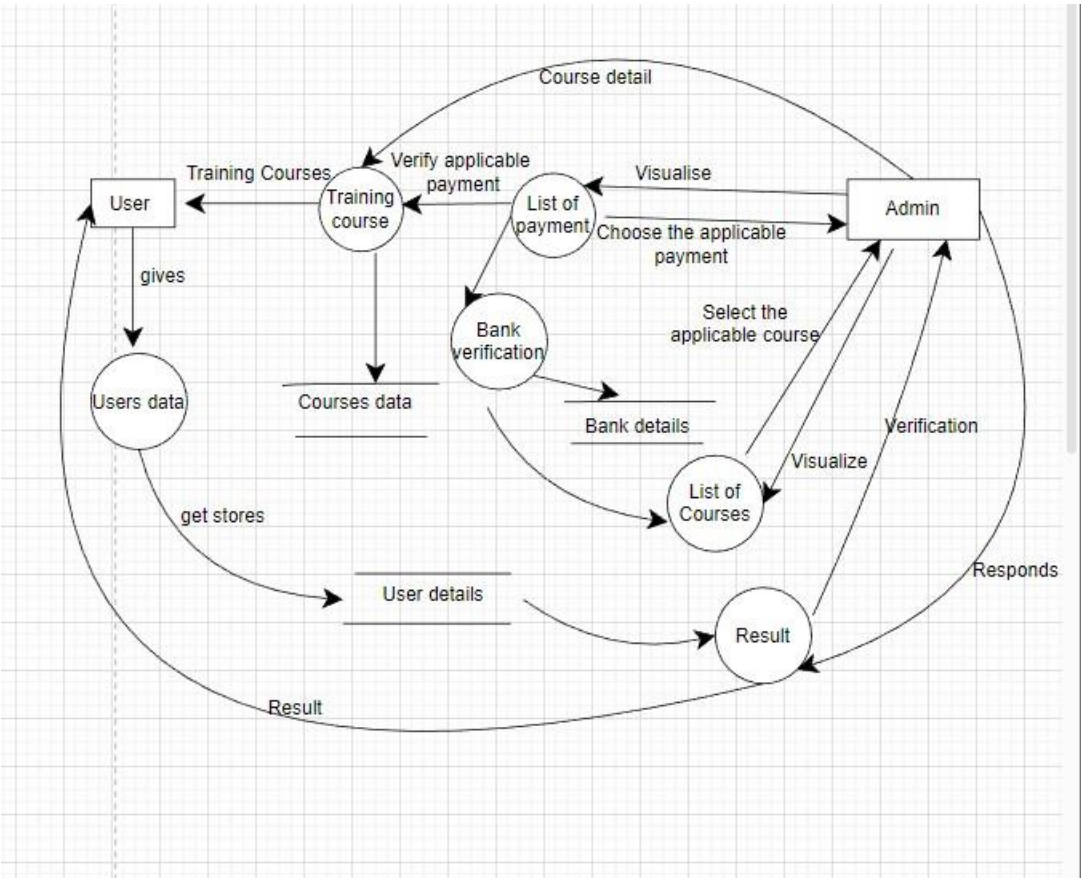


Figure 27: Level 2 DFD for training course

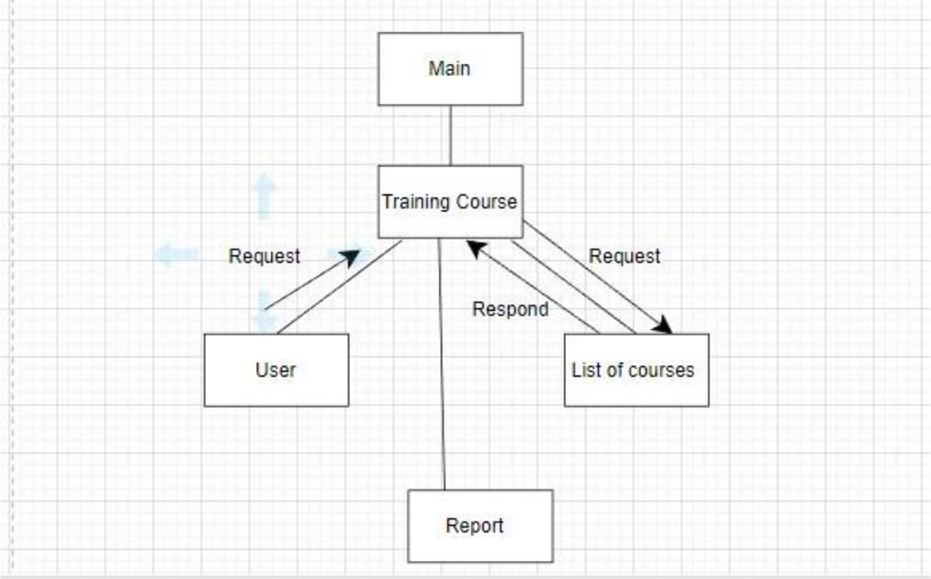


Figure 28: Structure chart for training course

Name Join Training Course

Purpose To Join a Training Course

Pseudocode

```
DO
    INPUT Course Name

    DO
    IF ((Course_Name) available )
    END DO
    PROVIDE  Details to Users
    GENERATE result
```

```
                    END DO
```

Input parameters	Course Name
Output parameters	Book a cab
Global Variable	None
Local Variable	Course name
Call	Join Training Course
Called by	Customer

Conclusion

The goal of this assignment was to design the "Allgemein" transportation application, which offers three essential services: cab booking, car hire and provide training services. Customers who use cab services can reserve rides to get them from one location to another, either inside the city or even from one Nepalese city to another. Like this, customers of vehicle rental services can hire vehicles like freight trucks, bulldozers, etc. for a set period of time after submitting a specific cost designated to a particular vehicle.

When finishing the coursework, we had both group and individual assignments. We created a few evaluations and design specifications for the entire system as part of group projects. For the system, we also created context level DFD, level 1 DFD, and level 2 DFD. The construction of the ERD, data dictionary, process specification, structure chart, and assignment diary were other things we worked on as a group. In our individual jobs, we were required to choose one of the five functions offered and create our own context level DFD, level 1 DFD, and level 2 DFD for that function. For the functions, we also created a structure chart and a module specification.

In order to complete the coursework, we learnt how to build a system and how to use data flow diagrams, structure charts, data dictionaries, ER diagrams, module specifications, and process specifications. We also gained expertise in project management and requirement analysis. As a result, the group's efforts to finish this homework were successful.

References

Agarwal, A., 2019. *Geeks for Geeks.* [Online]
Available at: https://www.geeksforgeeks.org/software-engineering-structure-charts/
[Accessed 26 December 2022].

Computer Hope, 2022. *Computer Hope.* [Online]
Available at: https://www.computerhope.com/jargon/d/drawio.htm [Accessed 20 December 2022].

Gillis, A. S., 2022. *Tech Target.* [Online]

Available at: https://www.techtarget.com/searchcio/definition/project-charter [Accessed 20 December 2022].

Lucid Chart, n.d. *Lucid Chart.* [Online]
Available at: https://www.lucidchart.com/blog/data-flow-diagramtutorial#:~:text=Level%201%20DFDs%20are%20still,stores%20to%20link%20them%20 together.
[Accessed 21 December 2022].

Miro, n.d. *Miro.* [Online]
Available at: https://miro.com/blog/context-diagram/ [Accessed 20 December 2022].

Singh, C., 2012. *Beginners Book.* [Online]
Available at: https://beginnersbook.com/2015/04/e-r-model-in-dbms/ [Accessed 21 December 2022].

Techopedia, 2012. *Technopedia.* [Online]
Available at: https://www.techopedia.com/definition/28636/process-specification [Accessed 25 December 2022].

UA Little Rock, 2022. *UA Little Rock.* [Online]
Available at: https://ualr.edu/itservices/applications/v/microsoft-word/ [Accessed 20 Decembetr 2022].

UC Merced Library, n.d. *UC Merced Library.* [Online] Available at: https://library.ucmerced.edu/data-dictionaries [Accessed 25 December 2022].

Visual Paradigm, n.d. *Visual Paradigm.* [Online]
Available at: https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-dataflow-diagram/
[Accessed 20 December 2022].

Wiki Books, 2020. *Wiki Books.* [Online]

Available at:
https://en.wikibooks.org/wiki/Alevel_Computing/AQA/Problem_Solving,_Programming,_Data_Representation_and_Pr actical_Exercise/Problem_Solving/Structure_charts
[Accessed 26 December 2022].