ABSTRACT

Coffee Shop Management System is based on a concept to maintain orders and management of a particular coffee shop.

The aim of the coffee management is to create communication between rural area people and coffee management.

This project automates the process of manually maintaining the records related to transaction flows, user details, payment details. Coffee Shops are very needy. This project helps the owners of Coffee Shops to maintain day to day transactions on the computer.

This project deals with management of the coffee. It deals with the purchase and sale of coffee.

TABLE OF CONTENT

List of Figures

INTRODUCTION

The **Coffee Shop Management** is based on a concept to maintain orders and management of a particular coffee shop. The administrator can handle the data and update information of the coffees . The Coffee Shop Management System is a web based project. The Aim of Coffee Shop Management is communication between rural area people and coffee shop management.

1.1 Background:-

The System "Coffee Shop Management" is an automated system. Coffee Shop Management is a web based project. In this system Admin can handle data. In this System admin can handle data.

The Customer can suggest the admin(distributor) to make any change in coffees.The administrator can add, update ,delete any information about the system.

1.2 Objective:-

- To develop a web based system that will help to manage the coffee's information about the coffees.
- It takes the suggestion from the user.
- This system is helpful for rural area people.
- It takes less processing time.
- To provide fresh coffee and healthy products to the people.

1.3 Purpose and Scope:-

1.3.1 Purpose:

- To handle records easily by saving time and cost.
- It is an automated process.
- The main purpose is to provide fresh coffee.
- Using this system the coffee products give earnings.

1.3.2 Scope:-

The Coffee Shop management helps to manage and run the Cafe shop systematically. In this management system we will provide a café employee to take order.So the owner of the shop can evaluate the whole system. Employees can take payment which will be managed into the software.The administrator can handle all records like employee ,product ,customer ,order and bill.

1.3.3 Applicability:-

- This system is user friendly.
- It provides high security.
- It is flexible to use as compared to the previous system.
- It contains all operations like delete ,view ,add.
- This system is transparent/clear to the user as compared to the previous system.
- This project is helpful for farmers and rural area people.

System Analysis

2.1 Existing system:-

- It was very tedious.
- Slow data processing.
- Not user friendly.
- No more orders for the product and benefits are less.

2.2 Fact Finding Requirement:-

1] What operations do you want to perform on the coffee shop Management?

- ✔ Update

- ✔ Delete

- ✔ Insert

- ✔ View

2] Did you want to link your website with social media?

- ✔ No

3] Did you need to take High security for your system?

- ✔ Yes

4] What is your cost budget for developing your website?

- ✔ 60,000

5] Do you want to give any notification on your website?
- ✔ Yes

6] Did you need to show all Coffee Shop information on your website?

    ✔ No

7] Do you want to get any suggestions from the user?

    ✔ Yes

8] Is your opening the Coffee Shop Management helpful for customers?

    ✔ Yes

9] Did you need to take feedback from customers?

    ✔ Yes

10] Did you want Graphical Representation in your Website?

    ✔ No

11] How do you want to generate your bill?

    ✔ Cash on delivery

2.3 Proposed System:-

- This system is user friendly.
- It provides high security.
- It contains all operations like add, update, delete, view etc.
- It is an automated system.
- This system helps to place orders for coffees.

2.4 Requirement Analysis:-

By doing fact finding techniques in my project i.e Coffee Shop Management . I gather Information For my project according to the requirements of my client, where the client can handle these systems easily. It provides high security . Coffee Shop Management is an automated system that is using customers to easily order coffees at any time.

2.5 Requirement Analysis:-

By doing fact finding techniques in my project i.e Coffee Shop Management . I gather Information For my project according to the requirements of my client, where the client can handle these systems easily. It provides high security . Coffee Shop Management is an automated system that is using customers to easily order coffees at any time.

2.6 Hardware Requirements:-

❖ Processor : Intel Pentium core i5

❖ RAM : 8 GB

❖ Hard Disk :512 GB

2.7 Software Requirement:

❖ Design Constraint:

- Operating System:- windows 10
- Front End:- Microsoft .net framework with Microsoft visual studio

2015

- Back End:- SQL Server 2012

- Language:- English ❖ Requirement User:

- Operating system:- Any Operating System
- Browser:- Any Browser, jdk file
- Back End:- SQL Server

2.8 Feasibility Study:-

A Feasibility study is performed by a company when they want to know whether a project is possible given certain circumstances. There are three types of feasibility study are:

- ☐ Technical
- ☐ Feasibility Time
- ☐ Feasibility

  Operational

  Feasibility

1] Technical Feasibility:-

- The technical feasibility in the proposed system deals with technology used in the system. It deals with the Hardware and Software used in the system whether they are of latest technology or not.

- Technical feasibility study is the complete of the project in terms of input, output, program and its language etc….

- In this project I am using language which are commonly use to all of us i.e English

- The whole project is made in Microsoft visual studio 2015.

- I am using this technology because it is easy to understand.

- In this project I am using .Net technology at the front end and sql server at the back end also c# programming language is used for coding.

2] Time Feasibility:-

      -Time feasibility means in simple words that the project is completed before the given time period.

- On this system time given is one year which is divided into two parts.

- Ensure that you drink coffee on time and fastly.

3] Operational Feasibility:-

-Operation feasibility refers to the measure of solving problems with the help of a new proposed system.

-This software is user friendly and does not require any technical person to operate.

-The requirements of the user/admin I have developed for this website.

-On this system all coffees are available because only the admin can handle or update, Delete date at any time.

-Users can see only the website and view the product, order product quantity and provide their own suggestions.

-also designed an interface which is easy to use and user/admin can access this website.

-Users can only see coffees and its quantity which helps to view. Insert operation we can also use this project.

2.9 Methodology:-

2.9.1 Spiral model

- ▯ Spiral model is a combination of an iterative nature of prototyping and systematic aspects of traditional waterfall models.

- ▯ The spiral model has four phases: Planning, Risk, Engineering and Evaluation.

- ▯ This model is best used for large projects.

- ▯ This model the same activities are repeated for all the spirals till the entire software is built.

1. Planning phase-
- Requirements are gathered during the planning phase.
- Requirements like business requirement specification and system requirement specification.

2.Risk Analysis:-

- Risk analysis includes identifying, estimating and monitoring the technical feasibility and management risk.
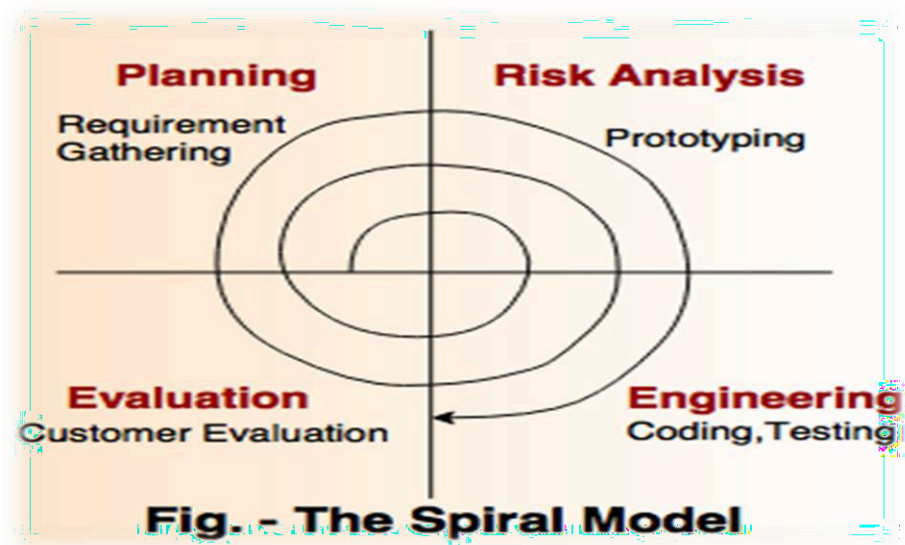
3.Engineering phase:-

- Actual development and testing the software take place in this phase.

4.Evaluation phase:-

- Customers evaluate the software and provide their feedback.

Diagram of Spiral Model:

Fig. - The Spiral Model

How this Model use in my project:-

1) Development is fast.
2) Software requires significant changes.
3) Risk analysis is proper.
4) Users see the system early.

System Design

3.1 MODULE DIVISION:-

3.1.1 Class Diagram:-

Customer
+Cust_ID
+Cust_Name
+Address
+E_id
+Password
+Phone_No
+insert()
+update()
+delete()

Admin
+Name
+E_id
+Password
+Address
+Phone_No
+insert()
+update()
+delete()
+view()
+edit()

Employee
+e_id
+e_name
+e_phoneno
+e_address
+insert()
+update()
+delete()

Bill
+Bill_id
+Customer_name
+Customer_address
+Customer_Email
+Product_name
+Product_quantity
+Cost_price
+Sell_price
+view()
+insert()
+update()
+delete()

Order
+Product_cattel
+Product_quantity
+Order_time
+Order_price
+view()
+insert()
+update()
+delete()

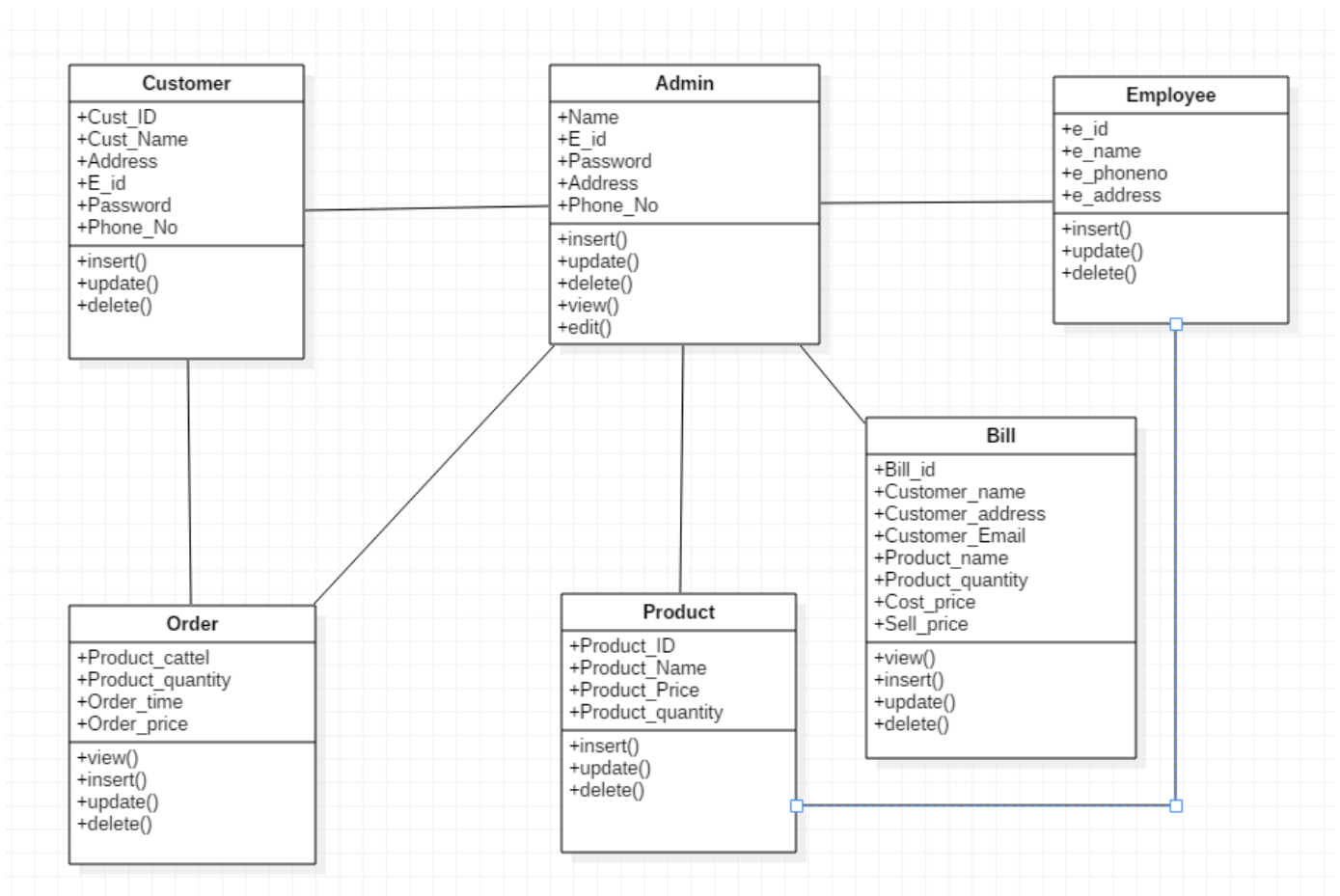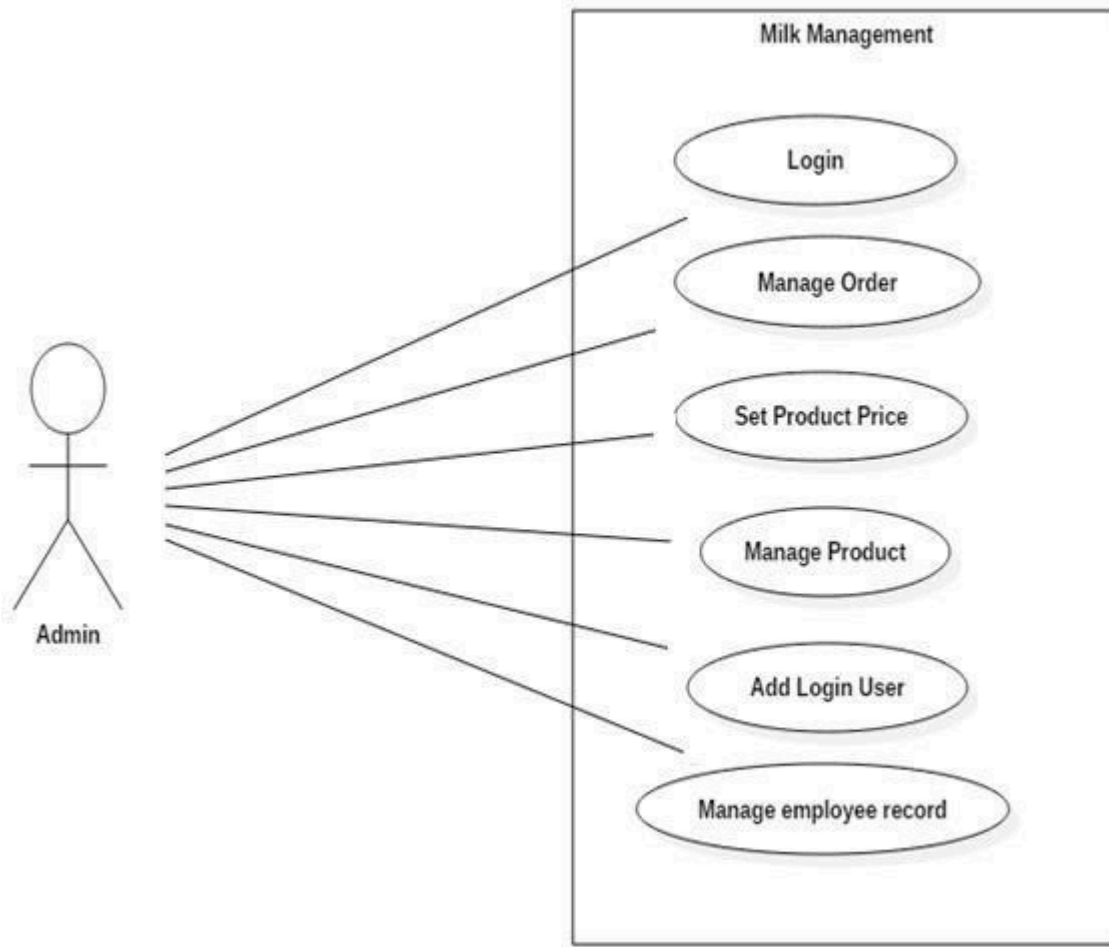| Product |
|---|
| +Product_ID |
| +Product_Name |
| +Product_Price |
| +Product_quantity |
| +insert() |
| +update() |
| +delete() |

Diagram shows how different parts of a business system are connected. Customers, admins, and employees interact with products, orders, and bills. For example, customers place orders for products, and bills are generated for those orders.
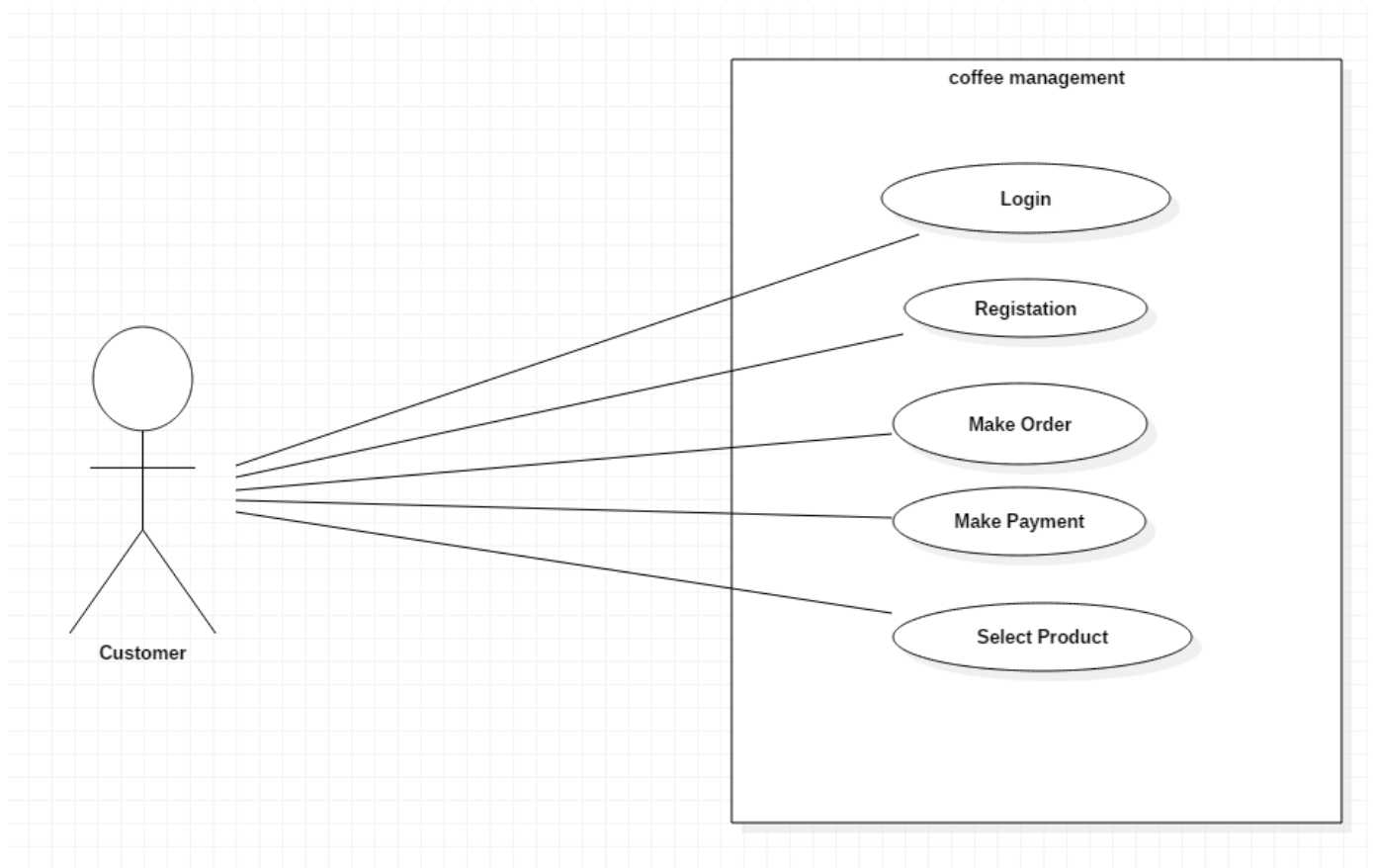
3.1.2 Use Case Diagram:-

3.1.2.1 Usecase diagram for Admin:

The use case diagram shows what an admin can do in the system. The admin can manage orders, set product prices, manage products, add login users, and manage employee records. The admin must first log in to access these functions.
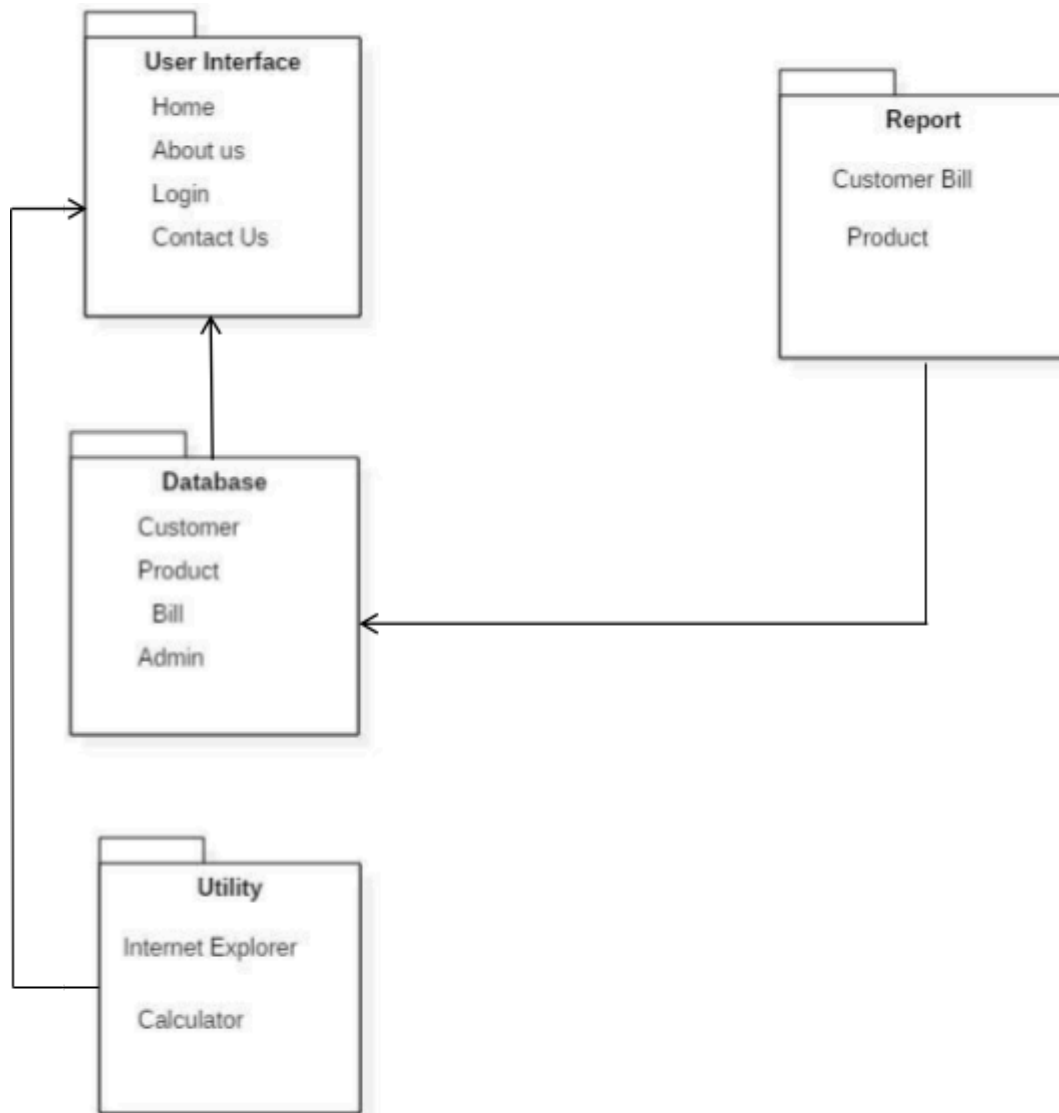
3.1.2.2 Use Case diagram for customer:

The use case diagram shows what a customer can do in the coffee management system. The customer can register, login, select products, make orders, and make payments.

3.2 DATA DICTIONARY:-
### 3.2.1 Package diagram:-

This diagram shows how different parts of the system work together.

1. **User Interface** is like the front door – it's what you see and use, with things like the Home page, About Us info, and the way you log in.

2. **Database** is like a filing cabinet – it stores all the important info, like customer details, product info, bills, and who's in charge (the Admin).

3. **Report** is like the summary page – it pulls out important info from the filing cabinet to make things like customer bills and product lists.

4. **Utility** is like the toolbox – it has handy tools like a web browser (Internet Explorer) and a calculator to help you out.

So basically, the diagram shows how all these parts work together to make the system run smoothly.

3. 2.2 Component diagram:
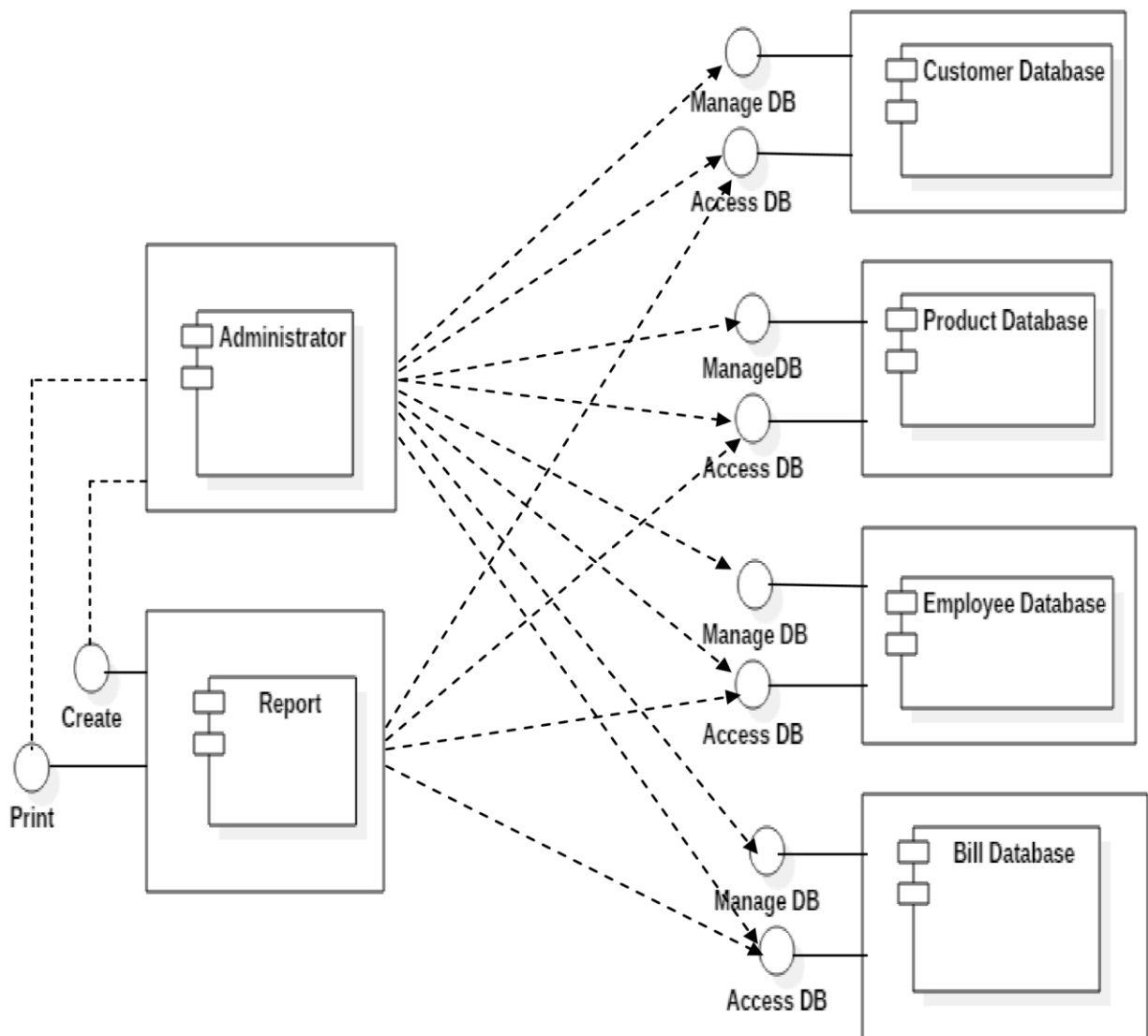
This diagram shows how different parts of a system, like databases and user functions, interact. The "Administrator" has full control over all the databases (Customer, Product, Employee, Bill). The "Report" function can create, print, and view data from the databases, but can't make changes to them. It's like a blueprint of who can do what with the information.

3.3 ER diagram:
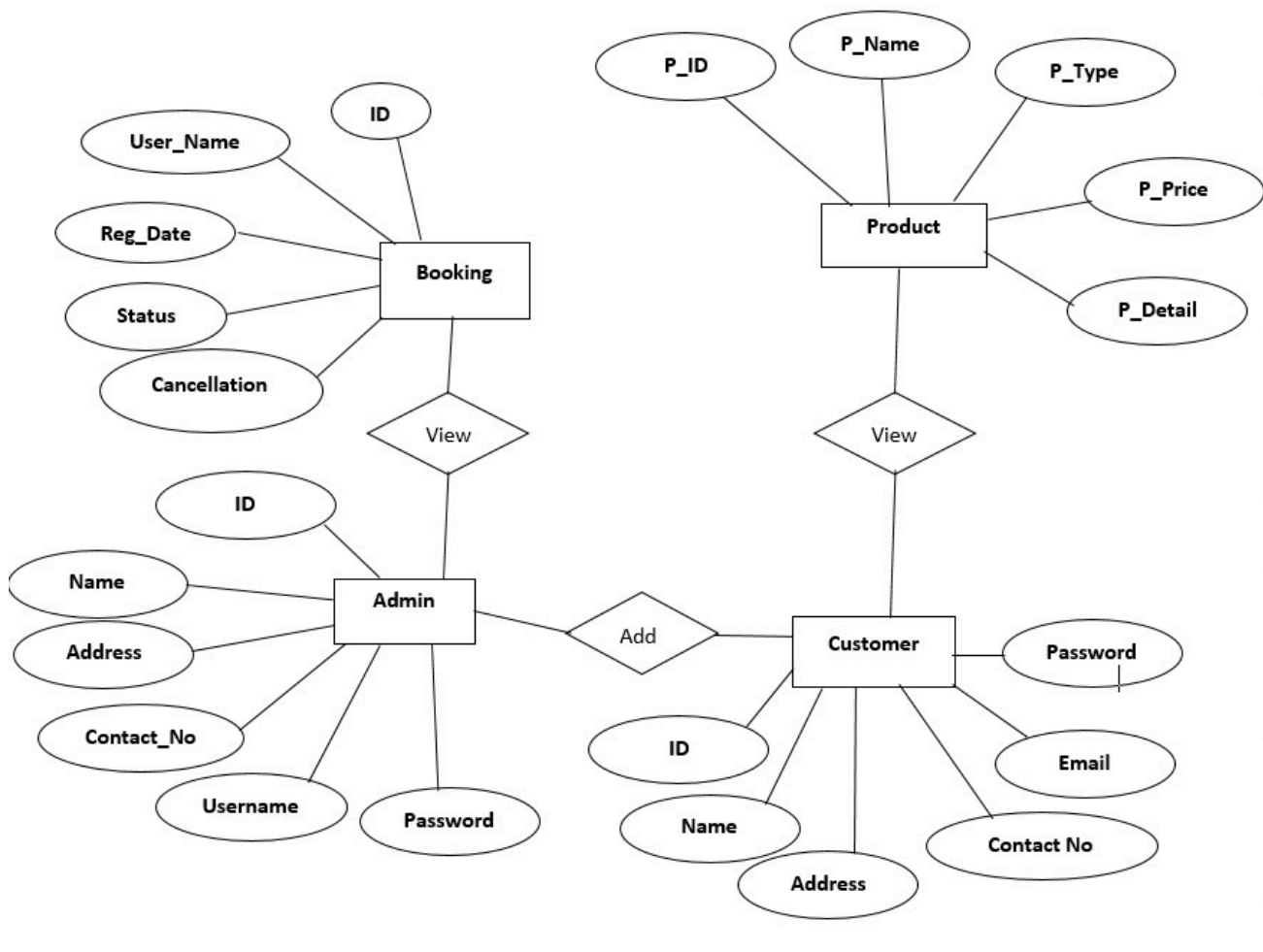
1. This ER diagram represents a system with Admin, Customer, Booking, and Product entities.

2. The Admin manages customers and bookings by adding customer details.

3. Customers can view Products and make Bookings, which have details like status and cancellation.

4. Each entity has specific attributes like ID, Name, Contact No, and Password.

5. The Product entity includes details such as P_ID, P_Name, P_Type, P_Price, and P_Detail.

3.4 : DATA FLOW DIAGRAMS / UML:- 3.4.1 Sequence diagram

3.4.1.1 Availability:
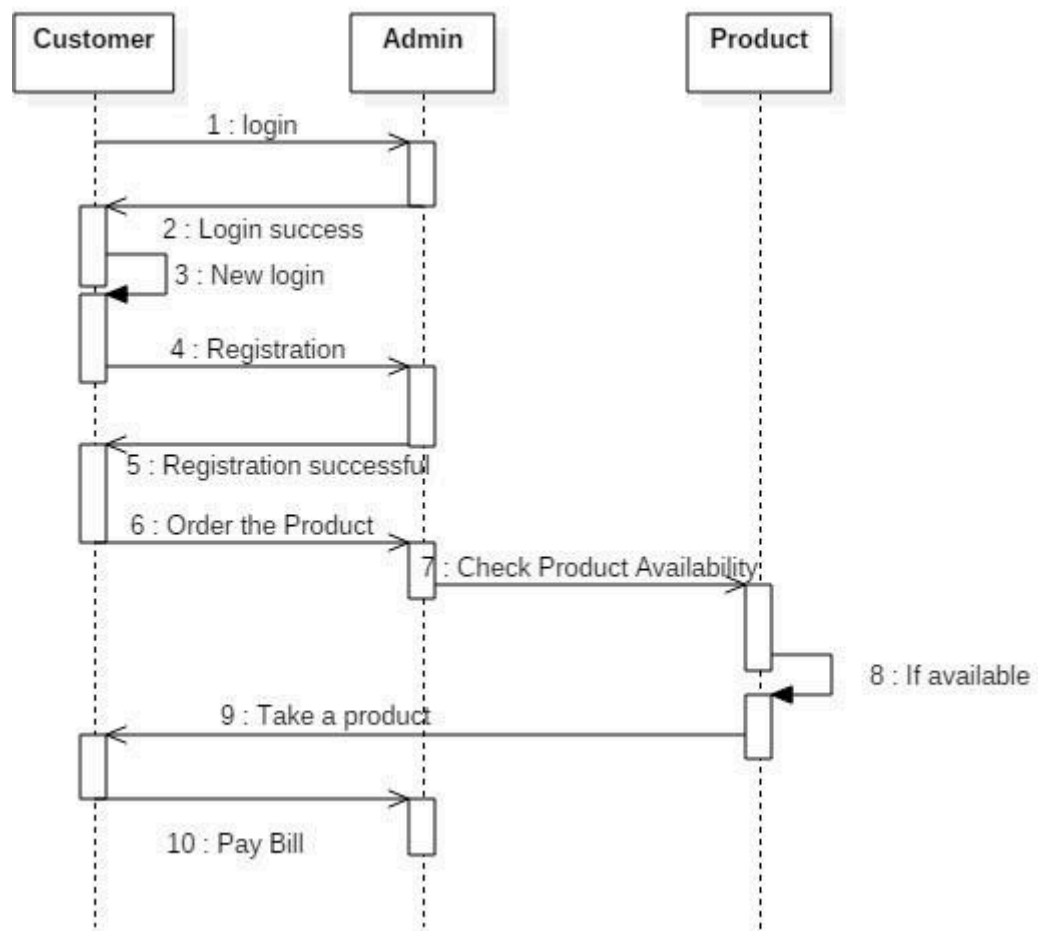
This sequence diagram depicts the process of a customer ordering a product, starting with login or registration.

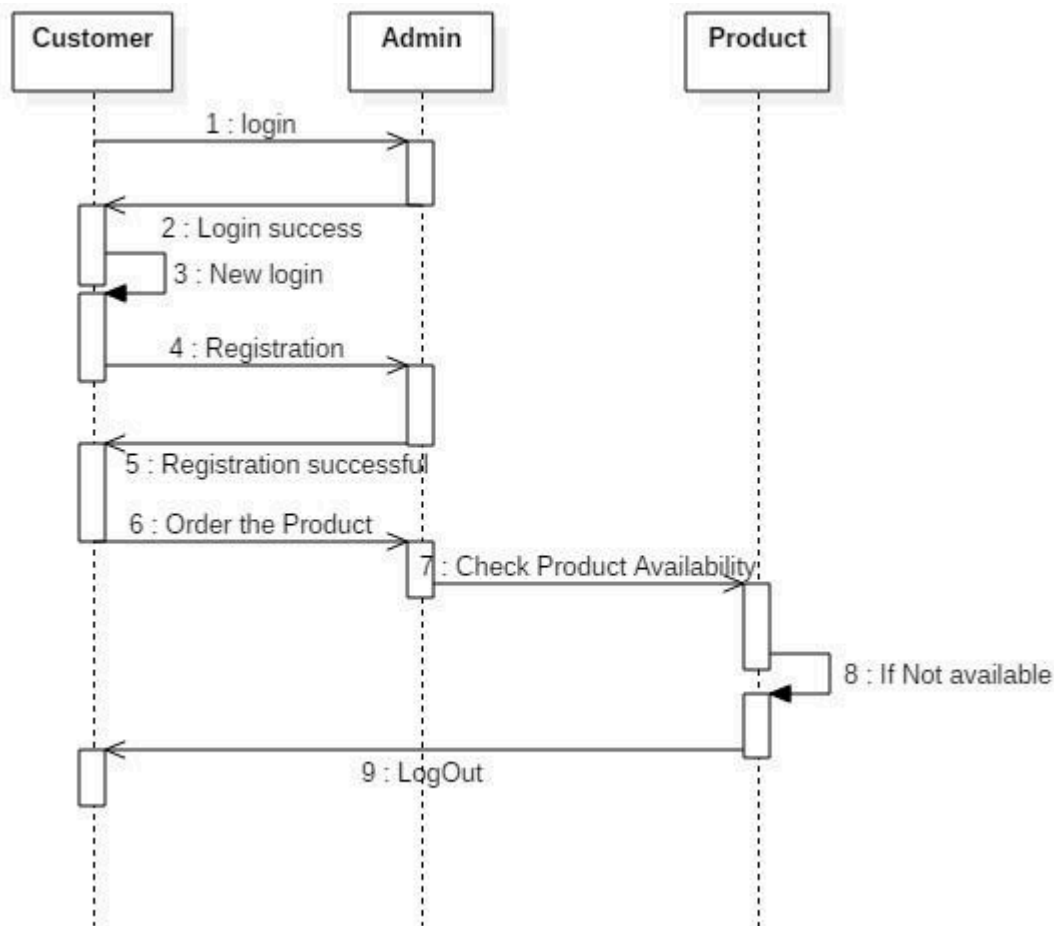The customer interacts with both the Admin (presumably the system) and the Product.

After successful login or registration, the customer places an order.

The Admin checks product availability.

If available, the customer is informed and proceeds to take the product.

Finally, the customer pays the bill, completing the purchase process.

3.4.1.2Not available:

This sequence diagram outlines the scenario when a customer attempts to order a product that is unavailable.

The customer begins by logging in or registering with the Admin system.

After successful authentication, the customer places an order for a specific product.
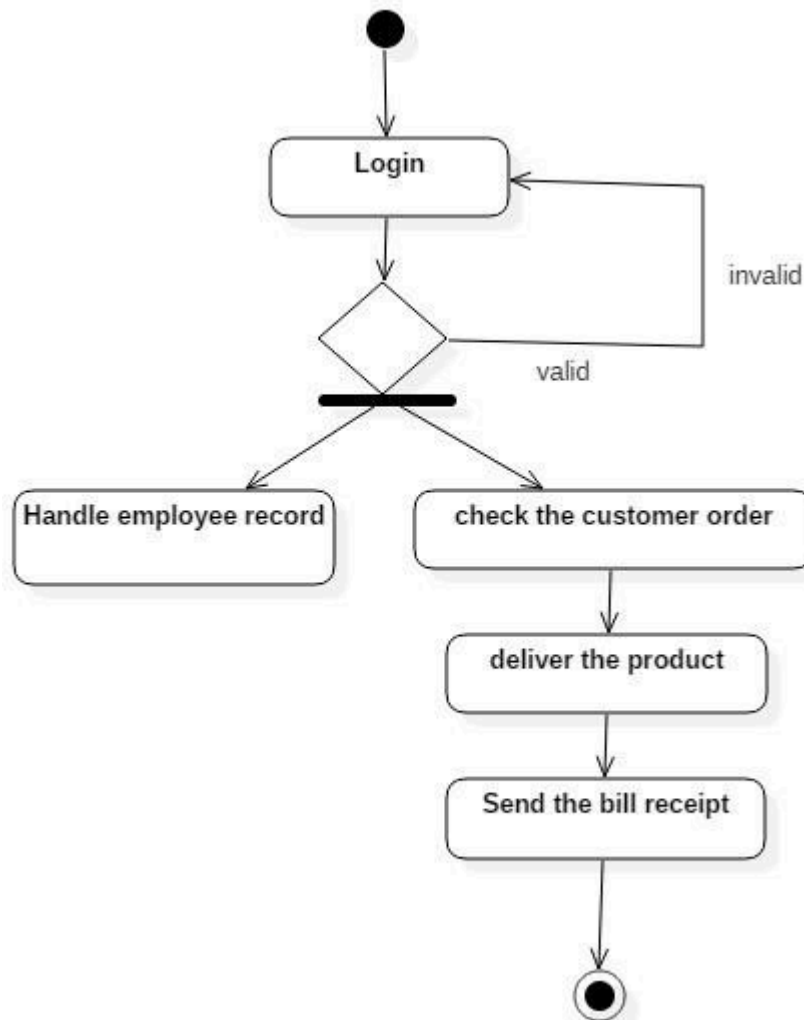
The Admin checks the product's availability with the Product system.

In this case, the Product system returns "Not Available".

Subsequently, the customer logs out, as the order cannot be fulfilled.

3.4.2 Statechart diagram

3.4.2.1 Admin:

This statechart diagram describes the workflow of an Admin user.
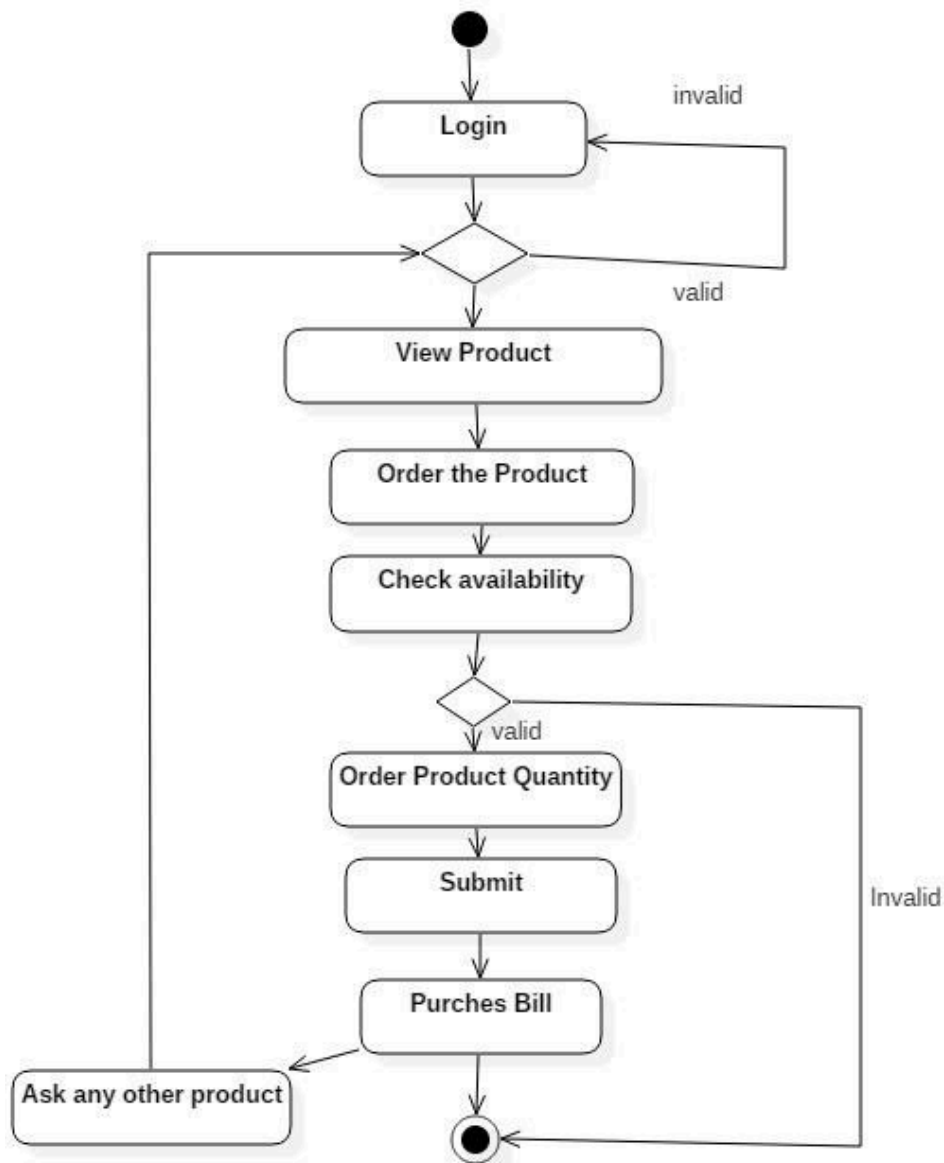
The Admin begins by attempting to log in.

A failed login keeps the Admin in the login state.

A successful login allows the Admin to choose between managing employee records or checking customer orders.

Checking customer orders leads to product delivery and sending a receipt.

The process ends after either task is completed.

3.4.2.2 Customer:

This shows what a customer does on a website.

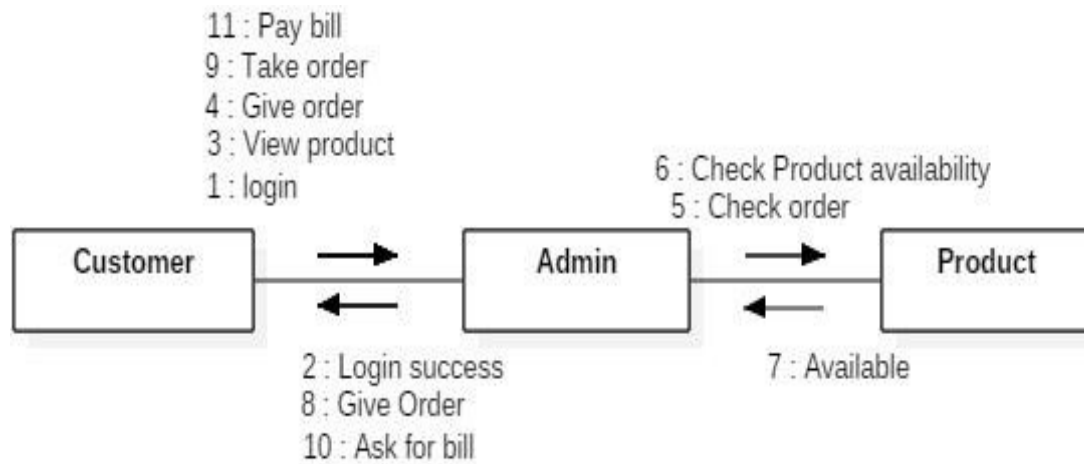First, they try to log in. If it's wrong, they try again.

If they log in correctly, they can look at products, choose one, and see if it's in stock.

If it's in stock, they say how many they want and order it. If the order has a problem, they fix it.

If the order is okay, they pay for it.

Afterward, they can ask about other things to buy, or they can leave the website

3.4.3 Collaboration Diagram:

11 : Pay bill
9 : Take order
4 : Give order
3 : View product
1 : login

6 : Check Product availability
5 : Check order

| Customer | Admin | Product |

2 : Login success
8 : Give Order
10 : Ask for bill

7 : Available

This collaboration diagram shows interactions between a Customer, Admin, and Product.

The Customer starts by logging in.

After successful login, the Customer views a product and places an order.
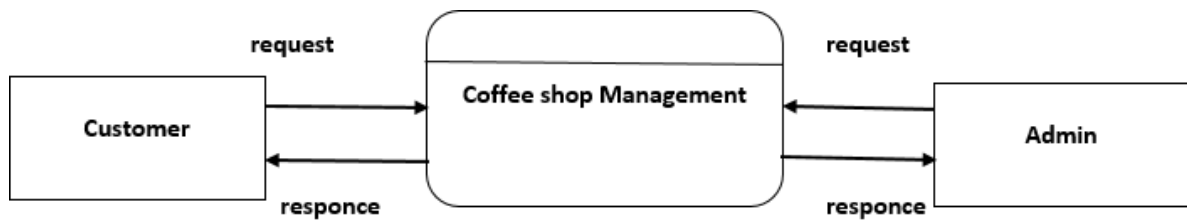
The Admin checks the product's availability.

If available, the Admin gives the order to the Customer.

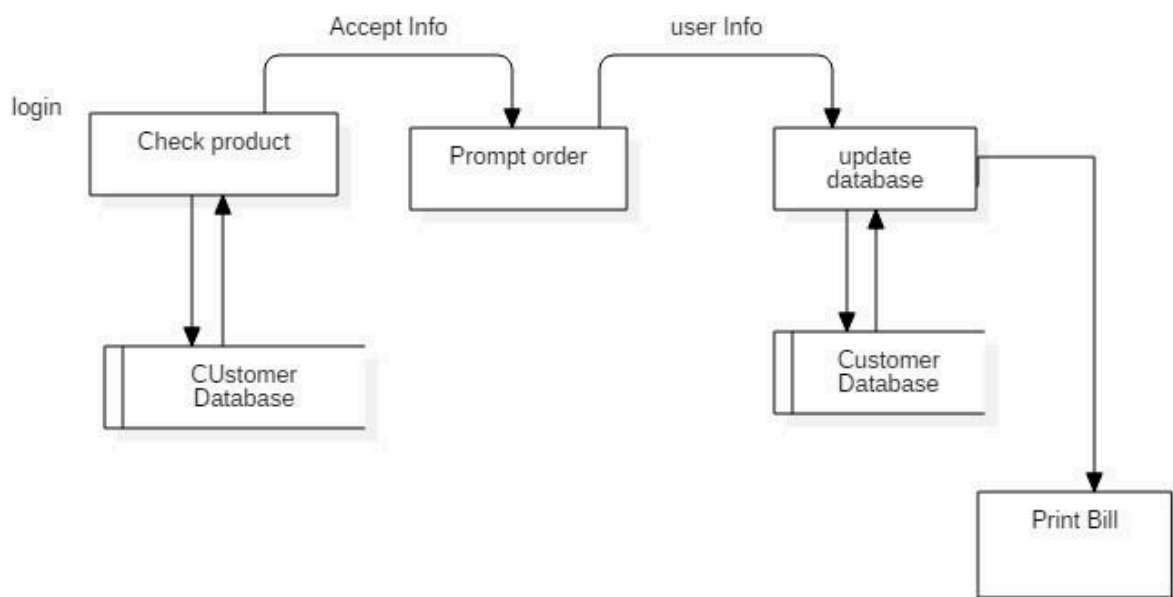The Customer takes the order and pays the bill.

Finally, the Customer may ask for the bill. The diagram highlights the sequence of messages exchanged between the actors.
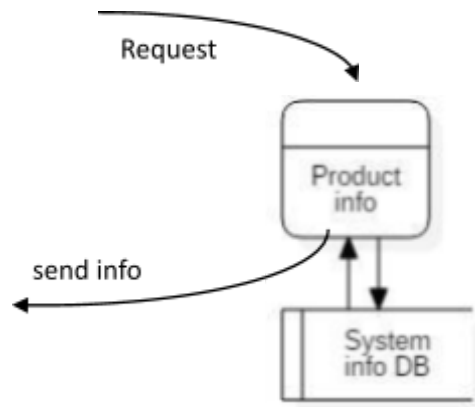
3.4.4 Data Flow

Diagram: Level 0:

Level 1:

Level 2:

Level 0: Context Diagram :

This is the highest level view, showing the entire system as a single process. It shows the "Coffee Shop Management" system interacting with external entities: the Customer and the Admin. Both entities send requests to and receive responses from the system.

Level 1: Overview Diagram :

This level breaks down the system into its major processes. It shows processes like "Check Product," "Prompt Order," "Update Database," and "Print Bill." It also shows data stores like the Customer Database, highlighting the flow of data between processes and these stores.

Level 2: Detailed Diagram :

This level further decomposes a specific process from Level 1, in this case, likely "Check Product." It shows the sub-processes and data flows within that process. It depicts the system requesting and receiving product information, and then sending relevant information to the system's database.

3.4.5 Activity diagram:
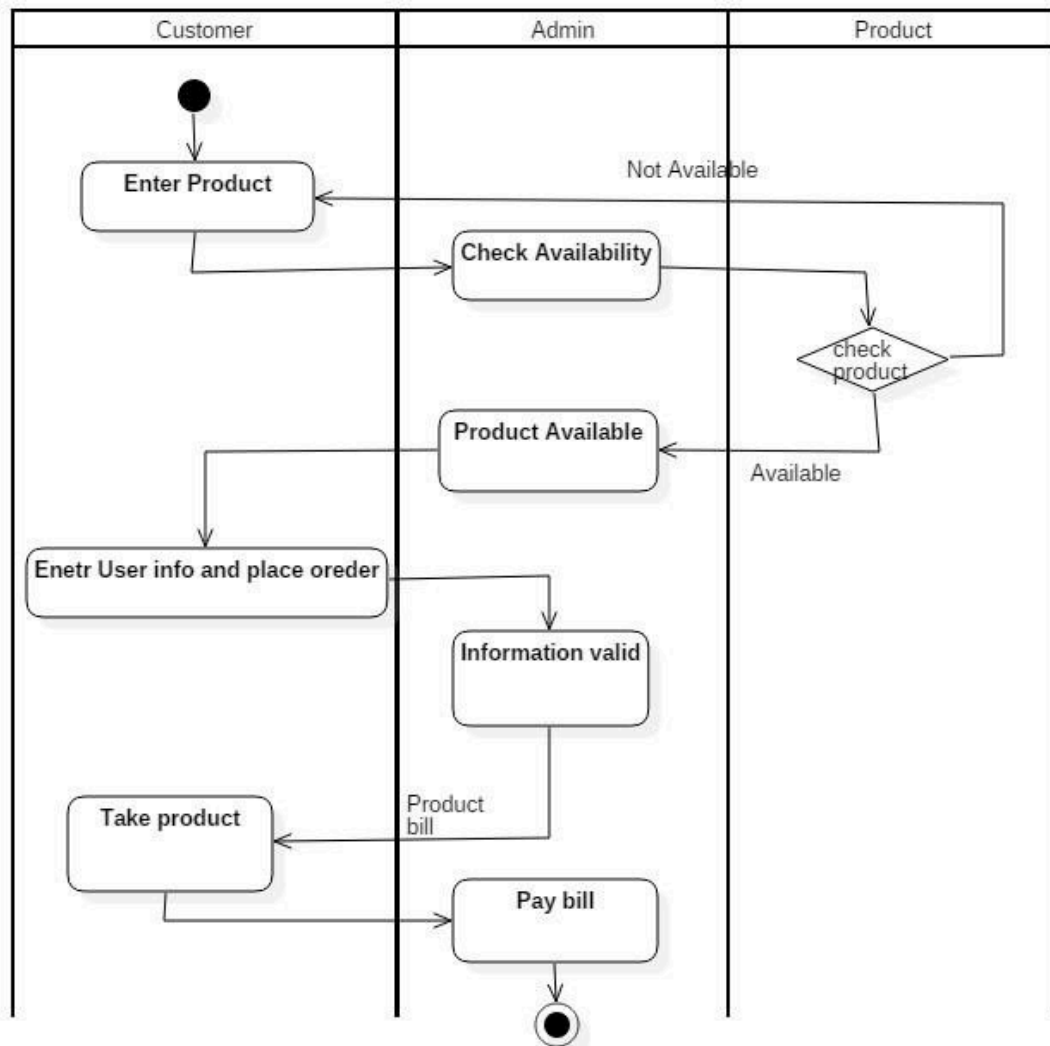
This activity diagram shows the steps involved in a customer placing an order.
The customer starts by entering product details.
The admin checks if the product is available. If not, the process may loop or end.
If available, the customer enters their information and places the order.
The admin validates the order.
Finally, the customer receives the product and pays.

3.4.6 Deployment:
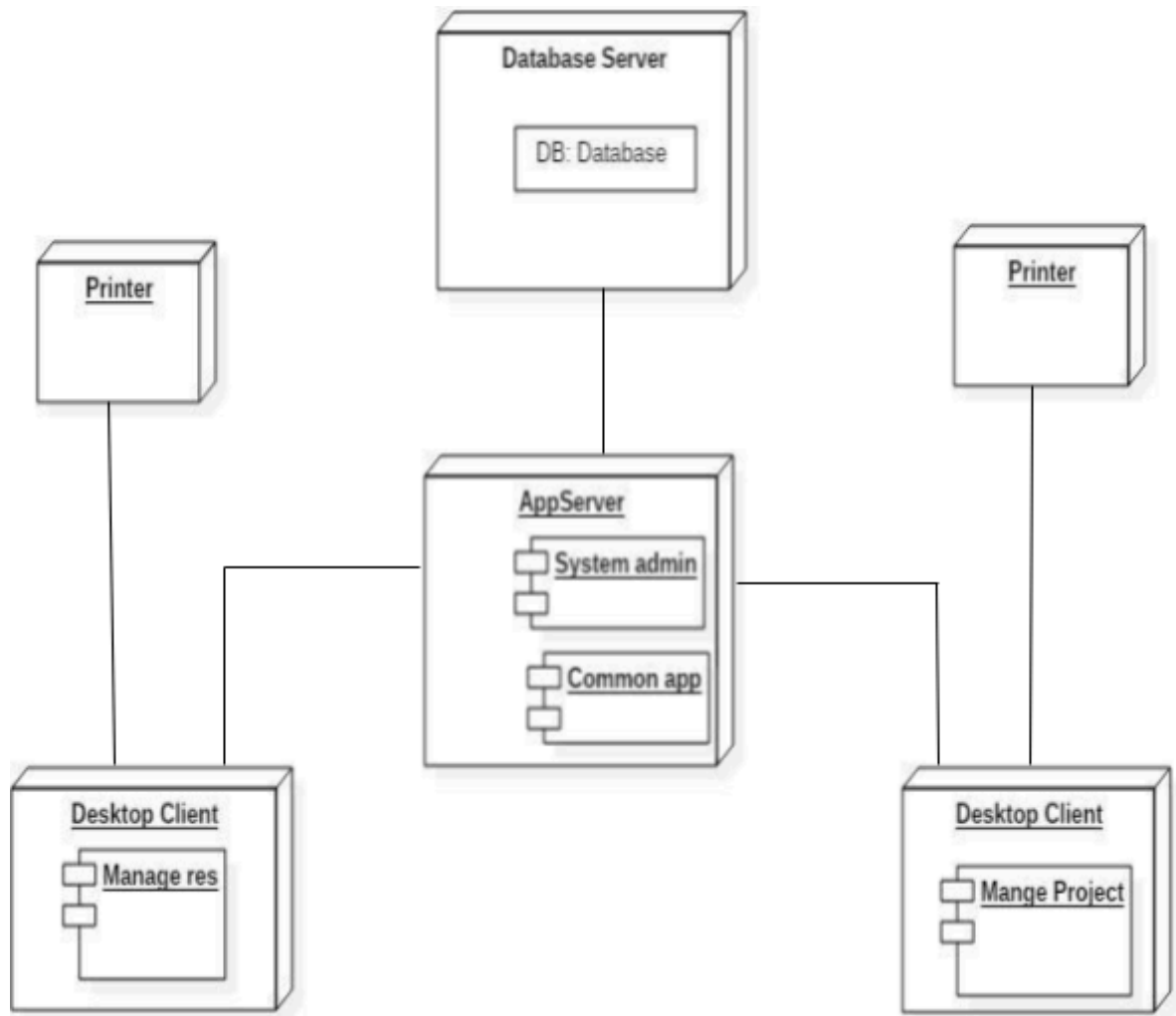
This deployment diagram shows the physical components and their connections for a software system.
It includes a Database Server hosting a database, an AppServer running system administration and common applications, and two Desktop Clients.
Each Desktop Client is associated with a specific function: "Manage resources" and "Manage projects," respectively.
Two Printers are connected to the system, likely accessible by the applications on the AppServer.
The diagram illustrates the hardware and software elements and how they are deployed and linked.
It highlights the physical architecture of the system, including servers, clients, and peripheral devices.

3.4.7 System Flowchart :
3.4.7.1 Admin Flowchart :
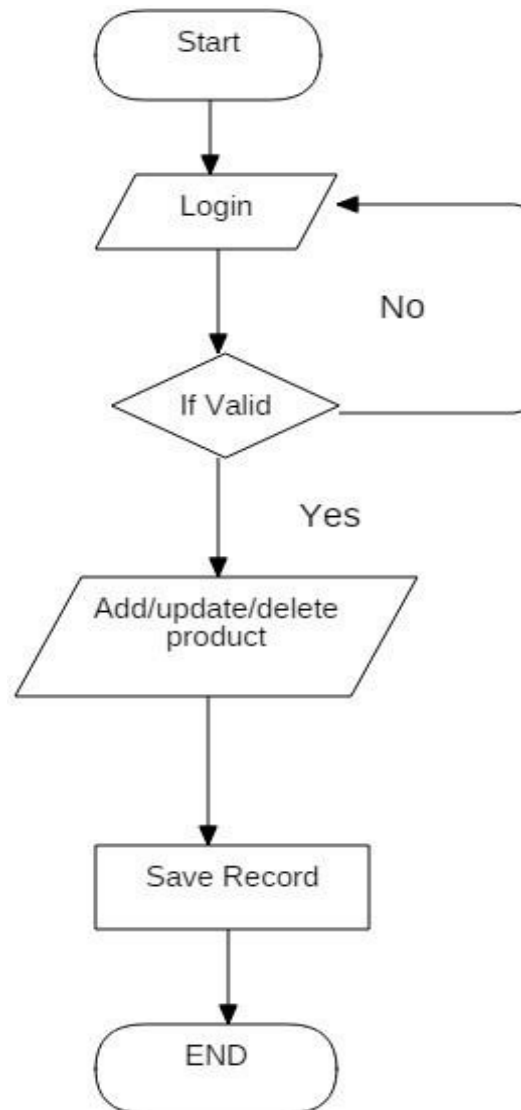
This flowchart describes the process for an admin to manage products in a system.
The process begins with the admin attempting to log in.
If the login is invalid, the admin is looped back to the login step.
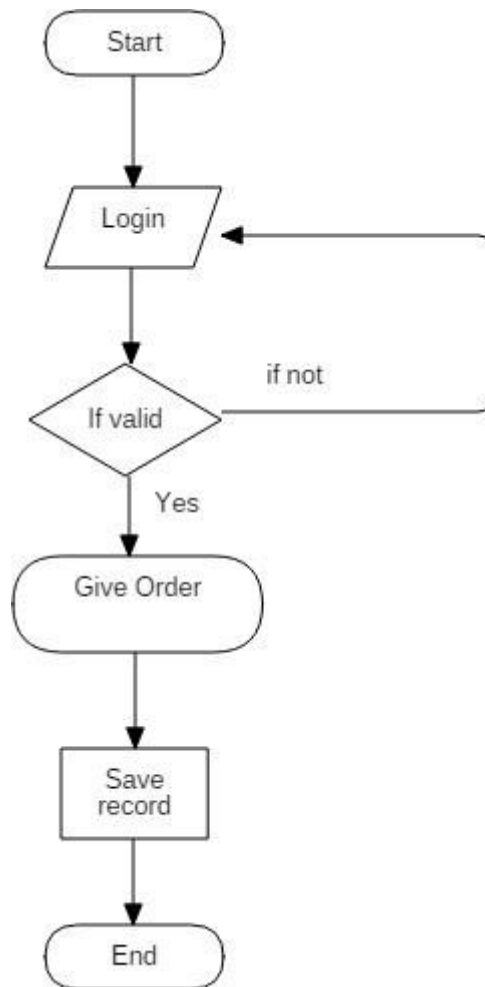Upon successful login, the admin is granted access to product management functions.
The admin can then add, update, or delete product information.
After making changes, the admin saves the record.
The process concludes after the record is saved.

3.4.7.2 User Flow Chart :

This flowchart outlines the steps a user takes to place an order.
The process begins with the user attempting to log in.
If the login is invalid, the user is prompted to log in again.
Upon successful login, the user proceeds to place an order.
The order details are then saved in the system.
After saving the order, the process concludes.
The flowchart clearly shows the login validation step and the subsequent order placement and saving actions.
It depicts a linear flow with a loop for login validation, leading to the final order confirmation.

<u>Implementation and Testing</u>

Testing Approach:-

4.1.1 Unit Testing:-

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

4.1.2 Integration Testing:-

INTEGRATION TESTING is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

Test Cases and Validation:-

Case no	Scenario	Sr. No	Action	Expected output	Actual output	Result
1	Login form	A	User don't enter user name	Message :you can't login without specific login name	Message :you can't login without specific login name	Pass
		B	User don't enter password	Message :you can't login without specific password	Message :you can't login without specific login password	Pass
2	User registration form	A	User enter numeric value in name text field	Message: Enter text character	Message: Enter text character	Pass
		B	User enter character in digit text field	Message: Enter valid number	Message: Enter valid number	pass
		C	User doesn't enter valid email	Message: Enter valid Email	Message: Enter valid Email	pass
3	Customer details	A	User enter numeric value in name text field	Message: Enter only text character	Message: Enter only text character	pass
		B	User enter character in digit text field	Message: Enter valid number	Message: Enter valid number	pass

		C	User doesn't enter valid email	Message: Enter valid Email	Message: Enter valid Email	pass
4	Product detail form	A	User save incomplete information	Message: Enter all field	Message: Enter all field	pass
5	Bill	A	Admin save character in digit text field.	Message: Enter valid number.	Message: Valid number entered by user	pass
		B	Admin save incomplete information.	Message: Enter all fields successfully.	Message: All fields entered by admin.	pass

References:

https://www.google.com/search?q=coffee+shop+management+system+project&rlz=1C1CHBD_e
nIN833IN833&oq=coffee+shop+manageme&aqs=chrome.0.0i19j69i57j0i19l6.14844j0j7&sourcid=c
hrome&ie=UTF-8

https://www.scribd.com/document/435834741/Coffee-shop-management-system-
https://www.freeprojectz.com/project-source-code-database-download/coffee-shop-managem
ent- system-project