COVIPEDIA

A PROJECT REPORT

Submitted by

Yogita Sharma (196400316132)

Ayushi Rathod (196400316121)

Devam Pathak (196400316109)

Parth Pawar (196400316111)

In partial fulfilment for the award of the

DIPLOMA ENGINEERING

in

Information Technology Department

R.C. Technical Institute, Ahmedabad

Gujarat Technological University, Ahmedabad

September, 2021

R.C. TECHNICAL INSTITUTE

Opposite Gujarat High Court, S.G. Highway, Sola,

Ahmedabad-380060

DEPARTMENT OF INFORMATION TECHNOLOGY

2021

CERTIFICATE

This is to certify that

Yogita Sharma(196400316132),

Ayushi Rathore(196400316121),

Devam Pathak(196400316109) and

Parth Pawar(196400316111)

has satisfactorily completed the project.

Work entitled COVIPEDIA assigned as UDP in partial fulfillment of Project-1 (3351605) during

5th semester of Academic Year 2021-22.

Internal Guide,	**H.O.D in IT Department**
Hirav J. Shah	**Mr. Manoj P Parmar**
Lecturer in IT Department	

ACKNOWLEDGEMENT

- We are pleased to "**COVIPEDIA**" project and take this opportunity to express our profound gratitude to all those people who helped us in completion of this project.

- We thank our college for providing us with supportive faculties that helped us to complete and present this project. We would also like to thank the staff members & lab-assistants for permitting us to use computers in the lab as and when required.

- We express our deepest gratitude toward our project guide for her valuable and timely advice during the various phases in our project. We would also like to thank her for providing us with all proper facilities and support as the project co-coordinator. We would like to thank her for support, patience and faith in our capabilities and for giving us flexibility in terms of working and reporting schedules.

- We would like to thank all our friends for their smiles and friendship making the college life enjoyable and memorable and family members who always stood beside us and provided the utmost important moral support. We would like to thank everyone who has helped us directly or indirectly in our project.

ABSTRACT

- Today the pandemic corona virus is going ahead and many people face more problem to prevent from this virus. So we're created a website which help people to make some knowledge about corona and help to take some prevention toward the virus. Our website give detail about corona like Wikipedia so that why we name it COVIPEDIA. In website we make some attractive part like blogs, COVI-store. Which show's some interaction toward viewer.

List of Figure

CHAPTER – 1: INTRODUCTION

It is website on covid-19. To give people all kind information on Covid-19 in way of explanation for better understanding. So instead of reading whole article and find a point, one can easily find what they need in this format. We will also give information on all about how vaccine, mask, social distancing, etc. can help reduce chances of getting infected. Customer well also be able to view product for corona infection prevention. Customer will be able to place order for that and also able to pay online.

1.1 Project Summary

Project Name: **COVIPEDIA**

COVIPEDIA is a project "which help people to take awareness from the namely virus corona". It's like Wikipedia which gave information about all over world but in this project only corona detail be provided like how to take prevention towards its.

Corona is a virus which cover all world by dangerously. In which people compulsory to know what is mask, PPKIT, face shield, Oximeter, Infrared thermometers, etc.

In this project we put some interested part like blogs, COVISHOP.

Thing which are included in our project like:

About: Covid -

1. Date of pandemic's start.
2. Origin Country.
3. Speed of spreading.
4. Ways of spreading.
5. Symptoms.

Vaccine -

1. Which company manufactured vaccines.
2. Time to make it, based on brands, release data.
3. No. of dose.

Public awareness -

1. Works of mask.
2. Need of social distances.
3. Effect of sanitisation and hygiene.

Blogspot -

Blogs based on covid-19 for latest news and updates.

Cases in world -

Links of latest news and cases information.

Video -

How to maintain self- awareness towards corona.

Quiz

Post Query

Buy or sell products related to covid

- **Functionality of the user**

1) Admin

- login
- manage users
- manage category
- manage subcategory
- manage product
- manage user order
- mange feedback
- manage corona updates

- manage advertisements
- manage blog
- view query
    - answer to query
- generate report

2) Customer

- login or registration
- Edit profile
- view category
- view subcategory
- view products
- place order
    - view order history
- payment
- view corona details
- Create or Read blogs
- post query
    - view query response
- give feedback

3) Visitor

- view category
- view subcategory
- view products
- view corona details
- view blogs
- view queries
- view feedback

1.2 Purpose

To develop this type of Platform our main motive is how to make some awareness in people toward corona virus.

Because people take it's lightly and from that many problems were face by people now a days.

So for that we created this website which give some kind of information about corona.

1.3 Scope

Following scope:

1. Registration:- In registration module, we will make an user page which allows them to register there-self in our website.
2. Login:- Login module allows user to log them self in our web site by using the registration id given to them while registration.
3. About us:- this page deliver information about COVIPEDIA group and some basic information covid 19
4. COVI-store:- This is an dynamic webpage which allows register user to purchase covid related item.
5. Blog stop:- This page allows register user to create an added there blogs in our website. And view and comment other blogs.

1.4 Technology Review

Front End: JSX (java script react)

Back End: Node JS

- **JSX**

- It is called JSX, and it is a syntax extension to JavaScript. We recommend using it with React to describe what the UI should look like. JSX may remind you of a template language, but it comes with the full power of JavaScript. JSX produces React "elements". We will explore rendering them to the DOM in the next section. Below, you can find the

basics of JSX necessary to get you started. React embraces the fact that rendering logic is inherently coupled with other UI logic: how events are handled, how the state changes over time, and how the data is prepared for display.

- Instead of artificially separating *technologies* by putting markup and logic in separate files, React separates *concerns* with loosely coupled units called "components" that contain both. We will come back to components in a further section, but if you're not yet comfortable putting markup in JS, this talk might convince you otherwise.
- React doesn't require using JSX, but most people find it helpful as a visual aid when working with UI inside the JavaScript code. It also allows React to show more useful error and warning messages.

➢ Node JS

- Nodejs is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux. Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent**.**

Features of Node.js

Following are some of the important features that make Node.js the first choice of software architects.

- **Asynchronous and Event Driven** − All APIs of Node.js library are asynchronous, that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.
- **Very Fast** − Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution.
- **Single Threaded but Highly Scalable** − Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.
- **No Buffering** − Node.js applications never buffer any data. These applications simply output the data in chunks.
- **License** − Node.js is released under the MIT license.

CHAPTER – 2 : PROJECT MANAGEMENT

2.1. Project Planning & Schedule

2.1.1. Project Development and Approach

- How to choose the right approach for a project is a large topic. The methodology We have choose depends on many thing, including the structure and location of the
  project team, the technologies being used on the project, and the degree to which collaboration is a part of the company's culture.
- The approach of our project is to provide the information to the peoples contains the
  vaccination data and its criteria, and we're selling providing the products i.e. sanitizer, mask, face shield, pp kit, etc. which help during covid.
- We are making our project using java script in front end jsx and node js in back end.

2.1.2. Project plan

- Project planning is one of the major tasks that are performed during the development of the project.

Using project planning, the task of finding the size of the project is done and with that total amount of

time and cost required for the project is calculated.

- The approach to developing the software system should follow some systematic way i.e. Software

Development Life Cycle. Using the upper-level analysis and the environment of the project, which

lifecycle model would fit properly for this project was judged. After deciding the proper software

development lifecycle model, the development of this project according to the model was done.

- This project also some basic parts like cart, shopping, covid live cases, blog.
- In cart, customer can add item which they want to purchase and also can increase quantity.
- In shopping, the customer can purchase the items using category and sub-category the category contains sections like what they exactly want i.e. sanitizer, mask, pp kit,
  face shield, etc. the sub-category contains for whom they want product i.e. man woman, children.

2.1.3. Milestones and Deliverables

- **Milestones**

  - ✓ Our first milestone is to understand the project sections which required for our covidpedia project.
  - ✓ We determined all the segments of our project definition and carried out different complexity by doing feasibility study.
  - ✓ After searching all the factors and knowing project definition we gathered project specification, it's requirements and features.
  - ✓ We analyzed all the specifications and requirements with taken care of every possible ways and all the loop holes.
  - ✓ We planned our project according to project requirements, we divided whole project in small segments, and we convert those
    segments into sprints and based on that we started working on our project.
  - ✓ After requirement gathering, analyzing and planning we started designing our project modules in which we created user interface according
    to wire frames which provides the information regarding covid 19, shopping of necessary products, blogspot.
  - We provide video which provides precautions regarded to covid 19.

- **Deliverables**

  - ✓ Shopping
  - ✓ Blogspot
  - ✓ Data Dictionary/Database
  - ✓ User Pages (UI)
  - ✓ Provider Pages (UI)

2.3 Estimation

Project estimation can estimate project size, cost and required effort. It can also form the basis for resource planning and scheduling. Estimation of resources, cost, and schedule for a software engineering effort requires experience, access to good historical information, and the courage to commit to quantitative predictions when qualitative information is all that exists. An error in estimation carries inherent risk and this risk leads to uncertainly. In a large program, cost estimation error can make the difference between profit and loss.

- **Registration**
  - ✓ An action register is a central place where the project manager can record all of the different actions associated with the project, its owners, due dates and other important information.
  - ✓ With an action register, it is simpler to keep track of all actions needed to keep the project running smoothly.
  - ✓ We are completing our registration portion in 20 days.
- **Log-In**
  - ✓ A logging is the entering of identifier information into a system by user in order to access that
    system.
  - ✓ A logging generally requires the user to enter two pieces of information, first user name and
    then password
    and reset password.
  - ✓ It generally requires 15 days to done this portion.
- **About**
  - ✓ The approach of our project is to provide the information to peoples contains the vaccination data
    and its criteria, and we're selling the products i.e. sanitizer, mask, face shield, ppt kit, etc. which help
    during covid 19.
  - ✓ We are also providing feedback, blogspot, store.
  - ✓ It requires 7 days to complete this portion.
- **Shop**

- ✓ It ahs all manufacturing in one location. All material and information has to come to this location, and the finished
  Product is then completed at this location.
- ✓ This approach is most often used very large and difficult to move products in small quantity.
- ✓ This requires 2 months/60 days.

- **Blogspot**
  - ✓ Blogger is a free publishing platform, while blogspot is a free domain service provider.
  - ✓ A blog is often write about their opinions and thoughts.
  - ✓ Generally 1 months requires for it.

CHAPTER - 3 : SYSTEM REQUIREMENT STUDY

3.1. User characteristic

- Admin

  - Admin will manage data of users, she/he will also maintain application. She/he will remove inactive accounts.
  - Admin can login.
  - Login: Admin can login for authentication.
  - Manage users: Admin can manage user for user list.
  - Manage category: Admin can manage category for category list.
  - Manage subcategory: Admin can manage subcategory for subcategory list.
  - Manage products: Admin can manage products for product list.
  - Manage users order: Admin can manage user order for accept/Reject order.
  - Manage feedback: Admin can manage feedback for get feedback list.
  - Manage corona updates: Admin can manage corona updates for corona updates list.
  - Manage advertisements: Admin can manage advertisements.
  - Manage blog: Admin can manage blog for blog list.
  - View query: Admin can view query for query status.
  - Generate report: Admin can generate report for report generate.

- Customer

  - Login: Customer can login for authentication.
  - View category: Customer can view category for category list.
  - View subcategory: Customer can view subcategory for subcategory list.
  - View product: Customer can view product for product list.
  - Place order: Customer can place order for confirmation order.
  - Payment: Customer can payment for payment successfully.
  - Give feedback: Customer can give feedback for updates feedback.
  - View corona details: Customer can view corona details for corona details list.
  - Create or read blogs: Customer can create or read blogs for blogs details.
  - Edit profile: Customer can edit profile for success profile edit.
  - Post query: Customer can post query for query list.

- Visitor –

  - View category: Visitor can view category for category list.
  - View subcategory: Visitor can view subcategory for subcategory list.
  - View product: Visitor can view product for product list.
  - View feedback: Visitor can view feedback for feedback list.
  - View corona details: Visitor can view corona details for corona details list.
  - View blogs: Visitor can view blogs for blogs details.
  - View query: Visitor can view query for query list.

3.2. Hardware & Software Characteristics

3.2.1. Development Tools & Technology

Tools & Technology	
Front-End	JSX
Back-End	Node JS
Web Server	Apace

3.2.2. Hardware Requirement

Hardware Requirement	
Memory	512MB
Hard Disk	10 GB
Monitor	Standard
Key Board	Standard
Mouse	Standard

3.2.3. Software Requirement

Software Requirement	
Frame Work	React Node
Front End	JSX
Back End	Node JS
Design Tools	HTML-5, CSS

CHAPTER – 4 : SYSTEM DESIGN AND ANALYSIS

4.1:- Study of Current System:

Existing System -

Customer does not have idea about new product available in market to prevent

corona infection. Currently customer need to find all Covid - 19 basic information in details on different media and interface. Currently all information need to search manually like.

1) Covid-19 -

1. Date of pandemic's start.
2. Origin Country.
3. Speed of spreading.
4. Ways of spreading.
5. Symptoms.

2) Genetic structure -

-Genetic design of Covid-19.

3) Vaccine -

1. Which company manufactured vaccines.
2. Time to make it, based on brands, release data.
3. No. of dose.

4.2 :-Problem and weakness of current system

Disadvantages of Existing System -

- It is time consuming to search for all information on different medium.

- Customer does not have true information in simple form.

- Customer need to visit shop to buy different products

4.3:- Requirement of New System.

Using our web site customer can view all different product available in our system. Customer can view feedback about quality of that product and also able to place order for

that. All the details about corona available in our system which can be access by customer from anywhere and anytime. Modules of our system are:

1) Public awareness -

1. Works of mask.

2. Need of social distances.

3. Effect of sanitisation and hygiene.

2) Blogspot -

Blogs based on covid-19 for latest news and updates.

3) How Covid-19 kills -

Detailed description on Covid-19 effect on human bodies.

4) Cases in world -

Links of latest news and cases information.

5) Video -

How to maintain hygiene

6) Quiz

7) Post Query

8) Buy or sell products related to covid.

4.4 Feasibility Study:-

❖Economic Feasibility

➢ The System should be time effective because of User Interaction.

➢ The result obtained contains minimum errors and are highly accurate as the data is required

. ❖Technical feasibility

➢ It will give the fast response.

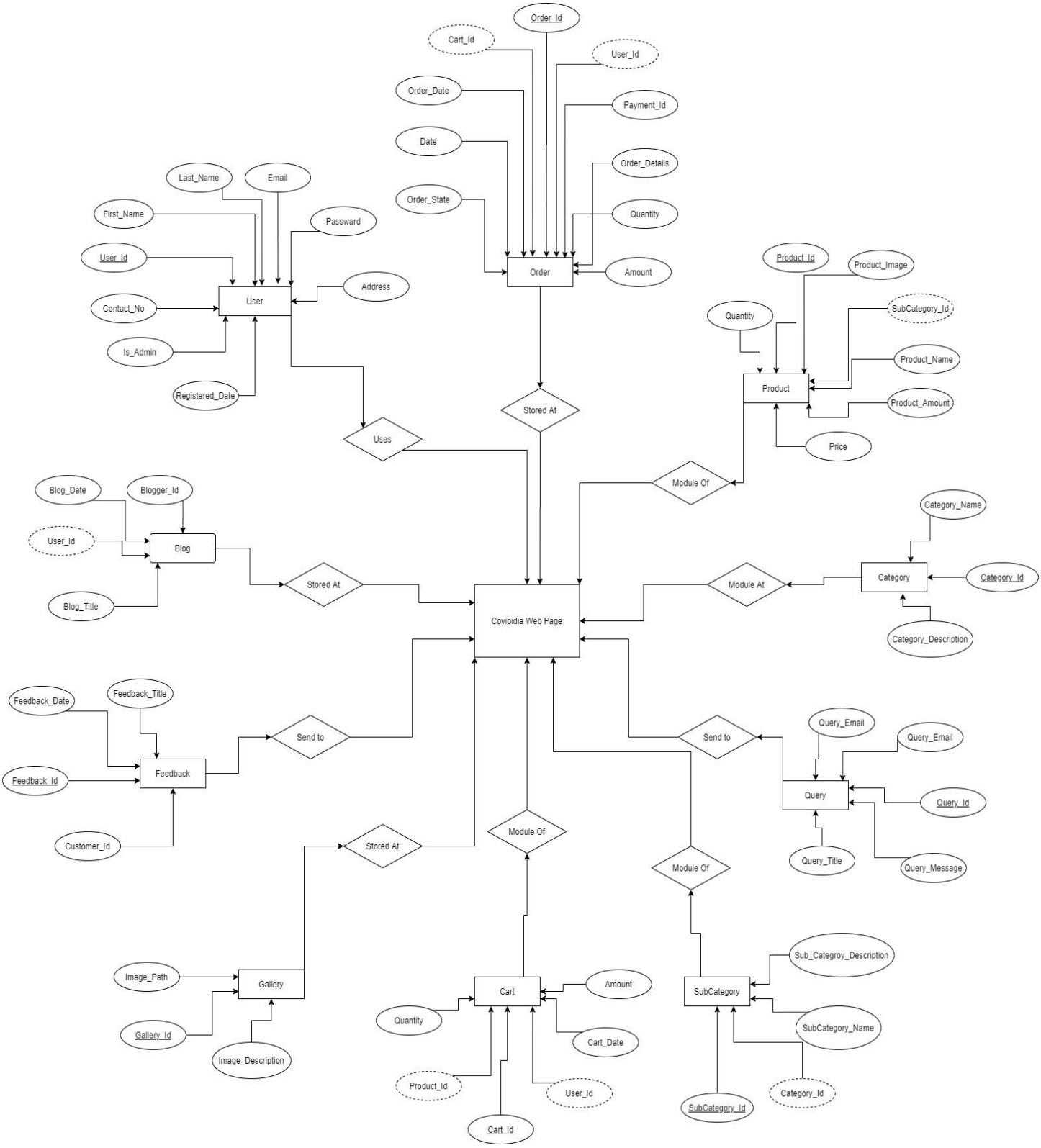
➢ It requires minimum dual core processor, 512 of RAM and 10 GB Storage.

➢ Of storage space.

➢ It should give expense report of user. So, it is easy to get a system of this kind of configuration. From that's on I can say that my system is technically feasible.

4.5 Data Modelling:-

4.5.1 E-R Diagram:-



ER Diagram –

1) **COVIPEDIA** :

This is the main module in ER Diagram

1)**User_Id**:

1 key attribute and 8 attributes

Following are the attributes

1. User_id (primary key):
   This field stores Unique Identification of user.
2. First_name:
   This field stores first name of user.
3. Last_name:
   this field store last name of user.
4. Email:
   This field store email of user.
5. Password:
   this field store password of user.
6. Address:
   this field store address of user.
7. Contact_no.:
   this field store contact no. of user.
8. Is_admin:
   this field store Is admin of user.
9. Register_date:
   this field store register date of user.

2) **Category:**

1 key attributes and 2 attributes

Following are the attributes

1.Category id:

This field stores Category Identification of category

2. category name:

This field stores Category name of category

3. category description:

This field stores Category description of category

3)sub-category:

1 key attribute, 1 key attribute and 2 attributes

1.SubCategory_id:

This field stores SubCategory Identification of subcategory.

2.Category_id:

This field stores Category Identification of subcategory.

3.Subcategory_name:

This field stores SubCategory name of subcategory.

4.Subcategory_description:

This field stores SubCategory description of subcategory.

4)Products:

1 key attribute, 1 key attribute and 5 attributes

1.Product_id:

This field stores Product Identification of product.

2.Product _name:

This field stores Product name of product.

3.Sub_category id:

This field stores Product category Identification of product.

4.Price:

This field stores price of product.

5.Quantity:
This field stores quantity of product.

6.Product_amount:
This field stores Product amount of product.

7.Product_image:
This field stores Product image of product.

5) Cart-

1 Foreign key , 1 Primary key and 3 attributes

Following are the attributes

1. User_Id -
   This field stores user identification of cart.
2. Cart_Id-
   This field stores cart identification of cart.
3. Quantity-
   This field stores quantity of cart.
4. Product_Id-

   This field stores product identification of cart.
5. Amount- This field stores amount of cart.
6. Cart_Date-
   This field stores cart date of cart.

6) Order-

2 Foreign key, 1 Primary key and 7 attributes

Following are the attributes

1. Order_Id –
   This field stores order identification of order.
2. Oder_Date–
   This field stores order date of order.

3. Cart_Id –
   This field stores cart identification of order.
4. User_Id-
   This field stores user identification of order.
5. Date-
   This field stores date of order.
6. Order_Status-
   This field stores order status of order.
7. Payment_Status-
   This field stores payment status of order.
8. Order_Details –
   This field stores order details of order.
9. Quantity-
   This field stores quantity of order.
10. Amount-
    This field stores amount of order.

7) Gallery:-

It has 1 key attribute and 2 normal attribute.

Following are attributes.

1. Gallery_Id.(Primary Key)
   - This field stores Gallery Identification of gallery.
2. Image_Path
   - This field stores image path of gallery.
3. Image_Description
   - This field stores Image description of gallery.

8) **Feedback:**

1 key attribute and 3 attributes

1. Feedback_Id:
   This field stores feedback identification of feedback.
2. Feedback_Date :

This field store feedback date of feedback.

3. Feedback_Title:
   This field stores feedback title of feedback.
4. Customer_Id:
   This field stores customer identification of feedback.

9) **Blogspot:**

1 foreign key and 3 attributes

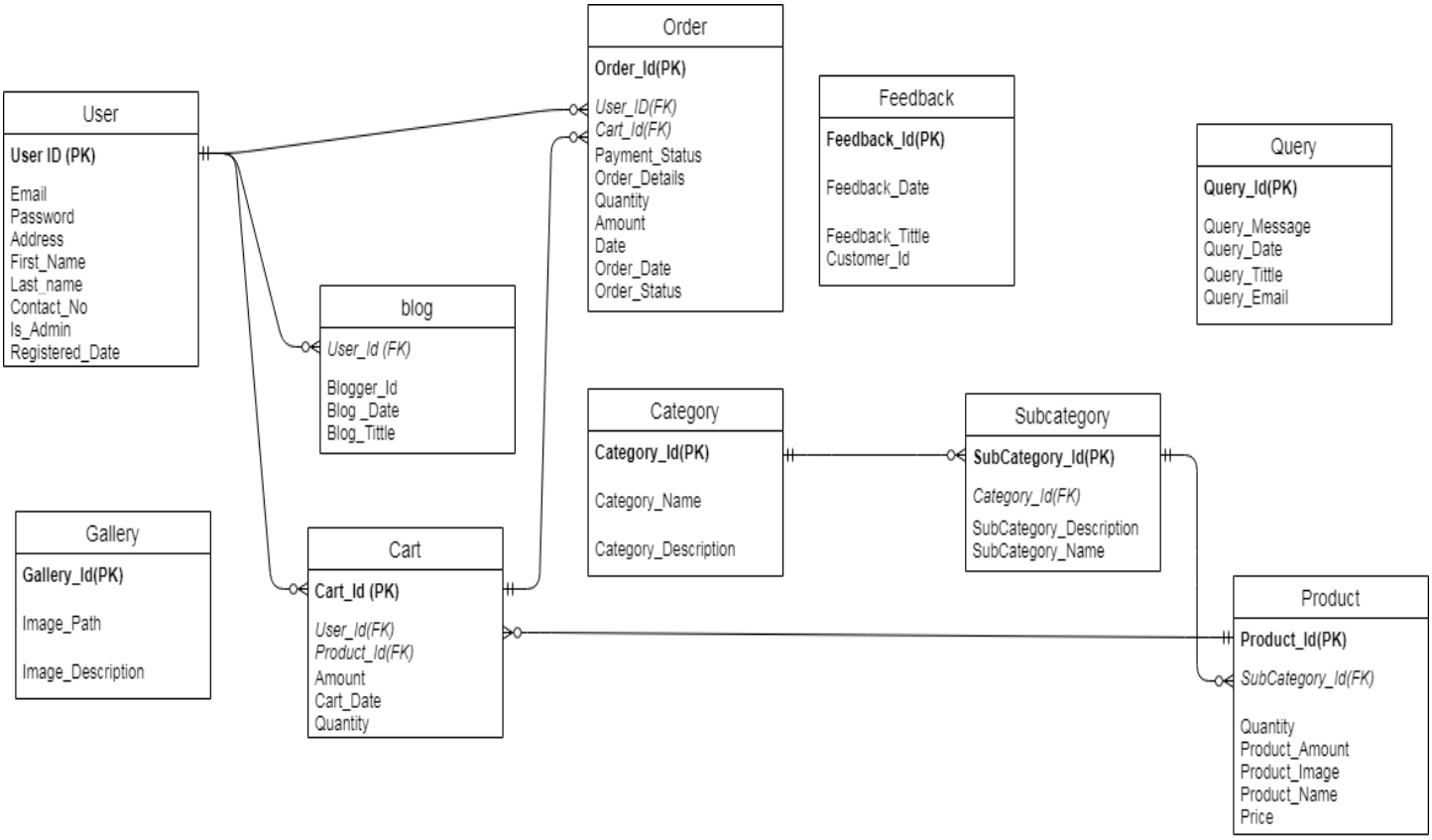
1. User_Id: This field stores user identification of blog.
2. Blog_Date: This field stores blog date of blog.
3. Blog_Title: This filed stores blog titke of blog.
4. Blogger_Id: This field stores blogger identification of blog.

10) **Query:**

1 primary key and 4 attributes

1. Query_Id: This field stores query identification of query.
2. Query_Eamil: This field stores query email of query.
3. Query_Message: This field stores query message of query.
4. Query_Date: This field stores query date of query.
5. Query_Title: This field stores query title of query.

4.5.2 E-R Diagram Based on Data Diagram:-



E-R Diagram Based on Data Dictionary

There are 10 Modules Based on 10 Data Dictionary

1)User:-

-It has 1 Primary Key User_Id.

2)Category:-

-It Has 1 Primary key Category_Id.

3)SubCategory:-

-IT has 1 Primary Key SubCategory_Id.

-It has 1 Foreign Key Category_Id.

4)Product:-

-It has 1 Primary Key Product_Id.

-It has 1 foreign Key SubCategory_Id.

5)Cart:-

-It has 1 Primary Key Cart_Id.

-It have 2 Foreign Key User_Id and Product_Id.

6)Order:-

-It has 1 Primary Key Order_Id.

-It have 2 Foreign Key Cart_Id and User_Id.

7)Gallery:-

-It has 1 Primary Key Gallery_Id.

8)Feedback:-

-It has 1 Primary key Feedback_Id.

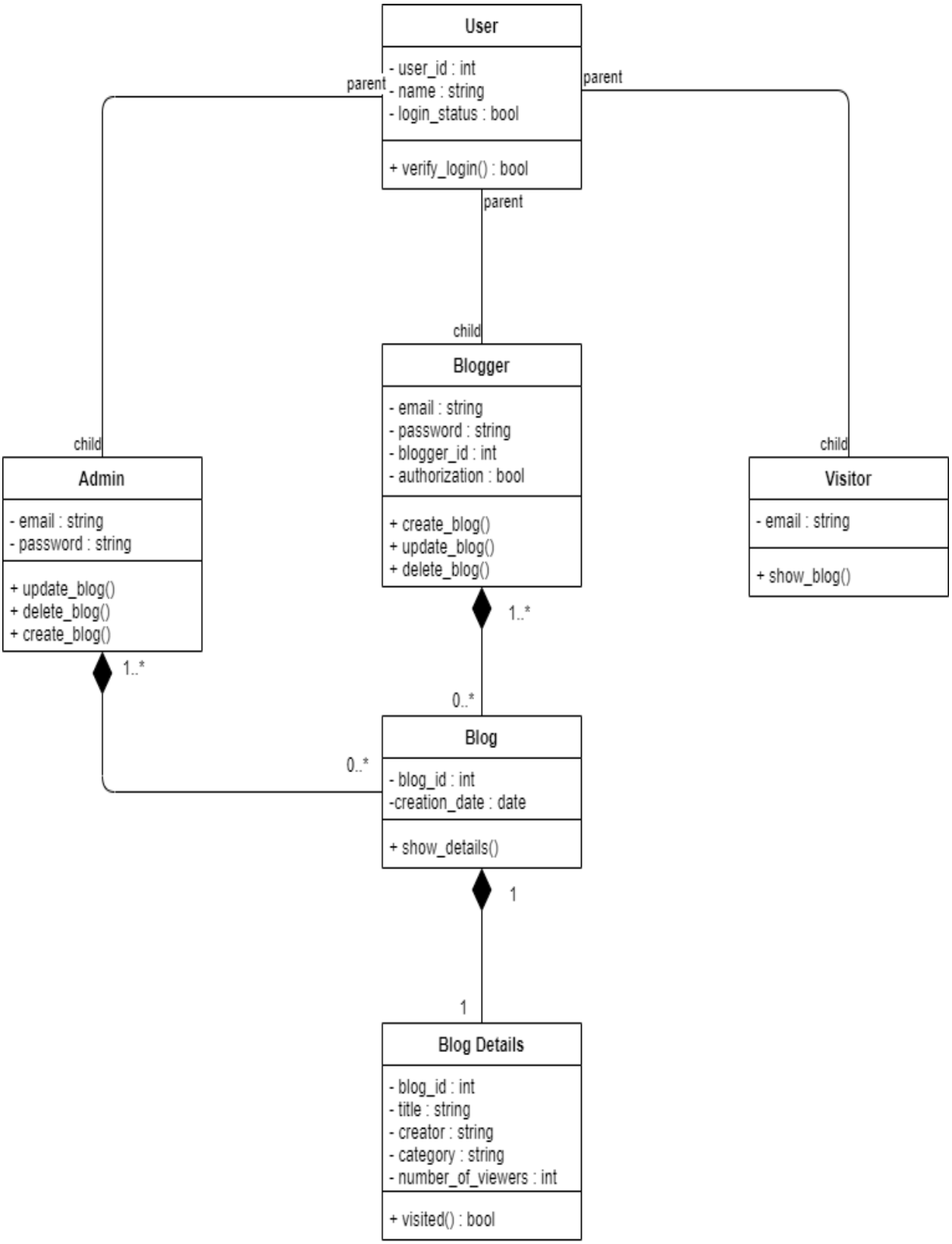
9)Blog:-

-It has 1 Foreign Key User_Id.

10)Query:-

-It has 1 Primary Key Query_Id.

4.5.4. Class Diagram:-



UML Class Diagram:-

There are 6 modules.

1)User:-

-User is a base Module which differs.

2)Admin:-

-It is a admin's blog, it describes the work of admin in blogspot.

3)Blogger:-

-This are users which are allowed to post blogs.

-This module describes the Works and entities of a blogger.

4)Visitor:-

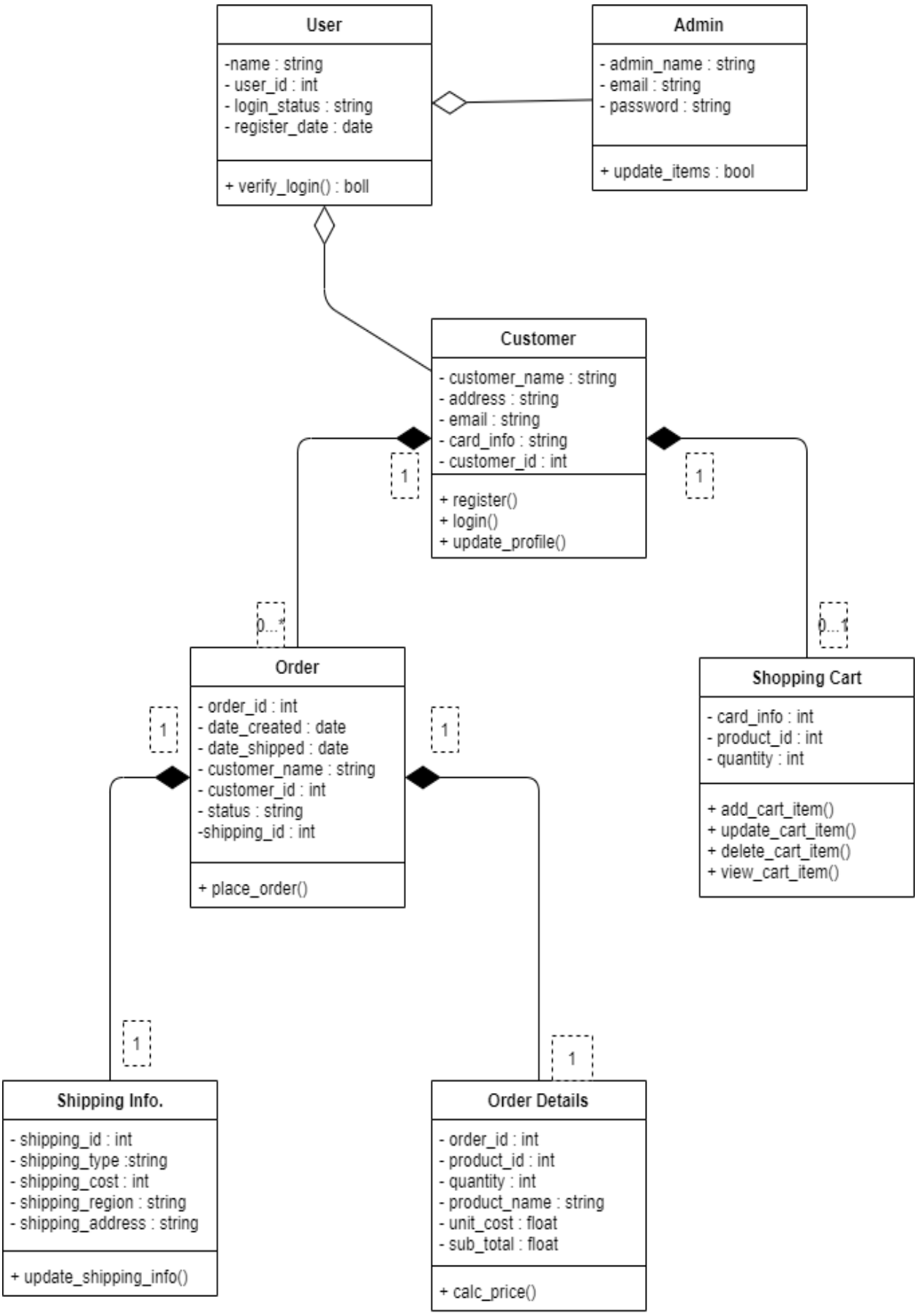
-This describes the propertied of visitor.

5)Blog:-

- This module describes the property of a blog.

6)Blog_Details:-

- This describes the blog's identification and creator details.


User
-name : string
- user_id : int
- login_status : string
- register_date : date
+ verify_login() : boll
Admin
- admin_name : string
- email : string
- password : string
+ update_items : bool
Customer
- customer_name : string
- address : string
- email : string
- card_info : string
- customer_id : int
+ register()
+ login()
+ update_profile()
1
1
0...*
0...1
Order
- order_id : int
- date_created : date
- date_shipped : date
- customer_name : string
- customer_id : int
- status : string
-shipping_id : int
+ place_order()
1
1
Shopping Cart
- card_info : int
- product_id : int
- quantity : int
+ add_cart_item()
+ update_cart_item()
+ delete_cart_item()
+ view_cart_item()
1
1
Shipping Info.
- shipping_id : int
- shipping_type :string
- shipping_cost : int
- shipping_region : string
- shipping_address : string
+ update_shipping_info()
Order Details
- order_id : int
- product_id : int
- quantity : int
- product_name : string
- unit_cost : float
- sub_total : float
+ calc_price()

-UML Class Diagram Covistore.

-There are 7 modules in diagram.

1)User:-

-This module Log's in and differs Admin and customer.

2)Admin:-

-This module Describes the properties of Admin.

3)Customer:-

-This module describes the process of purchasing product.

4)Shopping Cart:-

-This module describes the collection of item to buy.

5)Order:-

-This Describes the modules of order itself.

6)Shipping Info:-

- This describes the shipping process.

7)Order Details:-

-This describes the order details.

4.5.3 Data Dictionary:-

			Data Dictionary - User	

Sr.No	Field Name	Data Type	Constraint	Description
1	User_Id	int(5)	Primary Key	This field stores Unique Identification of user.
2	First_Name	varchar(10)	Not Null	This field stores First Name of user.
3	Last_Name	varchar(10)	Not Null	This field stores Last Name of user.
4	Email	varchar(30)	Not Null	This field stores email of user.
5	Password	varchar(20)	Not Null	This field stores Password of user.
6	Address	varchar(70)	Not Null	This field stores Address of user.
7	Contact_No	varchar(13)	Not Null	This field stores Contact No of user.
8	Is_Admin	int(1)	Not Null	This field stores Is Admin of user.
9	Registered_Date	Date	Not Null	This field stores Registered Date of user.

			Data Dictionary - Category	

Sr.No	Field Name	Data Type	Constraint	Description
1	Category_Id	int(5)	Primary Key	This field stores Category Identification of category.
2	Category_Name	varchar(15)	Not Null	This field stores Category Name of category.
3	Category_Description	varchar(70)	Not Null	This field stores Category Description of category.

			Data Dictionary - SubCategory	

Sr.No	Field Name	Data Type	Constraint	Description
1	SubCategory_Id	int(5)	Primary Key	This field stores SubCategory Identification of subcategory.
2	Category_Id	int(5)	Foreign Key	This field stores Category Identification of subcategory.
3	SubCategory_Name	varchar(15)	Not Null	This field stores SubCategory Name of subcategory.
4	SubCategory_Description	varchar(70)	Not Null	This field stores SubCategory Discription of subcategory.

Data Dictionary - Products

Sr.No	Field Name	Data Type	Constraint	Description
1	Product_Id	int(5)	Primary Key	This field stores Product Identification of product.
2	Product_Name	varchar(20)	Not Null	This field stores Product Name of product.
3	Sub_Category Id	int(10)	Foreign Key	This field stores SubCategory Identification of product.
4	Price	int(10)	Not Null	This field stores Price of product.
5	Quantity	varchar(10)	Not Null	This field stores Quantity of product.
6	Product_Amount	int(10)	Not Null	This field stores Product Amount of product.
7	Product_Image	varchar(200)	Not Null	This field stores Product Image of product.

Data Dictionary - Cart

Sr.No	Field Name	Date Type	Constraint	Description
1	Cart_Id	int(10)	Primary Key	This field stores Cart Identification of cart.
2	User_Id	int(10)	Foreign Key	This field stores User Identification of cart.
3	Quantity	varchar(20)	Not Null	This field stores Quantity Identification of cart.
4	Product_Id	int(10)	Foreign Key	This field stores Product Identification of cart.
5	Amount	int(10)	Not Null	This field stores Amount of cart.
6	Cart_Date	Date	Not Null	This field stores Cart Date of cart.

Data Dictionary - Order

Sr.No	Field Name	Data Type	Constraint	Description
1	Order_Id	int(10)	Primary Key	This field stores Order Identification of order.
2	Order_Date	Date	Not Null	This field stores Order Date of order.
3	Cart_Id	int(10)	Foreign Key	This field stores Cart Identification of order.
4	User_Id	int(10)	Foreign Key	This field stores User Identification of order.
5	Date	Date	Not Null	This field stores Date of order.
6	Order_Status	varchar(10)	Not Null	This field stores Order Status of order.
7	Payment_Status	varchar(10)	Not Null	This field stores Payment Status of order.
8	Order_details	varchar(15)	Not Null	This field stores Order Details of order.
9	Quantity	varchar(10)	Not Null	This field stores Quantity of order.
10	Amount	varchar(10)	Not Null	This field stores Amount of order.

Data Dictionary - Gallery

Sr.No	Field Name	Data Type	Constraint	Description
1	Gallery_Id	int(10)	Primary Key	This field stores Gallery Identification of gallery.
2	Image_Path	varchar(20)	Not Null	This field stores Image Path of gallery.
3	Image Description	varchar(30)	Not Null	This field stores Image Description of gallery.

Data Dictionary - Feedback

Sr.No	Field Name	Data Type	Constraint	Description
1	Feedback_ Id	varchar(10)	Primary key	This field stores Feedback Identification of feedback.
2	Feedback_Date	Date()	Not null	This field stores Feedback Date of feedback.
3	Feedback_Title	varchar(20)	Not null	This field stores Feedback Title of feedback.
4	Customer_Id	varchar(10)	Not null	This field stores Customer Identification of feedback.

Data Dictionary - Blog

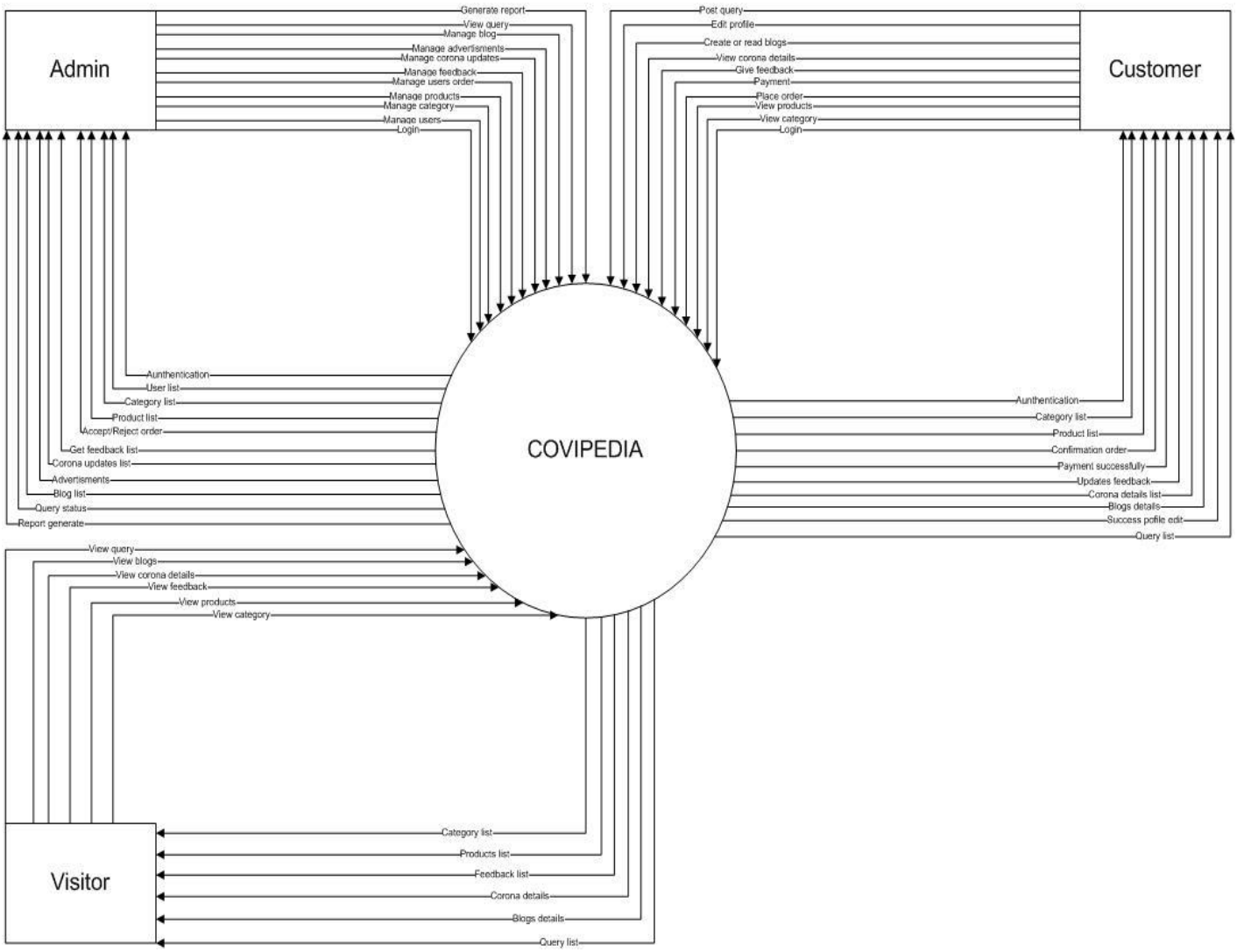
Sr.No	Field Name	Data Type	Constraint	Description
1	User_Id	int(10)	Foreign Key	This field stores User Identification of blog.
2	Blog_Date	Date()	Not Null	This field stores Blog Date of blog.
3	Blog_Title	varchar(20)	Not Null	This field stores Blog Title of blog.
4	Blogger_Id	varchar(10)	Not Null	This field stores Blogger Identification of blog.

Data Dictionary - Query

Sr.No	Field Name	Data Type	Constraint	Description
1	Query_Id	int(10)	Primary Key	This field stores Query Identification of query.
2	Query_Email	varchar(20)	Not Null	This field stores Query Email of query.
3	Query_Message	varchar(100)	Not Null	This field stores Query Message of query.
4	Query_Date	Date	Not Null	This field stores Query Date of query.
5	Query_Title	varchar(20)	Not Null	This field stores Query Title of query.

4.6 Functional & Behavioural Modelling:-

4.6.1 Context Level Diagram:-

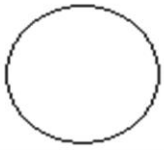
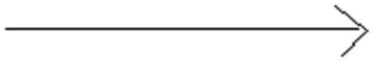


What is Data Flow Diagram?

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled.

They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually "say" things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO.

That's why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.

Symbol	Meaning
 Process	Single process: A circle is used to represent the entire system.
	Data flow: An arrow is used to represent the flow of data between the process and external entities.
 External entity	External entity: A square or rectangle represents any person or organisation that sends data to or receives data from the system.
 Data store	Data store: An open rectangle represents the location where data is stored. It could be a filing cabinet, hard disk.

Context Level Diagram (Level 0) :-

1) **Admin-**
   - Login: Admin can login for authentication.
   - Manage users: Admin can manage user for user list.
   - Manage category: Admin can manage category for category list.
   - Manage subcategory: Admin can manage subcategory for subcategory list.
   - Manage products: Admin can manage products for product list.
   - Manage users order: Admin can manage user order for accept/Reject order.
   - Manage feedback: Admin can manage feedback for get feedback list.
   - Manage corona updates: Admin can manage corona updates for corona updates list.
   - Manage advertisements: Admin can manage advertisements.
   - Manage blog: Admin can manage blog for blog list.

- View query: Admin can view query for query status.
- Generate report: Admin can generate report for report generate.

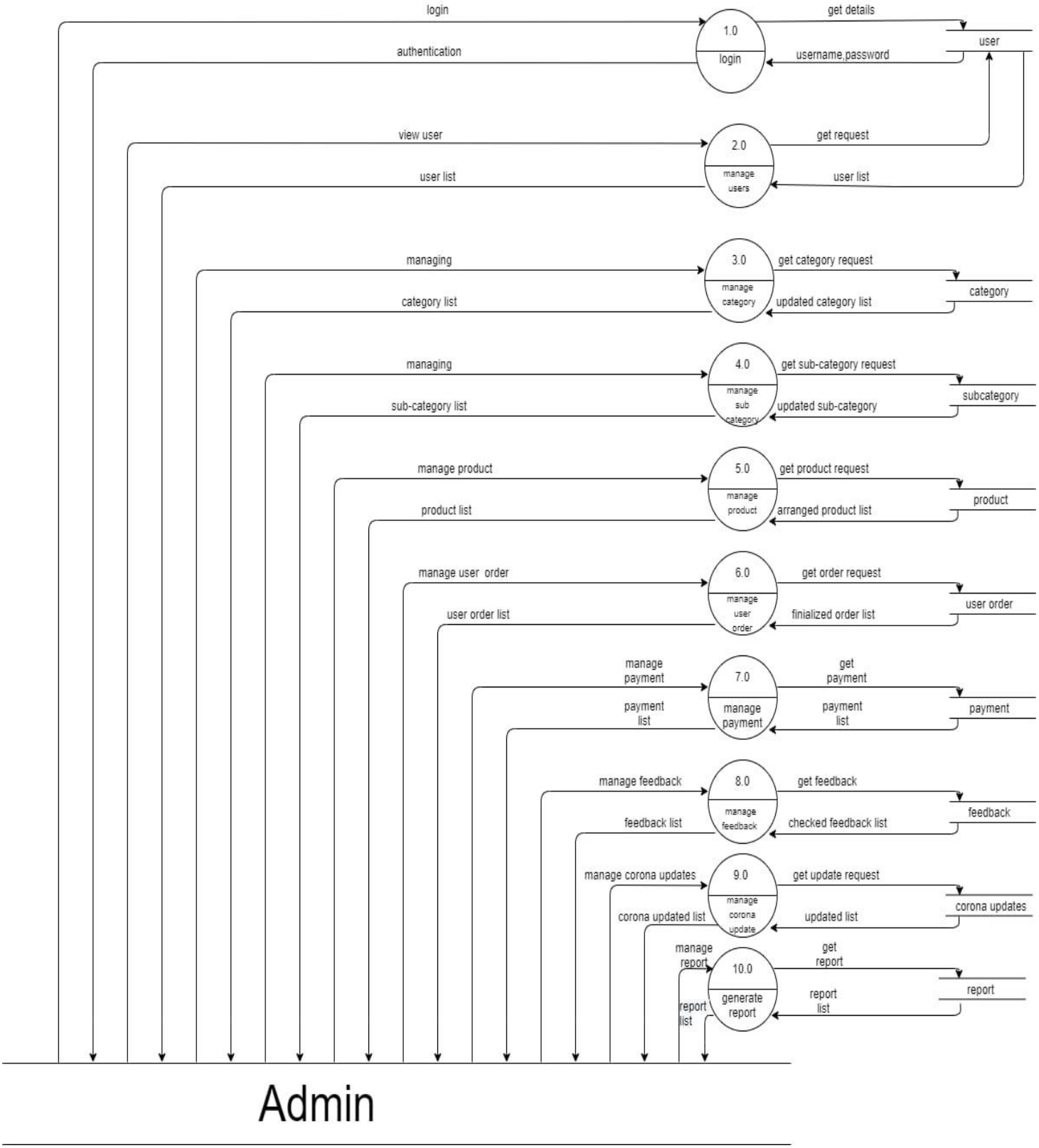
2) Customer -

- Login: Customer can login for authentication.
- View category: Customer can view category for category list.
- View subcategory: Customer can view subcategory for subcategory list.
- View product: Customer can view product for product list.
- Place order: Customer can place order for confirmation order.
- Payment: Customer can payment for payment successfully.
- Give feedback: Customer can give feedback for updates feedback.
- View corona details: Customer can view corona details for corona details list.
- Create or read blogs: Customer can create or read blogs for blogs details.
- Edit profile: Customer can edit profile for success profile edit.
- Post query: Customer can post query for query list.

3) Visitor –

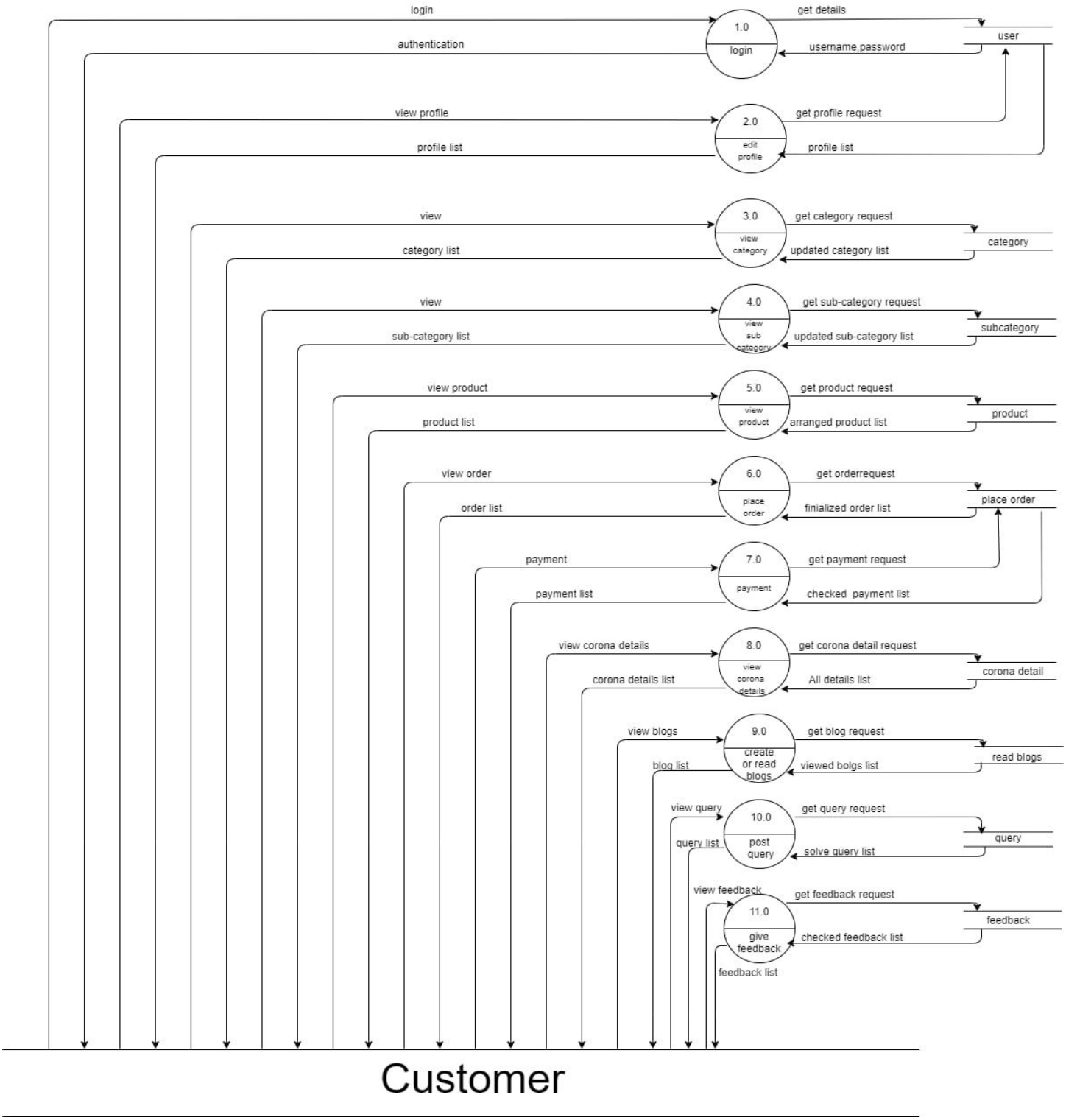
- View category: Visitor can view category for category list.
- View subcategory: Visitor can view subcategory for subcategory list.
- View product: Visitor can view product for product list.
- View feedback: Visitor can view feedback for feedback list.
- View corona details: Visitor can view corona details for corona details list.
- View blogs: Visitor can view blogs for blogs details.
- View query: Visitor can view query for query list.

4.6.2 Data Floe Diagram (Level-1)



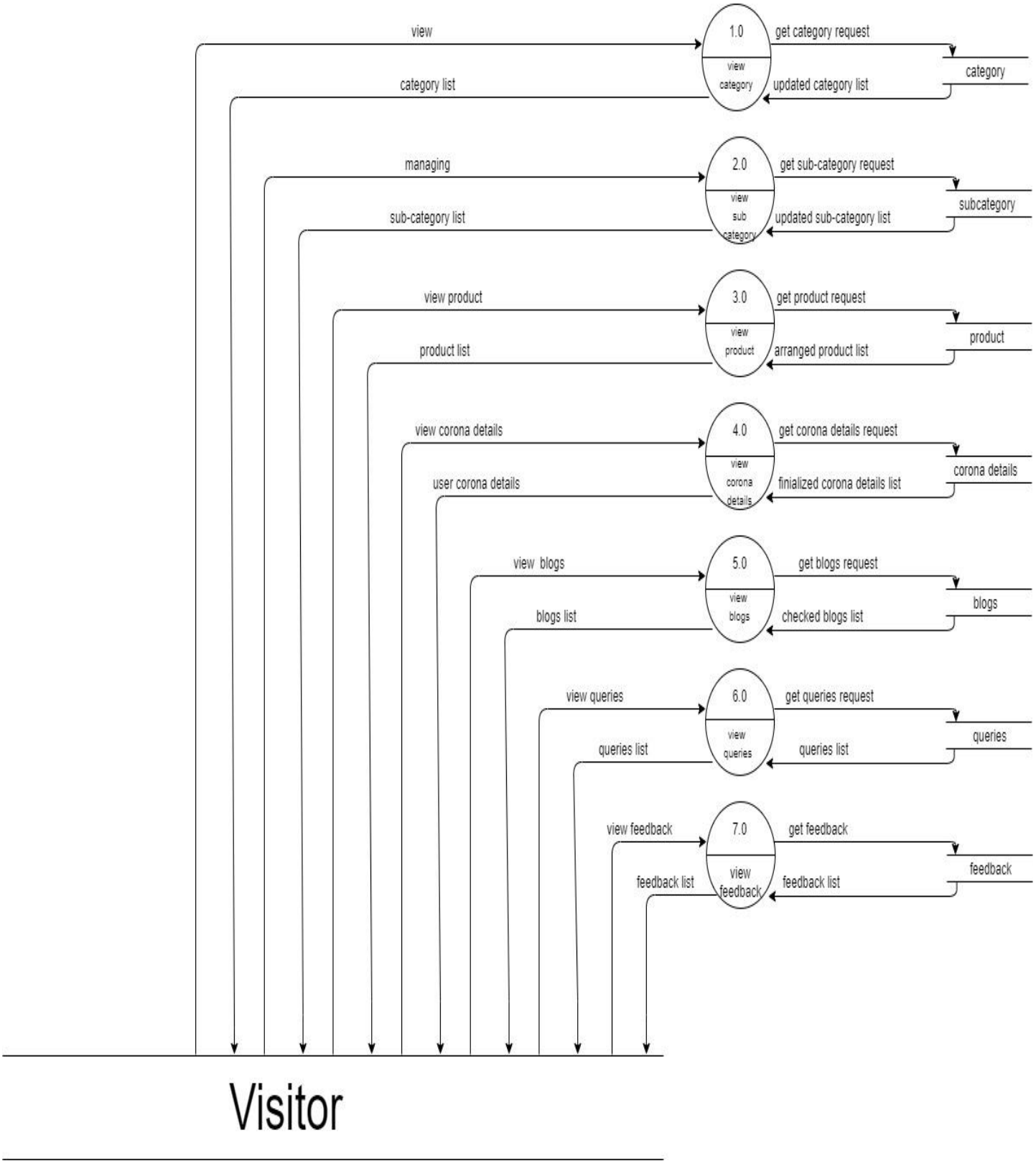
❖ Data flow diagram level-01(admin)

- Login: It is part of diagram which show how admin take it is first step towards website.
- Manage users: It is part of diagram which show how to manage user list.
- Manage category: It is part of diagram which show how to manage category.
- Manage sub-category: It is part of diagram which show how to manage sub-category.
- Manage product: It is part of diagram which show how to manage product.
- Manage user order: It is part of diagram which show how to manage user order.
- Manage feedback: It is part of diagram which show how to manage feedback.
- Manage corona updates: It is part of diagram which show how to manage corona updates.


login
authentication
1.0
login
get details
username,password
user
view profile
profile list
2.0
edit profile
get profile request
profile list
view
category list
3.0
view category
get category request
updated category list
category
view
sub-category list
4.0
view sub category
get sub-category request
updated sub-category list
subcategory
view product
product list
5.0
view product
get product request
arranged product list
product
view order
order list
6.0
place order
get orderrequest
finialized order list
place order
payment
payment list
7.0
payment
get payment request
checked payment list
view corona details
corona details list
8.0
view corona details
get corona detail request
All details list
corona detail
view blogs
blog list
9.0
create or read blogs
get blog request
viewed bolgs list
read blogs
view query
query list
10.0
post query
get query request
solve query list
query
view feedback
feedback list
11.0
give feedback
get feedback request
checked feedback list
feedback
Customer

❖ Data flow diagram leve-01(Customer)

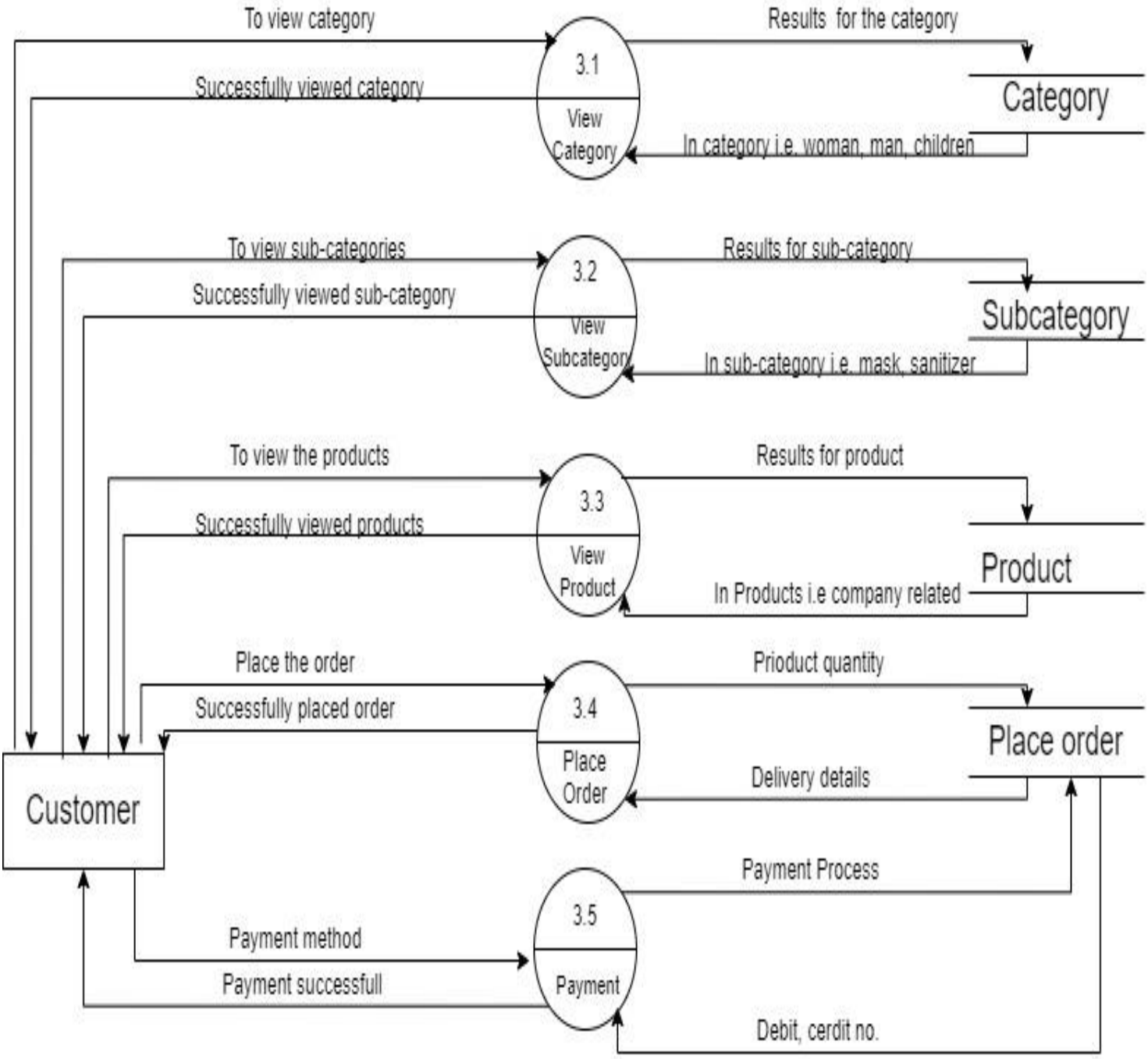
- Login: It is part of diagram which show how admin take it is first step towards website.
- Edit profile: It is part of diagram which show how to edit profile.
- View category: It is part of diagram which show how to view category.
- View sub-category: It is part of diagram which show how to view sub-category.
- View product: It is part of diagram which show how to view product.
- Place order: It is part of diagram which show how to place order.
- Payment: It is part of diagram which show how to payment.
- View corona details: It is part of diagram which show how to view corona details.
- Create or read blogs: It is part of diagram which show how to create or read blogs.
- Post query: It is part of diagram which show how to post query.
- Give feedback: It is part of diagram which show how to give feedback.


view
1.0
get category request
view category
category
category list
updated category list
managing
2.0
get sub-category request
view sub category
subcategory
sub-category list
updated sub-category list
view product
3.0
get product request
view product
product
product list
arranged product list
view corona details
4.0
get corona details request
view corona details
corona details
user corona details
finialized corona details list
view blogs
5.0
get blogs request
view blogs
blogs
blogs list
checked blogs list
view queries
6.0
get queries request
view queries
queries
queries list
queries list
view feedback
7.0
get feedback
view feedback
feedback
feedback list
feedback list
Visitor

❖ Data flow diagram level-01(visitor)

- View category: It is part of diagram which show how to view category.
- View sub-category: It is part of diagram which show how to view sub-category.
- View product: It is part of diagram which show how to view product.
- View corona details: It is part of diagram which show how to view corona details.
- View blogs: It is part of diagram which show how to view blogs.
- View queries: It is part of diagram which show how to view product.
- View feedback: It is part of diagram which show how to view feedback.

4.6.3 Data Flow Diagram(Level- 2):-



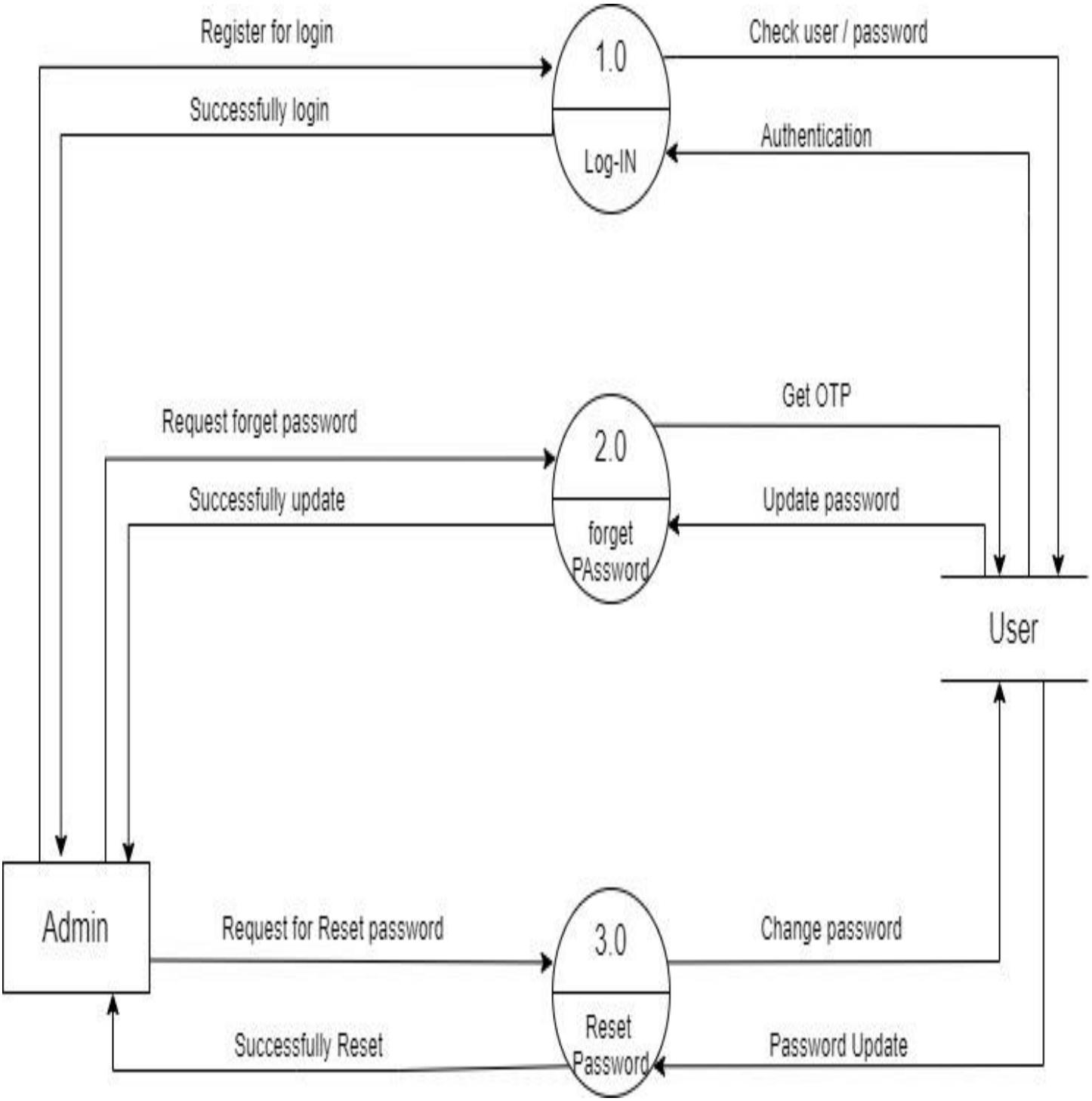
❖ Data Flow Diagram Level-2 (Order)

- View Category:- In this portion it will show to whom the product will needed i.e. woman, man, children, etc.

- View Sub-Category:- This section will view the sub-category what the particularly product Customer want i.e. mask, sanitizer, PPE kit, etc.

- View Product:- Which company/brand providing which product i.e. in sanitizer there's gel sanitizer, liquid sanitizer, etc.

- Place Order & Payment:- In this section it will process on the order the customer placed and the portion of payment like via credit card, debit card, net banking, etc.

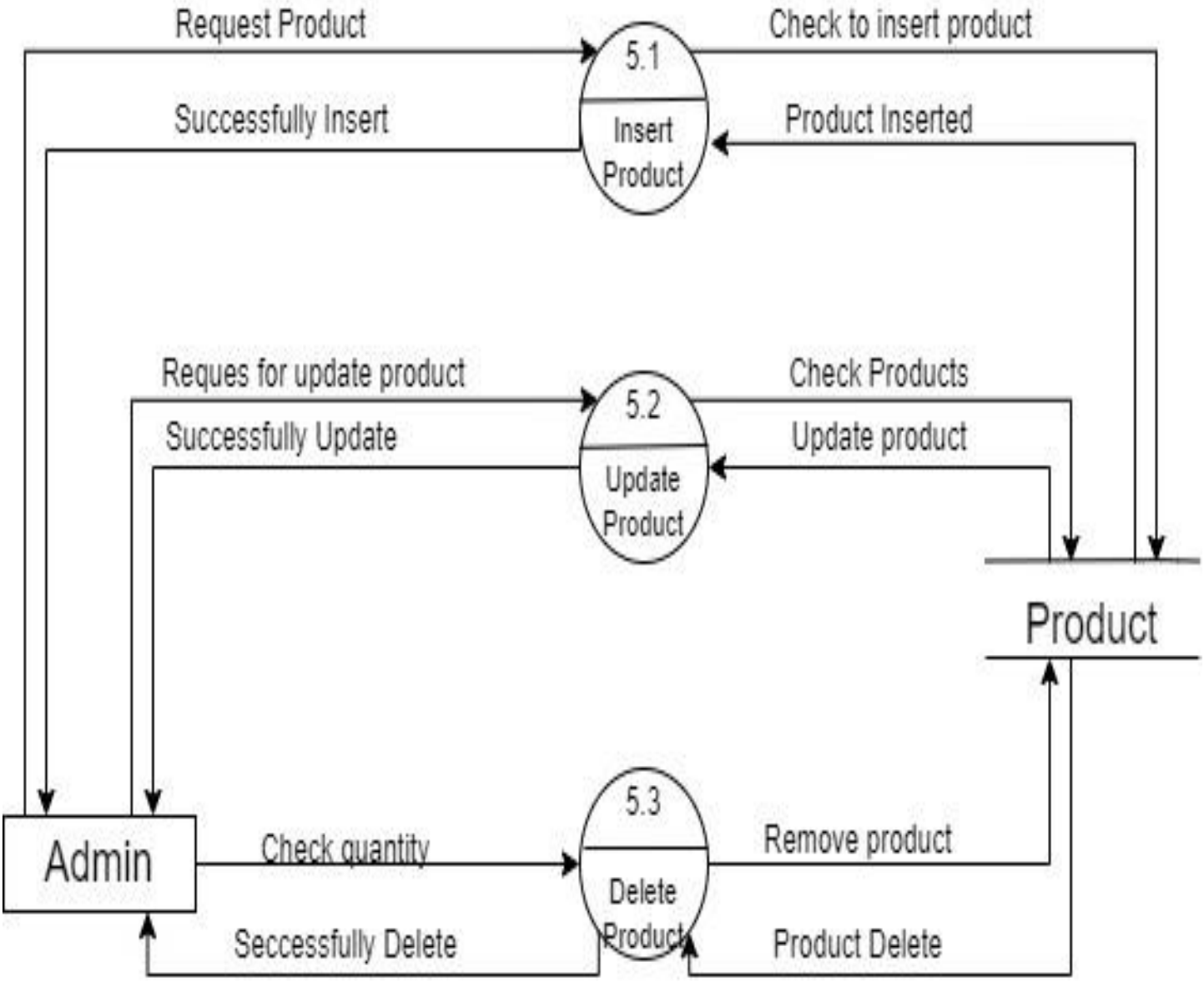

Register for login
1.0
Check user / password
Successfully login
Log-IN
Authentication
Get OTP
Request forget password
2.0
Successfully update
Update password
forget PAssword
User
Admin
Request for Reset password
3.0
Change password
Reset Password
Successfully Reset
Password Update

❖ Data Flow Diagram Level-2 (Admin)

- Log-In:- This portion particularly covers the registration for login after the request for login send it comes under check user/ password, when the user and password work it goes under authentication process and get successfully login.

- Forget Password:- When user request for forget password they will get OTP, after entering the OTP system ask to enter new password and after that the password will get updated successfully.

- Reset Password:- when the user request for reset password or change password the system will authenticate the old password and get modify the new password and finally the password will get successfully reset.


Request Product
Check to insert product
5.1
Insert Product
Successfully Insert
Product Inserted
Reques for update product
Check Products
5.2
Update Product
Successfully Update
Update product
Product
Admin
Check quantity
5.3
Delete Product
Remove product
Seccessfully Delete
Product Delete

❖ Data Flow Diagram Level-2 (Product)

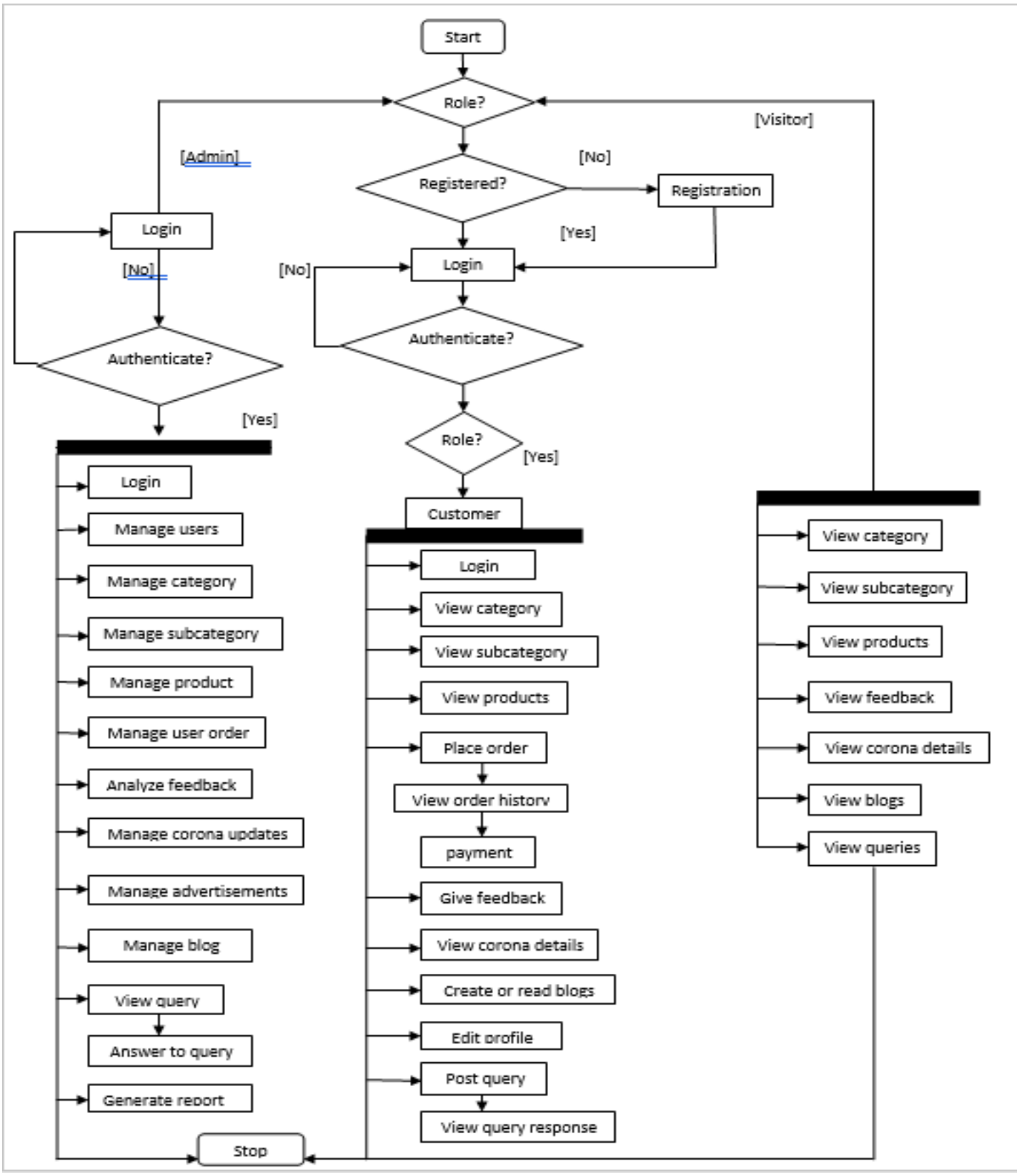
- Insert Product:- When there's need for any product, so to insert new product data this portion particularly works. After it the new quantity of data is successfully inserted.

- Update Product:- When there's request for new products, this will change and update the product quantity and some other information regarded to the product.

- Delete Product:- This portion will delete the product when there's not a single product in the stock.

4.7 Function Of System:-

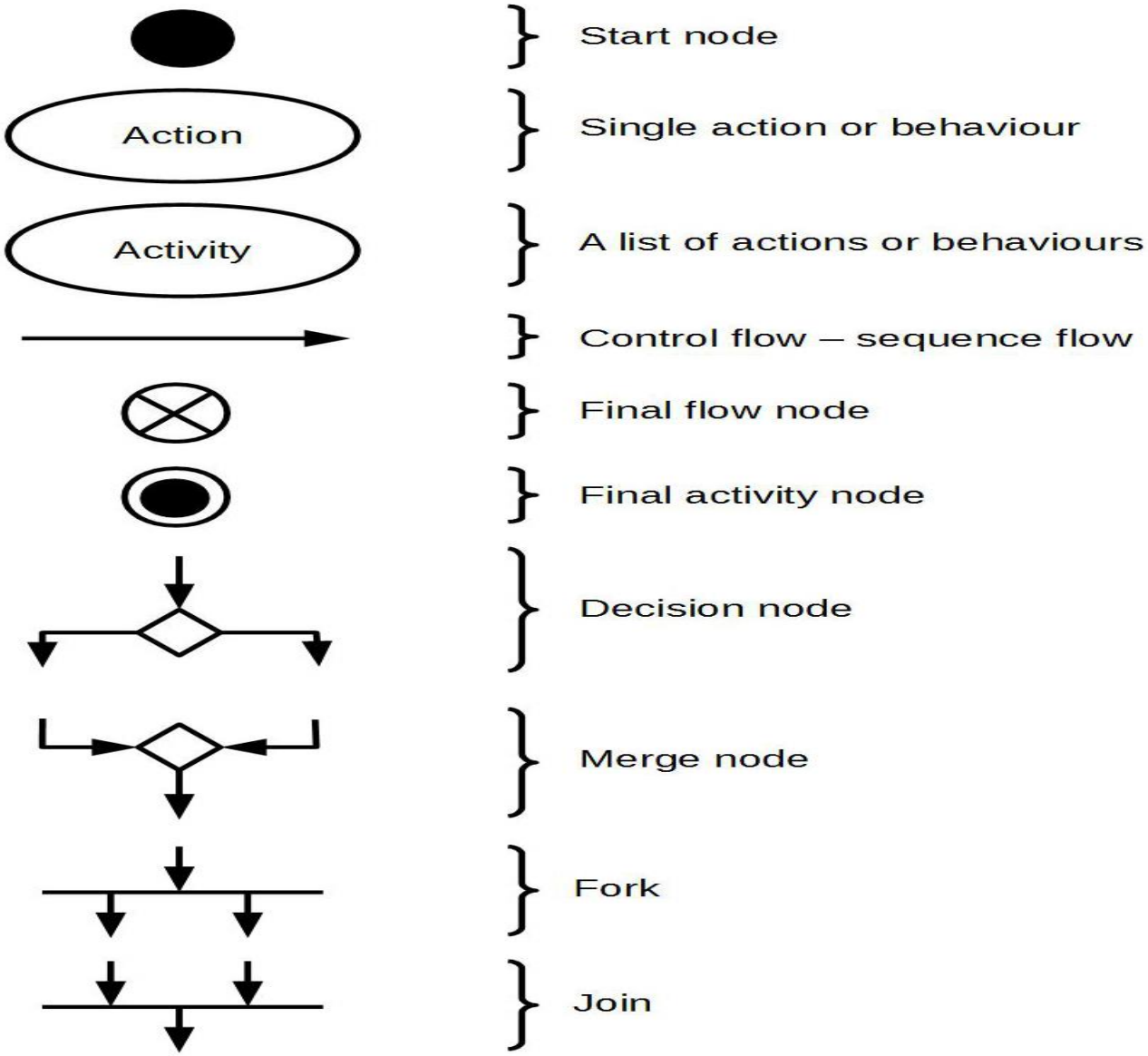
4.7.1. Flow Chart:-



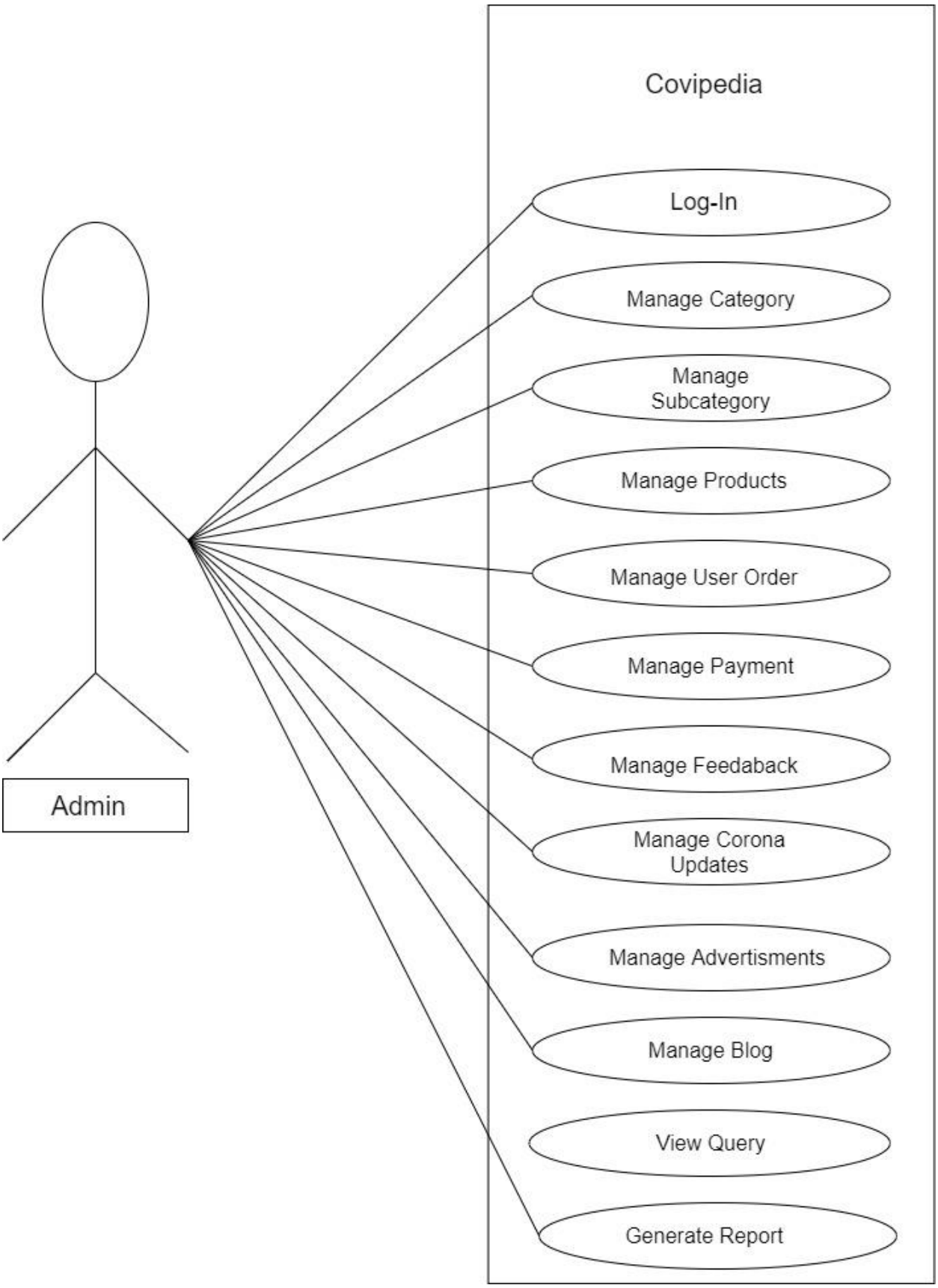
4.7.2 Use Case Diagram:-

What is Unified Modeling Language (UML)?

UML, short for Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.



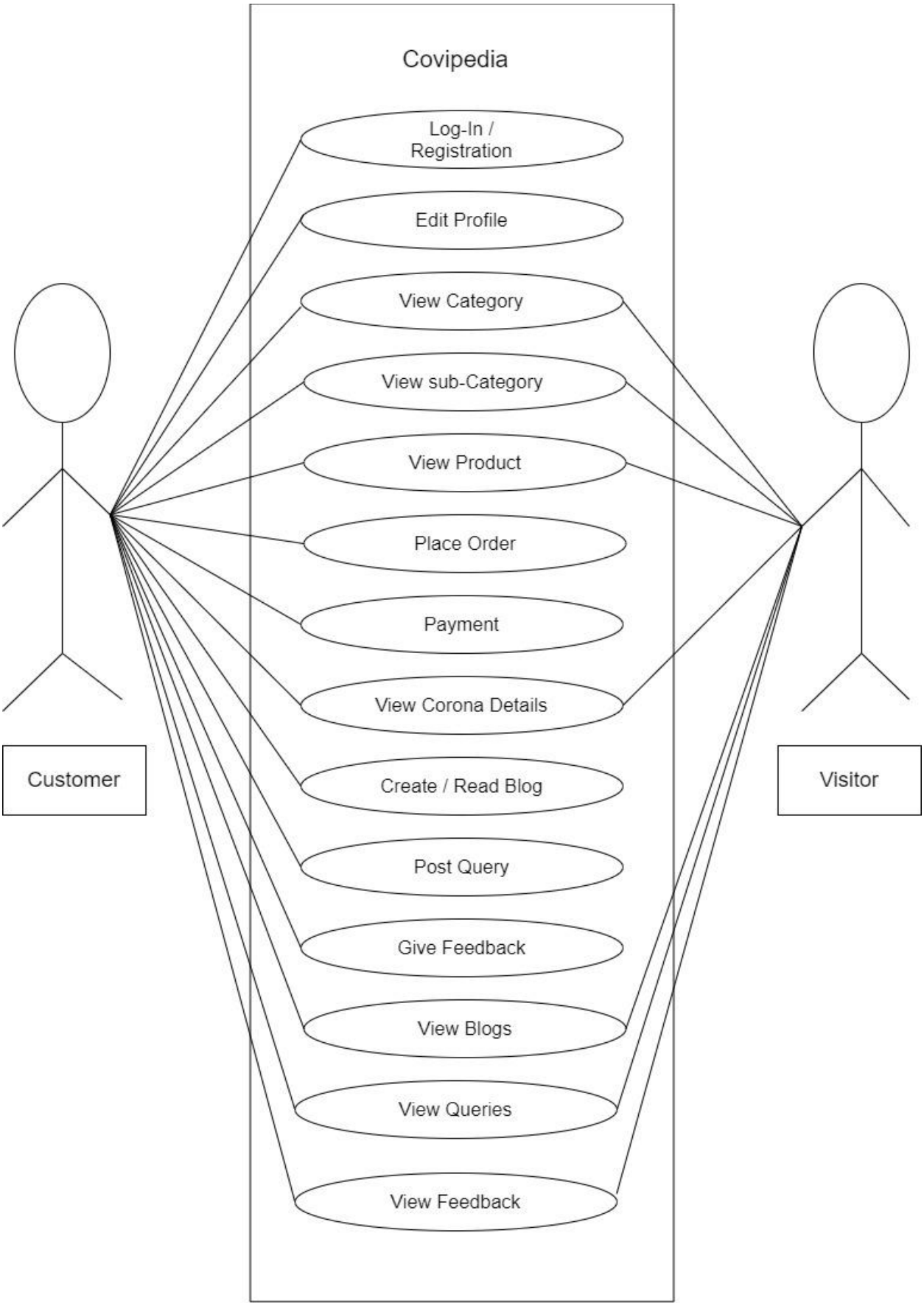
4.7.2. Use Case Diagram.



Use Case Diagram (Admin)

Admin:- This frame particularly describes the key features of admin like

i. Log-In
ii. Manage Category
iii. Manage Sub-Category
iv. Manage Products
v. Manage User Order
vi. Manage Payment
vii. Manage Feedback
viii. Manage corona Updates
ix. Manage Advertisements
x. Manage Blog
xi. View Query
xii. Generate Report


Covipedia
Log-In / Registration
Edit Profile
View Category
View sub-Category
View Product
Place Order
Payment
View Corona Details
Create / Read Blog
Post Query
Give Feedback
View Blogs
View Queries
View Feedback
Customer
Visitor

Use Case Diagram (Customer and Visitor)

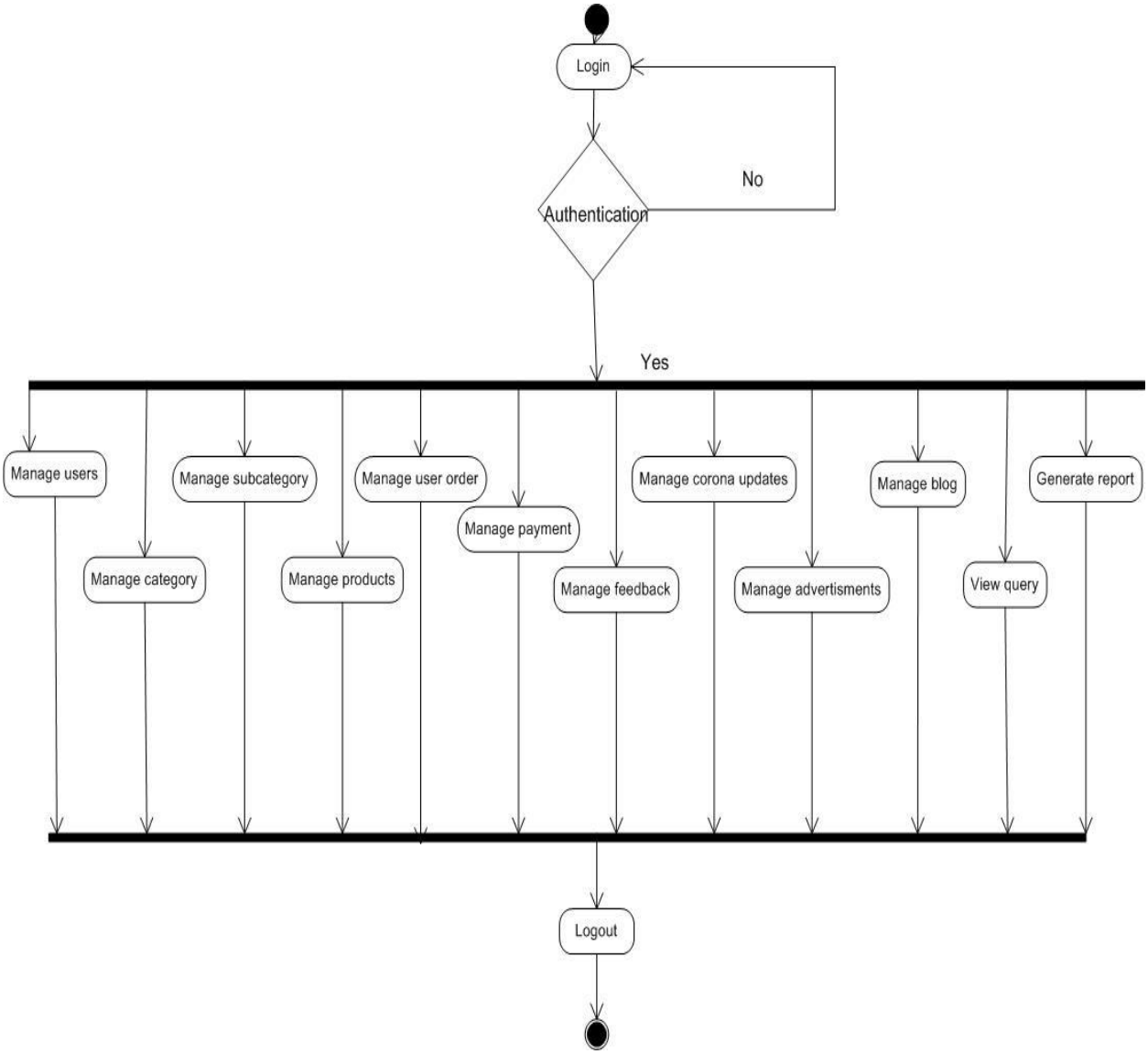
Customer:- This frame particularly describes the key features of customer like

i. Log-In/ Registration
ii. Edit Profile
iii. View Category
iv. View Sub-category
v. View Product
vi. Place Order
vii. Payment
viii. View Corona Details
ix. Create/Read Blogs
x. Post Query
xi. Give Feedback
xii. View Blogs
xiii. View Queries
xiv. View Feedback

Visitor:- This frame particularly describes the key features of visitor like

i. View Category
ii. View Sub-category
iii. View Product
iv. View Corona Details
v. View Blogs
vi. View Queries
vii. View Feedback

4.7.3 Activity Diagram.

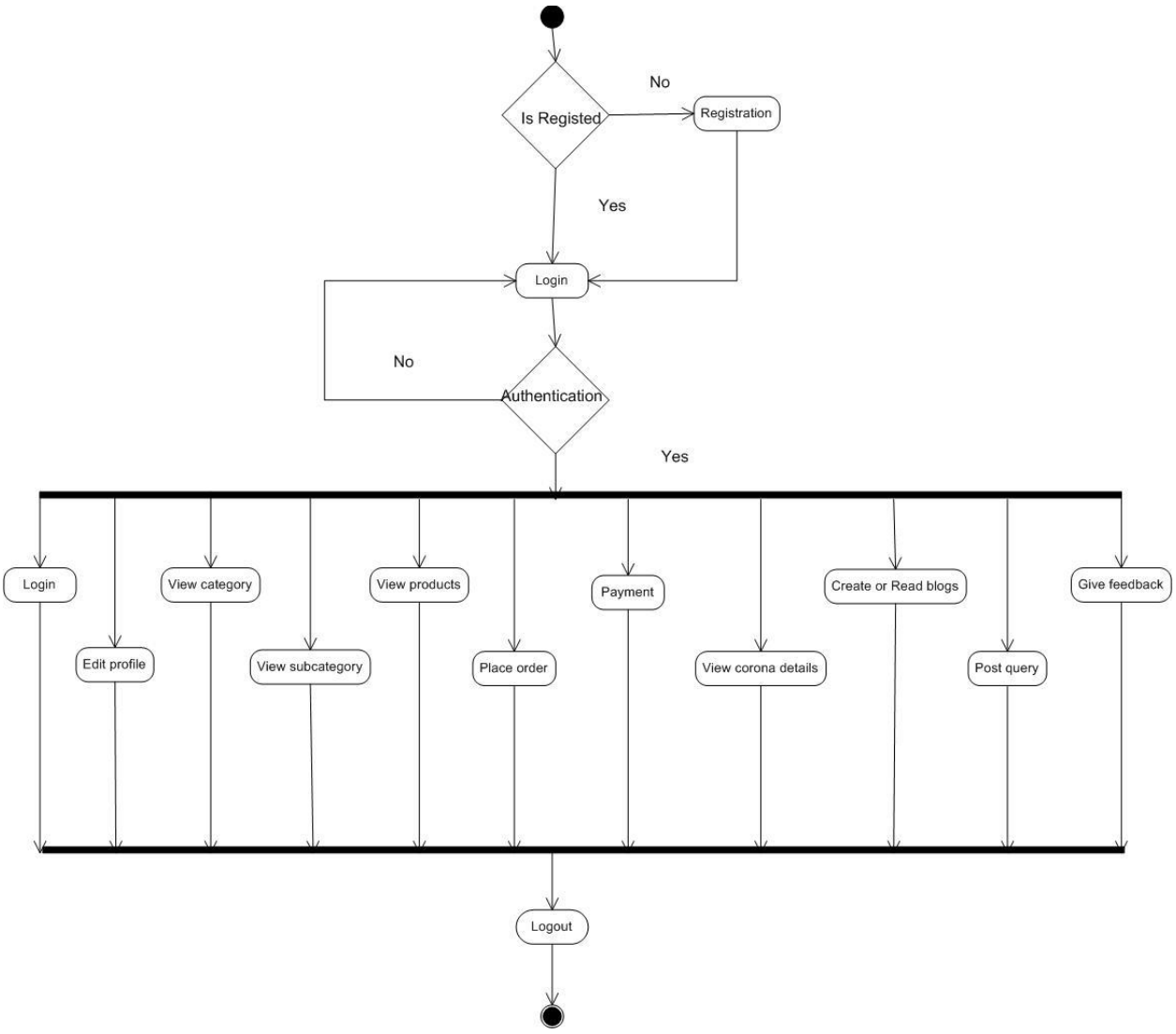


Activity Diagram: -

1) **Admin -**

- Manage users: Admin can manage users.
- Manage category: Admin can manage category.
- Manage subcategory: Admin can manage subcategory.
- Manage products: Admin can manage products.
- Manage user order: Admin can manage user order.
- Manage feedback: Admin can manage feedback.
- Manage corona updates: Admin can manage corona updates.

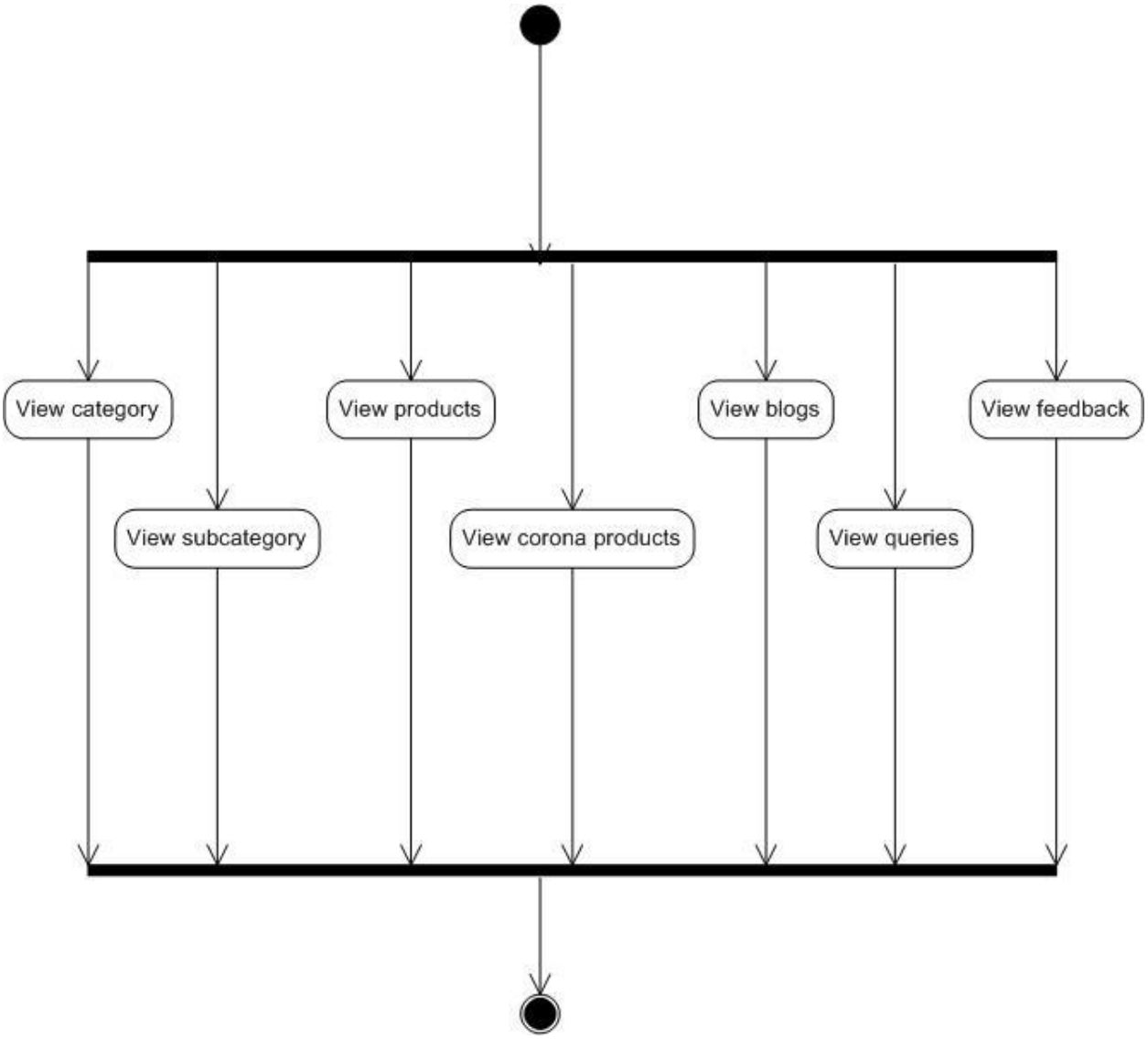
- Manage advertisements: Admin can manage advertisements.
- Manage blog: Admin can manage blog.
- View query: Admin can view query.
- Generate report: Admin can generate report.



2) Customer –

- Login: Customer can login or register.
- Edit profile: Customer can edit the profile.
- View category: Customer can view category.
- View subcategory: Customer can view subcategory.
- View products: Customer can view products.

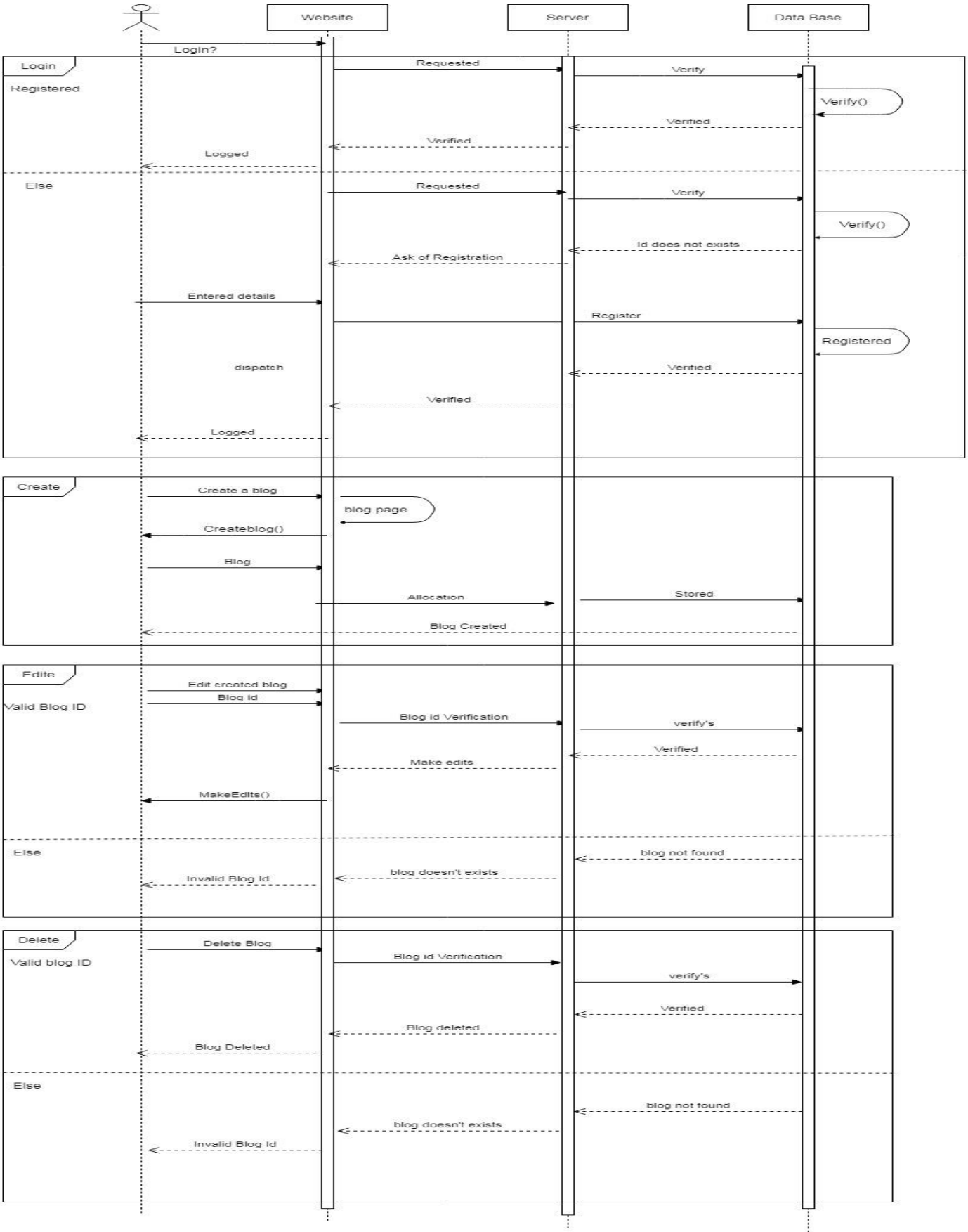
- Place order: Customer can place order.
- Payment: Customer can pay.
- View corona details: Customer can view corona details.
- Create or Read blogs: Customer can create or read blogs.
- Post query: Customer can post the query.
- Give feedback: Customer can give feedback.



3)Visitor –

- View category: Visitor can view category.
- View subcategory: Visitor can view subcategory.
- View products: Visitor can view products.
- View corona products: Visitor can view corona products.
- View blogs: Visitor can view blogs.
- View queries: Visitor can view queries.
- View feedback: Visitor can view feedback.

4.7.4 Sequence Diagram:



- Sequence diagram Admin and User Blog

-There are 3 modules.

-Website :- Represents web page

- Server :- Represents web server

- Database :- Represents Physical storage

-There is 1 Actor which represents user

- There are 3 frames.

1)Login:-

-This frame describes User's process of logging in.

-if User is register it will directly Log user to website.

-If not then, It will ask for registration.

-After registration server will allow the user to access website

2)Create:-

-When user clicks on the create a blog button, a blog creation will open.

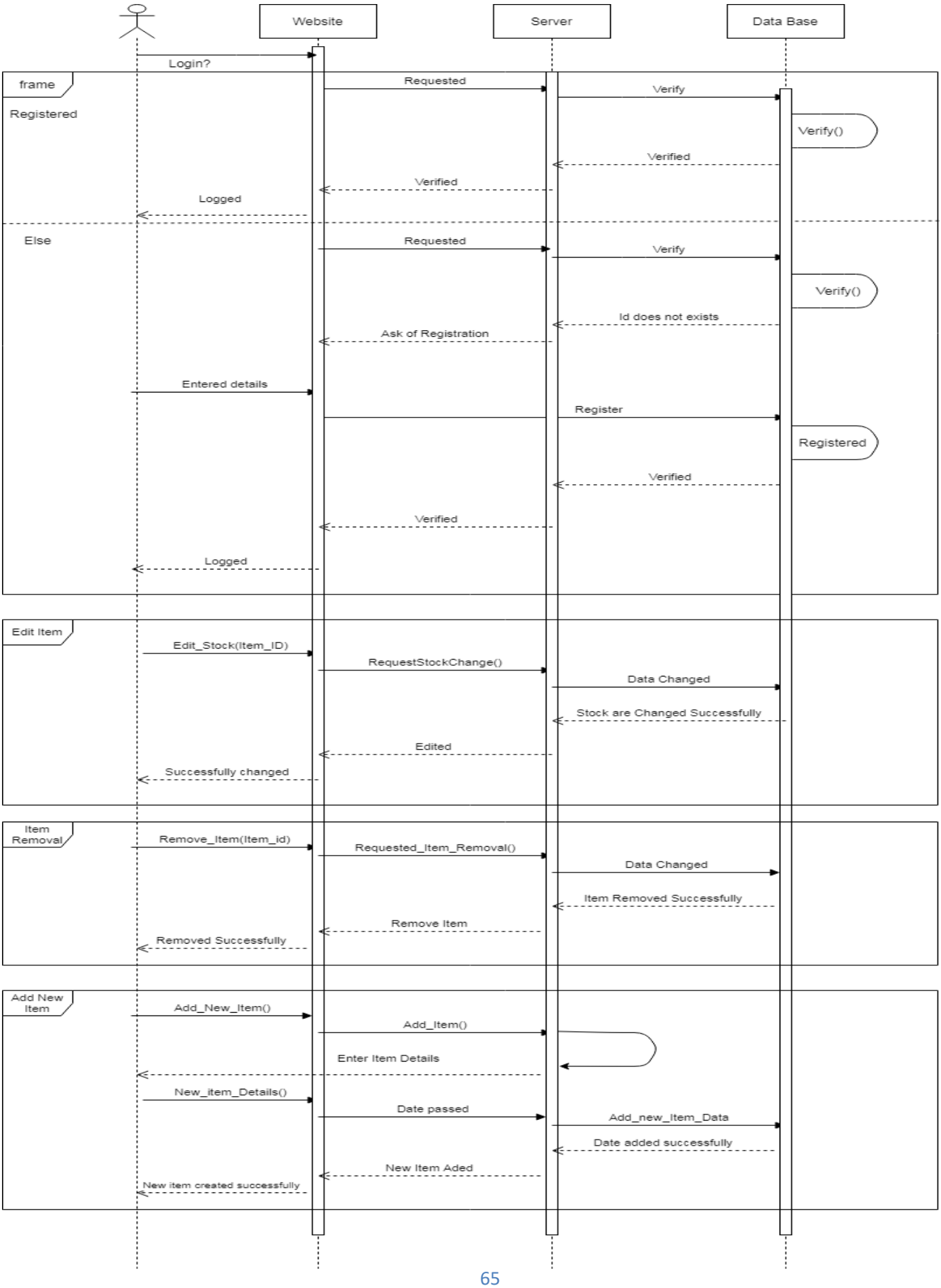
- And user can Create a blog and post it.

3) Edit:-

- This frame will allow the user to make edits in the blog that they have created.

4)Delete:-

- this frame will allow the user and admin to delete a blog.


Website
Server
Data Base
Login?
frame
Registered
Requested
Verify
Verify()
Verified
Verified
Logged
Else
Requested
Verify
Verify()
Id does not exists
Ask of Registration
Entered details
Register
Registered
Verified
Verified
Logged
Edit Item
Edit_Stock(Item_ID)
RequestStockChange()
Data Changed
Stock are Changed Successfully
Edited
Successfully changed
Item Removal
Remove_Item(Item_id)
Requested_Item_Removal()
Data Changed
Item Removed Successfully
Remove Item
Removed Successfully
Add New Item
Add_New_Item()
Add_Item()
Enter Item Details
New_item_Details()
Date passed
Add_new_Item_Data
Date added successfully
New Item Aded
New item created successfully

- Sequence diagram Admin in covistore.

-There are 3 modules.

-Website:- Represents web page

- Server:- Represents web server

- Data base :- Represents Physical storage

-There is 1 Actor which represents user

- There are 3 frames.

1)login:-

-This frame describes Admin's process of logging in.

2)Edit Item:-

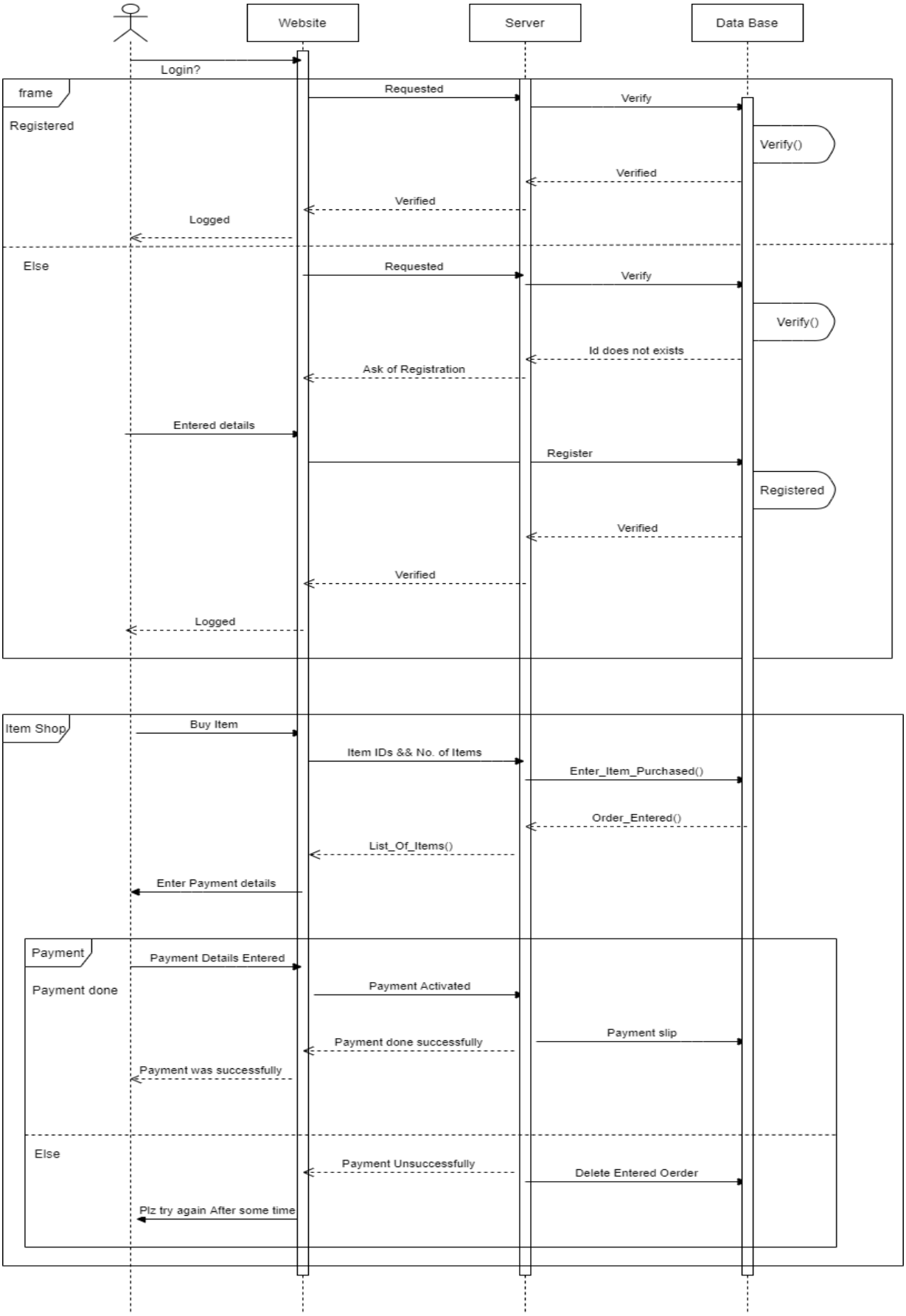
-This frame describes the process of changing the stocks of entered Item number.

3)Item removal:-

-This Frame describes the process of removing item.

4)Add New Item:-

-This frame describes the process of Adding a new item.



- Sequence diagram Customer Covistore

-There are 3 modules.

-Website:- Represents web page

- Server:- Represents web server

- Data base :- Represents Physical storage

-There is 1 Actor which represents user

- There are 3 frames.

1)login:-

-This frame describes User's process of logging in.

-if User is register it will directly Log user to website.

-If not then, It will ask for registration.

-After registration server will allow the user to access website

2)Item shop:-

- it 2 Sub Parts

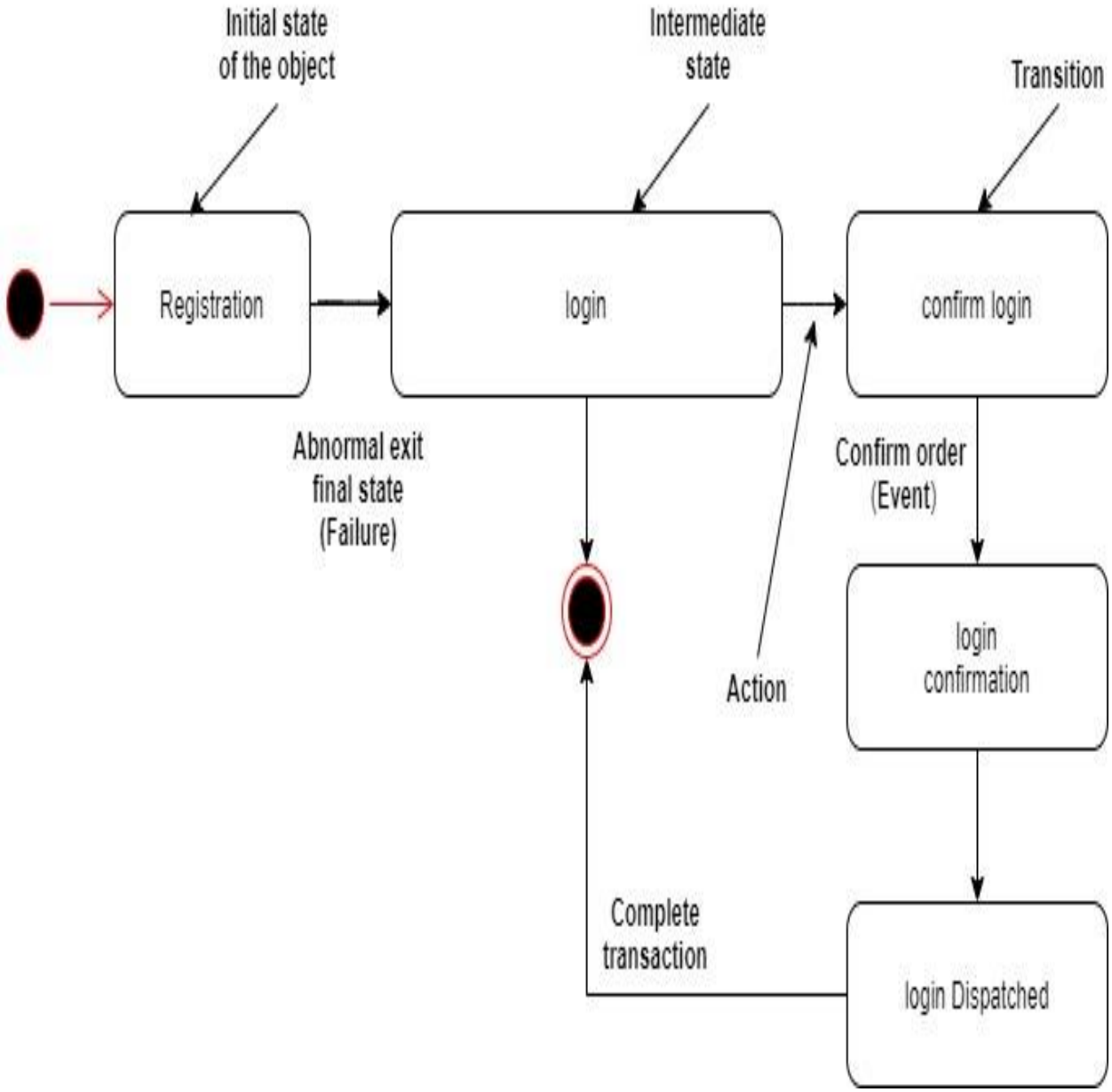
i)Item Shop:-

-This frame work describe the process of shopping and going to payment frame.

ii)Payment:-

- this frame describes the process of payment.

4.7.5 State Chart:-



- UML state chart

- Start
- In First initial state of the object comes in which registration part covered
- In Second intermediate state comes in which login part covered.
- In third transition comes in which confirm login part covered.
- In fourth confirm order(event) in which login confirmation part covered then login dispatched also done.
- After all this process done then transaction complete.
- End

Conclusion:-

We wanted to develop an web application which help Information of Covid-19 to spread.

Which will help in public awareness.

We also provide latest information through blog.

We will also sell product related covid protect.

Reference :-

FSD Books.