EX.NO:2
DATE :

ONLINE SHOPPING

AIM

To prepare necessary documents and to develop ONLINE SHOPPING with UML diagrams using Software Engineering Methodology.

PROGRAM ANALYSIS AND PROJECT PLANNING

Problem Statement:

This project ONLINE SHOPPING is to develop an application for shopping. Initially a user should choose the category whether he\she is a Customer or Administrator. If the person is an Administrator then they can add the products. If the user is a Customer then he\she can register an account, then choose and buy the products through online.

SOFTWARE REQUIREMENT ANALYSIS

The Modules in the Project:

1. Login.
2. Add Products.
3. User Registration.
4. Choose Products.
5. Calculate Total Amount.
6. Online Payment.

The first module is Login in which the user has to login to the system as a Customer or Administrator.

The next module is Add Products, where if the user is an administrator, then he\she add the new products.

The next module is User Registration the new customer can register to create an account.

The next module is Choose Products, the customer choose the products.

The next module is Calculate Total Amount, here the total amount for the products are calculated.

The last module is the Online Payment the payment is transferred through online.

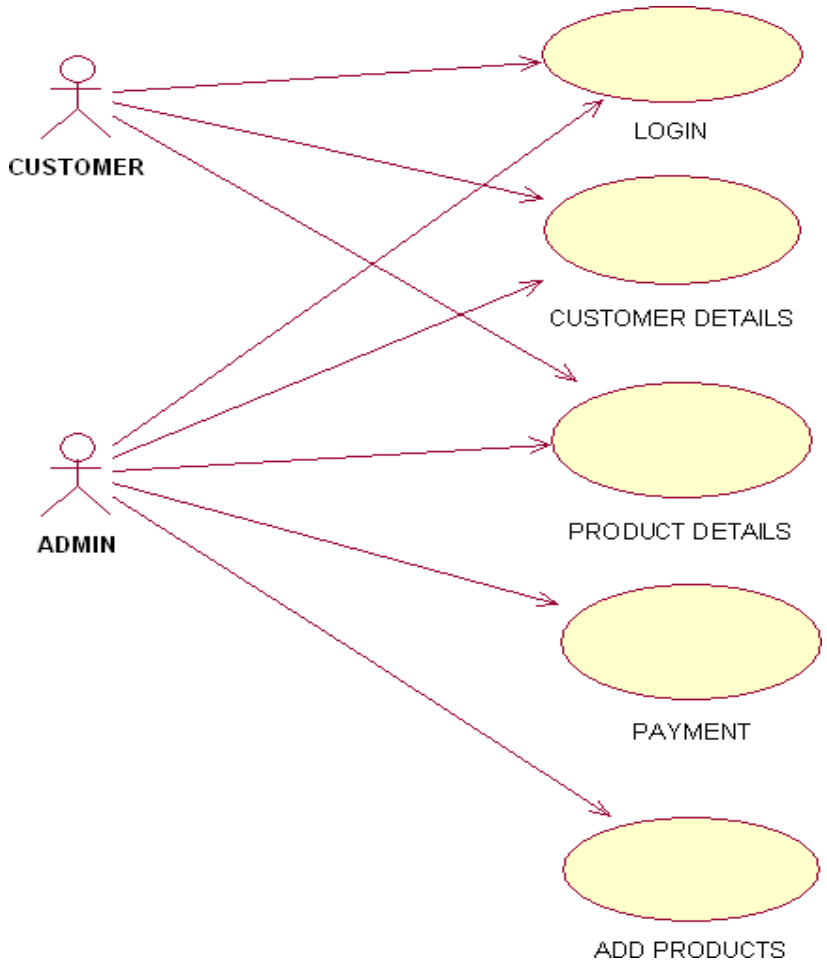
DATA MODELING

Data Dictionary:

Field Name	Description	Data Type	Field size	Default value	Validation
Name	Name of the customer	String	50	NULL	(A-Z) or (a-z)
Pass wd	Password for the customer	String	50	NULL	(A-Z) or (a-z)
dob	Date of Birth	Date/Time	8	NULL	(0-9)
Gender	Gender	String	50	NULL	(A-Z) or (a-z)
Address	Address of the customer	String	70	NULL	(A-Z) or (a-z)
Ph no	Phone no of the customer	Double	8	NULL	(0-9)
Mail id	Customer's Mail id	String	50	NULL	(A-Z) or (a-z)
Pi d	Product id	Integer	2	NULL	(0-9)
P name	Name of the Product	String	50	NULL	(A-Z) or (a-z)
P type	Type of the Product	String	50	NULL	(A-Z) or (a-z)
C name	Company Name	String	50	NULL	(A-Z) or (a-z)
price	Price of the Product	Currency	8	NULL	(0-9)
quantity	Quantity of the Product	Integer	2	NULL	(0-9)

t payment	Type of Payment	String	50	NULL	(A-Z) or (a-z)
b name	Name of the Bank	String	50	NULL	(A-Z) or (a-z)
acc no	Account number for the Bank	Integer	8	NULL	(0-9)
date	Current Date	Date	8	NULL	(0-9)

USE CASE DIAGRAM:



Use case Diagram:

Use case diagram is a graph of actors, set of use cases enclosed by a system boundary, communication association between the actors and the use cases and a generalization among the use cases.

Actor:

An actor represent a set of roles that user of a use case play when interacting with the use cases. Actor identified here is Administrator and Customer.

Use case:

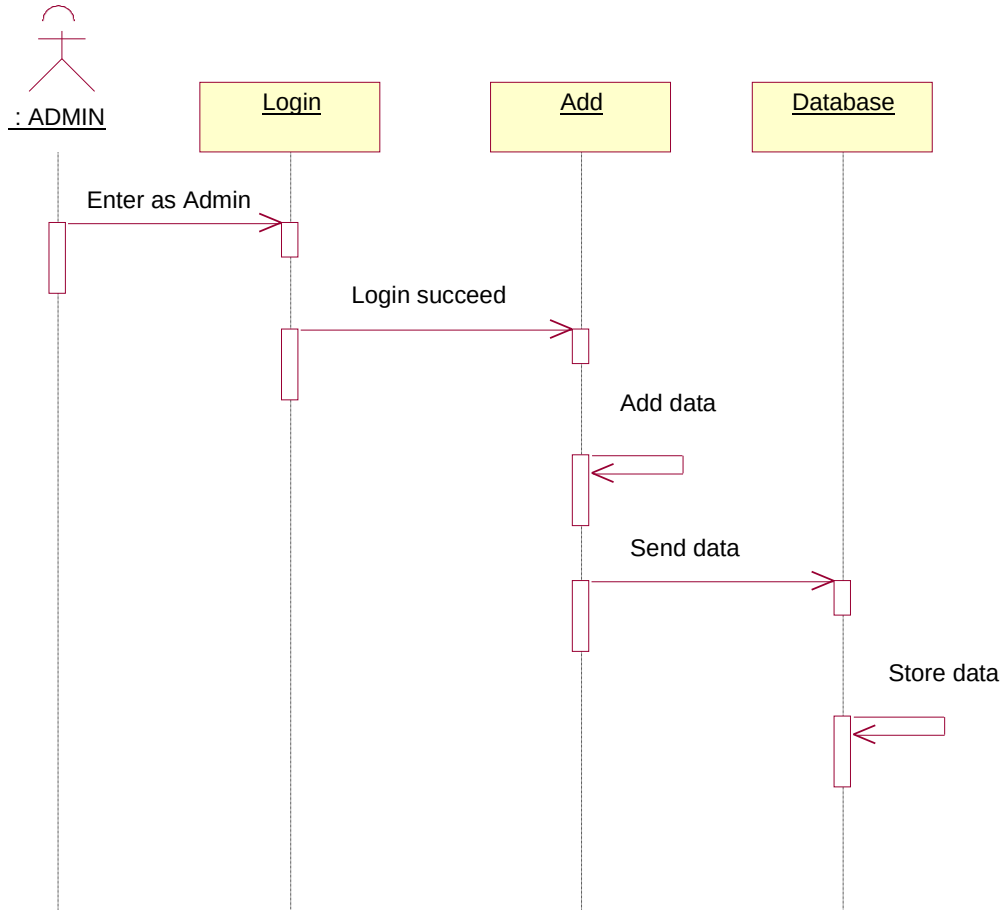
A use case is a description of a set of sequence of actions that a system performs to yield result of value to an actor.

The Use Cases described are,

1. Login
2. Customer Details
3. Product Details
4. Payment
5. Add Products

- The Login use case is to describe that, the user should choose his/her category whether he/she is a administrator or customer.
- The use case Customer Details describes that, the details of the customers are added into the database.
- The use case Product Details describes that, it displays the list of products from the database.
- The Payment use case describes that, the payment options for online shopping.
- The Add Products use case describes that the Administrator can add the new products.

SEQUENCE DIAGRAM
User: ADMIN



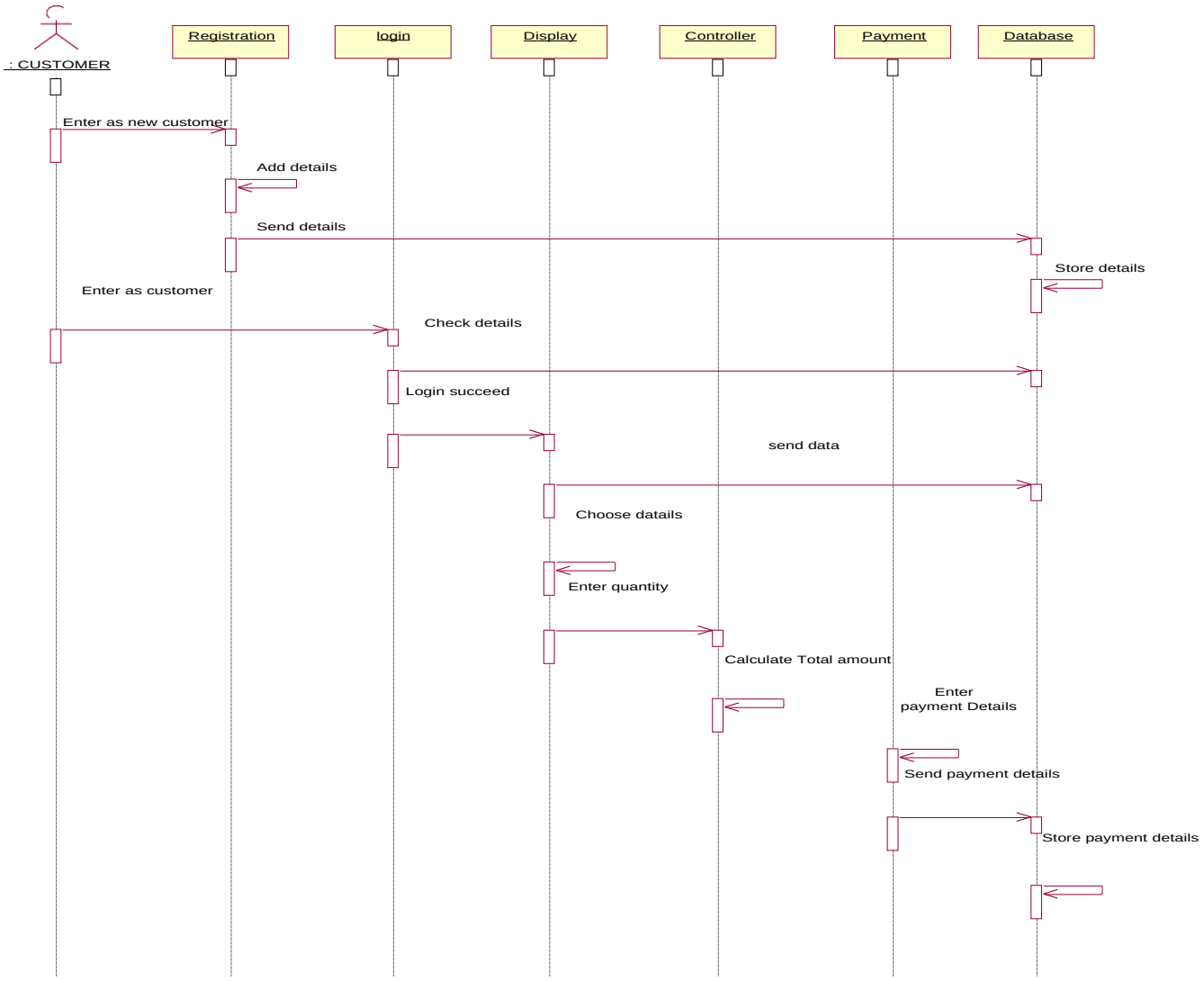
Sequence Diagram:

Sequence diagrams are easy and intuitive way of describing the behavior of a system by viewing the interaction between the system and its environment. A sequence diagram shows an interaction arranged in a time sequence.

The objects used in this sequence diagram are,

1. Login
2. Add
3. Display
4. Controller
5. Payment
6. Database

SEQUENCE DIAGRAM
User: CUSTOMER



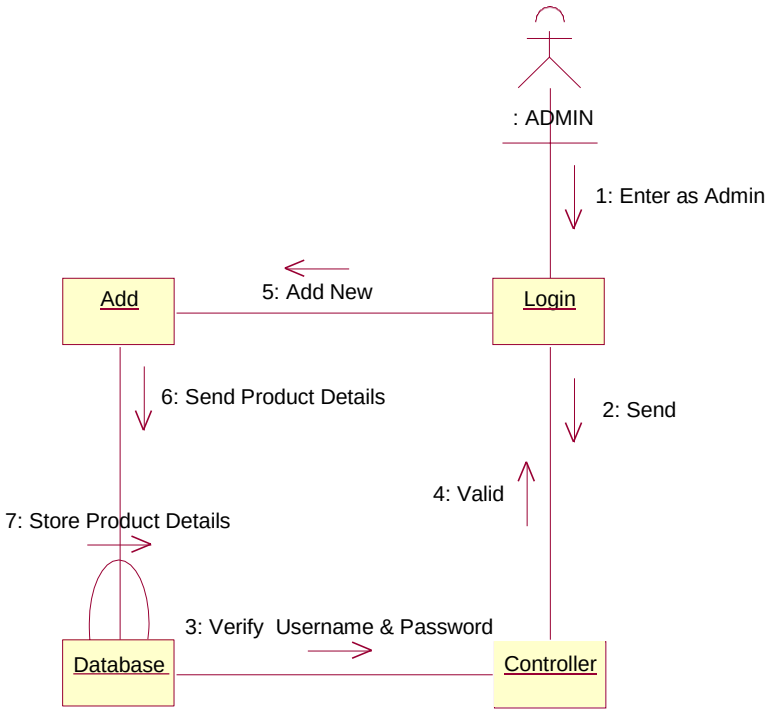
Basic Flow:

The administrator or customer enters their name and password, and the password gets checked by the system. After confirmation of the password the system allows them to access.

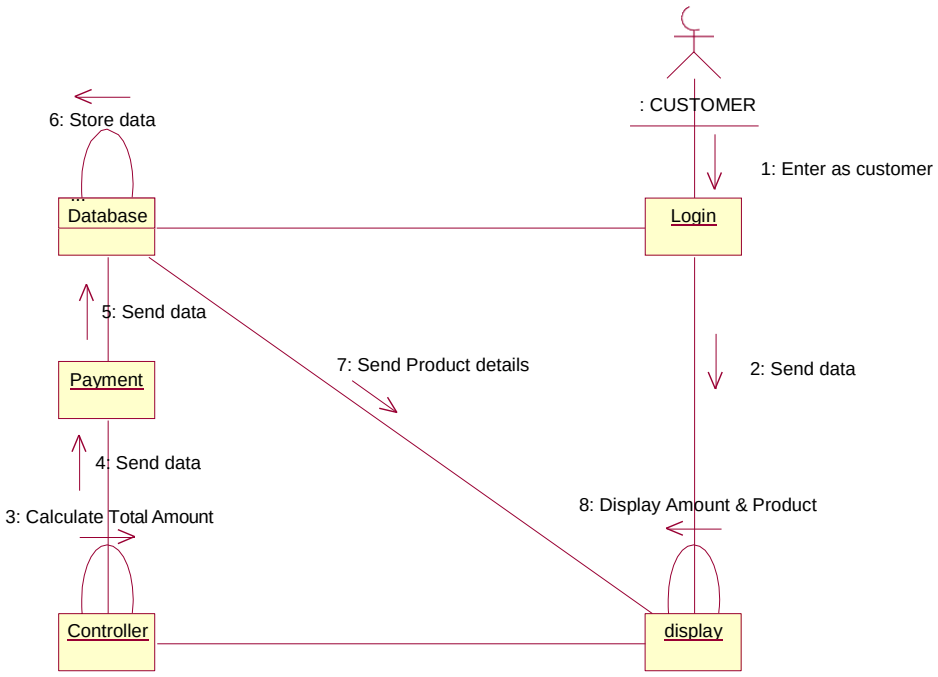
Alternate Flow:

- The actor Administrator and Customer are the persons who interacts with the system.
- The object Login makes the administrator/ customer to enter.
- The object Add will add the new products into the database.
- The object Display will display the product details from the database.
- The object Payment will allows to choose the mode of payment.
- The object Database will store all the product details & customer details.

COLLABORATION DIAGRAM :
USER: ADMIN



COLLABORATION DIAGRAM:
User: CUSTOMER



Collaboration Diagram:

A collaboration diagram represents a collaboration, which is a set of objects related in a particular context, and interaction, which is a set of messages exchanged among the objects within the collaboration to achieve a desired outcome.

Collaboration diagram shows exactly the same information as the sequence diagram. However, collaboration diagram shows this information in a different way and with different purpose.

In this collaboration diagram, the objects are represented as rectangle, the actors are stick figures. Whereas the sequence diagram illustrates the object and actor interaction overtime, the collaboration diagram shows the object and actor interaction without reference to time.

In our ONLINE SHOPPING each object interacts with each other or collaborates with each other; it gets represented by the solid line drawn between them.

ACTIVITY DIAGRAM:

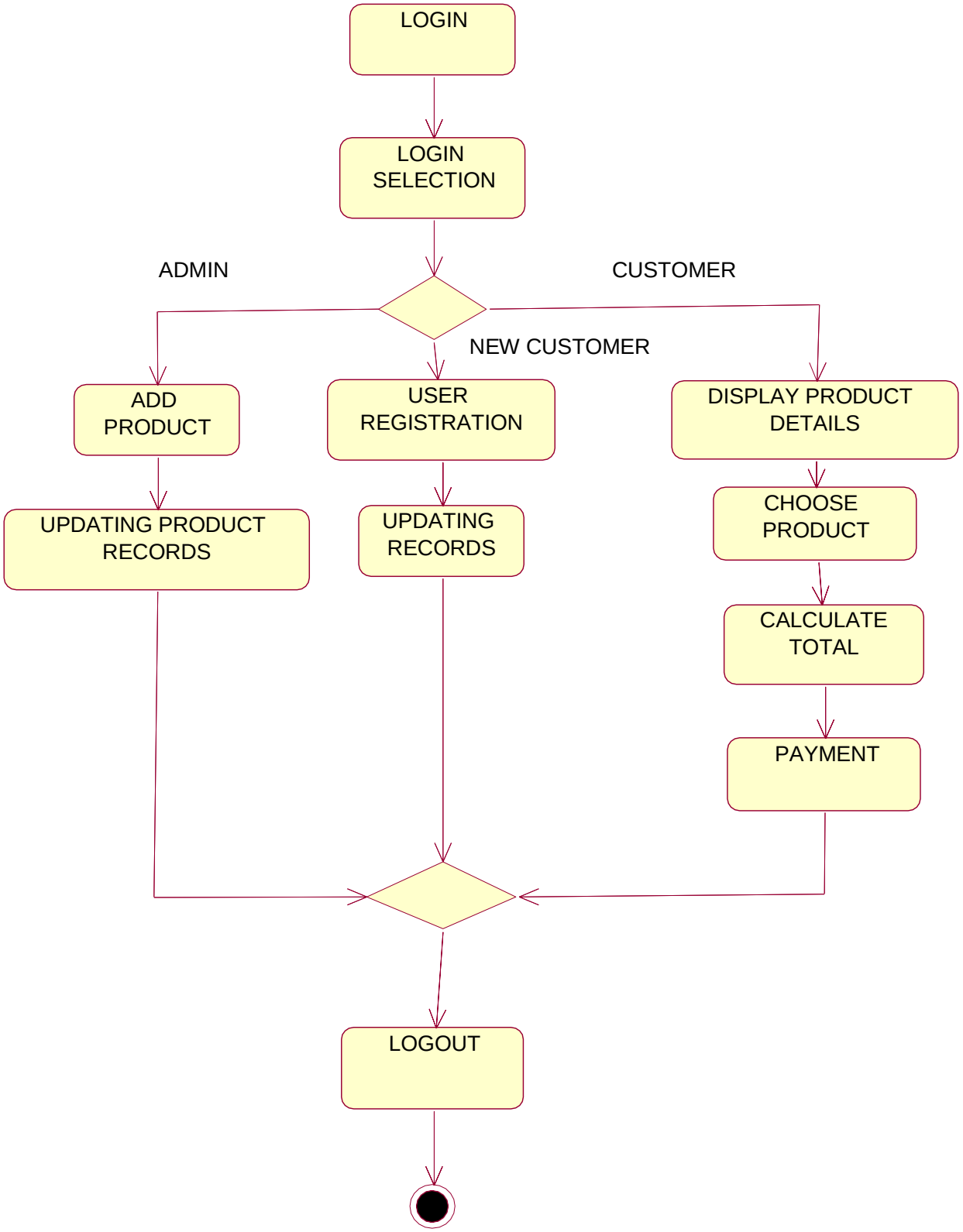
The activity diagram describes the sequencing of activities with support for both conditional and parallel behavior.

The Activity diagram is used to describe the various activities taking place in an application. Here in our ONLINE SHOPPING, we have various activities starting from login.

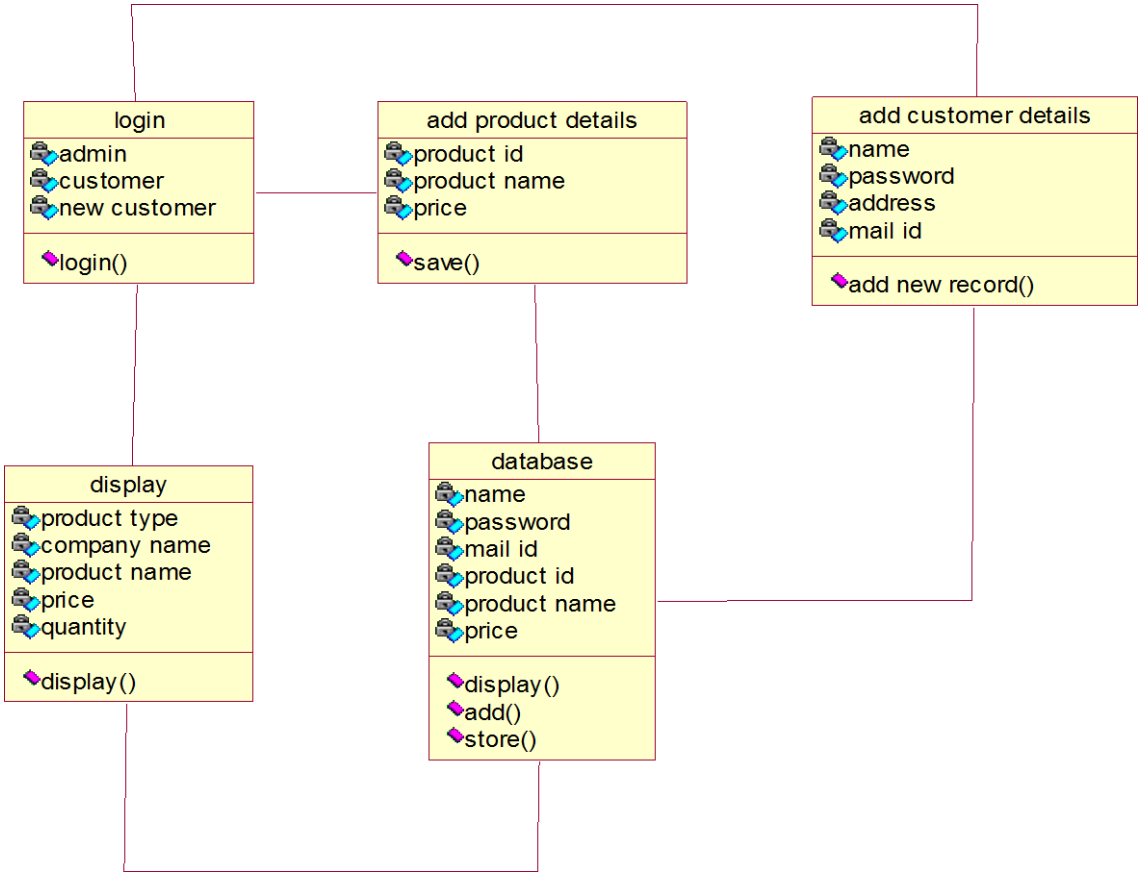
After login, the user selection activity gets performed, where the user can be an administrator or customer.

If the user is a administrator, then they have to enter their name and password and only when those details are valid they can access the system. They can add the new products which gets stored in the database.

If the user is a customer, then they login into the system using valid username and password, after login they can choose the products and also buy the products.


LOGIN
LOGIN SELECTION
ADMIN
CUSTOMER
NEW CUSTOMER
ADD PRODUCT
USER REGISTRATION
DISPLAY PRODUCT DETAILS
UPDATING PRODUCT RECORDS
UPDATING RECORDS
CHOOSE PRODUCT
CALCULATE TOTAL
PAYMENT
LOGOUT

CLASS DIAGRAM:



Class Diagram:

Class diagrams show the interactions between classes in the system. Class diagram also shows the attributes and operation of a class and the constraints that apply to the way objects are connected.

Classes contain information and behavior that acts on that information. Each class on class diagram is represented by rectangle divided into three sections. The first section shows the class name, second section shows the attributes the class contains and last section contains the operation of the class.

In our ONLINE SHOPPING, the classes identified are

1. Login
2. Add Product Details
3. Add Customer Details
4. Display
5. Database

Each class has its own attributes and operations.

Login class - The attributes defined are administrator, customer. and new customer .The method identified is login.

Add Product Details class - The attributes are product id, product name, Price.The operation defined is saving new record.

Add Customer Details class - The attributes are name, password, address, mail .The operation defined is adding new record.

Display class - The attributes are product name, company name, price, quantity.The operation identified is Display.

Database class - The attributes are name, password, mail id, product name, price.The operations defined are display, add and store.

The Solid line between the classes shows the Association relationship between them.

SOFTWARE DEVELOPMENT:

Login Form:

```
Private Sub
Command1_Click()
If Text1.Text = "rv" And Text2.Text = "mk" Then
  shopping.Show
Else
MsgBox ("Invalid input")
End If
End Sub
```

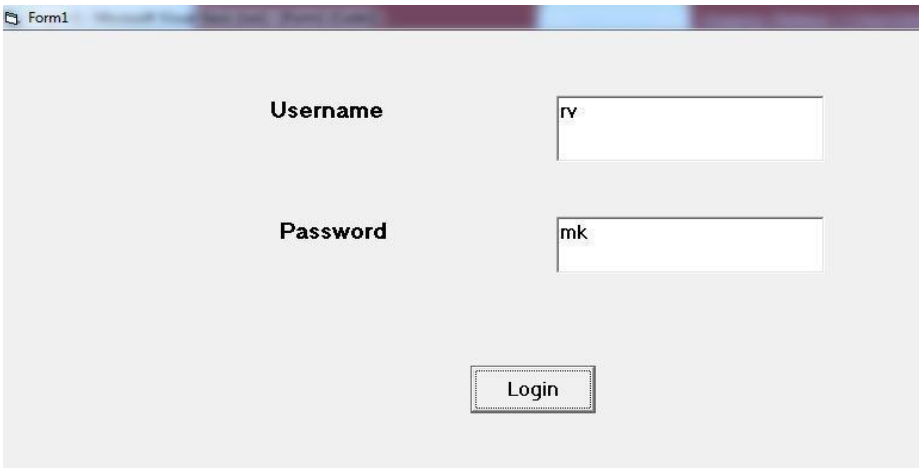
Shopping Form:

```
Private Sub buybtn_Click()
shoppingado.Recordset.Fields("name") = txtname.Text
shoppingado.Recordset.Fields("date of birth") = txtdob.Text
shoppingado.Recordset.Fields("gender") = txtgender.Text
shoppingado.Recordset.Fields("Address") = txtadd.Text
shoppingado.Recordset.Fields("Contact no") = txtphone.Text
shoppingado.Recordset.Fields("mail") = txtmail.Text
shoppingado.Recordset.Fields("product name") =
```

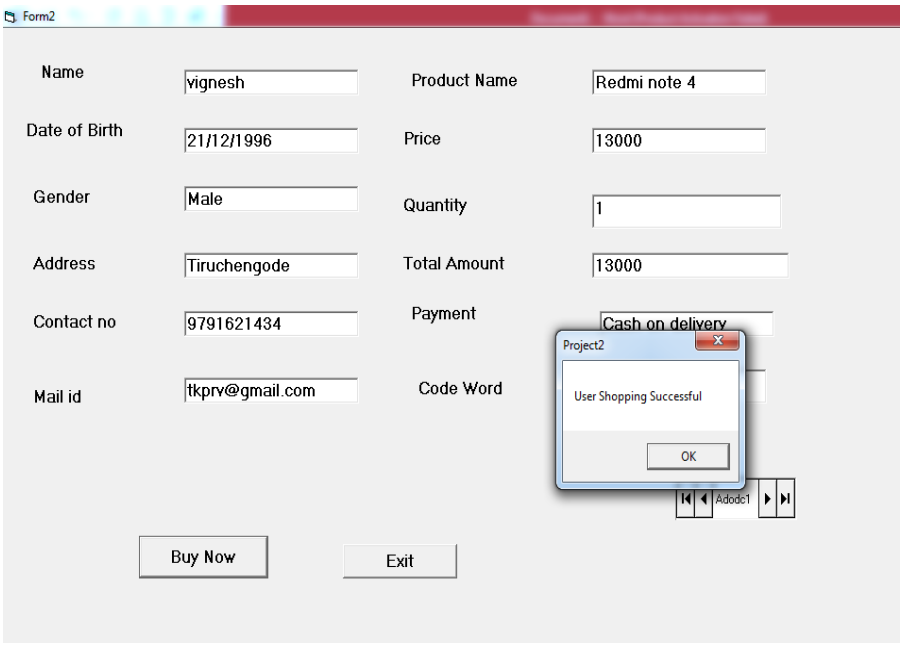
```
txtproduct.Text
shoppingado.Recordset.Fields("price") =txtprice.Text
shoppingado.Recordset.Fields("quantity") = txtquantity.Text
shoppingado.Recordset.Fields("total amount") =
txtamount.Text
shoppingado.Recordset.Fields("payment") = txtpayment.Text
shoppingado.Recordset.Fields("code word") = txtcode.Text
shoppingado.Recordset.Update
MsgBox "User Shopping Successful"
End Sub

Private Sub
Command2_Click()
End
End Sub

Private Sub Form_Load()
shoppingado.Recordset.AddNew
End Sub
```

OUTPUT:

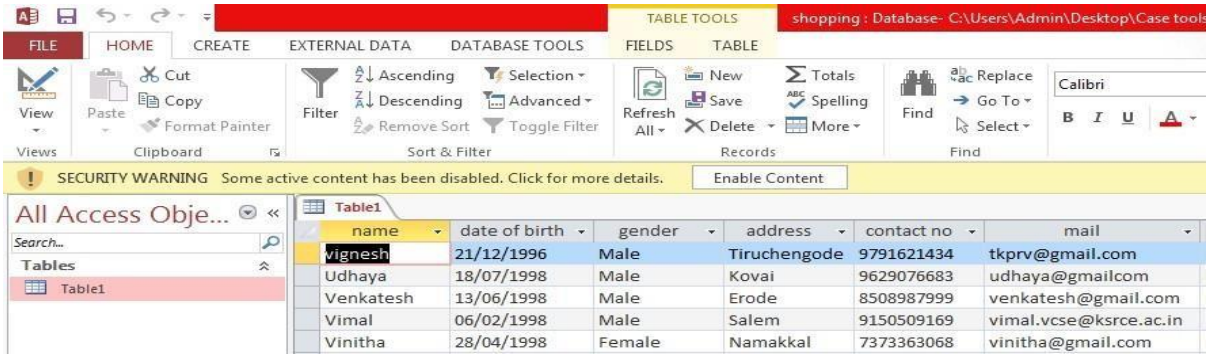
Form1:



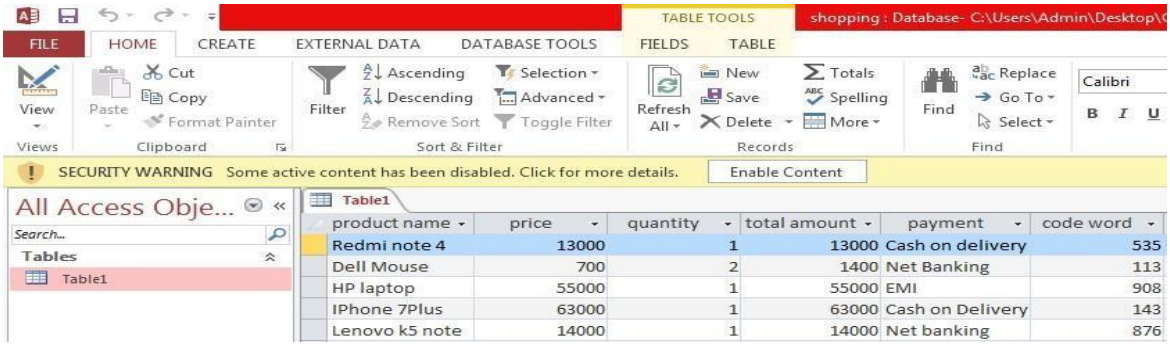
Form2:



Database Form:



name	date of birth	gender	address	contact no	mail
vignesh	21/12/1996	Male	Tiruchengode	9791621434	tkprv@gmail.com
Udhaya	18/07/1998	Male	Kovai	9629076683	udhaya@gmailcom
Venkatesh	13/06/1998	Male	Erode	8508987999	venkatesh@gmail.com
Vimal	06/02/1998	Male	Salem	9150509169	vimal.vcse@ksrce.ac.in
Vinitha	28/04/1998	Female	Namakkal	7373363068	vinitha@gmail.com



product name	price	quantity	total amount	payment	code word
Redmi note 4	13000	1	13000	Cash on delivery	535
Dell Mouse	700	2	1400	Net Banking	113
HP laptop	55000	1	55000	EMI	908
IPhone 7Plus	63000	1	63000	Cash on Delivery	143
Lenovo k5 note	14000	1	14000	Net banking	876

RESULT:

Thus the ONLINE SHOPPING is developed with all necessary documents and UML diagrams using Software Engineering methodology.