Online Grocery shopping system "banana market"

team#13	Student Name	section	ID	Responsibilities
1	Sara Al-meshari	46470	435202339	Description+usecase diagram
2	Al-zahra Al-howaish	46470	435201808	sequence diagram +use case description
3	Renad Al-malki	46472	435201240	composite diagram
4	Nora Al-khunaifer	47856	435201094	state diagram+class diagram

Instructor Name: L. Nora al-zamil

1-Description :

1.1-overview

The banana market is a grocery system deliver customer product choice to their current location .
The system consists of central server connecting several smart phones carried by customers .
And the market manager (admin)can add new product.
Customer can create account, view product ,add product to shopping cart ,place the order ,cancel order .
A market manager can add product to database in server by scan the barcodes Then The system will retrieve product info (perf description, component, calories..)from stowage system and let manager enter available quantity and product price ,mangers can delete product or update them.
The customer can view the lists of product by its categories before sign in but he/she can't add any of product to shopping cart when he/she in guest mode .
The customer can create account by email address ,name and phone number. When customer sign-up the system send verification message to his phone number then order him to enter the verify code to complete their sign-up .

1.2-Scope:

everyone this days need online shopping because several reasons and the most important reason of implementing this app is unified the price for all customers .
Objective of this system, is to provide Online Grocery Shopping solution to consumers and vendors. It will automate some of the basic operations of an online store. Scope would be to provide basic functionalities using an mobile application so that those manual process can be automated. It will include to provide administration access to vendors and admins and user specific access to customers. This system can be accessed by customers who are in Riyadh.

1.3-Functional requirement:

1. The system shall display detailed information of the selected products.
2. The system shall enable user to search for product .
3. The system shall send an order confirmation to the user through Email.
4. The system shall send an phone number confirmation to the user through SMS.
5. The system shall allow user to add products to the shopping cart.

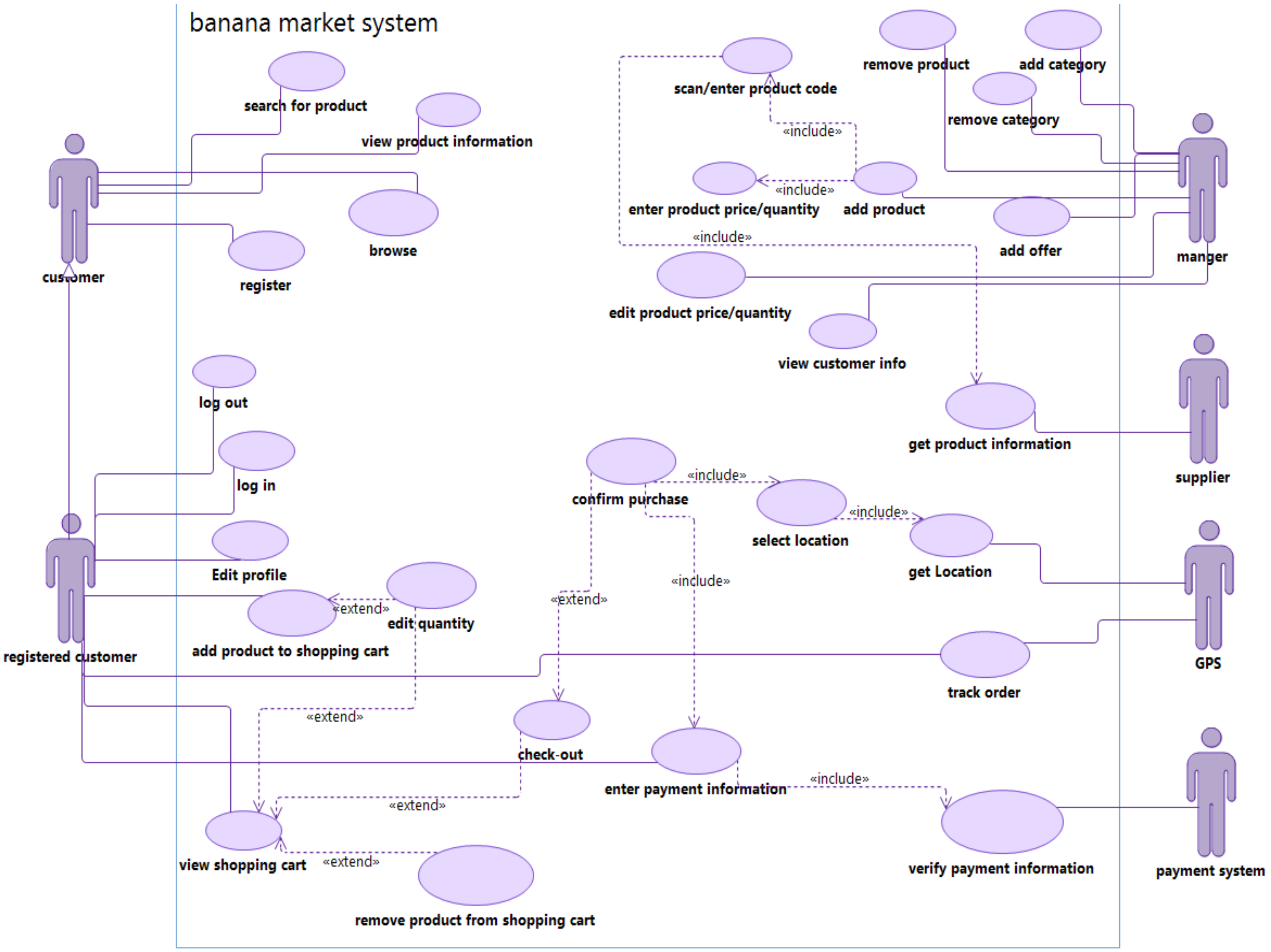
6. The system shall allow user to remove products from the shopping cart.
7. The system shall allow user to confirm the purchase.
8. The system shall enable user to enter the payment information.
9. The system shall display the detailed information about the selected order.
10. The system shall allow user to edit the profile information.
11. The system should allow unregistered user to search for view products.
12. The system shall allow user to edit quantity of the selected product.
13. The system should ask user to select specific location to ship to.
14. The system shall allow user to scan barcode of needed product.
15. The system shall provide a digital image for each product in the product catalog
16. The system shall allow users to track their order.

1.4-Non Functional requirement :

1. The system shall provide multi language support.
2. The users shall be able to add an item to the cart in fewer than 5 seconds .
3. The users shall be able to view information about an item in fewer than 5 seconds.
4. The system shall be available 99% per hour.
5. The application should be run in any Android and IOS environment .
6. The system should not consume more than 10% per hour from mobile battery.
7. The size of the application shall be 10 megaByte to save a phone storage .

2-System Use Case Diagram and Use-cases:

3.1 use case diagram:



2.2.use case description:

1. Add product to shopping cart use-case

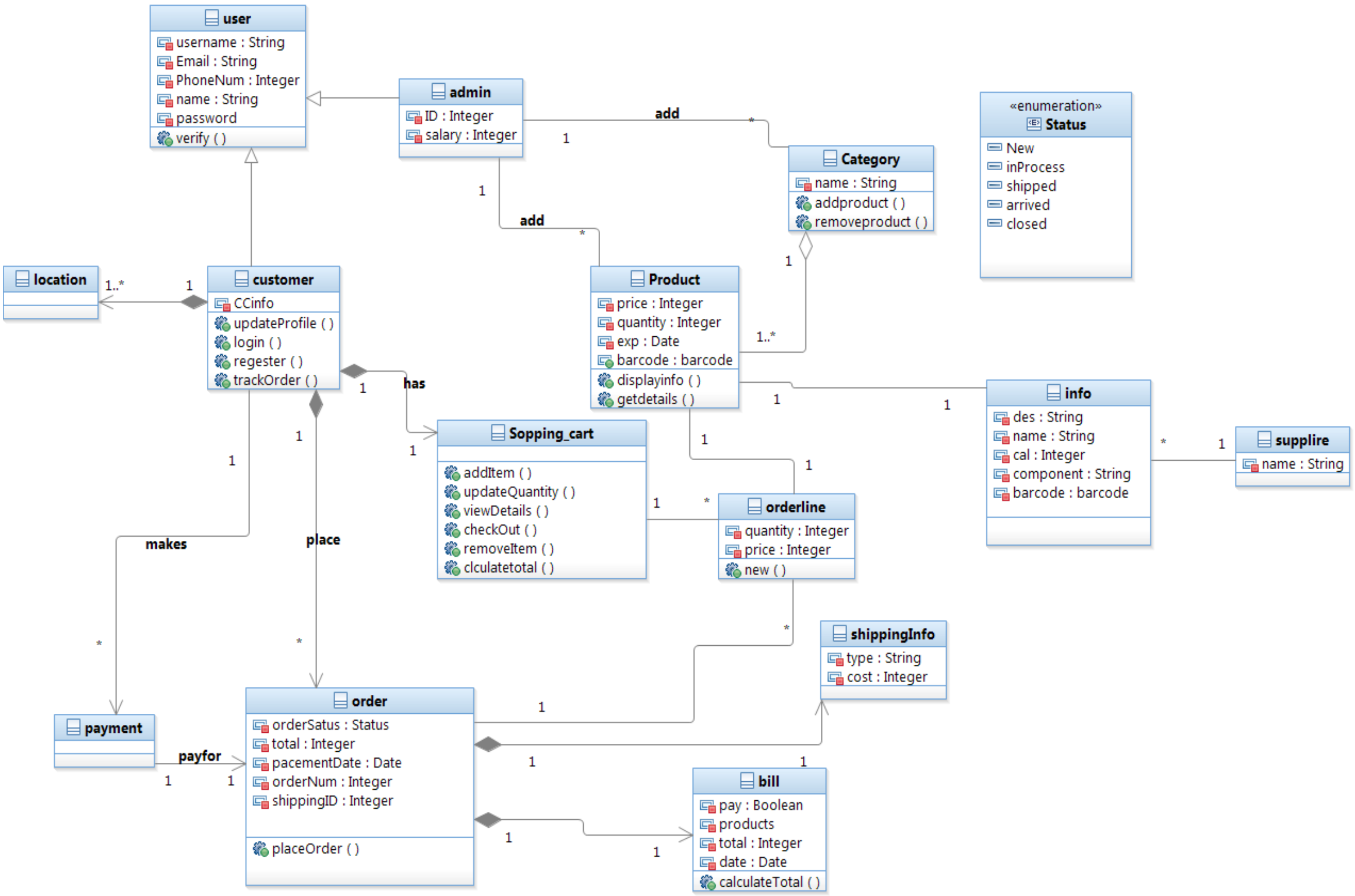
System: Online Grocery shopping system”banana market”	
Use Case name: Add product to shopping cart	
Primary Actor: Registered customer	**Other Actors:**
Description: This use case describes how the Register customer add items to shopping cart on the system.	
Relationships: **Includes:** none **Extends:** none	
Pre-conditions: 1.The Register customer must be logged in successfully.	
Steps:	
Actor: Registered customer	**System:** banana market application
1.The customer select one product . 3.The customer enter the quantity . 4.The customer select add to cart .	2.The system display the product information along with its price ,picture , available quantity and some details. 5.The system display success message with check-out and continue shopping options .
Alternative and exceptional flows: 1.The wanted quantity is not available if the customer in step 5 entered quantity number more the available or limit specified by manger . 1.1 The system display limit exceeded message with available quantity number . 2.Product has more than one size or shape	

after step 4 and before step 6 if product has more than one size or shape
2.1 The customer must select wanted product .

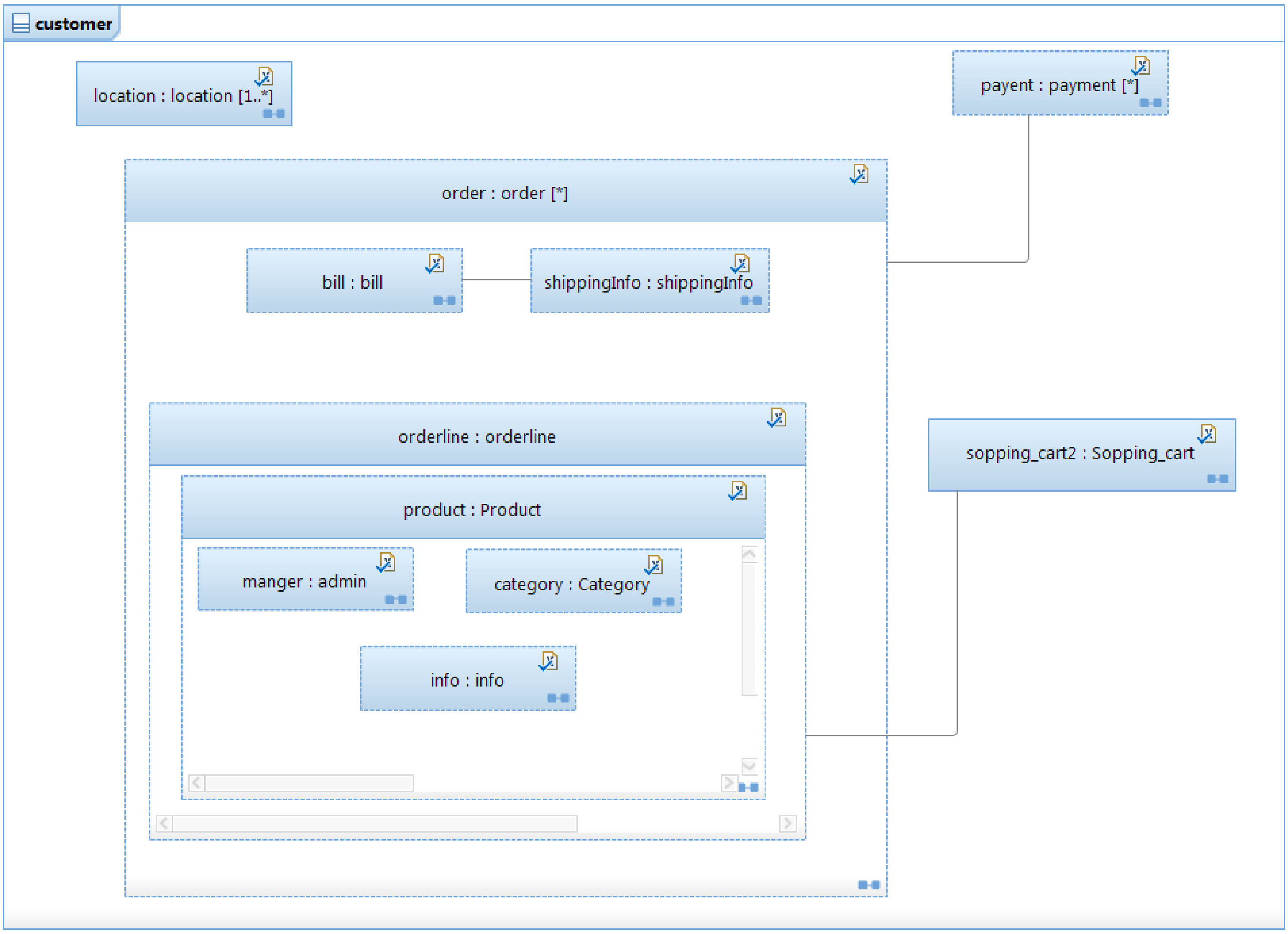
3. Registered customer quits
If at any step before step 4 the user selects cancel
3.1.The use case ends with a failure condition

Post Conditions:
●**Successful condition:** The system successfully Added the product to shopping cart on the Online Grocery shopping system .
●**Failure Condition:** No items added to shopping cart .

3-class diagram :



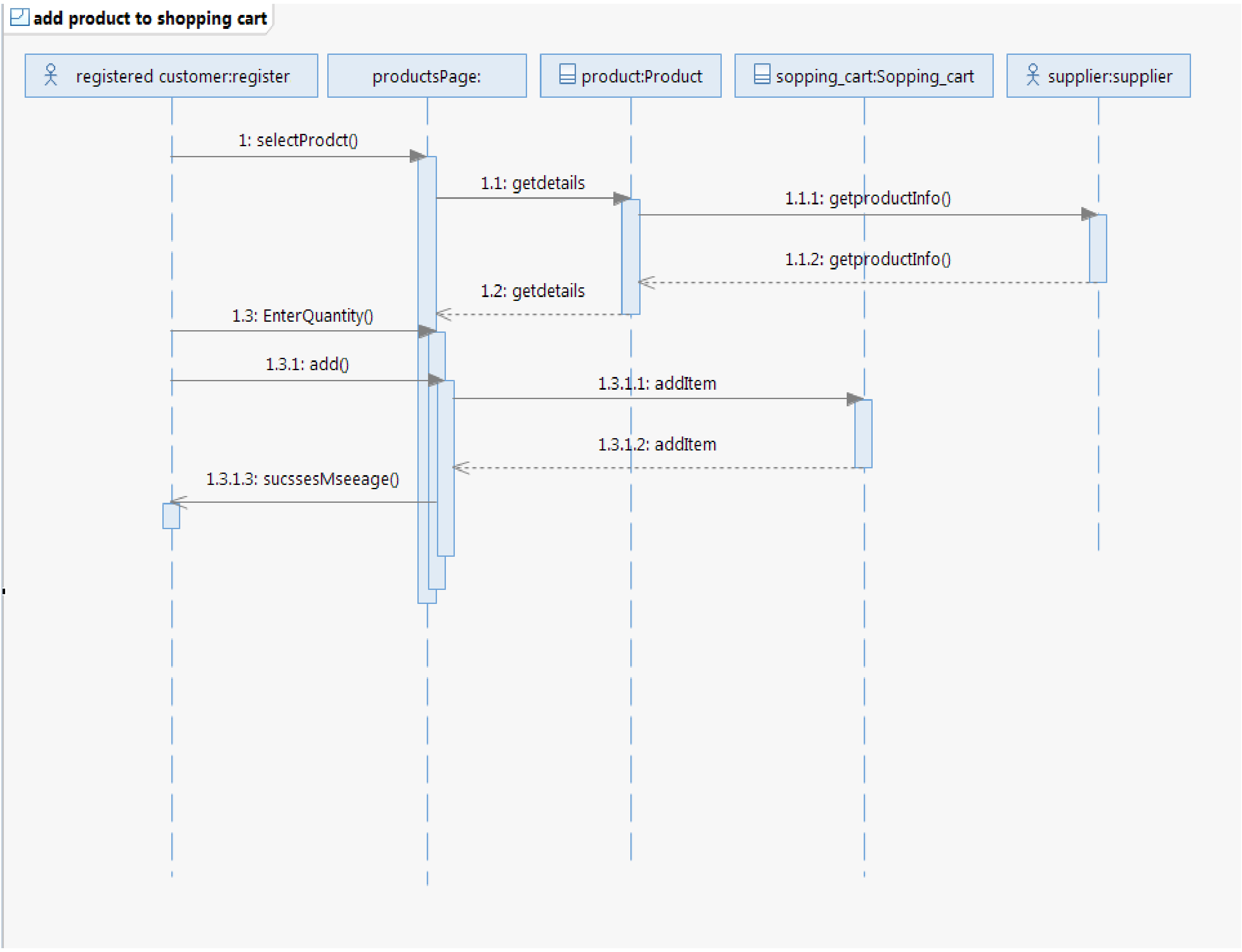
4-Composite Diagram :



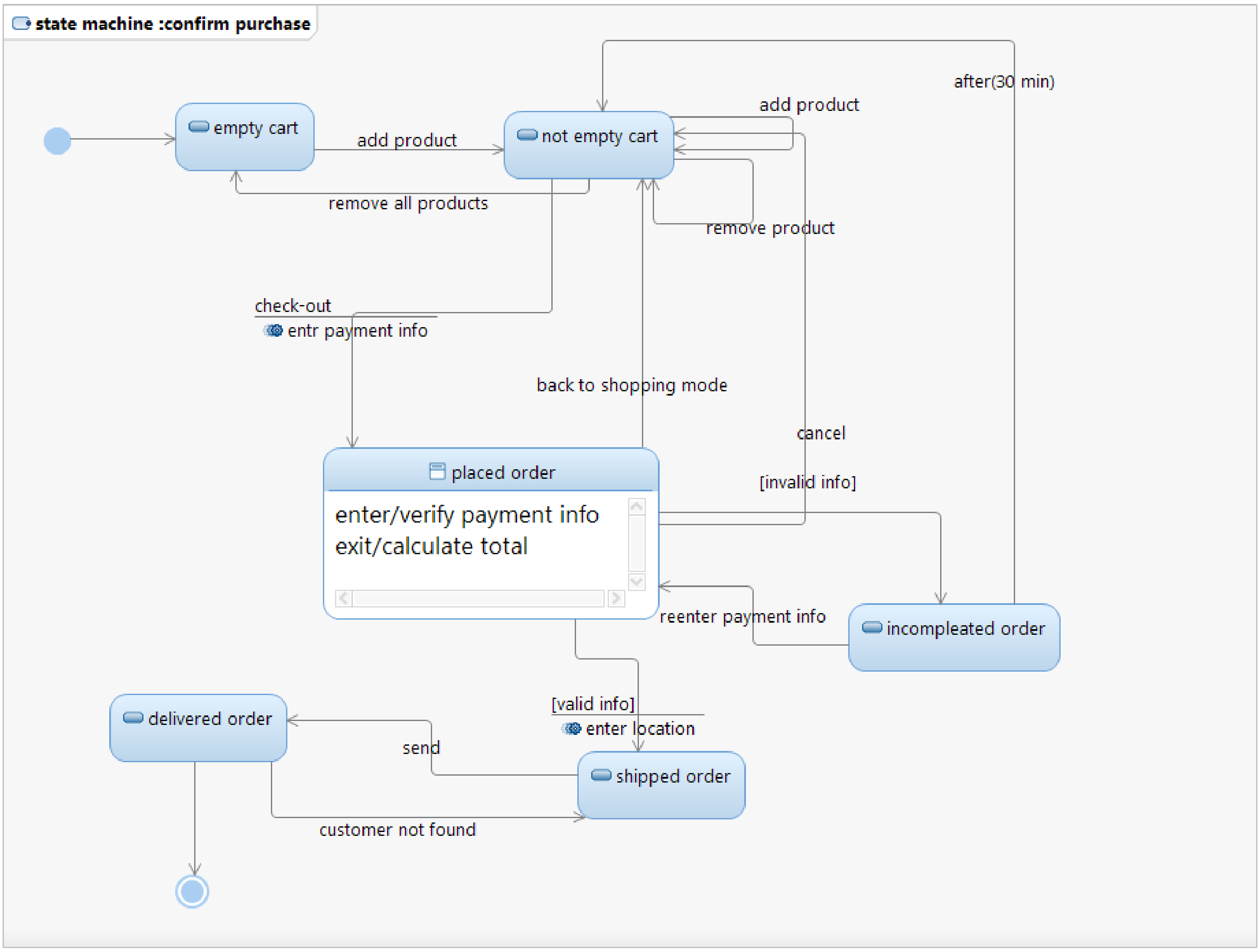
why ?

we need the composite diagram on “Customer” class because it’s complex class and has many different kind of association so we need it to show the internal structure including its interaction points and relationship to other parts of the system.

5- Sequence Diagram



6- State Diagram :



7- Java Code:

That are some classes and the rest in CD

```
package classD;

import java.util.Set;

/**
* @author nora
*/
public class user {

	private customer account;
	private admin manger;
	private String username;
	private String Email;
	private Integer PhoneNum;
	private String name;
	private Object password;
	public Boolean verify() {
		// begin-user-code
		// TODO Auto-generated method stub
		return null;
		// end-user-code
	}
	public void setPassword(Object password) {
		// begin-user-code
		this.password = password;
		// end-user-code
{
{

public class supplire {

	private String name;


	public String getName() {
		// begin-user-code
		return name;
		// end-user-code
	}


	public void setName(String name) {
		// begin-user-code
		this.name = name;
		// end-user-code
	}
```

```
    private Set<info> info;


    public Set<info> getInfo() {
        // begin-user-code
        return info;
        // end-user-code
    }


    public void setInfo(Set<info> info) {
        // begin-user-code
        this.info = info;
        // end-user-code
    }
}

public class Sopping_cart {

    private Set<Product> product;


    public Set<Product> getProduct() {
        return product;
    }


    public void setProduct(Set<Product> product) {
        this.product = product;
    }


    private Set<orderline> orderline;


    public Set<orderline> getOrderline() {
        return orderline;

    }


    public void setOrderline(Set<orderline> orderline) {
                this.orderline = orderline;
            }

    public void addItem(Integer id) {

    }

    public void updateQuantity(Integer n) {
    }

    public void viewDetails() {
```

```
        }

        public void checkOut() {
        }

        public void removeItem() {

        }

        public Integer clculatetotal() {
                return null;

        }
}
public class payment {

        private order order;


        public order getOrder() {
                // begin-user-code
                return order;
                // end-user-code
        }


        public void setOrder(order order) {
                // begin-user-code
                this.order = order;
                // end-user-code
        }
}
```

8-references :

1. nana direct https://www.nana.sa/en
2. al danube https://play.google.com/store/apps/details?id=com.app.danube&hl=en
3. TSC Online Grocery Shopping http://www.sultan-center.com/
4. Object-Oriented Software Engineering, 2nd edition , Timothy C.lethbridge, Robert Laganiere , March 2005.
5. Software Modeling & Design UML, Use Cases, Patterns and Software Architectures ,Hassan Gomaa,2011.