GROUP NAME: NEOPHYTES

PROJECT NAME: ONLINE GROCERY SYSTEM

**A textual description of the domain**

**1 Brief Description**

This use case describes how the user interact with the Online Grocery System (OGS).

**2 Actors**

2.1 New User

2.2 Registered User

2.3 System Admin

**3 Preconditions**

The New User and the Registered User has access to internet connection.

**4 Basic Flow of Events**

1. The use case begins when the new user or the registered user access the Online Grocery System and System Admin is able to manage the system.

2. The new user is able to view the different departments and can register with Online Grocery System but cannot place the order or make a payment.

3. The registered user cannot register again but can place the order and make a payment.

4. Both registered and new users can view the items listed under a specific Department.

5. Both registered and new users can search the items under a specific Department.

6. The system asks for the customer authentication for adding items to the cart.

7. Only the registered user can add the items to the cart.

8. Before making the payment, users can view or update the shopping cart, user Authentication is required before choosing a payment option.

9. Users can choose either of the two payment options i.e. Pay Using Card or Cash on Delivery.

10. User can logout of the system.

10. System Admin can View Registered Users, Modify Departments, Modify items and view History.

**5 Alternative Flows**

**5.1 User Already Exists**

If in step 2, the user registers with an already existing email address in the database, system prompts user already exists and the use case redirected to step 2.

**5.2 Item Unavailable**

If in step 4, the item listed under a department is not available at the store, then the system displays the message, “Item Temporarily Unavailable.”

**5.3 Invalid User**

If in step 6, if user validation is failed, then the user is redirected to step 6 and asked to re-login.

**5.4 Minimum Purchase**

If in step 9 of the basic flow the total amount of all the items in the cart is less than $30, then

1. The system shall display the message "Minimum purchase should be $30 or more".

2. The use case resumes at step 7.

**5.5 Payment and Delivery**

If in step 9 in the basic flow, the Customer chooses the cash on delivery or Pay using card and enters the address more than 10 km from the store, then

1. The system shall display the message “Cannot deliver to the address more than 10 km away.”

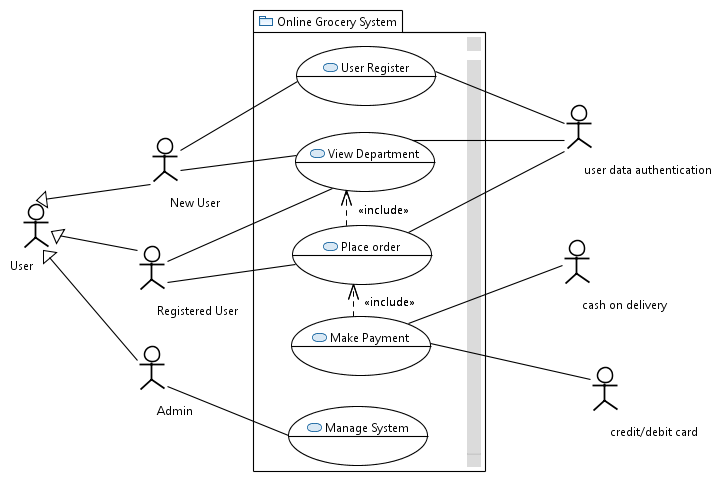
2. The use case resumes at step 9.

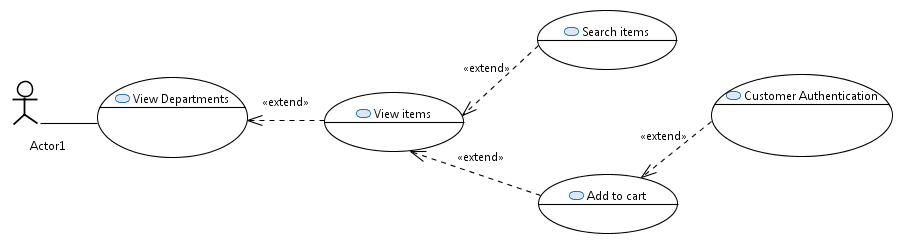
If in step 9 in the basic flow, the Customer chooses the cash on delivery or Pay using card and choose the expected time of delivery to be after 9 pm, then

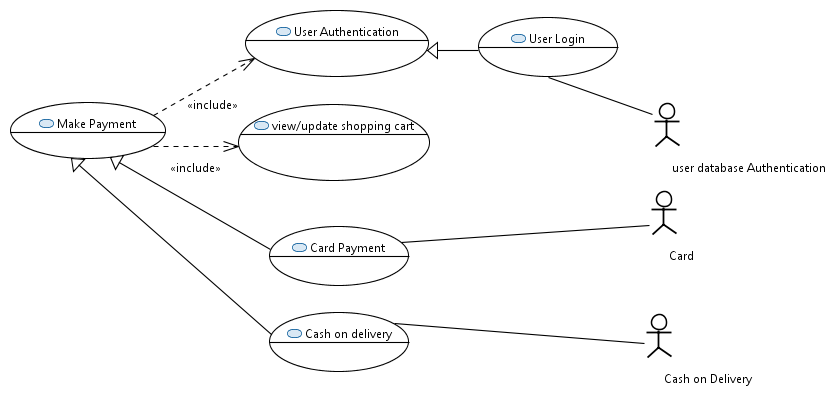
1. The system shall display the message, “We deliver till 9:00 pm”

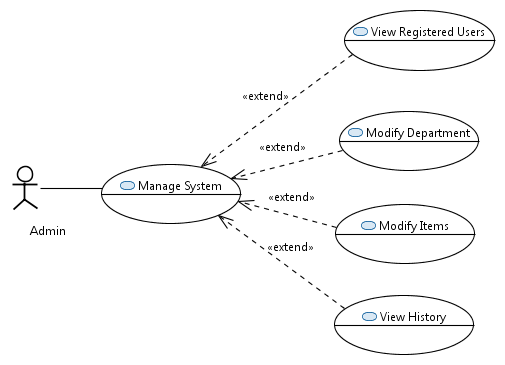
2. The use case resumes at step 9.

**USE CASE DIAGRAMS**

****

****

****

****

**Functional requirements:**

**User**

The users shall be able to view the departments on the home page of an application.

The users shall be able to view items in various departments.

The users shall be able to view details of the item.

The users shall be able to add items in the shopping cart.

The users shall be able to view the items added to the shopping cart.

The users shall be able to delete items from the cart.

The users shall login or register using user authentication form.

The users shall place an order only when login or registration is complete.

**Admin**

The admin shall be able to view all the registered users’ details.

The admin shall be able to add/delete/modify departments.

The admin shall be able to add/delete/modify items in various departments.

The admin shall be able to view the entire history of successfully placed orders.

**Non-functional requirements:**

**Reliability**: The system should be able to perform its required functions under stated conditions for a specific period of time.

**Performance**: The system should response quickly and not degrade in its operation with time.

**Usability:** The system should be easy to learn, operate and provide assistance to the user.

**Maintainability:** The system should be designed in a way that it supports maintainability.

**Security:** The system should not allow unauthorised access.