EXPERIMENT 2

Draw a coffe coffe day ordering system. A coffe coffee day shop vending machine dispenses coffee to customers. Customers order coffee by selecting a recipe from a set of recipes. Customers pay for the coffee using coins. Change is given back, if any, to the customers. The ‘service assistant’ loads ingredients (coffee powder, milk, sugar, water, chocolate) into the coffee machine. The ‘service assistant’ adds recipe by indicating the name of the coffee, the units of coffee powder, milk, sugar, water, chocolate to be added as well as the cost of the coffee. The service assistant can also edit and delete a recipe. Develop the use case diagram for the specification above.

AIM:

To design a **UML Use Case Diagram** for a **Coffee Day Ordering System**, representing the interactions between the **Customer, Vending Machine, and Service Assistant** in the process of ordering, payment, and recipe management.

PROCEDURE:

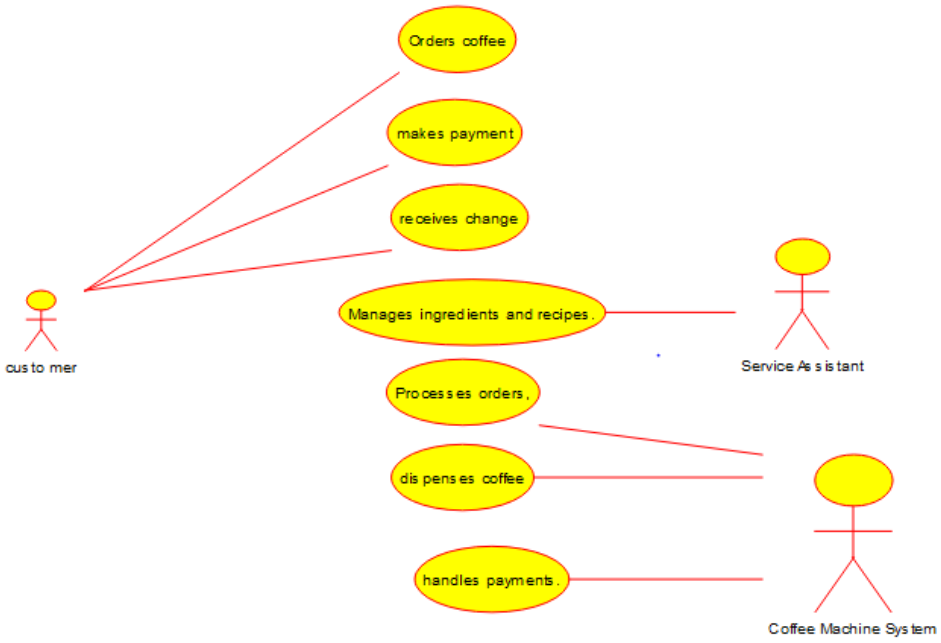
1. **Identify the Actors**
    - **Customer**: Orders coffee, makes payment, and receives change.
    - **Service Assistant**: Manages ingredients and recipes.
    - **Coffee Machine System**: Handles coffee dispensing, payment processing, and order management.
2. **Define the Use Cases**
    - **Orders Coffee**: The customer selects a coffee recipe.
    - **Makes Payment**: The customer pays for the selected coffee.
    - **Receives Change**: If there is an overpayment, the machine returns the change.
    - **Dispenses Coffee**: The machine serves coffee as per the selected recipe.
    - **Handles Payments**: The system processes the payment for the coffee.
    - **Processes Orders**: The coffee machine system handles the customer’s order.
    - **Manages Ingredients and Recipes**: The service assistant loads ingredients into the machine and updates coffee recipes.
3. **Connect the Use Cases to Actors**
    - Draw associations (lines) between actors and their relevant use cases.
    - For example, connect **Customer** to "Orders Coffee," "Makes Payment," and "Receives Change."
    - Connect **Coffee Machine System** to "Dispenses Coffee," "Processes Orders," and "Handles Payments."
    - Connect **Service Assistant** to "Manages Ingredients and Recipes."
4. **Highlight System Functions**

- Group related use cases under the **Coffee Machine System**.
- Ensure use cases like "Handles Payments" and "Dispenses Coffee" are part of the system's core functionalities.

5. **Review and Finalize**
   - Ensure all actors are connected to at least one use case.
   - Verify that all interactions and processes are logically represented.

OUTPUT:

USECASE DIAGRAM:



RESULT:

The UML Use Case Diagram for the Coffee Day Ordering System is successfully created.
It shows interactions between the Customer, Coffee Machine System, and Service Assistant.
The diagram clearly represents coffee ordering, payment, and recipe management processes.