

**Submitted By:**

**Uqba Gulzar**

**Submitted To:**

**Engr. Muhammad Shehzad**

**Reg No:**

**2023-BSE-067**

**Section:**

**BSE-IVB**

**Department:**

**Software Engineering**

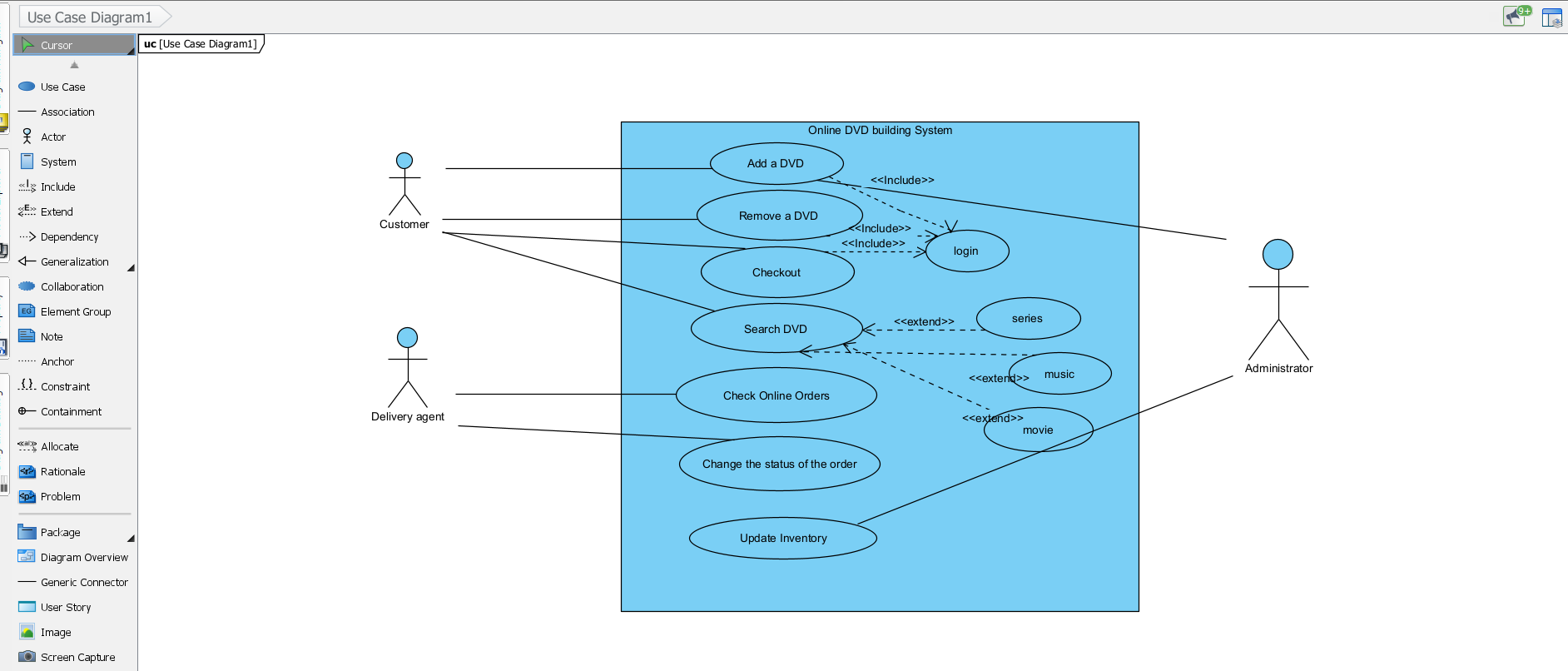
**Course Title:**

**SDA**

**LAB 04:**

1. Case Study

Create the UML Use Case Diagram for a system to buy DVDs online. Customers, Delivery Agent, and Administrators of the System will use the system. They can perform the following actions in the system: Customer: • Add a DVD to shopping cart (requires the user to log in) • Remove a DVD from shopping cart (requires the user to log in) • Check out shopping cart (requires the user to log in) • Review contents of shopping cart (requires the user to log in) • Search DVDs by Title and Category (Some examples of categories are: Series, Movies, Music) Delivery Agent: • Check online orders • Change the status of the order to delivered Administrator: • Add DVD • Update inventory of DVDs Note: Use the “include” and “extend” relationships as needed.



1. Case Study

When the Coffee Maker is not in use it waits for user input. There are six different options of user input: 1) add recipe, 2) delete a recipe, 3) edit a recipe, 4) add inventory, 5) check inventory, and 6) purchase beverage. Only three recipes may be added to the Coffeemaker. A recipe consists of a name, price, and units of coffee, units of milk, units of sugar, and units of chocolate. Each recipe name must be unique in the recipe list. Price must be handled as an integer. A status message is printed to specify if the recipe was successfully added or not. Upon completion, the Coffeemaker is returned to the waiting state. Software Design And Architecture 4th Semester-SE A recipe may be deleted from the Coffeemaker if it exists in the list of recipes in the Coffeemaker. The recipes are listed by their name. Upon completion, a status message is printed and the Coffee Maker is returned to the waiting state. A recipe may be edited in the Coffeemaker if it exists in the list of recipes in the Coffeemaker. The recipes are listed by their name. After selecting a recipe to edit, the user will then enter the new recipe information. A recipe name may not be changed. Upon completion, a status message is printed and the Coffee Maker is returned to the waiting state. Inventory may be added to the machine at any time from the main menu, and is added to the current inventory in the Coffeemaker. The types of inventory in the Coffeemaker are coffee, milk, sugar, and chocolate. The inventory is measured in integer units. Inventory may only be removed from the Coffeemaker by purchasing a beverage. Upon completion, a status message is printed and the Coffeemaker is returned to the waiting state. Inventory may be checked at any time from the main menu. The units of each item in the inventory are displayed. Upon completion, the Coffee Maker is returned to the waiting state. The user selects a beverage and inserts an amount of money. The money must be an integer. If the beverage is in the Recipe Book and the user paid enough money the beverage will be dispensed and any change will be returned. The user will not be able to purchase a beverage if they do not deposit enough money into the Coffeemaker. A user's money will be returned if there are not enough inventories to make the beverage. Upon completion, the Coffee Maker displays a message about the purchase status and is returned to the main menu. Identify: Actors of the system Identify Use Cases that appears as verbs. Use Include and Extend relationship where needed Draw a complete Use Case diagram of the system

▪ Actors: User

▪ Use Cases: Add Recipe, Delete Recipe, Edit Recipe, Add Inventory, Check Inventory, Purchase Beverage

▪ Include: "Check Inventory" is part of the "Purchase Beverage" use case.

▪ Extend: "Return Money" and "Return Change" are extensions to "Purchase Beverage" when there are issues with money or inventory

