

**Use Case Name:** Register

**Actors:** User

**Description:**

This use case describes the process by which a user registers for an account within the system.

**Preconditions:**

The user accesses the registration page of the system.

**Flow of Events:**

1. **User Registration:**

The user navigates to the registration page.

The system prompts the user to enter required information such as username, email address, and password.

1. **Enter Required Information:**

The user inputs the necessary information into the designated fields.

The required information typically includes username, email address, and password.

1. **Verify Information and Check Existing User:**

Once the user has entered all required information, the system verifies the information provided.

The system checks if the username and email address provided by the user are unique and not already registered in the system.

If the information provided is valid and unique, the system proceeds to create a new user account.

If the username or email address is already associated with an existing account, the system notifies the user to choose a different username or email address.

**Postconditions:**

If the registration is successful and the provided information is valid, a new user account is created in the system.

If the registration fails due to duplicate username or email address, the user is prompted to choose different credentials.

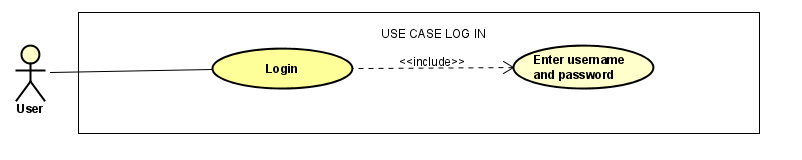
**Alternative Flows:**

If the user fails to provide any required information, the system prompts the user to fill in all necessary fields before proceeding with the registration.

If there are errors in the information provided (e.g., invalid email format), the system prompts the user to correct the errors before proceeding with the registration process.

**Exceptions:**

If there are technical issues or errors during the registration process, such as database connectivity problems, the system displays an error message and prompts the user to try again later or contact support for assistance.



**Use Case Name:** Login

**Actors:** User

**Description:**

This use case describes the process by which a user logs into their account within the system.

**Preconditions:**

The user accesses the login page of the system.

**Flow of Events:**

1. **User Login:**

The user navigates to the login page.

The system prompts the user to enter their username and password.

1. **Enter Required Information:**

The user inputs their username and password into the designated fields.

1. **Verify Information and Check Existing User:**

The system validates the entered username and password.

If the username and password combination is correct and matches an existing user account in the system, the system grants access to the user's account.

If the username or password is incorrect, the system displays an error message indicating that the login attempt failed.

**Postconditions:**

If the login is successful, the user gains access to their account and can proceed to use the system's features.

If the login fails due to incorrect username or password, the user is prompted to re-enter their credentials or reset their password.

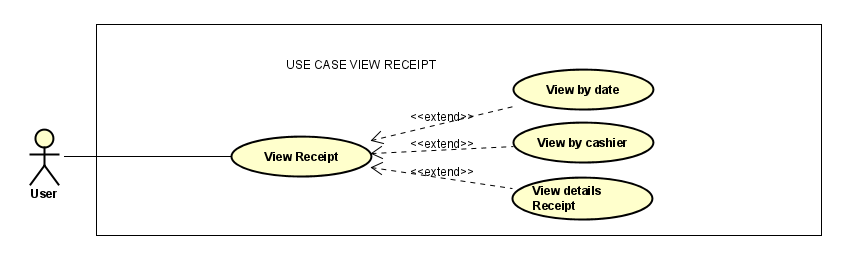
**Alternative Flows:**

If the user forgets their password, they can choose the "Forgot Password" option, which typically prompts the user to reset their password via email verification or security questions.

If the user has not yet registered for an account, they can navigate to the registration page to create a new account.

**Exceptions:**

If there are technical issues or errors during the login process, such as database connectivity problems, the system displays an error message and prompts the user to try again later or contact support for assistance.



**Use Case Name:** View Receipt

**Actors:** User

**Description:**

This use case describes the process by which a user views receipts within the system.

**Preconditions:**

The user is logged into their account and has access to the receipt viewing functionality.

**Flow of Events:**

1. **User View Receipt:**

The user navigates to the receipt viewing section of the system.

The system presents options for the user to view receipts, typically categorized by date, cashier, or receipt details.

1. **View by Date:**

The user selects the option to view receipts by date.

The system displays a calendar or date picker interface for the user to select a specific date or date range.

After selecting the date or date range, the system retrieves and displays receipts generated on the chosen date(s).

1. **View by Cashier:**

The user selects the option to view receipts by cashier.

The system provides a dropdown menu or list of cashier names for the user to choose from.

After selecting a cashier, the system retrieves and displays receipts associated with transactions processed by the selected cashier.

1. **View Details of Receipt:**

The user selects a specific receipt to view its details.

The system retrieves and displays the details of the selected receipt, including items purchased, prices, quantities, and any other relevant information.

Depending on the system's design, the user may have the option to print or download the receipt for their records.

**Postconditions:**

The user successfully views receipts according to the selected criteria (date, cashier, or receipt details).

**Alternative Flows:**

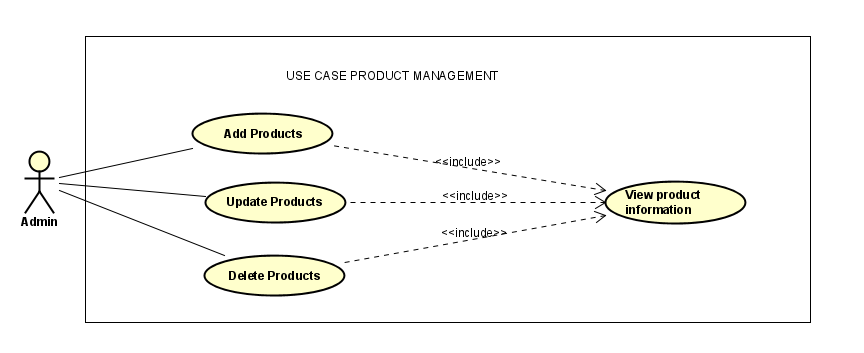
If the user wants to search for a specific receipt using keywords or transaction numbers, the system may provide a search bar or advanced search functionality.

If the user wants to filter receipts based on specific criteria such as payment method or product category, the system may offer additional filtering options.

**Exceptions:**

If there are no receipts matching the selected criteria, the system displays a message indicating that no receipts were found.

If there are technical issues or errors during the receipt retrieval process, such as database connectivity problems, the system displays an error message and prompts the user to try again later or contact support for assistance.



**Use Case Name:** Product Management

**Actors:** Admin

**Description:**

This use case describes the process by which an admin manages products within the system, including adding, updating, and deleting products, as well as viewing product information.

**Preconditions:**

The admin is logged into their account and has access to the product management functionality.

**Flow of Events:**

1. **Admin Adds Products:**

The admin selects the option to add new products.

The system presents a form or interface for the admin to input the details of the new product, such as name, description, price, category, and any other relevant information.

After the admin fills in the required product information, they submit the form to add the new product to the system's database.

1. **Admin Updates Products:**

The admin selects the option to update existing products.

The system displays a list of products currently available in the system.

The admin selects the product they want to update from the list.

The system presents a form or interface pre-filled with the existing product information for the admin to modify.

After making the necessary changes, the admin submits the form to update the product information in the system's database.

1. **Admin Deletes Products:**

The admin selects the option to delete existing products.

The system displays a list of products currently available in the system.

The admin selects the product(s) they want to delete from the list.

The system prompts the admin to confirm the deletion action.

After confirmation, the system removes the selected product(s) from the system's database.

1. **Admin Views Product Information:**

The admin selects the option to view product information.

The system displays a list of products currently available in the system, including details such as name, description, price, category, and any other relevant information.

The admin can filter or search for specific products based on criteria such as category or keyword.

**Postconditions:**

The admin successfully manages products within the system, including adding, updating, and deleting products, as well as viewing product information.

**Alternative Flows:**If the admin wants to add products in bulk, the system may provide an option to upload a CSV file containing product details.

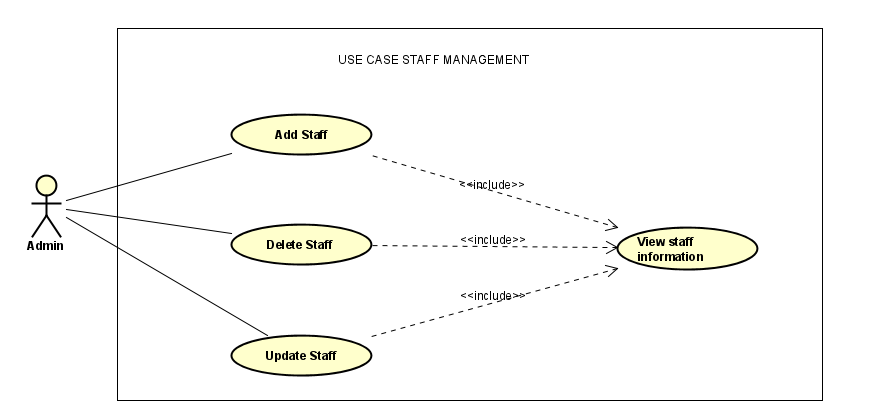
If the admin wants to update or delete multiple products at once, the system may provide batch editing or deletion functionality.

If the admin wants to view detailed analytics or sales data for specific products, the system may offer additional reporting features.

**Exceptions:**

If there are errors or missing information in the product details provided by the admin during the add or update process, the system prompts the admin to correct the errors before proceeding.

If there are technical issues or errors during the product management process, such as database connectivity problems, the system displays an error message and prompts the admin to try again later or contact support for assistance.

****

**Use Case Name:** Staff Management

**Actors:** Admin

**Description:**

This use case describes the process by which an admin manages staff members within the system, including adding, updating, and deleting staff, as well as viewing staff information.

**Preconditions:**

The admin is logged into their account and has access to the staff management functionality.

**Flow of Events:**

1. **Admin Adds Staff:**

The admin selects the option to add new staff members.

The system presents a form or interface for the admin to input the details of the new staff member, such as name, email address, role, contact information, and any other relevant details.

After filling in the required staff information, the admin submits the form to add the new staff member to the system's database.

1. **Admin Updates Staff:**

The admin selects the option to update existing staff members.

The system displays a list of staff members currently registered in the system.

The admin selects the staff member they want to update from the list.

The system presents a form or interface pre-filled with the existing staff member's information for the admin to modify.

After making the necessary changes, the admin submits the form to update the staff member's information in the system's database.

1. **Admin Deletes Staff:**

The admin selects the option to delete existing staff members.

The system displays a list of staff members currently registered in the system.

The admin selects the staff member(s) they want to delete from the list.

The system prompts the admin to confirm the deletion action.

After confirmation, the system removes the selected staff member(s) from the system's database.

1. **Admin Views Staff Information:**

The admin selects the option to view staff information.

The system displays a list of staff members currently registered in the system, including details such as name, email address, role, contact information, and any other relevant details.

The admin can filter or search for specific staff members based on criteria such as role or keyword.

**Postconditions:**

The admin successfully manages staff members within the system, including adding, updating, and deleting staff, as well as viewing staff information.

**Alternative Flows:**

If the admin wants to add staff members in bulk, the system may provide an option to upload a CSV file containing staff details.

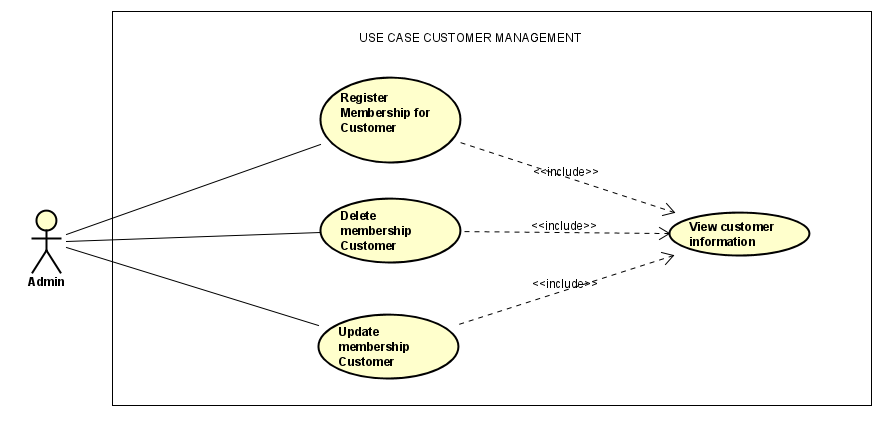
If the admin wants to update or delete multiple staff members at once, the system may provide batch editing or deletion functionality.

If the admin wants to view detailed performance metrics or attendance records for specific staff members, the system may offer additional reporting features.

**Exceptions:**

If there are errors or missing information in the staff details provided by the admin during the add or update process, the system prompts the admin to correct the errors before proceeding.

If there are technical issues or errors during the staff management process, such as database connectivity problems, the system displays an error message and prompts the admin to try again later or contact support for assistance.



**Use Case Name:** Customer Management

**Actors:** Admin

**Description:**

This use case describes the process by which an admin manages customer memberships within the system, including registering memberships, deleting memberships, updating memberships, and viewing customer information.

**Preconditions:**

The admin is logged into their account and has access to the customer management functionality.

**Flow of Events:**

1. **Admin Registers Membership for Customer:**

The admin selects the option to register a new membership for a customer.

The system prompts the admin to enter the customer's details, such as name, email address, contact information, and any other relevant information.

The admin assigns a membership level or type to the customer, which may include benefits or privileges associated with the membership.

After filling in the required customer information and selecting the membership type, the admin submits the form to register the customer's membership in the system's database.

1. **Admin Deletes Membership for Customer:**

The admin selects the option to delete a customer's membership.

The system displays a list of customers who are currently registered as members in the system.

The admin selects the customer whose membership they want to delete from the list.

The system prompts the admin to confirm the deletion action.

After confirmation, the system removes the customer's membership from the system's database.

1. **Admin Updates Membership for Customer:**

The admin selects the option to update a customer's membership.

The system displays a list of customers who are currently registered as members in the system.

The admin selects the customer whose membership they want to update from the list.

The system presents a form or interface pre-filled with the customer's existing membership information for the admin to modify.

The admin can update details such as membership level, expiration date, or associated benefits.

After making the necessary changes, the admin submits the form to update the customer's membership information in the system's database.

1. **Admin Views Customer Information:**

The admin selects the option to view customer information.

The system displays a list of customers registered in the system, including details such as name, email address, contact information, membership status, and any other relevant details.

The admin can filter or search for specific customers based on criteria such as membership status or keyword.

**Postconditions:**

The admin successfully manages customer memberships within the system, including registering, deleting, and updating memberships, as well as viewing customer information.

**Alternative Flows:**

If the admin wants to register memberships for multiple customers at once, the system may provide an option to upload a CSV file containing customer details.

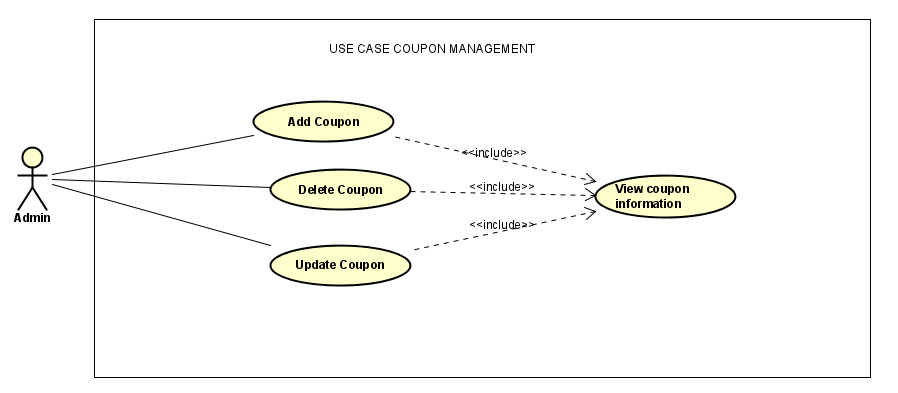
If the admin wants to update or delete multiple customer memberships at once, the system may provide batch editing or deletion functionality.

If the admin wants to view detailed purchase history or preferences for specific customers, the system may offer additional reporting features.

**Exceptions:**

If there are errors or missing information in the customer details provided by the admin during the registration or update process, the system prompts the admin to correct the errors before proceeding.

If there are technical issues or errors during the customer management process, such as database connectivity problems, the system displays an error message and prompts the admin to try again later or contact support for assistance.

****

**Use Case Name:** Coupon Management

**Actors:** Admin

**Description:**

This use case outlines the process by which an admin manages coupons within the system, including adding, deleting, updating coupons, and viewing coupon information.

**Preconditions:**

The admin is logged into their account and has access to the coupon management functionality.

**Flow of Events:**

1. **Admin Adds Coupon:**

The admin selects the option to add a new coupon.

The system prompts the admin to input details for the new coupon, such as coupon code, discount amount or percentage, expiration date, and any other relevant information.

After filling in the required coupon details, the admin submits the form to add the new coupon to the system's database.

1. **Admin Deletes Coupon:**

The admin selects the option to delete an existing coupon.

The system displays a list of coupons currently available in the system.

The admin selects the coupon they want to delete from the list.

The system prompts the admin to confirm the deletion action.

After confirmation, the system removes the selected coupon from the system's database.

1. **Admin Updates Coupon:**

The admin selects the option to update an existing coupon.

The system displays a list of coupons currently available in the system.

The admin selects the coupon they want to update from the list.

The system presents a form or interface pre-filled with the existing coupon details for the admin to modify.

The admin can update details such as discount amount, expiration date, or any other parameters associated with the coupon.

After making the necessary changes, the admin submits the form to update the coupon information in the system's database.

1. **Admin Views Coupon Information:**

The admin selects the option to view coupon information.

The system displays a list of coupons currently available in the system, including details such as coupon code, discount amount or percentage, expiration date, and any other relevant information.

The admin can filter or search for specific coupons based on criteria such as coupon code or discount amount.

**Postconditions:**

The admin successfully manages coupons within the system, including adding, deleting, updating coupons, and viewing coupon information.

**Alternative Flows:**

If the admin wants to add coupons in bulk, the system may provide an option to upload a CSV file containing coupon details.

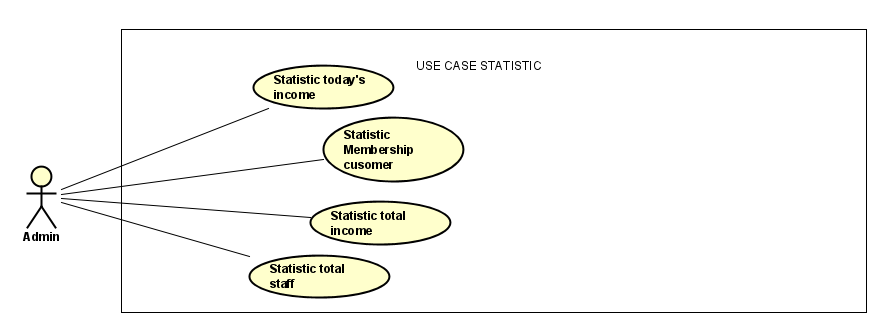
If the admin wants to update or delete multiple coupons at once, the system may provide batch editing or deletion functionality.

If the admin wants to track coupon usage or redemption statistics, the system may offer additional reporting features.

**Exceptions:**

If there are errors or missing information in the coupon details provided by the admin during the add or update process, the system prompts the admin to correct the errors before proceeding.

If there are technical issues or errors during the coupon management process, such as database connectivity problems, the system displays an error message and prompts the admin to try again later or contact support for assistance.



**Use Case Name:** Statistic

**Actors:** Admin

**Description:**

This use case outlines the process by which an admin views various statistics related to income, customer membership, total income, and total staff within the system.

**Preconditions:**

The admin is logged into their account and has access to the statistic viewing functionality.

**Flow of Events:**

1. **Admin Views Today's Income:**

The admin selects the option to view today's income statistics.

The system calculates and displays the total income generated by the business on the current day.

The admin may also have the option to view income breakdowns by different sources, such as sales, services, or products.

1. **Admin Views Membership Customer Statistics:**

The admin selects the option to view statistics related to membership customers.

The system calculates and displays various metrics related to membership customers, such as the total number of members, membership levels distribution, and membership enrollment trends over time.

The admin may also have the option to view additional details about individual members or membership transactions.

1. **Admin Views Total Income Statistics:**

The admin selects the option to view total income statistics.

The system calculates and displays the total income generated by the business over a specified period, such as daily, weekly, monthly, or yearly.

The admin may also have the option to view income trends over time and compare income between different time periods.

1. **Admin Views Total Staff Statistics:**

The admin selects the option to view statistics related to total staff.

The system calculates and displays various metrics related to staff members, such as the total number of staff, staff roles distribution, and staff turnover rates.

The admin may also have the option to view additional details about individual staff members or staff performance metrics.

**Postconditions:**

The admin successfully views various statistics related to income, customer membership, total income, and total staff within the system.

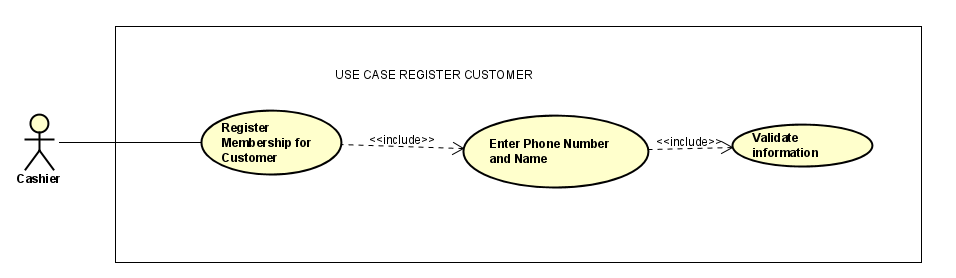
**Alternative Flows:**

If the admin wants to view more detailed or customized statistics, the system may provide options for filtering, sorting, or drilling down into specific data sets.

If the admin wants to export statistics data for further analysis, the system may offer options for exporting data in formats such as CSV or Excel.

**Exceptions:**

If there are technical issues or errors during the statistic calculation or retrieval process, such as database connectivity problems or data processing errors, the system displays an error message and prompts the admin to try again later or contact support for assistance.

****

**Use Case Name:** Register Customer

**Actors:** Cashier

**Description:**

This use case describes the process by which a cashier registers a customer for a membership within the system.

**Preconditions:**

The cashier is logged into their account and has access to the customer registration functionality.

**Flow of Events:**

1. **Cashier Registers Membership for Customer:**

The cashier selects the option to register a new customer for a membership.

The system prompts the cashier to enter the customer's phone number and name, which are typically used as unique identifiers for the customer's account.

1. **Enter Phone Number and Name:**

The cashier enters the customer's phone number and name into the designated fields.

The system may prompt the cashier to verify the phone number and name to ensure accuracy.

1. **Validate Information:**

After the cashier enters the customer's phone number and name, the system validates the information provided.

The system checks if the phone number is unique and not already associated with an existing customer account.

If the phone number is unique and valid, the system proceeds to register the customer for a membership.

If the phone number is already associated with an existing customer account, the system may prompt the cashier to verify if the customer wants to update their membership or register as a new member with a different phone number.

**Postconditions:**

The cashier successfully registers the customer for a membership within the system.

**Alternative Flows:**

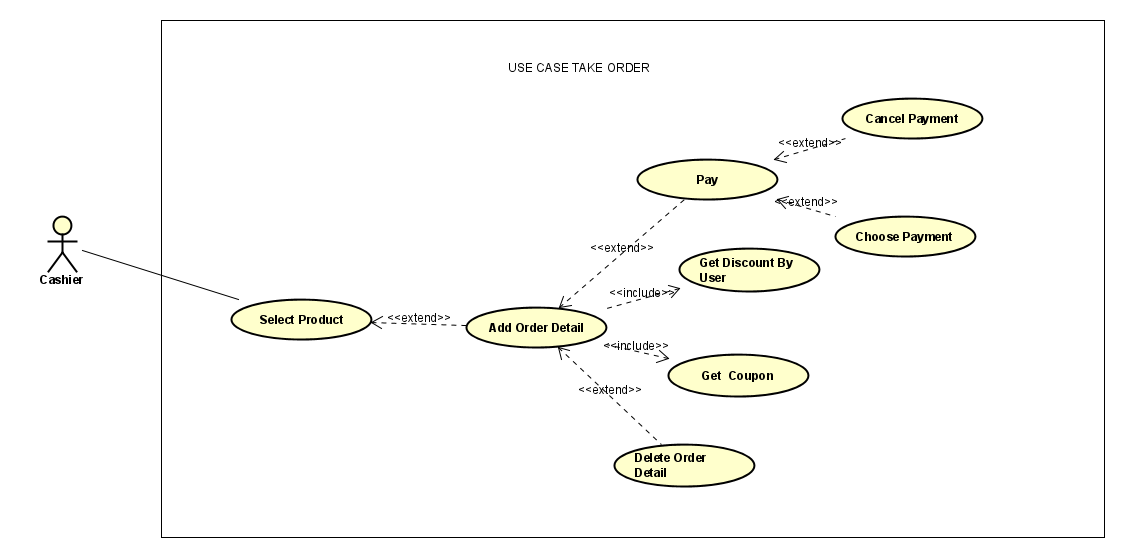
If the customer already has an existing membership, the system may prompt the cashier to verify if the customer wants to renew or update their membership instead of registering as a new member.

If the customer provides additional information such as email address or address, the system may prompt the cashier to enter this information for a more comprehensive customer profile.

**Exceptions:**

If there are errors or missing information in the customer details provided by the cashier during the registration process, the system prompts the cashier to correct the errors before proceeding.

If there are technical issues or errors during the registration process, such as database connectivity problems, the system displays an error message and prompts the cashier to try again later or contact support for assistance.



**Use Case Name:** Take Order

**Actors:** Cashier

**Description:**

This use case describes the process by which a cashier takes an order from a customer within the system.

**Preconditions:**

The cashier is logged into their account and has access to the order management functionality.

**Flow of Events:**

1. **Cashier Selects Product:**

The cashier selects the product(s) that the customer wishes to order from the menu.

The system may display a list of available products with options for customization or additional details.

1. **Add Order Detail:**

After selecting the product(s), the cashier adds the order detail(s) to the customer's order.

The system may prompt the cashier to specify quantity, size, or any other relevant options for each product.

1. **Get Discount By User:**

The system automatically checks if the customer is eligible for any discounts based on their membership status or purchase history.

If the customer is eligible for a discount, the system applies the discount to the order total.

1. **Get Coupon:**

If the customer presents a coupon, the cashier selects the option to apply the coupon to the order.

The system verifies the validity of the coupon and applies the discount specified by the coupon to the order total.

1. **Pay:**

The cashier calculates the total amount due for the customer's order.

The customer pays for the order using their preferred payment method, such as cash, credit card, or mobile payment.

The system processes the payment transaction and updates the order status accordingly.

1. **Choose Payment (Optional):**

If the customer wants to change their payment method or split the payment between multiple methods, the cashier selects the option to choose payment.

The system allows the cashier to select or enter different payment methods for the order.

1. **Cancel Payment (Optional):**

If the customer decides to cancel the order or payment transaction for any reason, the cashier selects the option to cancel payment.

The system cancels the payment transaction and updates the order status accordingly.

1. **Delete Order Detail (Optional):**

If the customer decides to remove any items from their order, the cashier selects the option to delete the corresponding order detail.

The system removes the selected item(s) from the order and adjusts the order total accordingly.

**Postconditions:**

The cashier successfully takes the customer's order, applies any discounts or coupons, processes the payment transaction, and updates the order status accordingly.

**Alternative Flows:**

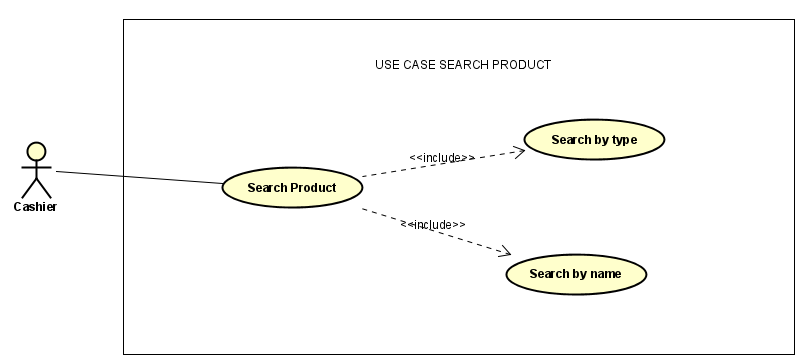
If the customer wants to add or modify their order after it has been submitted, the cashier may edit the order details before finalizing the payment transaction.

If the customer wants to apply a discount or coupon that they forgot to mention earlier, the cashier can manually apply the discount or coupon before finalizing the payment transaction.

**Exceptions:**

If there are errors or discrepancies in the order details, payment transaction, or application of discounts/coupons, the system alerts the cashier and prompts them to resolve the issue before finalizing the order.

If there are technical issues or errors during the order-taking process, such as payment processing failures, the system displays an error message and prompts the cashier to try again later or contact support for assistance.

****

**Use Case Name:** Search Product

**Actors:** Cashier

**Description:**

This use case describes the process by which a cashier searches for a product within the system.

**Preconditions:**

The cashier is logged into their account and has access to the product search functionality.

**Flow of Events:**

1. **Cashier Searches Product:**

The cashier selects the option to search for a product.

The system presents options for the cashier to search by type or by name.

1. **Search by Type:**

If the cashier chooses to search by type, the system displays a list of product categories or types.

The cashier selects the desired product type from the list.

The system filters and displays products that belong to the selected product type.

1. **Search by Name:**

If the cashier chooses to search by name, the system prompts the cashier to enter keywords or the exact name of the product.

The cashier enters the search query into the search bar.

The system filters and displays products whose names match the search query.

**Postconditions:**

The cashier successfully searches for a product within the system based on the chosen search method (by type or by name).

**Alternative Flows:**

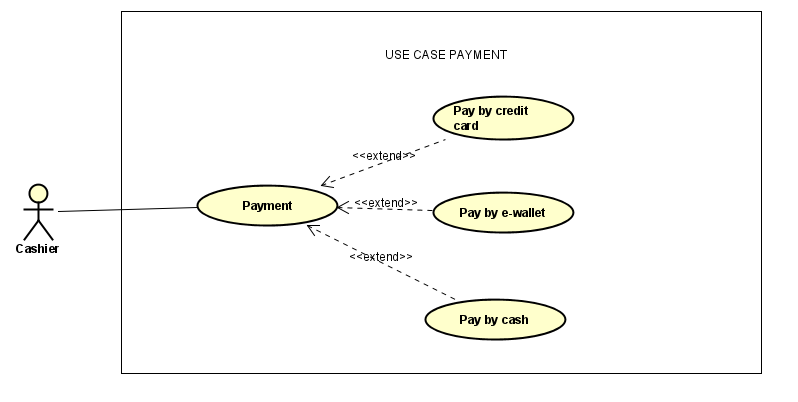
If the cashier wants to search for products using advanced filters or criteria, the system may provide options for additional parameters such as price range, brand, or availability.

If the search query returns multiple results, the system may display pagination or scrolling options for the cashier to navigate through the search results.

**Exceptions:**

If there are no products matching the search criteria, the system displays a message indicating that no products were found and prompts the cashier to refine their search query.

If there are technical issues or errors during the product search process, such as database connectivity problems or search engine failures, the system displays an error message and prompts the cashier to try again later or contact support for assistance.

****

**Use Case Name:** Payment

**Actors:** Cashier

**Description:**

This use case describes the process by which a cashier processes a payment for an order within the system.

**Preconditions:**

The cashier has completed the order details and the customer is ready to pay.

**Flow of Events:**

1. **Cashier Initiates Payment:**

After completing the order details, the cashier initiates the payment process.

The system prompts the cashier to select the payment method.

1. **Pay by Credit Card:**

If the customer chooses to pay by credit card, the cashier selects the option to process payment by credit card.

The system prompts the cashier to enter the credit card information, including card number, expiration date, CVV, and billing address if required.

The cashier enters the credit card information into the system.

The system processes the credit card payment transaction and updates the order status accordingly.

1. **Pay by E-wallet:**

If the customer chooses to pay by e-wallet, the cashier selects the option to process payment by e-wallet.

The system prompts the cashier to select the e-wallet service provider and enter the customer's e-wallet account information.

The cashier enters the e-wallet account information into the system.

The system processes the e-wallet payment transaction and updates the order status accordingly.

1. **Pay by Cash:**

If the customer chooses to pay by cash, the cashier selects the option to process payment by cash.

The customer hands the cash payment to the cashier.

The cashier enters the amount received into the system and calculates the change if necessary.

The cashier provides the change to the customer and finalizes the cash payment transaction.

The system updates the order status accordingly.

**Postconditions:**

The cashier successfully processes the payment for the customer's order using the selected payment method.

**Alternative Flows:**

If the customer wants to split the payment between multiple methods (e.g., part cash, part credit card), the cashier can select the option to split payment and process each payment method separately.

If there are issues with the payment transaction (e.g., declined credit card, insufficient funds), the system alerts the cashier and prompts them to take appropriate action, such as selecting a different payment method or contacting the customer for alternative payment arrangements.

**Exceptions:**

If there are technical issues or errors during the payment processing, such as connectivity problems with payment service providers, the system displays an error message and prompts the cashier to try again later or contact support for assistance.

If there are discrepancies in the amount received for cash payments or issues with credit card authorization, the system alerts the cashier and prompts them to resolve the issue before finalizing the payment transaction.