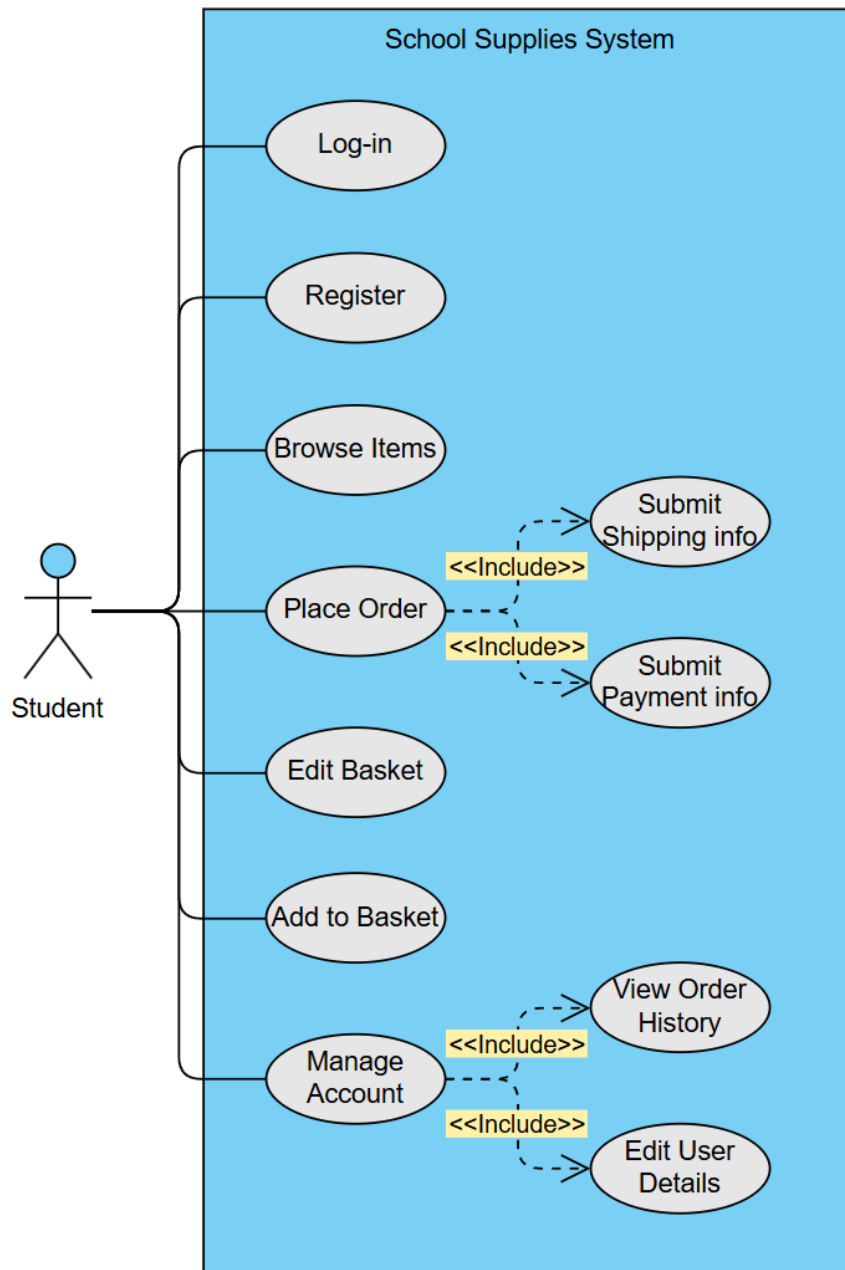


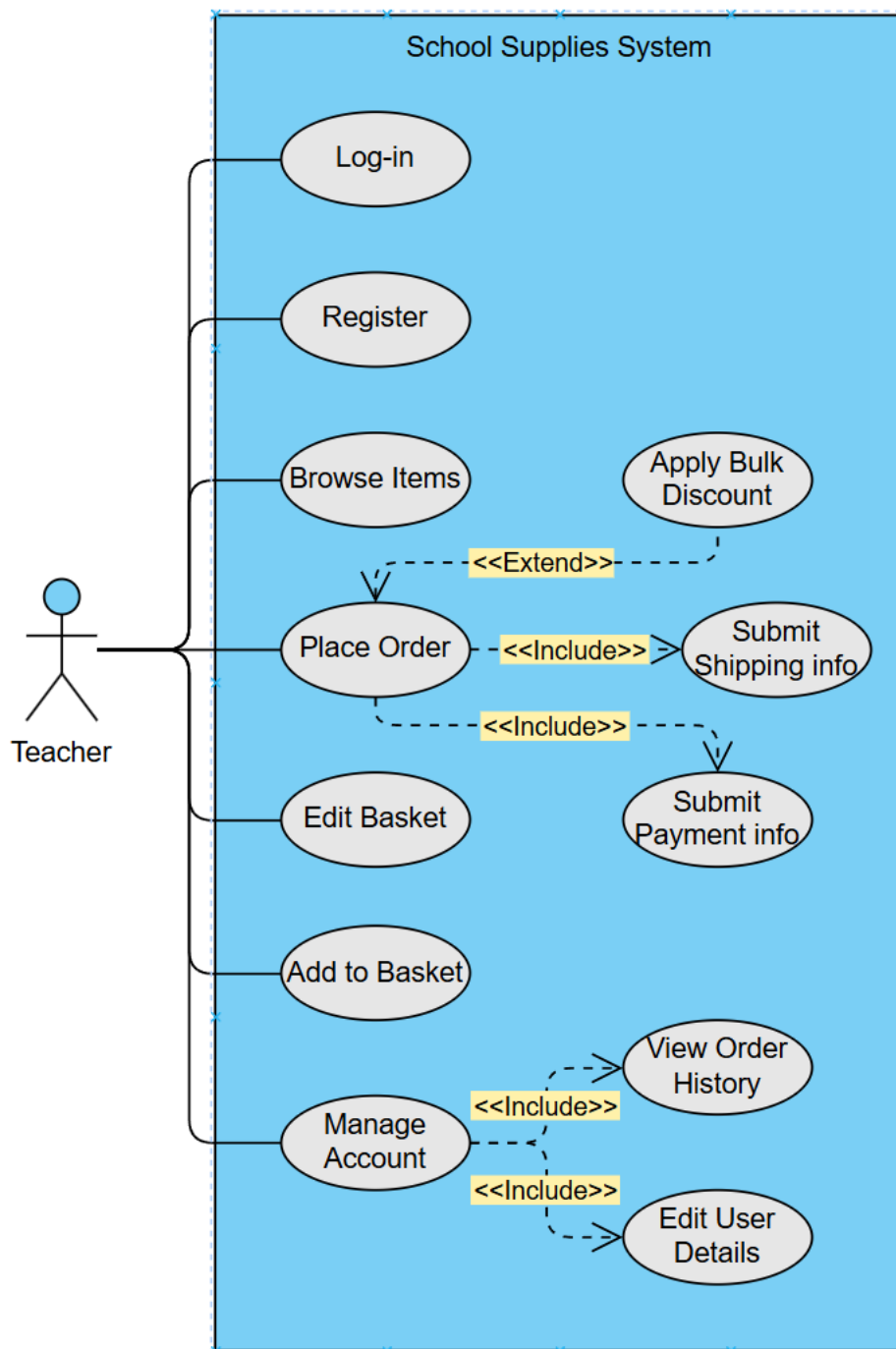
Web framework development

Deliverable 2: Use Cases

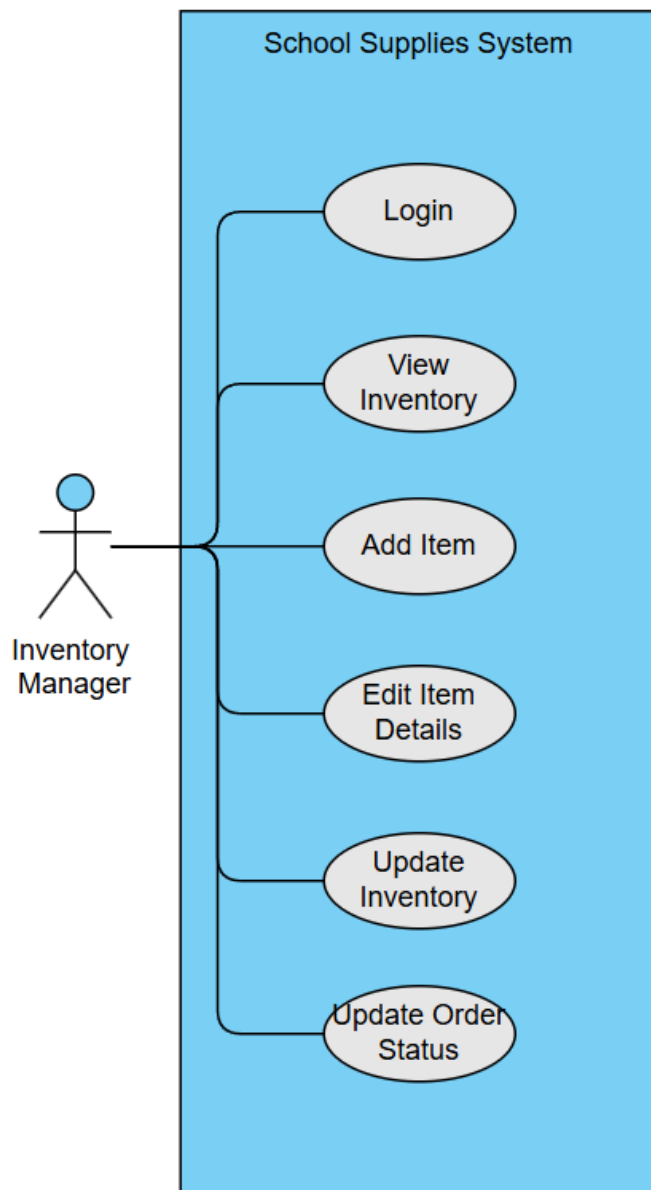
Student Use Cases



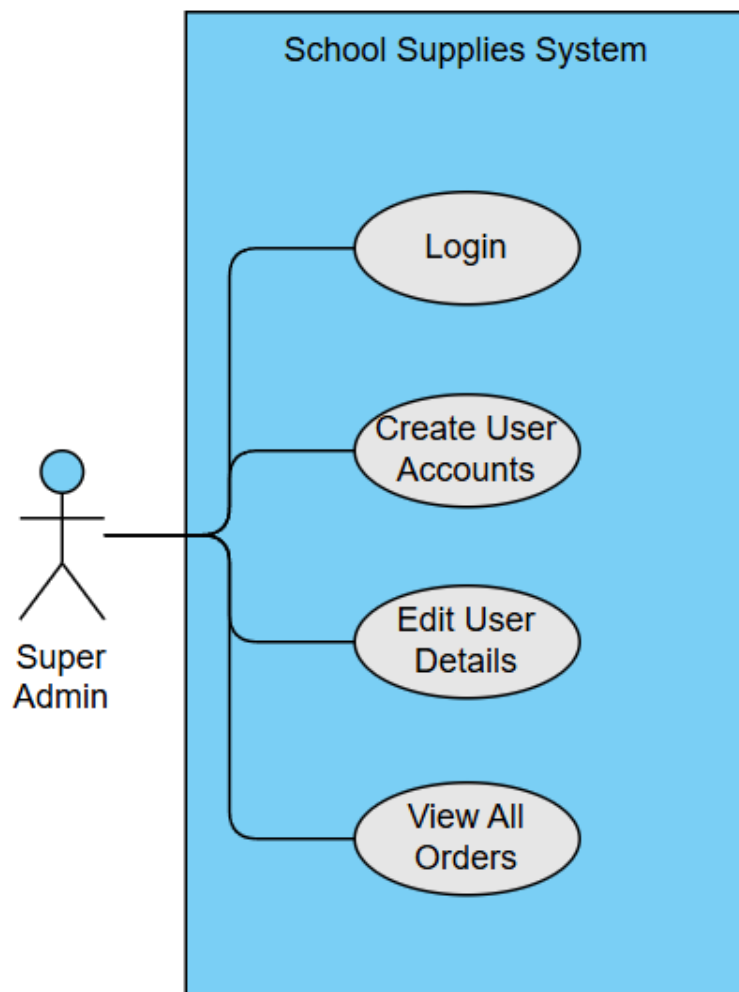
Teacher Use Cases



Inventory Manager Use Cases



Super Admin Use Cases



Explanation of use case diagrams:

Register:

- **Goal:** Allow users to create an account
- **Actors:** Student, teacher
- **Preconditions:** User is not already registered
- **Postconditions:** User account is created and successfully stored in the database
- **Main flow:**
 - User selects account type (student/teacher)
 - User enters email, name, password
 - User clicks submit
 - Information stored in database
 - On successful completion confirmation message shown

Log in:

- **Goal:** Allow users to log in into the system
- **Actors:** Student, teacher, inventory manager, system admin (super admin)
- **Preconditions:** User has registered
- **Postconditions:** User is authenticated and can access additional features (e.g. place order)
- **Main flow:**
 - User enters email and password
 - System verifies credentials according to the database
 - User is logged in and redirected

Browse items:

- **Goal:** Allow users to view all available inventory
- **Actors:** Student, teacher
- **Preconditions:** *None*
- **Postconditions:** Items are displayed from the database
- **Main flow:**
 - User navigates to items page
 - System fetches and displays items from the database

Add to basket:

- **Goal:** Allow users to add items to their basket
- **Actors:** Student, teacher
- **Preconditions:** User is logged in
- **Postconditions:** Items are stored in the user's basket
- **Main flow:**
 - User defines quantity to add to the basket
 - User clicks "*Add to basket*" button
 - Item is added to the user's basket stored in the database

Edit basket

- **Goal:** Allow users to edit the content of their basket
- **Actors:** Student, teacher
- **Preconditions:** User is logged in, has items in their basket
- **Postconditions:** User's basket content is changed
- **Main flow:**
 - User navigates to basket content
 - User modifies their basket content
 - Basket of the user is updated in the database

Manage account

- **Goal:** Allow users to view their orders and edit their account
- **Actors:** Student, teacher
- **Preconditions:** User has registered and logged in
- **Postconditions:** User retrieves information on their orders, can edit their information
- **Main flow:**
 - User navigates to manage account page
 - User selects view order history or edit account page
 - List of previous orders displayed on order history
 - *or* User can change information about their account (e.g. password)
 - newly updated user information is stored in the database

View inventory

- **Goal:** Allow inventory manager to view their inventory
- **Actors:** Inventory manager
- **Preconditions:** Inventory manager has logged in
- **Postconditions:** Inventory manager views all items stored in the database by their account including information on stock, etc.
- **Main flow:**
 - User navigates to view inventory page
 - System fetches all data posted by the inventory manager into the database and displays it

Add item

- **Goal:** Allow inventory manager to add new items into the system
- **Actors:** Inventory manager
- **Preconditions:** Inventory manager has logged in
- **Postconditions:** New item is added into the items table and can be viewed by users
- **Main flow:**
 - User navigates to add new items page
 - User fills out a form regarding their product
 - New item is then added to the database with reference to who added the item
 - Confirmation is returned to the inventory manager

Update Inventory

- **Goal:** Allow inventory manager to update their inventory
- **Actors:** Inventory manager
- **Preconditions:** Inventory manager has logged in, has items added to the database
- **Postconditions:** Content of the item table is updated
- **Main flow:**
 - User navigates to update inventory page
 - Items created by the user are shown by the system
 - User can edit information or quantity regarding their products
 - System updates item table

Update order status

- **Goal:** Allow inventory manager to update the status of an order
- **Actors:** Inventory manager
- **Preconditions:** Inventory manager has logged in, has items added to the database
- **Postconditions:** Status of an order changes
- **Main flow:**
 - User views list of orders placed
 - User updates the status of the order
 - System notifies the buyer regarding the order
 - Status is updated in the database

Create user account

- **Goal:** Allow system admin (super admin) to create users
- **Actors:** System admin
- **Preconditions:** Admin has logged in
- **Postconditions:** Admin creates an account with role specified that is added to the database
- **Main flow:**
 - Admin submits email, password, role information
 - Admin presses "*Submit*"
 - New account is created and stored in the database

Edit user details

- **Goal:** Allow system admin (super admin) to view and edit user information
- **Actors:** System admin
- **Preconditions:** Admin has logged in
- **Postconditions:** Admin updates data about users and is stored in the database
- **Main flow:**
 - Admin views all user accounts from the database
 - Admin selects account to edit
 - Admin edits desired information about the user or deletes the account
 - System updates the database

View all orders

- **Goal:** Allow system admin (super admin) to view all orders placed
- **Actors:** System admin
- **Preconditions:** Admin has logged in
- **Postconditions:** All orders placed are viewed by the admin
- **Main flow:**
 - Admin navigates to Orders page
 - System fetches all orders
 - Admin can view all orders