**LLD (Low-Level Design )Document for Flipkart:**

Low-Level Design (LLD) is the process of converting a High-Level Design (HLD) or system architecture into a detailed design that can be implemented by software developers. In the software development life cycle, LLD comes after the completion of the High-Level Design phase and before the actual coding or implementation phase.

**Component: Product Catalog Module**

1. **Class: ProductManager**

* **Responsibilities:**
  + Manage product information.
  + Handle CRUD operations on products.
* **Attributes:**
  + **productId**: Unique identifier for each product.
  + **productName**: Name of the product.
  + **productDescription**: Description of the product.
  + **productPrice**: Price of the product.
  + **productCategory**: Category to which the product belongs.
* **Methods:**
  + **addProduct(productDetails)**: Adds a new product to the catalog.
  + **updateProduct(productId, updatedDetails)**: Updates the details of an existing product.
  + **deleteProduct(productId)**: Deletes a product from the catalog.
  + **getProductDetails(productId)**: Retrieves details of a specific product.
  + **getProductsByCategory(category)**: Retrieves a list of products in a specific category.

2. **Class: InventoryManager**

* **Responsibilities:**
  + Manage product availability in the inventory.
  + Track stock levels.
* **Attributes:**
  + **productId**: Unique identifier for each product.
  + **availableQuantity**: Quantity of the product available in stock.
* **Methods:**
  + **checkStockAvailability(productId, quantity)**: Checks if the requested quantity is available in stock.
  + **updateStock(productId, quantity)**: Updates the stock after a purchase.

**Component: User Authentication Module**

1. **Class: UserManager**

* **Responsibilities:**
  + Manage user accounts.
  + Handle user authentication and authorization.
* **Attributes:**
  + **userId**: Unique identifier for each user.
  + **username**: User's username.
  + **passwordHash**: Hashed password for security.
* **Methods:**
  + **createUser(userData)**: Creates a new user account.
  + **authenticateUser(username, password)**: Authenticates a user during login.
  + **authorizeUser(userId, role)**: Grants specific roles and permissions to a user.

2. **Class: SessionManager**

* **Responsibilities:**
  + Manage user sessions for authentication.
* **Attributes:**
  + **sessionId**: Unique identifier for each user session.
  + **userId**: Identifier of the associated user.
  + **expiryTime**: Time when the session will expire.
* **Methods:**
  + **createSession(userId)**: Creates a new session for a logged-in user.
  + **validateSession(sessionId)**: Validates the user's session during requests.
  + **expireSession(sessionId)**: Expires a user session.

**Component: Shopping Cart and Checkout Module**

1. **Class: ShoppingCartManager**

* **Responsibilities:**
  + Manage user shopping carts.
  + Handle operations related to adding, updating, and removing items from the cart.
* **Attributes:**
  + **cartId**: Unique identifier for each shopping cart.
  + **userId**: Identifier of the associated user.
  + **cartItems**: List of items in the shopping cart.
* **Methods:**
  + **addItemToCart(cartId, productId, quantity)**: Adds an item to the user's shopping cart.
  + **updateCartItem(cartId, productId, quantity)**: Updates the quantity of a product in the cart.
  + **removeItemFromCart(cartId, productId)**: Removes an item from the user's shopping cart.
  + **getCartDetails(cartId)**: Retrieves the details of the user's shopping cart.

2. **Class: CheckoutManager**

* **Responsibilities:**
  + Handle the checkout process.
  + Calculate the total cost, apply discounts, and finalize the order.
* **Attributes:**
  + **orderId**: Unique identifier for each order.
  + **userId**: Identifier of the associated user.
  + **orderItems**: List of items in the order.
  + **orderTotal**: Total cost of the order.
* **Methods:**
  + **processCheckout(cartId)**: Initiates the checkout process for the user's shopping cart.
  + **applyDiscounts(orderId)**: Applies discounts and promotions to the order.
  + **finalizeOrder(orderId)**: Completes the order processing.

**Flow chart:**

**A paper with writing on it

Description automatically generated**