

| Student Authentication and Profile Management |  |
|---|--|
| <b>Function</b>                               | User Authentication, Registration, and Profile Management  |
| <b>Description</b>                            | Handles secure student registration and login using an external identity provider compliant with OpenID Connect standards. Manages user sessions, supports password reset through the identity provider, and enables students to update their personal information such as name, ID, email, and profile picture. |
| <b>Inputs</b>                                 | Student credentials (email, password, or federated login token); profile data fields (name, ID, photo).  |
| <b>Source</b>                                 | User input through registration/login form and identity provider response.   |
| <b>Outputs</b>                                | Authentication token and active session; success or error messages; updated user profile data.   |
| <b>Destination</b>                            | User dashboard and profile management interface.   |
| <b>Action</b>                                 | When a student logs in or registers, the system authenticates credentials using the provider, establishes a secure session, and loads the student dashboard. Profile edits are stored in the user database after validation.   |
| <b>Requirements</b>                           | Stable internet connection; valid institutional account; secure backend database.  |
| <b>Pre-condition</b>                          | Authentication service must be available; student account must exist or be newly created.  |
| <b>Post-condition</b>                         | Session established or updated; modified profile saved to the database.  |
| <b>Side effects</b>                           | Automatic logout occurs after session timeout for security.  |

| Menu Display and Browsing |  |
|---------------------------|--|
| <b>Function</b>           | Display and manage the cafeteria menu with detailed information and filters.   |
| <b>Description</b>        | Displays daily categorized menus such as Breakfast, Lunch, Snacks, and Beverages. Each menu item includes its name, description, price, image, availability status, preparation time, nutritional facts, and allergen warnings. Students can search or filter items by category. |
| <b>Inputs</b>             | Menu item data (name, description, price, category, image, availability, nutritional info, allergen info).   |
| <b>Source</b>             | Cafeteria menu database or content management system.  |
| <b>Outputs</b>            | Dynamic menu displayed on the user interface; search and filter results; real-time updates.  |
| <b>Destination</b>        | Frontend web interface - Menu Page.  |
| <b>Action</b>             | The system retrieves and displays menu items grouped by category. When the user searches or applies filters, the system updates visible items. Real-time availability is refreshed automatically when an item becomes unavailable.   |
| <b>Requirements</b>       | Active backend API; structured menu database; connection to the availability tracking module.  |
| <b>Pre-condition</b>      | System connected to the backend database; user on the menu page.   |
| <b>Post-condition</b>     | Filtered or updated menu displayed to the user.  |
| <b>Side effects</b>       | None.  |

| Meal Pre-Ordering     |  |
|-----------------------|--|
| <b>Function</b>       | Meal Pre-Ordering  |
| <b>Description</b>    | Enables students to select and pre-order meals from the available menu before arriving at the café. Students can choose desired items, select portion sizes (Small, Medium, Large), and specify optional special instructions. The feature streamlines the ordering process and reduces waiting times at pickup. |
| <b>Inputs</b>         | Selected menu items, portion size, and optional special instructions.  |
| <b>Source</b>         | User selections from the menu browsing interface.  |
| <b>Outputs</b>        | Order preview showing selected items, size, price, and preparation time.   |
| <b>Destination</b>    | Cart system and order database.  |
| <b>Action</b>         | When the student selects an item and size, the system verifies availability, applies any size-based price modifiers, and adds the item to a temporary pre-order list. The order details are stored for confirmation before checkout.   |
| <b>Requirements</b>   | Real-time menu data; pricing and size mapping; order validation mechanism.   |
| <b>Pre-condition</b>  | User must be logged in and menu items must be available for the current day.   |
| <b>Post-condition</b> | Pre-ordered items are temporarily saved in the user's session or cart.   |
| <b>Side effects</b>   | Unavailable items may be automatically removed if their availability changes before checkout.  |

| Shopping Cart Management |   |
|--------------------------|---|
| <b>Function</b>          | Shopping Cart Item Modification and Management  |
| <b>Description</b>       | Allows students to view, update, and manage their cart before checkout. They can change quantities, remove items, or clear the entire cart. The cart automatically recalculates the total price (including tax) after every update, ensuring accurate order totals. |
| <b>Inputs</b>            | Cart items, updated quantities, item deletions, and recalculated totals.  |
| <b>Source</b>            | User interactions on the cart interface.  |
| <b>Outputs</b>           | Updated cart summary reflecting changes in item count and total cost.   |
| <b>Destination</b>       | Checkout module and order summary page.   |
| <b>Action</b>            | Each modification (add, edit, delete) triggers an immediate update to the cart state and recalculates totals. Once finalized, the updated cart data is sent to the checkout module for further processing.  |
| <b>Requirements</b>      | Active user session; consistent pricing data; responsive cart interface.  |
| <b>Pre-condition</b>     | User logged in with existing cart data; menu items available.   |
| <b>Post-condition</b>    | Cart state updated and stored until checkout or manual clearance.   |
| <b>Side effects</b>      | If an item becomes unavailable, the system automatically removes it and updates the total.  |

| Order Checkout & Customization |   |
|--------------------------------|---|
| <b>Function</b>                | Order checkout with pickup scheduling, automatic tax calculation, and customization options.  |
| <b>Description</b>             | Facilitates the final checkout process by allowing students to choose a preferred pickup time slot with at least 15 minutes' advance notice. The system automatically calculates the total order cost, including applicable taxes, and provides a field for any special instructions related to the order (e.g., "extra sauce," "no onions"). |
| <b>Inputs</b>                  | Cart items, selected pickup time, quantity, size, and optional special instructions.  |
| <b>Source</b>                  | User input from the cart and checkout interface.  |
| <b>Outputs</b>                 | Finalized order summary showing item details, total cost with tax, and scheduled pickup time confirmation.  |
| <b>Destination</b>             | Order confirmation module and kitchen order queue.  |
| <b>Action</b>                  | Once the user proceeds to checkout, the system validates cart items, ensures the selected pickup slot is available, computes the total price (base price + tax), and attaches any special notes to the order. After confirmation, the order details are recorded and sent to the kitchen for preparation.                                     |
| <b>Requirements</b>            | Accurate tax configuration in the backend; order timing module to manage pickup slots; active database for storing order data.  |
| <b>Pre-condition</b>           | Student must be logged in with a valid cart and selected items. The chosen pickup time must be at least 15 minutes from the current time.   |
| <b>Post-condition</b>          | Order is saved in the database with complete details including total cost, selected pickup slot, and special instructions.  |
| <b>Side effects</b>            | If multiple users choose the same time slot, the system may adjust availability dynamically or suggest an alternate slot.   |