FlavourA

The application aims to simplify and enhance the food discovery and purchase process for users, offering a visually engaging interface. Users can explore diverse food options, add items to a basket, and make informed purchases, all while enjoying a seamless and intuitive experience.

Software Specification

Upon launching the application, users are welcomed by a captivating **loading template** with a briefanimation. After the loading completes, the interface seamlessly transitions to the **home template**, presenting an array of available food options. Users can effortlessly scroll through these choices, adding items to their **basket** directly from the **home template**.

Accessing the **cart template** is convenient, directly from the **home template**. Users can also navigate to the **menu** options from both the **home and chart templates**. When ready to make a purchase, users inspect their selected items in the **cart template** before proceeding to the **buy template**. Clear information about the total price is provided before finalizing the purchase.

Additionally, users have the flexibility to remove items from their **basket**, and the option to adjust their currency preference is available in the **menu**. This comprehensive process ensures a user-friendly and seamless experience from exploring food options to making a purchase.

loading-template:

This template serves the dual purpose of instilling a psychological reward effect, intended to evoke excitement in the user upon initiating the application, and signaling that the application is in the process of launching, nearing readiness for use.

It features a dynamic loading sequence, showcasing a green loading bar beneath three lively food vector graphics displayed at the center of the screen, that are animated in asynchronous jumping motion. The loading process is optimized to complete within 1-2 seconds, seamlessly transitioning to the **home template**. A fade-out animation gracefully concludes the loading phase, affecting all displayed elements on the screen.

home-template:

The home template is the starting point of the application, offering users a visually appealing interface to explore a diverse range of available food options.

Icon Buttons:
- **Menu**   : Opens menu **(described at the next section - menu)**
- **Cart**   : Redirects the user to the **cart template** to inspect added by the user food items before making a purchase. The cart icon also shakes with each new item added to the basket.

Labels:
- **Items count**   : The quantity of items added to the basket. NUMBER, is set on top of the **cart** icon.
- **Alert box**   : This is a small message box to inform the user of critical actions. TEXT, and is restricted to 50 characters.

Food Item Container:
- **Image**   : Item's picture, (size: contain as square). IMAGE
- **Title**   : LABEL, is restricted to 20 characters. TEXT
- **Description**   : LABEL, is restricted to 250 characters. TEXT
- **Price**   : LABEL, is restricted to 100 units. (For: Euro, GB Pounds and ZLOTY). NUMBER
- **Add**   : BUTTON, adds an item to the basket, and updates the **Items count** label at the **cart** icon. And also, displays briefly **Alert box** label with the message: "added item to the basket".

- By default, the food-item-container exhibits an image, title, and description. Users can toggle the container to expand, revealing additional details such as the price label and add button.

- When a different container is selected,
  the expanded container smoothly hides with a transition animation.

- If expanded containers cannot fit side by side,
  they elegantly break down onto a new line for optimal visibility.

- Users can also scroll through the displayed food-item-containers vertically,
  while also maintaining the access to the menu and a cart button.

- Expanded container? Why some containers are expanded? What are they? Promotions, if so what should we do if we press on the box? Should it be animated to emphasise user-interation?
- Image at the center? What is it, logo, some food, an advert?

menu:

- The menu features two options: "Home" for returning to the **home template**, and "Currency",
  a dropdown with currency options including: Euro, British Pound, and Polish Zloty.

- Adjusting the currency settings will modify the pricing type displayed on food products across the application found at the **home**, **basket, and buy templates**.

- The "Currency" dropdown by default is set to Euro, and by pressing this dropdown, it will toggle,
  (slide in/out from below), showing the aforementioned currency options.

- The menu is accessible through an icon located at the top-left corner. Upon tapping the icon, the menu smoothly slides in from the left side, covering approximately 50-70% of the standard mobile device width in portrait mode.

- Users can toggle (hide/show) the menu using the same icon, and also dismiss it by tapping anywhere outside the menu with a slide animation. The menu is designed to be instantly hidden when users access a new template.

cart-template:

The purpose of the cart template is to provide users with an overview of the selected food items, enabling them to review and manage their choices before proceeding to the final purchase. It serves as a central location within the application where users can adjust the contents of their shopping cart and confirm their selections before completing the buying process.

Buttons:
- **Menu**   : ICON, Opens menu **(Previously described at section - menu)**
- **Buy**   : Redirects the user to the **buy template** to provide address details before
                        confirming the purchase.

Labels:
- **Basket** (title) : Template's title at the top of the screen. TEXT
- **Items count**   : The quantity of items added to the basket. NUMBER
- **Total cost**   : (0 <= num_lenght <= 1000) NUMBER
                        Final purchase price for the listed products in the basket

Food Item Container:
- **Image**   : Item's picture, (size: contain as square). IMAGE
- **Title**   : LABEL, is restricted to 20 characters. TEXT
- **Description**   : LABEL, is restricted to 250 characters. TEXT
- **Price**   : LABEL, is restricted to 100 units. (For: Euro, GB Pounds and ZLOTY). NUMBER
- **Remove**   : BUTTON, removes item from the basket, and updates the **Items count** label.

Items in the basket are positioned beneath the "Basket" title label, arranged for vertical scrolling.
The "Total Cost" label and "Buy" button remain fixed at the bottom of the screen for easy access.

buy-template:

The purpose of the buy template is to facilitate the final purchase process by guiding the user through the necessary steps, including providing an address and confirming the selected items from the cart.

Buttons:
- **Return arrow** : The return button redirects the user back to the **cart template** for easy review and adjustments before completing the purchase.

- **Confirm purchase** : The return button displays **loading template** and then redirects the user back to the **home template** with a popup message at the top that emphasises the user-made purchase.

  Any added items to the basket should be automatically removed, and total costs reset to initial default value - "0" upon making a purchase.

Form Inputs:
- **Full name** : (0 < char_lenght < 100) TEXT
  To record the complete name of the user or recipient associated with the delivery.

- **Phone number** : (standard ITU-T E.164: 10 <= num_lenght <= 15) NUMBER, **disclude region code**
  To provide a contact point for communication or delivery-related queries.

- **Postal code** : (2 <= char_lenght <= 11) TEXT
- **City** : (0 < char_lenght < 1000) TEXT
  To specify the exact location for accurate delivery, including the postal code and city information.

- **Address line 1** : (0 < char_lenght < 255) TEXT
  To capture the primary address details, such as street names or apartment information.

- **House number** : (0 < num_lenght < 5000) NUMBER
  To further refine the address details by including the specific house or building number

Labels:
- **Address info** (title) : Template's title at the top of the screen. TEXT
- **Total cost** : (0 <= num_lenght <= 1000) NUMBER
  Final purchase price for the listed products in the basket

- Payment Method?
  How and what do we provide, is this a button for a third party payment service like paypal?

All fields are **REQUIRED** before making a purchase, and must disclude symbols like @!$£" etc. And also, User-provided inputs will be temporarily stored for a duration of 12 hours, with the time limit resetting with each purchase. No input data shall be removed upon exiting the application or submitting the form.