|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RAZDA Co. | | | | |
| **Filename: [add\_product.html]** | | | | |
| **Summary***:*  This file serves as a structured interface for users, such as administrators or sellers, to add new products to Razda Market’s catalog. It includes input fields for product details, validation prompts, error handling, and submission of both textual and image data, ensuring a seamless product addition experience. | | | | |
| ***Processes*** | | | | |
| * **Header and CSS Imports** | **Links to Bootstrap and custom CSS stylesheets.** | | | **Ensures consistent styling and layout, providing a user-friendly interface with modern, responsive elements.** |
| * **Container Structure** | **Wraps content in a Bootstrap container.** | | | **Maintains clean and centered layout while supporting responsiveness on various devices.** |
| * **Flash Messages** | **Displays success/error messages from previous actions.** | | | **Provides immediate feedback on actions, helping users correct mistakes or confirm successful submissions.** |
| * **Product Addition Form** | **Gathers product details through fields like name, description, price, category, etc.** | | | **Central data collection form designed to capture all essential product information efficiently.** |
| * **Form Validation** | **Error messages next to fields for real-time validation.** | | | **Guides users on correct input format, reducing chances of errors and ensuring data integrity.** |
| * **Submit Button** | **Sends form data to the server when clicked.** | | | **Finalizes the product addition process, submitting all provided data for backend processing.** |
| * **JavaScript Imports** | **Includes Bootstrap’s JavaScript dependencies.** | | | **Supports interactive elements, such as modals, tooltips, and alerts, enhancing user experience.** |
| **Files it Gets Information From:** | | | **Files it Sends too:** | |
| * product\_creation.py: Manages backend logic to process and save product data received from the form submission. | | * product\_creation.py: Sends product details for processing, including saving to the database and validating data. | | |
| * product\_catalog.py: Retrieves existing category options for the dropdown menu, enhancing user selection accuracy. | | * Database: Stores all validated product information such as name, description, price, category, etc. | | |
| **Expected input into file:** | | | **Expected output from file:** | |
| **Input**: Textual data for product name, description, and category selection; numeric data for price and inventory count; file data for the image upload. | | | **Output**: Successfully collected and validated data submitted to the server; error messages where applicable. | |
| **Things that need to be taking place:** | | | | |
| |  | | --- | | **- Step 1: User navigates to the form and enters all required product information.** |  |  | | --- | | **- Step 2: User submits the form by clicking Add Product; error messages display if needed.** |  |  | | --- | | **- Step 3: Upon successful submission, a success message appears, confirming the product addition.** |  |  | | --- | |  | | | | | |
| Edit log (update each time you make changes to doc or file). | | | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: | | | | |