|  |  |  |
| --- | --- | --- |
| RAZDA Co. | | |
| **Filename: [verticalcat.py]** | | |
| **Summary***:*  This file handles the creation, display, updating, and deletion of vertical categories on the platform. These categories are displayed in a vertical arrangement, enhancing the browsing experience by allowing users to view and select categories organized for easy navigation. This functionality provides a structured way to categorize products or content in a user-friendly format. | | |
| ***Processes*** | | |
| * **Manage Vertical Categories (GET/POST)** | **Allows administrators to create new vertical categories and view existing ones. Validates category data, logs actions, and handles any errors during data submission.** | |
| * **Delete Vertical Category** | **Enables the deletion of a specific category, removing it from the database and providing feedback to the user.** | |
| * **Edit Vertical Category** | **Facilitates updating category details like category\_name and description, enabling admins to keep categories relevant and up-to-date. Retrieves category data for editing and commits updates to the database.** | |
| **Files it Gets Information From:** | | **Files it Sends too:** |
| * **Database**: Retrieves existing categories from the database for display and editing. | * **Log File** (verticalcat.log): Logs category creation, deletion, updates, and retrievals for monitoring and auditing. | |
| * **User Login** (user\_login.py): May restrict access to certain admin functions if necessary. | * **Database**: Inserts, updates, and deletes category records as per user actions. | |
| * **Session Data** (optional): Accesses session data to customize admin/user-specific actions. |  | |
| **Expected input into file:** | | **Expected output from file:** |
| Form Data: Receives category\_name and description for new categories or updates to existing ones.  Category ID: Passed in the URL path for edit and delete actions to identify specific categories. | | Template Rendering: Renders templates manage\_verticalcategories.html and edit\_verticalcategory.html for displaying and editing categories.  Flash Messages: Provides feedback (success or error) to users after category operations (creation, update, delete).  Log Entries: Logs the creation, update, deletion, and retrieval actions for tracking and debugging purposes. |
| **Things that need to be taking place:** | | |
| |  | | --- | | **● Session Management: If admin-only access is required, session data should be used to enforce permission restrictions.** |  |  | | --- | | **● Logging: Each action (create, edit, delete, view) should be logged to ensure proper monitoring and facilitate troubleshooting.** |  |  | | --- | | **● Error Handling: Database errors (like connection issues or failed queries) should be managed gracefully, with messages indicating action results to the user.** |  |  | | --- | | **● Feedback Mechanisms: Flash messages should inform the user/admin of the success or failure of each operation.** |  |  | | --- | | **● Database Integrity: Ensure category creation, updates, and deletions are correctly committed to the database to prevent data inconsistency.** |  |  | | --- | |  | | | |
| Edit log (update each time you make changes to doc or file). | | |
| **Contribution to the Entire Project:**  The **verticalcat.py** file is essential for the site's category management system. By allowing structured management of vertical categories, it enhances site organization, contributing to a more intuitive and categorized browsing experience for users. This file plays a critical role in providing a streamlined and user-friendly navigation system. | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: Set up vertical category management functionality, enabling admins to add, edit, and delete categories, with logging for each operation and flash messages for user feedback. | | |