|  |  |  |
| --- | --- | --- |
| RAZDA Co. | | |
| **Filename: [product\_sync.py]** | | |
| **Summary***:*  The product\_sync.py file synchronizes product data between Razda Market and SquareSpace, ensuring listings, descriptions, prices, and inventory counts are consistent across both platforms. By regularly updating product information, it provides a cohesive shopping experience and minimizes discrepancies that might confuse customers. It detects updates on either platform, applies changes as needed, and logs each action, preserving consistency across product catalogs. | | |
| ***Processes*** | | |
| * **Enviroment and Logging Setup** | **Configures logging in product\_sync.log to capture synchronization attempts, including successful updates and errors. Loads necessary credentials and environment variables to maintain secure API access and logging consistency.** | |
| * **Authentiication Validation** | **Imports SquareSpaceAuth for authentication with SquareSpace’s API, ensuring all synchronization actions are authorized. Checks for a valid access token to avoid unauthorized updates** | |
| * **Product Data Retrieval** | **Retrieves all products from ProductCatalog, including details like product ID, name, description, price, category, inventory, and image URL. Ensures product data is complete and accurate before syncing with SquareSpace. Logs successful retrieval or errors if issues occur.** | |
| * **SquareSpace Product Sync API:** | **Sends formatted product data to SquareSpace’s API, logging each sync action and capturing API responses. Detects any API errors, logs issues, and ensures that updates are recorded accurately for consistent catalog management.** | |
| **Files it Gets Information From:** | | **Files it Sends too:** |
| * **SquareSpaceAuth (squarespace\_auth.py)**: Manages authentication, providing an access token to securely connect with SquareSpace’s API. Prevents unauthorized access by validating tokens before sync actions. | * **Log file (product\_sync.log)**: Records each synchronization attempt, including timestamps, product data details, status messages, and any errors. Provides an audit trail to monitor sync actions and troubleshoot issues. | |
| * **ProductCatalog (product\_catalog.py)**: Retrieves up-to-date product data, including IDs, names, descriptions, categories, prices, and inventory counts. Ensures SquareSpace receives accurate information for every sync. |  | |
| * **ProductImages (product\_images.py)**: Fetches URLs for product images, ensuring SquareSpace displays current and consistent visuals for each product. |  | |
| **Expected input into file:** | | **Expected output from file:** |
| * **Environment Variables**: .env configuration for SquareSpace API (SQUARESPACE\_API\_KEY, SQUARESPACE\_API\_URL) and database connection details (DB\_HOST, DB\_USER, DB\_PASSWORD, DB\_NAME).  **Product Data**: Structured data from ProductCatalog containing product\_id, name, description, price, category\_name, inventory\_count, and image\_url. **User Session**: Valid session data, confirming the user is authenticated to initiate synchronization. * Example Input Code: * ```python ```python * products = catalog.get\_all\_products() * headers = squarespace\_auth.get\_headers() * squarespace\_api\_url = f".../products" * response = requests.post(...) * response.raise\_for\_status() * logging.info("Sync successful") * ``` | | * **Product Sync Log Entry**: Logs each product sync attempt in product\_sync.log, capturing product ID, sync status, timestamp, and error messages for auditing.  **SquareSpace API Response**: Confirms success or failure of product synchronization, logging details for successful updates or issues to support diagnostics. * Example Output Code: * def sync\_products(): * ... * for product in products: * product\_data.append({}) * response = requests.post(...) * if response.status\_code == 200: * logging.info("Sync successful") * else: * logging.error("Sync failed" |
| **Things that need to be taking place:** | | |
| |  | | --- | | **1. Environment Variable Verification: Ensure that all required credentials (SquareSpace API, database) are present and correctly loaded from .env. Confirm access to SQUARESPACE\_API\_KEY and that it is correctly set for API authentication.** |  |  | | --- | | **2. Logging Verification: Check product\_sync.log to ensure each product sync attempt is recorded, capturing order ID, user ID, product details, status, and error information. Verify log security, ensuring sensitive information is not stored in plaintext.** |  |  | | --- | | **3. Product and Image Data Accuracy:** |   ** Validate that ProductCatalog and ProductImages provide complete and current product data, preventing discrepancies between Razda Market and SquareSpace.**  ** Test for missing product fields and log retrieval errors to ensure all data elements are accurately synced. | | 4. API Key and Authentication Security: Ensure SQUARESPACE\_API\_KEY is securely managed, stored only in .env, and not hardcoded in code. Protect against unauthorized access by restricting key visibility to authorized users. | | 5. Data Structuring for SquareSpace API:**  ** Confirm that all fields (product\_id, name, description, price, etc.) are correctly formatted according to SquareSpace’s API requirements, preventing sync failures due to data mismatches.**  ** Exclude sensitive or unnecessary fields from logging, maintaining security without compromising functionality. | | 6. API Response and Error Handling:**  ** Verify that successful API responses are logged and that errors are captured in detail, including status codes and SquareSpace error messages, aiding in diagnostics.**  ** Implement retry logic for network errors or rate-limited API responses, ensuring robust sync functionality. | | 7. User Session Validation: Ensure only authenticated users can trigger the product sync, recording session and user ID for auditing and preventing unauthorized access. Log session validation outcomes for security auditing. | | 8. Data Consistency Audits:**  ** Periodically review and compare product data between Razda Market and SquareSpace, identifying and resolving any discrepancies.**  ** Develop monitoring tools to alert for unsynced or partially synced products, reducing customer impact from inconsistencies. | | 9. Network Security and API Request Encryption: Ensure all API communications with SquareSpace are encrypted via HTTPS, securing data in transit. Protect access tokens and data from exposure, adhering to security standards. | | 10. User Feedback and Error Transparency:**  ** Provide user-friendly feedback on sync status, especially if errors prevent successful syncs. Guide users to retry or check connection status as needed.**  ** Ensure that logs include specific SquareSpace error messages for debugging, while protecting sensitive data from exposure. | | 11. Schedule and Real-Time Sync Testing:**  ** Test both scheduled and on-demand syncs, confirming that each scenario reliably updates products in SquareSpace.**  ** Verify that logging and error handling work consistently across sync types, ensuring full visibility for administrators. | | 12. Review and Maintenance:**  ** Regularly review product\_sync.log for recurring sync issues, updating error handling and data retrieval practices as necessary.**  ** Document sync procedures, error management, and security practices, ensuring compliance with Razda Market’s data management standards and evolving best practices. |**   |  | | --- | |  | | | |
| Edit log (update each time you make changes to doc or file). | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: | | |