|  |  |  |
| --- | --- | --- |
| RAZDA Co. | | |
| **Filename: [order\_automation.py]** | | |
| **Summary***:*  Automates order processing, retry logic, status updates, and alerts, synchronizing order data between Razda’s system and the factory. Handles discrepancies, cancellations, and validation for a smooth order fulfillment workflow. | | |
| ***Processes*** | | |
| * **Fetch Pending Orders** | **fetch\_pending\_orders retrieves all orders marked as pending from Squarespace for processing.** | |
| * **Process Orders** | **process\_orders manages pending orders in batches, invoking process\_order for each.** | |
| * **Process Single Order** | **process\_order submits each order to the factory, updating status or retrying upon failure. Uses exponential backoff for retries and alerts admins after multiple failures.** | |
| * **Update Order Status** | **update\_order\_status updates the order's status in Squarespace after each stage in the process.** | |
| * **Log Order History** | **log\_order\_history saves order status, details, and timestamps for future reference and tracking.** | |
| * **Notify Admin** | **notify\_admin sends alerts to admins when orders encounter issues that may require intervention, such as multiple failures** | |
| * **Check Order Discrepancies** | **check\_order\_discrepancies finds orders with unexpected statuses, logging and alerting admins if any are found.** | |
| * **Periodic Order Sync** | **periodic\_order\_sync runs the automation at specified intervals, re-running process\_orders and checking for discrepancies.** | |
| * **Cancel Order** | **cancel\_order allows manual order cancellation if necessary, updating Squarespace with the cancellation status and logging the action.** | |
| **Files it Gets Information From:** | | **Files it Sends too:** |
| * **Squarespace API**: Retrieves pending orders and sends updates on order status and cancellations. | * **Factory API**: Submits orders and checks completion status. | |
| * **Environment Variables**: Loads settings for retry limits, batch size, and alert thresholds. | * **Alerts Module**: Sends notifications to admins when orders encounter critical issues. | |
| **Expected input into file:** | | **Expected output from file:** |
| * **Pending Orders Data**: Retrieved from Squarespace, containing order details (SKU, quantity, etc.) ready for processing. * **Order API URL and Keys**: Used to connect and authenticate with Squarespace and factory APIs. | | * **Order Status Updates**: Changes status in Squarespace upon completion or cancellation, improving transparency. * **Order Processing Log**: Logs each step of processing and any issues for auditing and troubleshooting. * **Discrepancy Log**: Flags and alerts any orders not meeting expected status criteria. |
| **Things that need to be taking place:** | | |
| |  | | --- | | **- Regular Order Processing: periodic\_order\_sync ensures orders are checked and processed on schedule, minimizing backlogs.** |  |  | | --- | | **- Admin Alerting: Ensure notify\_admin is correctly configured to notify staff if an order encounters multiple retries or discrepancies, preventing workflow disruptions.** |  |  | | --- | | **- Order Validation: Use validate\_order\_data to confirm all required order fields are present before processing to avoid submission errors.** |  |  | | --- | | **- Exponential Backoff for Retries: process\_order implements backoff for efficient retrying in case of API failures, reducing strain on systems and improving success rates.** |  |  | | --- | | **- Order History Logging: Maintain accurate logs in order\_history for all order changes, supporting transparency and auditing.** |  |  | | --- | |  | | | |
| Edit log (update each time you make changes to doc or file). | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: | | |