|  |  |  |
| --- | --- | --- |
| RAZDA Co. | | |
| **Filename: [schedule\_manager.py]** | | |
| **Summary***:*   |  | | --- | |  |   Manages scheduling and execution of tasks, both recurring and one-time, with priority control, error handling, and alert notifications for failed executions.   |  | | --- | |  | | | |
| ***Processes*** | | |
| * **Task Scheduling** | **add\_task initializes a new task, setting its parameters like name, priority, and recurrence settings. Tasks are stored in a dictionary for retrieval by name. Scheduling for recurring tasks is handled through a dedicated function schedule\_recurring\_task.** | |
| * **Recurring Tasks with Interval Control** | **Recurring tasks are managed in separate threads and executed at specified intervals. schedule\_recurring\_task maintains the next execution time and reinitializes threads if the task’s parameters are updated.** | |
| * **Error Handling and Alerting** | **execute\_task executes tasks and includes error handling with alerts to admins if the task fails. Email alerts contain error details and can be sent if SMTP settings are configured, ensuring prompt attention to critical scheduling issues.** | |
| * **Dynamic Task Priority Management** | **Tasks are executed in ascending order of priority, ensuring high-priority tasks are executed first. update\_task can dynamically modify priority, interval, and other parameters, resetting the schedule for recurring tasks as needed.** | |
| * **Event-Driven Stop Mechanism** | **Each task uses a threading Event to manage its lifecycle, allowing tasks to stop on demand via stop\_all\_tasks. This ensures safe halting of all tasks during shutdown or error scenarios.** | |
| * **Admin Alert Integration via SMTP** | **Alerts are sent to admin emails configured through environment variables. In the event of a task failure, alerts detail the error, traceback, and task parameters for quick resolution. SMTP configurations ensure secure transmission of alerts.** | |
| **Files it Gets Information From:** | | **Files it Sends too:** |
| * **Function Integrations**: Can execute various task functions across modules needing scheduled execution, such as data syncs or alerts. | * **Alerts Module**: Can integrate with a broader alerting system for real-time notifications. | |
| **Expected input into file:** | | **Expected output from file:** |
| * **Task and Parameters**: Receives task parameters such as function to execute, arguments, priority, and interval for scheduling. * **Recurring Task Details**: Recurring tasks require an interval for scheduling, which triggers the task at regular intervals in its thread. | | * **Execution Logs**: Logs each task’s execution time, success/failure status, and next execution time (for recurring tasks). * **Error Alerts**: Admin alerts for failed tasks provide details and are sent via SMTP for immediate attention. |
| **Things that need to be taking place:** | | |
| |  | | --- | | **- Flexible Task Scheduling: add\_task enables flexible task addition with priority, recurrence, and interval settings, supporting varied scheduling needs for Razda’s operations.** |  |  | | --- | | **- Efficient Error Handling with Admin Alerts: execute\_task captures exceptions and triggers an alert to admins, facilitating swift response to critical task failures.** |  |  | | --- | | **- Controlled Task Execution: Threads and event controls ensure smooth scheduling and stopping of recurring tasks, providing reliability across varied task loads.** |  |  | | --- | | **- Dynamic Task Updates: update\_task enables on-the-fly modifications to task parameters, adapting to changing requirements without needing a complete reschedule.** |  |  | | --- | |  | | | |
| Edit log (update each time you make changes to doc or file). | | |
| | **Possible Enhancements:** | | --- |  |  | | --- | | - **Enhanced Dependency Management**: Allow tasks to depend on the completion of other tasks, supporting complex workflows with interdependent scheduling. |  |  | | --- | | - **Advanced Scheduling Patterns**: Expand recurring task intervals to support cron-style schedules (e.g., specific days, times), enhancing schedule flexibility. |  |  | | --- | | - **Task Execution Reporting**: Generate periodic reports summarizing task success/failure rates, execution times, and delays for performance monitoring. |  |  | | --- | | - **Admin Notification Integrations**: Support additional channels like SMS or chat notifications (e.g., Slack) to broaden the reach of failure alerts. | | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: | | |