**Use Case UC-014**

Version 1.00

**Revision History**

| Date | Author | Description of change |
| --- | --- | --- |
| 03/25/2024 | Dominic | Initial version of Use Case |
| 04/03/2024 | Dominic | Modified Main success scenario to match current implementation |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Use Case:** System Job Scheduling Change Requested

**Id:** UC - 014

**Description:** System CRON jobs are requested to be changed for different times of the week by a user of the system

**Level:** User Goal

**Primary Actor**  
Valid user of the system with authorization

**Supporting Actors**

System Administrator

**Stakeholders and Interests**

Data Analyst: Interested in data being collected in a timely logical fashion.

System Administrator: Interested in modifying the job scheduling to meet a demand.

**Pre-Conditions**

1. System Administrator or user is logged into the user interface.
2. The web hosted user interface is available to log into.

**Post Conditions**

Success end condition:

1. Modifications to the CRON jobs

Failure end condition:

1. CRON jobs fail to update or fail to change to requested time

Minimal Guarantee:

1. The system is accessible regardless of changes.
2. The system continues to retrieve data at default settings

**Trigger**

1. Changes to job scheduling is requested or required.

**Main Success Scenario TODO**

1. Authenticated system user navigates to the job scheduling page on the user interface
2. User selects custom or default selection.
   1. On default selection the user then selects which script default should be selected.
      1. The user clicks apply changes.
      2. The back end logic rewrites the CRON jobs to adhere to the default job scheduling time
   2. On custom selection the user must selects which script default should be selected
      1. User inputs data in each box they wish to modify.
      2. User then clicks apply changes
      3. Any unchanged or default fields will maintain the default values for the CRON job
3. User Clicks apply changes button and the CRON jobs are updated

**Frequency:**

On demand

**Assumptions**

1. The UI provides an intuitive understanding of what fields can be changed and for what time scaling.
2. The jobs are running and have access to all the resources they need.

**Special Requirements**

**Security**

1. Access to IONOS hosted user interface with valid credentials

**Change Logs**

1. Changes made to CRON jobs must be logged and by whom.

**Issues**

1. CRON jobs fail to update due to faulty logic or bugs
2. Connection to web host fails