Capstone proposal

Project Proposal: Batch Job Lifecycle Monitor

- Batch jobs tend to be "set and forget"; once created, they run indefinitely and are often not reviewed.
- Over time, this leads to jobs that are outdated, redundant or failing silently
- With hundreds of jobs in iEASA alone, it is difficult to know which ones are still relevant, which one needs maintenance and which ones should be retired
- A wise man once told me, " Jobs that are forgotten or done daily should be automated"

Proposed Solution

The Batch Job Lifecycle Monitor will be a rules-driven system that automatically tracks, classifies and alerts on batch jobs.

Key Features

1. Job Registry: Central record of all jobs, their owners, and definitions
2. Job Activity Tracker: Captures last run date, error history, and modification timestamps
3. Rules Engine: Applies lifecycle rules such as:
    - Inactive for > 3 months -> Stale
    - Consecutive failures > n times -> Needs Review
    - Owner missing or inactive (Left company) -> Ownerless
4. Automated Alerts: Emails/logs when jobs cross thresholds
5. Admin Dashboard: View jobs by lifecycle status, with ability to archive, reassign or review

Complexity

- Database Schema Design: New structured tables
- Rules Engine Development: Generic framework to evaluate jobs dynamically against configurable rules instead of hard coded if-else statements - allow changes to the rules without having to rewrite the code
- Extensibility: New rules can be added without code changes
- Automation: Background processes to continuously check, reclassify jobs and send out notifications
- Integration: Works with existing batch job registry to track run history
- Visualization: Frontend dashboard to provide actionable insights for users
- Quartz Scheduling: Enterprise-grade scheduler ensures jobs are evaluated and alerts are sent automatically

End State Benefits

- Operational Hygiene: Prevent forgotten jobs from running indefinitely
- Reliability: Early detection of failing jobs
- Governance: Clear visibility of job status
- Efficiency: Helps to prioritize and pick out jobs that requires attention, saving time and resources

Novelty

While enterprise job scheduling platforms exists, they are not feasible for iEASA due to complexity and integration overhead. This Capstone Project proposes a lightweight, domain-specific lifecycle monitor that provides similar governance benefits - stale job detection, rule-based classification and automated reminders tailored to our existing batch job architecture.

Tentative Technology Stack

- Backend - Java Spring Boot
    - Supports REST APIs and database access (JDBC)
    - Cleaner and more modern than Servlet/Ant (What iEASA is using)
    - Java-based, similar to iEASA
- Frontend - Vue.js + Bootstrap
    - Supports reactive dashboards
    - Clean and professional UI
- Scheduler - Quartz
    - Authentic - iEASA uses Quartz for batch jobs
    - Industry standard for scheduled tasks in Java
- Rules Engine - Custom Metadata-Driven Evaluator (Java)
    - Rules stored in DB
    - Evaluator applies rules dynamically
- Database - mySQL
    - Mimics iEASA's Oracle-based database to allow design migration
    - Free, widely available and easy to set up
    - May change to Oracle XE if required (closer to iEASA production but free version)