AUGUST, 2024

SCHEDULING APPLICATION

DOCUMENTATION

EZZ AHMED
ORANGE EGYPT

Table of Contents

Objective:

As a project manager, I want to efficiently schedule tasks across multiple departments and resources, so that I can improve our team's workflow and meet project deadlines.

1. I can input tasks with their durations (in days or half-days) and priorities from an Excel file.
2. I can specify available resources and their departments from an Excel file.
3. The system automatically schedules tasks considering dependencies and resource availability.
4. The system achieves a balanced workload while adhering to the constraints.
5. The system generates an Excel output, showing task assignments for each resource by day.
6. Each resource is grouped under his department in the output Excel file.
7. Tasks are color-coded for easy identification across different resources.
8. The output Excel file opens automatically after generation for immediate review.

Technical Document:

System Overview:

This document provides a technical overview of the task-scheduling application, which sorts and schedules tasks based on their dependencies and durations. The system reads tasks, resource data from an Excel file, processes it to decide the best order of execution, and outputs the scheduled tasks in an organized Excel format.

Architecture

- **Frontend:** The user interface is built using Tkinter for a desktop GUI, allowing users to interact with the application, select files, manage dependencies and view the output.
- **Backend:** The core functionality is implemented in Python, including task sorting, dependency resolution, and Excel file handling.

Data Input:

- Input is a single excel file. The app automatically picks the first 2 sheets
- Tasks are read from the Excel file (Sheet1)
- Task information includes name, priority, durations for different departments.
- Example:

Feature	Priority	BE API	Android	IOS	QC creation	QC execution
VCN - Listing	1	3	4	4	1.5	2
VCN - creating	3	3	4	4	0.5	1
Meeza	5		1	1	0.5	1.5
....	...	...	...	...	...	...

- Resources are read from the same Excel file (Sheet2).
  - Example

BE API	Android	IOS	QC
Ezz	Hour	George	Abdullah
Sama	David	Bahey	Mohamed
			Ruba

- Dependencies can be changed by altering dependency_rules.JSON file or through the GUI
  - Example: right side is dependent on left

```
{
        "Creation": ["Execution"],
        "BE API": ["IOS", "Android"],
        "IOS": ["QC execution"],
        "Android": ["QC execution"],
        "QC Creation": ["QC execution"]
}
```

Scheduling Algorithm:

- Task dependency is read from the JSON file and is set by default as follows.
  - BE API → Android/IOS → QC execution.
  - QC creation → QC execution
- Tasks are sorted topologically and by priority.
  - If multiple tasks have the same priority they are sorted by their total time
- Resources are sorted by their earliest available time.
  - If multiple resources have the same start time available, they are then sorted by their workload.
- The scheduler iterates through sorted tasks, assigning them to a resource.
- Dependencies are respected by ensuring dependent tasks start after their prerequisites finish.
- Special handling for QC tasks ensures the same resource assignment for both creation and execution.

Output Generation:

- An Excel file is generated with the following features:
  - Auto-generated colors for color-coding tasks for easy identification
  - Merged cells for department headers.
  - Automatically adjusted column widths
  - Centered and wrapped text for readability

Usage:

To run the application,

1. Open the .exe file.
2. Select the correct input excel file Ensure that the file is with properly formatted task and resource data.
3. If any dependency needs changing, click on the Dependency Rules button
   a. In the Dependency Rules window, you'll see a list of departments along with checkboxes for other departments.
   b. For each department, select the checkboxes corresponding to the departments that must be completed before it can start.
4. Select the desired name and path for your output
5. Click on the Run Scheduler button.

App Link:

https://drive.google.com/file/d/1tkyTSy6trAhBVTWmeQChTMbFYY1B2sc3/view?usp=sharing