

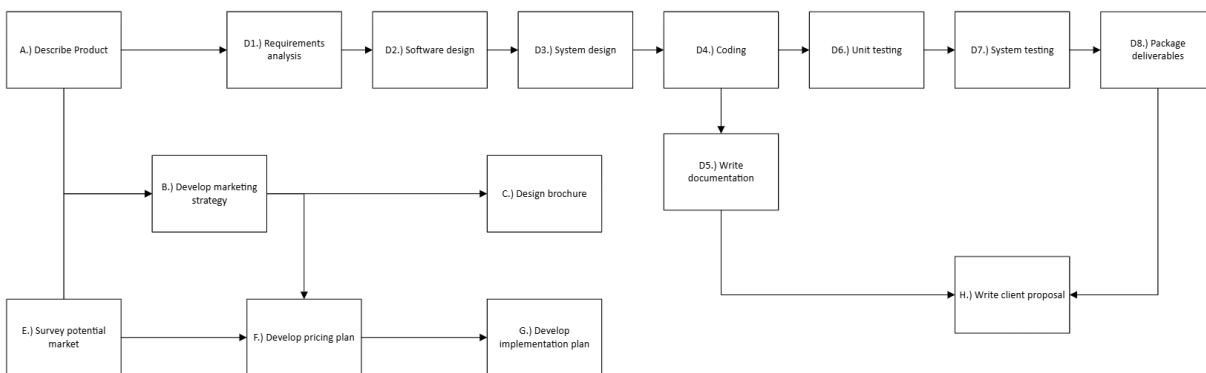
## Assignment 2: Network Models – Project Management

NWU MSDS 460

Thomas B Young

This write-up explains a hypothetical business situation for a freelance development project. It develops a project schedule optimization system using linear programming (LP) and critical path analysis (CPA). Its intent is to efficiently allocate resources, minimize completion time, and visualize dependencies via Gantt charts. Besides basic PuLP scripts, I have included a desktop and web-based solution. Lastly, I made a prototype of the desired application, which matches the requirement minus being hosted and data being stored in a Postgres DB- the version I built uses saved JSON files and the Yelp Fusion API.

The initial Excel workbook I modified per my thoughts on the project roles necessary. For example, instead of separating front and backend development, I went with 2 full-stack developers. My greatest area of uncertainty for the project is on the desired functionality. By “recommendation system” would a form for the end user to input parameters suffice? If it is more complex (viz., ML) how are we sourcing the user psychographics (maybe intake form on signup)? Also, if we are only concerned with this one city, I don’t see data collection from Yelp to be that difficult. A scheduled task running nightly seems sufficient.



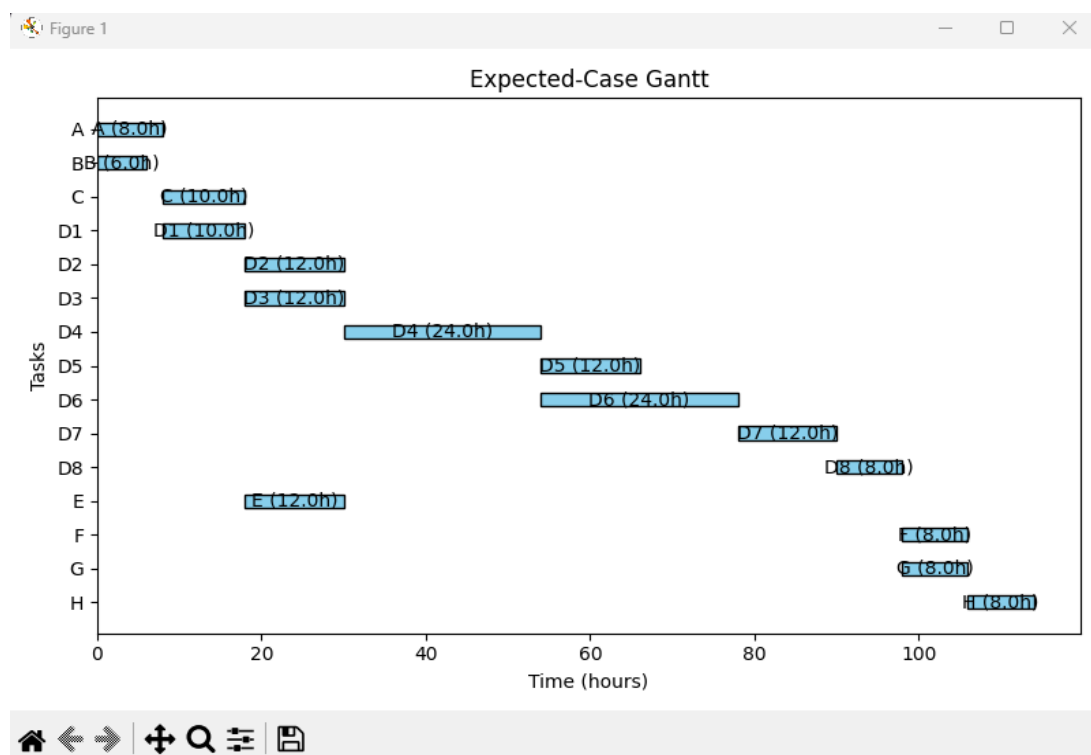
Above is a directed graph diagram (Visio) that visualizes these dependencies and shows where tasks can proceed in parallel. For instance, marketing activities (B) can commence independently from product design (D) and then feed into F (Develop pricing plan). Parallelization is thus clearly viable, but it depends on having the right roles and working concurrently.

I began working on the LP model by identifying each task in the set and noting their predecessor requirements. Since minimizing the project duration is the focus I added two primary variables for each task- start time and completion time- then defined a fixed duration for each task. The latter can be set to best-case, expected, or worst-case hours, depending on which scenario is needed. The model also links start and completion times and ensures non-negativity. To capture the total project time, there is an auxiliary variable that must be at least as large as every task’s completion time (imposing constraints). The objective is then to shrink the project’s finish time to its smallest feasible length given the task durations and dependencies. Consequently, a solution that yields the minimum also yields the minimum overall cost.

I began working on the LP model by identifying each task in the set and noting their predecessor requirements. Since minimizing the project duration is the focus I added two primary variables for each

task- start time and completion time- then defined a fixed duration for each task. The latter can be set to best-case, expected, or worst-case hours, depending on which scenario is needed. The model also links start and completion times, and ensures non-negativity. To capture the total project time, there is an auxiliary variable that must be at least as large as every task's completion time (imposing constraints). The objective is then to shrink the project's finish time to its smallest feasible length given the task durations and dependencies. Consequently, a solution that yields the minimum also yields the minimum overall cost.

To implement the linear programming model, I used the PuLP library. For quick Gantt visualization I used matplotlib. To attempt more interactive functionality, I built a simple tkinter gui where users can change hours in the table. Further, I built a vanilla JS web version thinking a real time interface for forecasting. The assumption was that team members would be inputting their hours which would update a dashboard for our project managers.



The model (script version verbose.py) has three sets of assumed durations. The first being the best-case scenario and the completion times aligning with optimistic projections. There is the expected duration, which is the most realistic (time and cost) given the tasks. The worst-case projection takes a pessimistic stance and pushes the upper bound estimates. This would be helpful when wanting to gauge the risk of overshooting our schedule.

Ultimately, this project serves as a POC, demonstrating how network models and LP can be used in streamlining project management. While the prototype isn't fully hosted and persists data via JSON rather than Postgres, it lays the groundwork for a scalable solution. Lastly, there is a drafted mock proposal for the hypothetical client. Feedback on desired functionality—especially around the recommendation system's complexity—would help clarify the project scope moving forward.