LINKEDIN

Northwestern University

MSDS 460-DL : Decision Analytics

ASSIGNMENT 2: Network Models

The Project Management (Spring 2024)

April 21, 2023

Humberto Consolo Holanda

▶ Table of Contents

Contact

Humberto Consolo Holanda - https://www.linkedin.com/in/hconsolo - humbertoconsoloholanda2025@u.northwestern.edu

Project Link: https://github.com/Hconsolo/MSDS460-Assignment_01_

(back to top)

Introduction

This assignment will cover the application of linear programming, a mathematical technique, to solve a network model for optimizing a project's time-cost trade-off using the critical path method (CPM). A network used to represent a project is called a project network, consisting of nodes and arcs, containing three essential pieces of information to describe a project: activities, precedence relationships, and time information in terms of duration of each activity and when it is done (start and end time). In this assignment, we will use the activity-on-node (AON) notation, which is more accessible for inexperienced users. The goal is to determine the critical path for three scenarios: best case, expected, worst case. We have provided code to calculate the critical path for those scenarios. Additionally, we will provide some brief information about crashing activities, although the paper's objective is to explore only the critical path. Project crashing is a schedule compression technique in which you bring in additional resources to complete two tasks simultaneously. The Project Management Body of Knowledge (PMBOK® Guide) defines the crashing technique as a way to shorten any project schedule for the least incremental cost.

Our project will use Yelp data of restaurants located in Marlborough, Massachusetts, to build an app that directed to the population of Marlborough and nearby cities, including Berlin, Hudson, Sudbury, Framingham, Southborough, and Northborough, totaling about 173 thousand people. We will update the list of restaurants monthly, while Yelp reviews, the lifeblood of our app, will be updated daily. To access large quantities of Yelp data, we will secure a contract with GraphQL API, underscoring the importance of this data source in our project.

We will use the following technical professional and tools to develop the app:

- Front-end developer: Alpine.js and Tailwind
- Backend developer: GraphQL API and a Go web and database server
- Data scientist: Go, Python, or R for recommender system analytics on the backend, with persistent storage provided by PostgreSQL, EdgeDB, or PocketBase
- Data engineering aspect of our project, a critical component, will be hosted on a major cloud platform such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP). This strategic decision ensures the scalability and reliability of our project, instilling confidence in its success.

A project manager will oversee the entire process of planning, marketing, pricing, and work management from start to finish. All professionals working on the project are independent contractors. This means that they will only charge for the time they spend working directly on the project. Therefore, if a particular task is ongoing, other team members will not charge the project for their time. The IRS defines an individual as an independent contractor if the payer has the right to control or direct only the result of the work and not what will be done or how it will be done ("Independent Contractor Defined" 2024).

(back to top)

Method

For this task, we employed Python and the following third part libraries to help with basic data handling tasks and to solve the the LP problem.

- Pandas - Data Wrangling and Processing
- Matplotlib, Seaborn - General Data Visualization
- Plotly.express - Advanced visualization: Gannt Chart and HTML figures rendering
- Networkx - Graph tool and Network visulization
- PulP - Provides a modeling framework and LP optimization tool

The first task was to complete the project planning table and adapt it to three app development scenarions. The table contains information in working hours.

1. ***bestCaseHours***: earliest possible duration time for a task
2. ***expectedHours***: most likely duration time for a task

3. ***worstCaseHours***: latest estimated duration time for a task

We researched the requirements for developing web-based apps and created a proposal outlining each team member's scope of work. We broke down coding tasks per role and developed a project plan to estimate the time required for each technical professional (front-end developer, back-end developer, data scientist, data engineer) to complete their coding tasks.

The tasks have been estimated using the T-shirt size method. A large task requires 40 working hours or an entire week, a medium task requires 20 working hours or about 2.5 days, and a small task requires 10 hours or 1.2 days. A working day is defined as 8 hours. All professionals will be paid 60 USD per hour as per ZipRecruiter's 2024 report ("Data Science Developer Salary"). The code activities per role are summarized in Table 01, with more details available in the appendix in Table 05.

Table 01: Summary of the expected effort (in hours) to develop the code of each design feature (ID starting with F) and technical feacture (ID starting with T).

Role	Effort	Design_ID
Backend Developer	180	F01, F02, F04, F05, F06, F07, F08, F10, F11, F12, F13, T05, T06
Data Engineer	160	F11, F12, T01, T02, T03, T04, T05, T06
Frontend Developer	150	F01, F02, F04, F05, F06, F07, F08, F10, F13
Data Scientist	80	F11, F12

Backend development has the busiest role, while front-end tasks pose the most considerable risk due to possible changes in aesthetics or added functionality. These factors are considered in calculating the critical path for each scenario. We used the Pulp package and Linear Programming techniques to solve the problem of minimum flow network.

Our restaurant recommendation app will feature unique functionalities such as browsing menus, online ordering, table reservations, loyalty programs, push notifications, reviews, and social sharing. The app will use an advanced machine-learning recommendation engine to provide personalized recommendations and will store user preferences and order history. Restaurant owners can post marketing content, and the app will have information on over 100 and less than 1000 restaurants in Marlborough. It will support up to 1000 users and use monthly updated Yelp reviews accessed through GraphQL API. A detailed list of features and requirements is available in Table 01, and the client, a group of restaurants from Marlborough, will decide the complete list of features.

After breaking down the code activities, we created a network or graph diagram for the whole project. For a comprehensive understanding of the tasks, their duration, and whether they are performed individually or in a team of one or more people, please refer to Table 06 in the appendix.

Network **Figure 01: directed graph diagram for the project.**

(back to top)

Results

We applied LP simplex algorithm from Pulp to determine the total duration of the project. The critical path activities were the ones with slack variables equal to 0, meaning that they had to start the earliest possible. Using the results of the problem, we plotted the Gannt chart for each of the scenarios.

Gannt 01. Expected project tasks

The coding of the application is the most important part of the project and the back-end developer's coding is expected to be critical if there are many features to be developed. However, the front-end development could be the most risky part if the restaurants' group has to approve the design proposal. This is shown in Figure 02, which indicates that the Front-End is the key to deliver the project within two months. Therefore, a time-cost trade-off analysis should be conducted, and it may be necessary to hire an additional front-end developer if the project scope is too complex. The scope of data engineers and back-end developers is more predictable, but the front-end development heavily depends on customer feedback.

Gannt 02. Comparison of All Scenarios

The results of the optimization problem for each scenario are presented in Table 02. Our estimate is consistent with the anticipated timeline for web-based application development as outlined in Hryshchenko (2024). According to our findings, a basic application usually takes 2-3 months to develop and costs around $100,000. Since independent contractors follow a business model, the primary error of the solution is not in the total cost (with a variance of approximately 20% between expected and worst-case scenarios) but in terms of project duration (with a discrepancy of approximately 40% between expected and worst-case scenarios).

Table 02: The project scenario comparison in terms of effort, project duration, and total costs

Scenario	Total Work (days)	Project Duration (days)	Total Cost (USD)
Best Case	194.25	49.375	93240
Expected	217.5	55.625	104400
Worst Case	259.875	74.75	124740

(back to top)

Conclusion

In this paper, we used linear programming techniques to analyze a simplified project planning, which is a well-known task to determine the critical path. We determined the coding level of effort for each professional (front-end developer, back-end developer, data scientist, and data engineer) based on the expected features of the restaurant recommendation application. We expect that the back-end developer will perform most of the code effort in the best-case and expected scenarios, but the front-end developer will require most of the time to complete the worst-case scenario. According to the LP solution using Pulp, the project duration should take between two to three months, and it should cost roughly $100,000. The most uncertaint aspect of the project is its completion time.

To monetize a restaurant recommendation app or website, we considered various strategies, including a commission-based model where we charge restaurants a commission fee for transactions made through our platform; a subscription model paid by consumers to have access to virtual coupons and special offer premium plans for additional features; advertising revenue generated through targeted ads; premium listings where we charge a fee for premium listing packages for restaurants; sponsored content where we collaborate with restaurants for sponsored content, and machine learning data monetization using curated analysis about the reviews from Yelp, demographics, and other insight data to provide insights to restaurants.

(back to top)

Reference

1. "Data Science Developer Salary". ZipRecruiter. Accessed on April 21, 2024
2. Hryshchenko, Myroslav. "How Long Does It Take to Build an App (Full Breakdown)". SPD Load. March 14, 2024
3. "Independent Contractor Defined". Internal Revenue Service (IRS) of the United States. Accessed on April 21, 2024
4. Project Management Institute. 2017. A Guide to the Project Management Body of Knowledge (PMBOK Guide). 6th ed. Newton Square, PA: Project Management Institute.
5. Sliger, Michele. Agile estimation techniques. October 23, 2012. PMI® Global Congress 2012—North America, Vancouver, British Columbia, Canada. Newtown Square, PA: Project Management Institute.

(back to top)

Appendix

Reference for the design features

Table 05: Details of the design features to be developed

Design_ID	Design_Description	Details
F01	Desktop and mobile version	Users can access the app on desktop or mobile. Build an attractive and easy-to-use website for both.
F02	Menu browsing	Users can view the restaurant's menu and make selections per restaurant.
F04	Reservations	Users can reserve a table at the restaurant or order for pickup.
F05	Loyalty program	Users can earn points or rewards for repeat business.
F06	Push notifications	Restaurants can send notifications to users, such as special promotions or discounts, using recommendation system.
F07	Reviews and ratings	Users can rate their dining experience and read reviews from other customers.
F08	Social sharing	Users can share their dining experience on social media platforms.
F10	Integration with other platforms	Users can order food from the restaurant app through platforms like Google Maps and Yelp.
F11	ML recommendation	ML recommendation tool personalized suggestion engine that uses Yelp, order history, GPS, and deals data.
F12	ML sentiment analysis	Our app provides sentiment analysis to restaurants based on Yelp reviews, highlighting their main strengths and weaknesses.
F13	Marketing content	Our app lets restaurant owners post marketing content and maintain a blog. It also stores users' order history and preferences.
T01	Size of restaurants DB	The app must store info for 100-1000 Marlborough, MA restaurants.
T02	Scalability	The application should be scalable enough to handle potential increases in the size of operations.
T03	Size of users DB	Users should be 1,000 or 0.06% of the population in Marlborough and its neighboring towns: Berlin, Hudson, Sudbury, Framingham, Southborough, and Northborough.
T04	ELT external data - Yelp	The project will use Yelp reviews of restaurants as external data, which will be updated monthly. The Yelp reviews will be accessed daily via GraphQL API.
T05	Site Infrastructure	Hosted site on a major cloud platform, run batch jobs, and ML algorithms periodically.
T06	Monitoring system	Store activity log, and trigger alerts when events happen

Table 06: Project plan to develop the app restaurant

	taskID	task	predecessorTaskIDs	bestCaseHours	expectedHours	worstCaseHours	projectManager	frontendDeveloper	backen
0	A	Describe product		18	20	22	1	0	
1	B	Develop marketing strategy		36	40	44	1	0	
2	C	Design brochure	A	18	20	22	1	0	
3	D	Develop product prototype	A	0	0	0	0	0	
4	D1	Requirements analysis	D	40	40	40	1	1	
5	D2	Software design	D1	40	40	40	1	1	
6	D3	System design	D1	40	40	40	1	0	
7	D4.0	Begin Code	D2, D3	0	0	0	0	0	
8	D4.1	Coding - Front-end dev	D4.0	130	150	300	0	1	
9	D4.2	Coding - Back-end dev	D4.0	160	180	200	0	0	
10	D4.3	Coding - Data Scientist	D4.0	60	80	120	0	0	
11	D4.4	Coding - Data Engineer	D4.0	140	160	180	0	0	
12	D4.5	End Code	D4.1, D4.2, D4.3, D4.4	0	0	0	0	0	

	taskID	task	predecessorTaskIDs	bestCaseHours	expectedHours	worstCaseHours	projectManager	frontendDeveloper	backen
13	D5	Write documentation	D4.5	18	20	22	1	1	
14	D6	Unit testing	D5	30	40	50	0	1	
15	D7	System testing	D6	30	40	50	1	1	
16	D8	Package deliverables	D5, D7	18	20	22	1	1	
17	E	Survey potential market	B, C	28	30	32	1	0	
18	F	Develop pricing plan	D8, E	13	15	17	1	0	
19	G	Develop implementation plan	A, D8	28	30	32	1	0	
20	H	Write client proposal	F, G	13	15	20	1	0	