Assignment 2: Network Models
Project Management (Fall 2024)

Emily Ekdahl

MSDS 460: Decision Analytics

Dr. Thomas W. Miller

2024-10-27

Introduction

Project planning is a classic example of a network model. In this assignment, I am simulating a scenario where I'm proposing and delivering a recommendation system for a group of restaurant owners. The clients have already made technical selections which largely dictate the tools I can use to deliver the project. While I could perform most of these functions as I've worn most of these hats in my tech career, I'm going to hire much of this work out to other capable contractors to expedite the timeline. The goal is to minimize time to delivery while taking into account project dependencies, as well as time estimates.

Method

To begin, I took the excel project planning spreadsheet and estimated best, worst, and most likely scenarios for various elements of the project. I then made a directed graph that visualized dependencies to help me understand better how my plan was shaping up.

Next, I set up a linear programming model for the project plan using the Python library, PuLP. For the objective function, I'm attempting to minimize delivery time while accounting for project dependencies and time estimates. I had the good fortune of being granted a reprieve in this simulation in that I can pay everyone the same hourly rate. The decision variables are the start time of the task and the duration of the task. Each task has a best-case, worst-case and expected duration. If there's anything I've learned in tech, it is that I should probably double my worst case estimate. I'll run the scenario for best, worst and expected time estimates optimizing for minimum time to delivery . As far as constraints go, we have constraints around task dependencies, task completion, and of course, non-negativity since we're dealing with real work in real time. The code and output is saved in the associated github repository under code.txt and output.txt.

Using those scenarios, I generated gantt charts to help me visualize the timelines so I'll know as soon as they start slipping. It will also help the client understand my plan via a different medium.

Results

Optimizing for time to deliver, I estimate that in the best case scenario, I can deliver the project in 480 working hours, worst case is 980 hours, and most likely scenario is 735 hours, given project dependencies pictured in Figure 1.

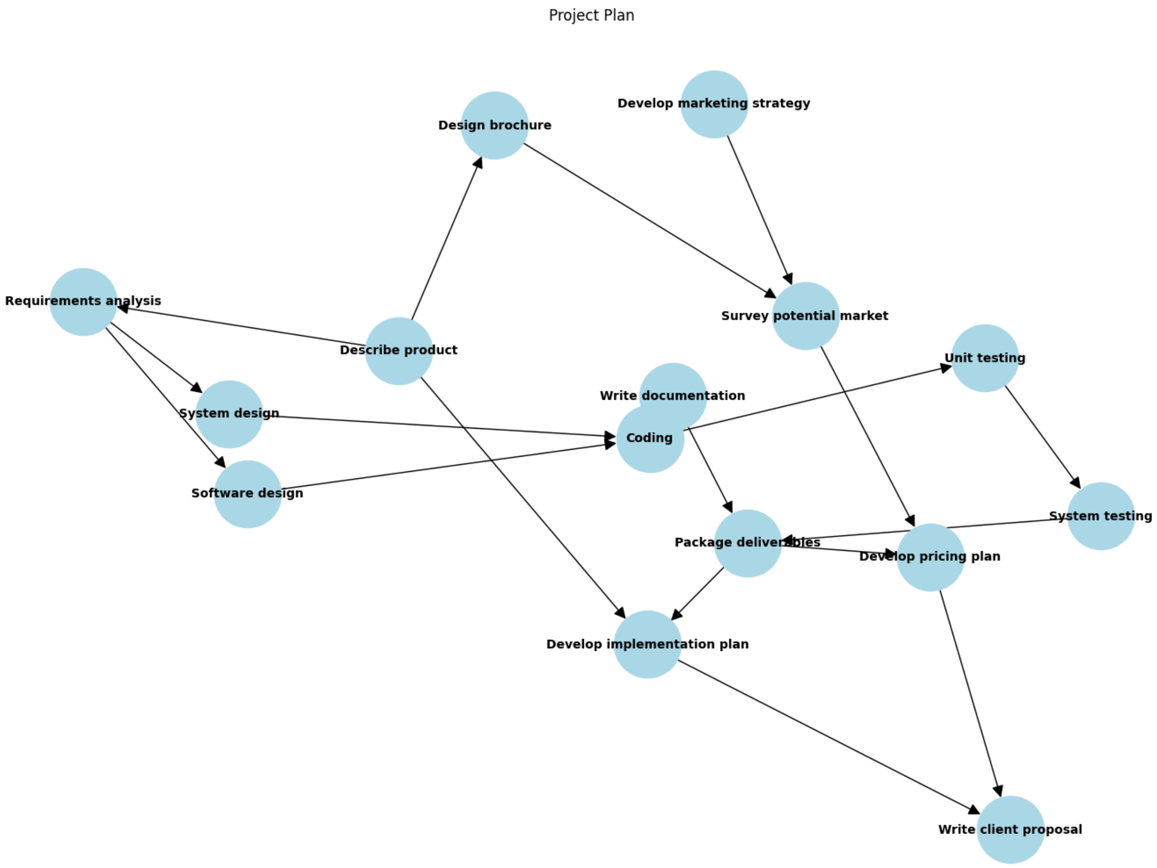

Figure 1

I've never been on a software project that came in on time and on budget, but here's hoping this one will be different. Figure 2 represents the gantt chart for the expected timelines.

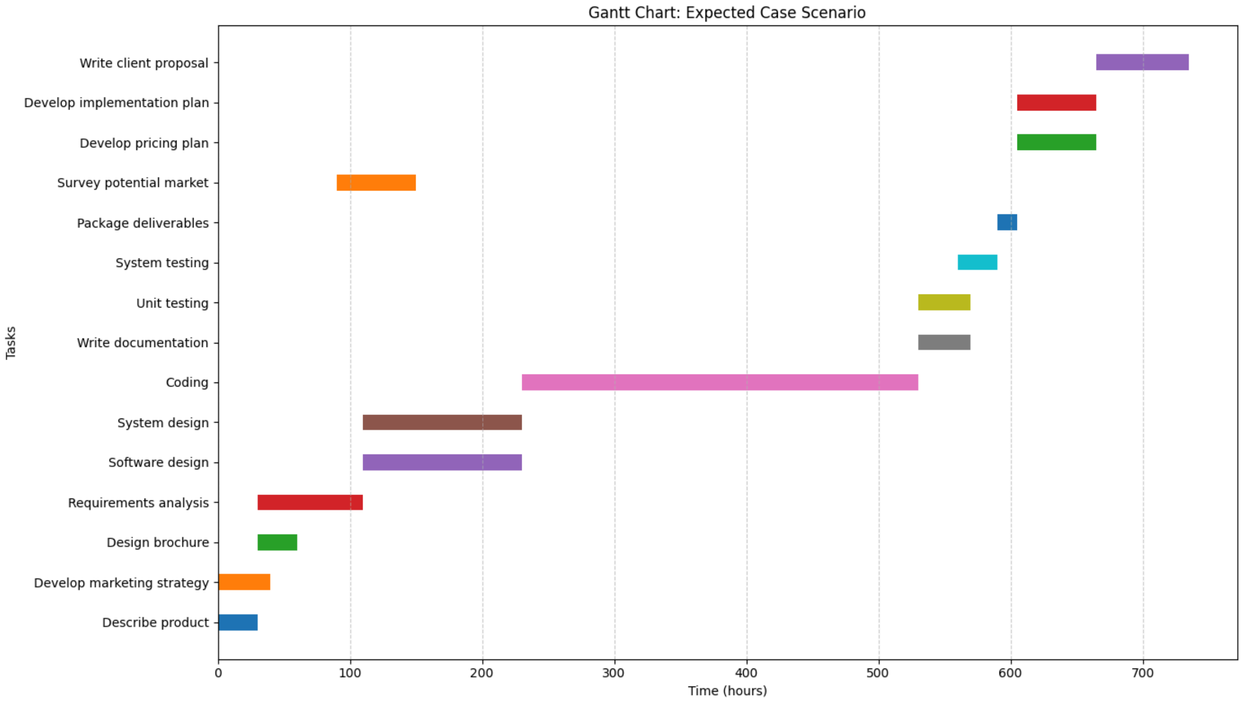

Figure 2

Project Overview

My project proposal is to develop a customer-focused recommendation system for 100 local restaurants in Marlborough, using modern technologies and infrastructure. Given this is a full stack ML application and we aim to deliver on an aggressive timeline, I will be contracting for frontend, backend, data engineer, data science, design, and project management. We will deliver the technologies specified by the restaurant group including Javascript, GraphQL, Go backend, as well as a Python recommendation system and a PostgresQL database. The most likely cost and timeline for the project is 735 hours and $73,500 for labor, with a worst case scenario of 980 hours and $98,000 in labor costs. (I would not share a best case scenario with the client; I find it highly improbable that it would occur and it will set me up for a scenario where I overpromise and under-deliver.) If we add additional contractors, we estimate that we might be able to deliver in 500 hours and costs would increase accordingly.

References

Dinwiddie, George. Software Estimation Without Guessing: Effective Planning in an Imperfect World. Dallas, TX: Pragmatic Bookshelf. [ISBN-13:978-1680506983]

Falk, Kim. 2019. Practical Recommender Systems. Shelter Island, NY: Manning. [ISBN-13: 9781617292705]

Hillier, Frederick S., and Gerald J. Lieberman. 2021. Introduction to Operations Research (eleventh edition). New York: McGraw-Hill. [ISBN-13: 978-125987299-0] Online resources, including Chatper 22 Project Management withPERT/CPM at https://highered.mheducation.com/sites/dl/free/1259872998/1126268/Hillier_IOR_11e_Ch022_WebChapter.pdfLinks to an external site.

Majumdar, Partha. 2021. Linear Programming for Project Management Professionals: Explore Concepts, Techniques, and Tools to Achieve Project Management Objectives. India: BPB Publications. [ISBN-13: 978-9355511164]

Williams, H. Paul. 2013. Model Building in Mathematical Programming (fifth edition). New York: Wiley. Section 5.3.7 Critical Path Analysis, pages 94–98. [ISBN-13: 978-1-118-44333-0] We provide a Python PuLP solution to Williams' home building example using an activities dictionary and derived start_time and end_time decision variables: Williams-Critical-Path-Analysis.pyDownload Williams-Critical-Path-Analysis.py

Wisniewski, Mik, and Jonathan H. Klein. 2001. Linear Programming/Critical Path Analysis. New York: Palgrave. [ISBN-13:978-0-333-76355-1] Chapter 8, Critical Path Network Analysis Techniques, pages 119–147. Available from Course Reserves.