# Marlborough Recommendation Project Proposal

Madison Trimble

MSDS460 - Spring 2024

## I. Problem Setup

Whether it be a special occasion, Sunday brunch, or just a quick bite to eat after work, people want to make sure they will get quality food and service at a chosen restaurant. The purpose of this project proposal is to recommend the best course of action in hiring a team for the development and implementation of a recommendation system that will allow customers to explore eatery options based on other customers' reviews. The project involves eight activities; describing the product, developing marketing strategy, designing marketing materials, developing a prototype of the product, surveying potential users, establishing a pricing structure, planning for implementation, and a final proposal to be presented to the client. Due to the highly technical nature of the product, developing the prototype involves its own special set of sub-tasks. These include analyzation of the requirements necessary for a successful product, designing the software and systems, writing code and documentation, testing at a unit and system level, and defining package deliverables. Several tasks can be completed concurrently with others; for instance, surveying the potential customers is an activity that can take place at any step in the project prior to the final determinations of pricing and implementation.

Developments in the early stages of the process include task assignment and time estimations, and cost projections. Each of which can be accessed in Figure 1 at the end of this document. The pay rate for the team members will be $55 per hour and the roles include project manager, backend and frontend developers, data scientist, and data engineer. Uncertainty could be introduced into this process and impact the final product in several ways. An aspect that could potentially cause time delays and cost increases would be unforeseen factors such as illness or hardware and software issues altering our task durations. Another element is the quality of data.

Since the data is sourced from Yelp reviews written by customers, there will be a significant amount of bias at play which could potentially alter the user experience.

## II. Model Specification

The objective of this analysis is to minimize total project costs, so the programming model chosen was a linear programming model. The objective function defied to achieve this optimization is as follows:

$$\text{Cost} = \sum_{\text{task}} \text{Duration} * \text{Rate}$$

The constraints that are relevant to this problem are vital to ensuring that not only is the project completed at an optimal cost, but in a logical manner. The first of these constraints is the time duration. Although and obvious element, the measures of best, expected, and worst-case scenarios introduce the potential for a lot of variability in the project timeline. The tasks dependency on each other provides an additional constraint to the problem at hand. A project with this many moving parts requires careful completion of certain deliverables before moving onto the next step. Introducing these considerations into the objective function ensures that the project flows in an efficient way.

## III. Linear Programming

The linear programming problem is solved using the Python PuLP library in Jupyter Notebook. First, the tasks and their dependencies are defined according to their task ID in Figure1 before defining their durations for each of the scenarios and the hourly rate for the work that is to be done. Next the problem is created by using LpProblem and LpMinimize tools from the PuLP library. The decision variables for the problem are defined using the LpVariable function. These include the start and end time for each task in each of the best, expected, and worst-case scenarios. The objective function and constraints are defined and applied to each task. Finally, the problem is solved using the solve() function, and the results are printed for each task within each scenario.

## IV. Solution

Solving the linear programming problem not only provides the minimized cost for the scenarios, but also uncovers the critical path. In each scenario the path follows the activities A – B – C – D1 – D2 – D3 – D4 – D5 – D6 – D7 – D8 – F – G – H. This tells the team that most of the activities in our project do not have flexibility in their timeline. The breakdowns of the optimal durations for best-case, expected, and worst-case scenario can be found in the Gantt charts in Figures 3, 4, and 5. The minimal costs and optimal schedule for each case provide useful information for the ways in which tasks can be sequenced and resources can be allocated. Best-case scenario cost is $10,780 and lasts a total of 132 hours. The expected scenario has a total duration of 165 hours and cost of $13,475. The worst-case scenario will take 198 hours to complete the project and cost $16,170.

## V. Overview

This proposal introduces an optimized schedule for a project aiming to develop a recommendation system for restaurants in Marlborough, MA using data from online Yelp reviews. With requirements of specialized technology such as Alpine.js, Tailwind, GraphQL API, and Go Web, this project plan provides a sequenced approach to product development and in-depth analysis of the tasks and costs involved. Linear programming techniques were leveraged to optimize the predicted project duration while minimizing the total cost. In addition to the optimized project schedule, this proposal includes critical path analysis and visualizations outlining all pertinent information. The anticipated project timeline spans 2-3 weeks and the total project cost can be expected to be between $11,000 and $17,000. At project completion the system will be fully functional and easily integrated with existing systems and software.

Appendix:

Figure 1: Project planning table listing best case, expected, and worst-case scenario activity durations and costs.

| taskID | task | predecessor TaskIDs | bestCaseHours | expectedHours | worstCase Hours | projectManager | frontend Developer | backendDeveloper | dataScientist | dataEngineer | BestCaseCost | ExpectedCost | WorstCaseCost |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | Describe product | | 4 | 5 | 6 | 1 | | | | | $ 220.00 | $ 275.00 | $ 330.00 |
| B | Develop marketing strategy | | 8 | 10 | 12 | 1 | | | | | $ 440.00 | $ 550.00 | $ 660.00 |
| C | Design brochure | A | 4 | 5 | 6 | 1 | | | | | $ 220.00 | $ 275.00 | $ 330.00 |
| D | Develop product prototype | | | | | | | | | | | | |
| D1 | Requirements analysis | A | 8 | 10 | 12 | | 1 | 2 | | | $ 440.00 | $ 550.00 | $ 660.00 |
| D2 | Software design | D1 | 16 | 20 | 24 | | 1 | 2 | | | $ 880.00 | $ 1,100.00 | $ 1,320.00 |
| D3 | System design | D1 | 20 | 25 | 30 | | 1 | 2 | | | $ 1,100.00 | $ 1,375.00 | $ 1,650.00 |
| D4 | Coding | D2, D3 | 24 | 30 | 36 | | 1 | 1 | 1 | 1 | $ 1,320.00 | $ 1,650.00 | $ 1,980.00 |
| D5 | Write documentation | D4 | 16 | 20 | 24 | 1 | | | | | $ 880.00 | $ 1,100.00 | $ 1,320.00 |
| D6 | Unit testing | D4 | 20 | 25 | 30 | | | 2 | | | $ 1,100.00 | $ 1,375.00 | $ 1,650.00 |
| D7 | System testing | D6 | 24 | 30 | 36 | | 1 | 1 | | | $ 1,320.00 | $ 1,650.00 | $ 1,980.00 |
| D8 | Package deliverables | D5, D7 | 12 | 15 | 18 | | 1 | 2 | | | $ 660.00 | $ 825.00 | $ 990.00 |
| E | Survey potential market | B, C | 16 | 20 | 24 | | | | 1 | 1 | $ 880.00 | $ 1,100.00 | $ 1,320.00 |
| F | Develop pricing plan | D8, E | 4 | 5 | 6 | 1 | | | | | $ 220.00 | $ 275.00 | $ 330.00 |
| G | Develop implementation plan | A, D8 | 12 | 15 | 18 | | 1 | | | | $ 660.00 | $ 825.00 | $ 990.00 |
| H | Write client proposal | F, G | 8 | 10 | 12 | 1 | | | | | $ 440.00 | $ 550.00 | $ 660.00 |

Figure 2: Directed Graph for activities in Marlborough Project

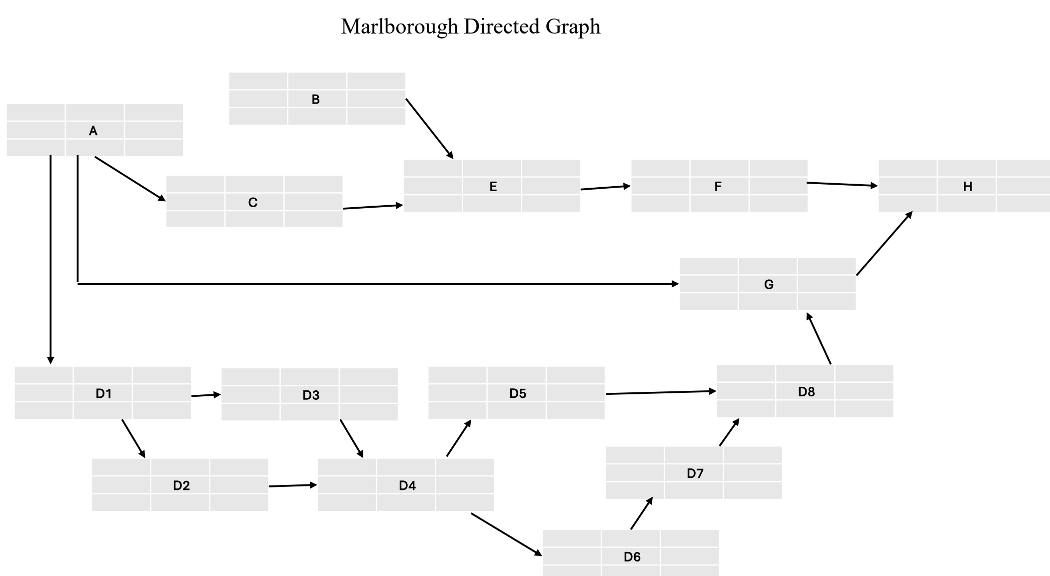


Figure 3: Best Case Gantt Chart

| taskID | task | predecessorTaskIDs | bestCase Hours | ES | LS | EF | LF | Slack |
|---|---|---|---|---|---|---|---|---|
| A | Describe product | | 4 | 0 | 0 | 4 | 4 | 0 |
| B | Develop marketing strategy | | 8 | 0 | 0 | 8 | 8 | 0 |
| C | Design brochure | A | 4 | 4 | 4 | 8 | 8 | 0 |
| D | Develop product prototype | | | | | | | |
| D1 | Requirements analysis | A | 8 | 4 | 4 | 12 | 12 | 0 |
| D2 | Software design | D1 | 16 | 12 | 12 | 28 | 28 | 0 |
| D3 | System design | D1 | 20 | 12 | 12 | 32 | 32 | 0 |
| D4 | Coding | D2, D3 | 24 | 32 | 32 | 56 | 56 | 0 |
| D5 | Write documentation | D4 | 16 | 56 | 56 | 72 | 72 | 0 |
| D6 | Unit testing | D4 | 20 | 56 | 56 | 76 | 76 | 0 |
| D7 | System testing | D6 | 24 | 76 | 76 | 100 | 100 | 0 |
| D8 | Package deliverables | D5, D7 | 12 | 100 | 100 | 112 | 112 | 0 |
| E | Survey potential market | B, C | 16 | 8 | 104 | 24 | 112 | 96 |
| F | Develop pricing plan | D8, E | 4 | 112 | 112 | 116 | 116 | 0 |
| G | Develop implementation plan | A, D8 | 12 | 112 | 112 | 124 | 124 | 0 |
| H | Write client proposal | F, G | 8 | 124 | 124 | 132 | 132 | 0 |

Duration (hours)

| 4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 | 40 | 44 | 48 | 52 | 56 | 60 | 64 | 68 | 72 | 76 | 80 | 84 | 88 | 92 | 96 | 100 | 104 | 108 | 112 | 116 | 120 | 124 | 128 | 132 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Critical Path = A - B - C - D1 - D2 - D3 - D4 - D5 - D6 - D7 - D8 - F - G - H

# Figure 4: Expected Case Gantt Chart

| taskID | task | predecessorTaskIDs | expected Hours | ES | LS | EF | LF | Slack |
|---|---|---|---|---|---|---|---|---|
| A | Describe product | | 5 | 0 | 0 | 5 | 5 | 0 |
| B | Develop marketing strategy | | 10 | 0 | 0 | 10 | 10 | 0 |
| C | Design brochure | A | 5 | 5 | 5 | 10 | 10 | 0 |
| D | Develop product prototype | | | | | | | |
| D1 | Requirements analysis | A | 10 | 5 | 5 | 15 | 15 | 0 |
| D2 | Software design | D1 | 20 | 15 | 15 | 35 | 35 | 0 |
| D3 | System design | D1 | 25 | 15 | 15 | 40 | 40 | 0 |
| D4 | Coding | D2, D3 | 30 | 40 | 40 | 70 | 70 | 0 |
| D5 | Write documentation | D4 | 20 | 70 | 70 | 90 | 90 | 0 |
| D6 | Unit testing | D4 | 25 | 70 | 70 | 95 | 95 | 0 |
| D7 | System testing | D6 | 30 | 95 | 95 | 125 | 125 | 0 |
| D8 | Package deliverables | D5, D7 | 15 | 125 | 125 | 140 | 140 | 0 |
| E | Survey potential market | B, C | 20 | 10 | 120 | 30 | 140 | 110 |
| F | Develop pricing plan | D8, E | 5 | 140 | 140 | 145 | 145 | 0 |
| G | Develop implementation plan | A, D8 | 15 | 140 | 140 | 155 | 155 | 0 |
| H | Write client proposal | F, G | 10 | 155 | 155 | 165 | 165 | 0 |

Duration (hours)

| 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 | 105 | 110 | 115 | 120 | 125 | 130 | 135 | 140 | 145 | 150 | 155 | 160 | 165 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Critical Path = A - B - C - D1 - D2 - D3 - D4 - D5 - D6 - D7 - D8 - F - G - H

# Figure 5: Worst Case Gantt Chart

| taskID | task | predecessorTaskIDs | worstCaseHours | ES | LS | EF | LF | Slack |
|---|---|---|---|---|---|---|---|---|
| A | Describe product | | 6 | 0 | 0 | 6 | 6 | 0 |
| B | Develop marketing strategy | | 12 | 0 | 0 | 12 | 12 | 0 |
| C | Design brochure | A | 6 | 6 | 6 | 12 | 12 | 0 |
| D | Develop product prototype | | | | | | | |
| D1 | Requirements analysis | A | 12 | 6 | 6 | 18 | 18 | 0 |
| D2 | Software design | D1 | 24 | 18 | 18 | 42 | 42 | 0 |
| D3 | System design | D1 | 30 | 18 | 18 | 48 | 48 | 0 |
| D4 | Coding | D2, D3 | 36 | 48 | 48 | 84 | 84 | 0 |
| D5 | Write documentation | D4 | 24 | 84 | 84 | 108 | 108 | 0 |
| D6 | Unit testing | D4 | 30 | 84 | 84 | 114 | 114 | 0 |
| D7 | System testing | D6 | 36 | 114 | 114 | 150 | 150 | 0 |
| D8 | Package deliverables | D5, D7 | 18 | 150 | 150 | 168 | 168 | 0 |
| E | Survey potential market | B, C | 24 | 12 | 144 | 36 | 168 | 132 |
| F | Develop pricing plan | D8, E | 6 | 168 | 168 | 174 | 174 | 0 |
| G | Develop implementation plan | A, D8 | 18 | 168 | 168 | 186 | 186 | 0 |
| H | Write client proposal | F, G | 12 | 186 | 186 | 198 | 198 | 0 |

Duration (hours)

| 6 | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60 | 66 | 72 | 78 | 84 | 90 | 96 | 102 | 108 | 114 | 120 | 126 | 132 | 138 | 144 | 150 | 156 | 162 | 168 | 174 | 180 | 186 | 192 | 198 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

Critical Path = A - B - C - D1 - D2 - D3 - D4 - D5 - D6 - D7 - D8 - F - G - H

In [1]:
```python
import pulp

# Define tasks
tasks = {
    "A": [],
    "B": [],
    "C": ["A"],
    "D1": ["A"],
    "D2": ["D1"],
    "D3": ["D1"],
    "D4": ["D2", "D3"],
    "D5": ["D4"],
    "D6": ["D4"],
    "D7": ["D6"],
    "D8": ["D5", "D7"],
    "E": ["B", "C"],
    "F": ["D8", "E"],
    "G": ["A", "D8"],
    "H": ["F", "G"]
}

# Task durations for best case, expected, and worst case in hours
task_durations = {
    "A": {"best": 4, "expected": 5, "worst": 6},
    "B": {"best": 8, "expected": 10, "worst": 12},
    "C": {"best": 4, "expected": 5, "worst": 6},
    "D1": {"best": 8, "expected": 10, "worst": 12},
    "D2": {"best": 16, "expected": 20, "worst": 24},
    "D3": {"best": 20, "expected": 25, "worst": 30},
    "D4": {"best": 24, "expected": 30, "worst": 36},
    "D5": {"best": 16, "expected": 20, "worst": 24},
    "D6": {"best": 20, "expected": 25, "worst": 30},
    "D7": {"best": 24, "expected": 30, "worst": 36},
    "D8": {"best": 12, "expected": 15, "worst": 18},
    "E": {"best": 16, "expected": 20, "worst": 24},
    "F": {"best": 4, "expected": 5, "worst": 6},
    "G": {"best": 12, "expected": 15, "worst": 18},
    "H": {"best": 8, "expected": 10, "worst": 12}
}
```

In [2]:
```python
hourly_rate = 55  # Hourly rate of employees
```

In [3]:
```python
# Create LP problem
prob = pulp.LpProblem("Scheduling", pulp.LpMinimize)
```

In [4]:
```python
# Define decision variables
start_time = {task: {scenario: pulp.LpVariable(f"Start_Time_{task}_{scenario}", lowBou
                     for scenario in ["best", "expected", "worst"]} for task in tasks}
end_time = {task: {scenario: pulp.LpVariable(f"End_Time_{task}_{scenario}", lowBound=0
                   for scenario in ["best", "expected", "worst"]} for task in tasks}
```

In [5]:
```python
# Define objective function
prob += pulp.lpSum((end_time["H"][scenario] - start_time["H"][scenario]) * hourly_rate
```

In [6]:
```python
# Add constraints for each scenario
for task in tasks:
```

```
    for pred_task in tasks[task]:
        for scenario in ["best", "expected", "worst"]:
            prob += end_time[pred_task][scenario] <= start_time[task][scenario]
    for scenario in ["best", "expected", "worst"]:
        prob += end_time[task][scenario] - start_time[task][scenario] == task_duratior
```

In [10]:

```
# Solve
prob.solve()

# Print results
for scenario in ["best", "expected", "worst"]:
    print(f"{scenario} case:")
    total_cost = 0
    for task in tasks:
        duration = task_durations[task][scenario]
        cost = duration * hourly_rate
        total_cost += cost
        print(f"{task}: Start Time = {start_time[task][scenario].varValue}, End Time =

    print(f"{scenario} case cost:", total_cost)
```

```
best case:
A: Start Time = 0.0, End Time = 4.0, Cost = 220
B: Start Time = 0.0, End Time = 8.0, Cost = 440
C: Start Time = 4.0, End Time = 8.0, Cost = 220
D1: Start Time = 4.0, End Time = 12.0, Cost = 440
D2: Start Time = 12.0, End Time = 28.0, Cost = 880
D3: Start Time = 12.0, End Time = 32.0, Cost = 1100
D4: Start Time = 32.0, End Time = 56.0, Cost = 1320
D5: Start Time = 56.0, End Time = 72.0, Cost = 880
D6: Start Time = 56.0, End Time = 76.0, Cost = 1100
D7: Start Time = 76.0, End Time = 100.0, Cost = 1320
D8: Start Time = 100.0, End Time = 112.0, Cost = 660
E: Start Time = 8.0, End Time = 24.0, Cost = 880
F: Start Time = 112.0, End Time = 116.0, Cost = 220
G: Start Time = 112.0, End Time = 124.0, Cost = 660
H: Start Time = 124.0, End Time = 132.0, Cost = 440
best case cost: 10780
expected case:
A: Start Time = 0.0, End Time = 5.0, Cost = 275
B: Start Time = 0.0, End Time = 10.0, Cost = 550
C: Start Time = 5.0, End Time = 10.0, Cost = 275
D1: Start Time = 5.0, End Time = 15.0, Cost = 550
D2: Start Time = 15.0, End Time = 35.0, Cost = 1100
D3: Start Time = 15.0, End Time = 40.0, Cost = 1375
D4: Start Time = 40.0, End Time = 70.0, Cost = 1650
D5: Start Time = 70.0, End Time = 90.0, Cost = 1100
D6: Start Time = 70.0, End Time = 95.0, Cost = 1375
D7: Start Time = 95.0, End Time = 125.0, Cost = 1650
D8: Start Time = 125.0, End Time = 140.0, Cost = 825
E: Start Time = 10.0, End Time = 30.0, Cost = 1100
F: Start Time = 140.0, End Time = 145.0, Cost = 275
G: Start Time = 140.0, End Time = 155.0, Cost = 825
H: Start Time = 155.0, End Time = 165.0, Cost = 550
expected case cost: 13475
worst case:
A: Start Time = 0.0, End Time = 6.0, Cost = 330
B: Start Time = 0.0, End Time = 12.0, Cost = 660
C: Start Time = 6.0, End Time = 12.0, Cost = 330
D1: Start Time = 6.0, End Time = 18.0, Cost = 660
D2: Start Time = 18.0, End Time = 42.0, Cost = 1320
D3: Start Time = 18.0, End Time = 48.0, Cost = 1650
D4: Start Time = 48.0, End Time = 84.0, Cost = 1980
D5: Start Time = 84.0, End Time = 108.0, Cost = 1320
D6: Start Time = 84.0, End Time = 114.0, Cost = 1650
D7: Start Time = 114.0, End Time = 150.0, Cost = 1980
D8: Start Time = 150.0, End Time = 168.0, Cost = 990
E: Start Time = 12.0, End Time = 36.0, Cost = 1320
F: Start Time = 168.0, End Time = 174.0, Cost = 330
G: Start Time = 168.0, End Time = 186.0, Cost = 990
H: Start Time = 186.0, End Time = 198.0, Cost = 660
worst case cost: 16170
```