***Train Simulation***

*SPMP*

Version No. 4.1.0

Table of Contents

[**1. Introduction**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.rd3qzjiv3or6) **4**

[**2.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.r303qcwlk38l) **Project Estimates** [4](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.a6yt6ldexo5)

[**3.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.eixv12g19xcn) **Risk Management 6**

[**4.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.z05d7tfa2w70) **Project Schedule 6**

[**5.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.op56jdm1fs7g) **Staff Organization 6**

[**6.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.5gk5jlyk1lzh) **Tracking and Control Mechanisms 7**

[**7.**](https://docs.google.com/document/d/1UgbAWyP9FZoIJBPv1KOcs0rtZ8ZHQNtmYYlqaMknMn4/edit#heading=h.gytwouahysm3) **Appendix**

1. **Introduction**
   1. **Problem Statement**

Design an application to optimize a railroad business’s efficiency. The application should be able to use real time simulations of railroad activities to collect data and provide insight on potential problems. Potential solutions will be given and simulated to customers in an attempt to reduce railway down time.

* 1. **Project Scope**
     1. **Inclusions**
* Inputs:

User Input

* Weather ~~Condition~~ Type
* Weather Severity
* Add / Remove Trains
* Add / Remove Hubs
* Add / Remove Stations
* Cargo Cost
* Fuel
* Day the user it will view
* Duration for a breakpoint
* Input files / directory

File Input

* Hubs
* Stations
* Tracks
* Train Types
* Schedule Routes
* Daily Routes
* Configuration
* Distance of each track
* Maintenance
  + - * Processing Functionality:
* Graph Creation
* Design Simulation
* Recoding Data
* Recording Statistics
* Optimize simulation
* Comparing old data to new data with each other
* Comparing end of days to each other
  + - * Outputs:
* User interface
* Edit Configuration
* ~~Pseudo real-time simulation~~
* Graph
* Files
* Errors and exception handling
* Metrics from simulation
* Comparison of picked days
  + 1. **Exclusions**
       - N/A
  1. **Major Software Functions**
* Adding trains, stations, railway, and hubs from a file that the user will provide
* Creating a logical graph extracting from file provided by user
* Apply train schedule and daily routes from a file provided by the user
* A Maintenance schedule that will be extracted from a file that the user will provide
* Configuration information that the user will provide
* Passenger train capacity
* ~~Create pseudo real-time simulation~~
* Collect Data from the simulation
* Automate suggestion of potential changes based on collected data
* Open data to external .csv file for data modeling
* Weather delay system that would be controlled by user to test various weather patterns
* Track cost of trains through distance and weight
* Train priority for the category of the trains
* Record traffic through individuals that would be shown in data
* When a ~~the~~ train has ~~a~~ less than 10% gas left ~~amount of 10% of gas~~ the train. It will reroute itself to the nearest hub to refuel and then is allowed to go back to its old route ~~remains train cannot leave station until it is refueled at a hub~~
* Display new data to user
* Edit Railway
* Add/remove trains from simulation according to Configuration Settings by User
  1. **Performance / Behavior constraints**
* N/A
  1. **Management and Technical Constraints**
     1. **Management Constraints**
        + Work
        + Transportation
        + Weather
        + Spring Recess: March 2-10th
        + Class Exams:
* Exam 1: Feb 27th
* Exam 2: April 17th
* Exam: April 24th
  + - * Due Dates:
* SPMP: Feb 25th
* Final Delivery: April 26th
* Use Case Summary: March 11th
* Use Case Document: March 18th
* Software Analysis Specification April 8th
  + - * General Availability:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Sun** | **Mon** | **Tues** | **Wed** | **Thur** | **Fri** | **Sat** |
| 8:00 AM |  |  |  |  |  |  |  |
| 9:00 AM |  |  |  |  |  |  |  |
| 10:00 AM |  |  |  |  |  |  |  |
| 11:00 AM |  |  |  |  |  |  |  |
| 12:00 PM |  |  |  |  |  |  |  |
| 1:00 PM |  |  |  |  |  |  |  |
| 2:00 PM |  |  |  |  |  |  |  |
| 3:00 PM |  |  |  |  |  |  |  |
| 4:00 PM |  |  |  |  |  |  |  |
| 5:00 PM |  |  |  |  |  |  |  |
| 6:00 PM |  |  |  |  |  |  |  |
| 7:00 PM |  |  |  |  |  |  |  |
| 8:00 PM |  |  |  |  |  |  |  |
| 9:00 PM |  |  |  |  |  |  |  |

* + 1. **Technical Constraints**
       - Limited Programming Languages
       - GitHub
       - Files

1. **Project Estimates**
   1. **Historical Data used for Estimates**

* School Projects
* Work Projects
* Vacation Time
* Weather Conditions
* School Scheduling
* Work Scheduling
  1. **Initial Estimate**
     1. **SPMP Completion Estimation**
        + 140 Hours
     2. **Overall Project Estimation**

**2.2.2.1. Line-of Code Estimate**

* 2000 Lines of Code

**2.2.2.2. Function Estimate**

* 150 Hours

**2.2.2.3. Task Estimate**

* 150 Hours

**2.2.2.4. Total overall project time estimate in hours of effort**

* 600 Hours
  1. **Estimation techniques applied and results**
     1. **Estimation technique 1 - lines of code**
        + Graph Creation = 500
        + Applying train schedule = 150
        + Simulation Design = 150
        + Run simulation and collect data = 50
        + Generate automated suggestions = 200
        + Visually display track data and usage = 100
        + Output data to .csv file for data modeling = 100
        + RNG based weather system = 250
        + Tracking running time and cost = 200
        + Multiple trains per track = 150
        + Assigning train priority = 150
        + Recognize gas in train and prevent trains with 10% of gas remaining from leaving station = 200
        + User interface to show comparison of current and previous days = 300
        + Track of fuel for train = 50
        + Edit track lines between nodes = 200
        + Add/remove trains from simulation = 75
        + Previous Estimate = 2310 Lines-of-Code
        + New Estimate = 2585 Lines-of-Code
     2. **Estimation for technique 1 - lines of code**
        + Revised Estimation = 300 Hours
     3. **Estimation technique 2 - function points**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Complexity** | | |
| **Component** | **Low** | **Average** | **High** |
| **EI** | **3** | **4** | **6** |
| **EO** | **4** | **5** | **7** |
| **WQ** | **3** | **5** | **6** |
| **EIF** | **5** | **7** | **10** |
| **ILF** | **7** | **10** | **15** |

* + - * 4 \* 3 EI
      * 5 \* 5 EO
      * 4 \* 4 EQ
      * 10 \* 1 EIF
      * 7 \* 1 ILF
      * 38hr \* 8 = **304 Hours total**

**Revised Function Point Estimate**

* + - * 70 \* 8 = **560 Hours total**
    1. **Estimate of technique 2 - function points**
       - 560 Hours
    2. **Estimation technique 3 - process / task**
       - Graph Creation = 30 hours
       - Applying train schedule = 9 hours
       - Simulation Design = 15 hours
       - Run simulation and collect data = 6 hours
       - Generate automated suggestions = 30 hours
       - Visually display track data and usage = 12 hours
       - Output data to .csv file for data modeling = 10 hours
       - RNG based weather system = 6 hours
       - Tracking running time and cost = 30 hours
       - Multiple trains per track = 20 hours
       - Assigning Train Priority = 6 hours
       - Recognize gas in train and prevent trains with 10% of gas remaining from leaving station = 20 hours
       - User interface to show comparison of current and previous days = 10 hours
       - Edit track lines between nodes = 10 hours
       - Add/remove trains from simulation = 5 hours
    3. **Estimate for technique 3 - process / task**
       - 188 hours

**Revised Process/Task Estimate**

* + - * 219 hours
  1. **Reconciled Estimate**
* Our final decision ended up being 300 hours total. We chose to make our final estimate close to our function point estimate because it was the largest value. We are worried about time constraints due to other external factors. Therefore we chose a more pessimistic value in order to ensure we get the job done
* Our revised final estimate increased from our original estimate of 300 hours, to 500 hours. As we spent more time discussing the project and refining our specifications, we realized that our initial estimates were on the optimistic side. We chose to increase our original estimate by 200 hours because we added extra functionalities to our program and revisited our previous requirements with better understanding.
  1. **Project Resources**
* GitHub
* Visual Studio
* NetBeans
* Discord
* Skype
* Google Docs
* Whats App
* Canvas
* Scene Builder
* Draw.io
* Problem Step Recorder

1. **Risk Management**
   1. **Project Risk Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risk | Probability | Impact | Mitigation Plan | Contingency Plan |
| Poor Estimation on Additional Functionalities | Medium | Low | Get regular updates and design main functionality to not rely on any additional functionalities | Drop support and expectations for respectable additional functionalities |
| Poor Estimation on Main Functionality | Medium | High | Get regular updates from team members to gauge progress, re-evaluate estimations regularly | Drop additional functionalities and focus on main |
| Project Specification changes | High | Medium | Focus on designing the project instead of programming it, design components modularly so if something changes the whole project doesn’t have to be redone | Change components that are affected by specification change |
| Underestimation of program complexity | Medium | Medium | Keep program functionality simple, have algorithms planned before creation of code | Reduce or remove functionality causing problems |
| Absence of Team Leader | Low | Medium | Finish Planning as early as possible, stay in contact | Assign new team leader and work off of what we have |
| Absence of Other Team Members | Low | Medium | Communicate regularly | Drop additional functionality of respective teammate |
| Inclement weather | Medium | Low | Adhere to schedule | Attempt digital meeting |
| Appearance of new technologies needed for functionalities | Medium | Medium | Attempt to learn early | Redesign plan to omit new technologies |

* 1. **Overview of Risk Management, Monitoring, Management**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Impact | | | | |
| Probability |  | Low | Medium | High | Critical |
| Low |  |  |  |  |
| Medium |  |  |  |  |
| High |  |  |  |  |
| Legend | | | | | |
| Monitor | | Execute Mitigation Plan | | Execute Contingency Plan | |

* Re-evaluate risks once every one-two weeks

1. **Project Schedule**
   1. **Project task set**

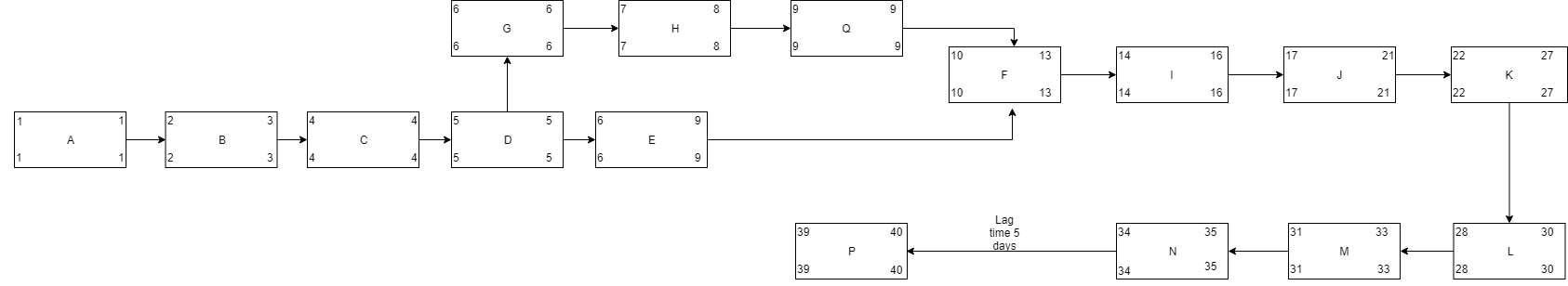
|  |  |
| --- | --- |
| Tasks | Duration (estimate days) |
| A | ~~1~~ 3 |
| B | 2 |
| C | ~~1~~ 4 |
| D | ~~1~~ 2 |
| E | 3 |
| F | 3 |
| G | ~~1~~ 3 |
| H | ~~2~~ 5 |
| I | 2 |
| J | 4 |
| K | ~~5~~ 6 |
| L | ~~3~~ 2 |
| M | 3 |
| N | ~~2~~ 3 |
| O | ~~3~~ 1 |
| P | ~~2~~ 6 |
| Q | 10 |

1. ~~Data extraction from files~~  Validating Requirements
2. ~~Creating a graph from the data~~  Decomposing Requirements
3. ~~Creating freight and passenger trains~~ Test strategy
4. ~~Train Priority \*~~ Modeling Requirements
5. ~~Shortest path for trains~~ UI Design
6. ~~Simulate the data~~ Graph Design
7. ~~Passenger check for capacity~~ Simulation Design
8. ~~Prevent collision with any trains~~ Test Plan
9. ~~Give option to enable weather simulation \*~~  File Validation Coding
10. ~~Collect metrics through simulation~~ UI Coding
    1. ~~Individual track usage counters~~
    2. ~~Train uptime and downtime~~
    3. ~~Collective uptime and downtime~~
    4. ~~Track train running costs \*~~
11. ~~Generate automated suggestions to decrease downtime \*~~ Run Simulation
12. Enable modifications to rail system
    1. Adding and removing track
    2. Adding and Remove hubs
    3. Adding and Remove stations
    4. Adding and Remove freight or passenger train
13. Update Graph ~~from new data~~
14. ~~Create visual representation of tracks used \*~~ Creating routes for the trains
15. Output metrics recorded to .csv file for data modeling \*
16. ~~Track fueling for cargo trains~~ Track Statistics
17. Testing
    1. **Task Network**

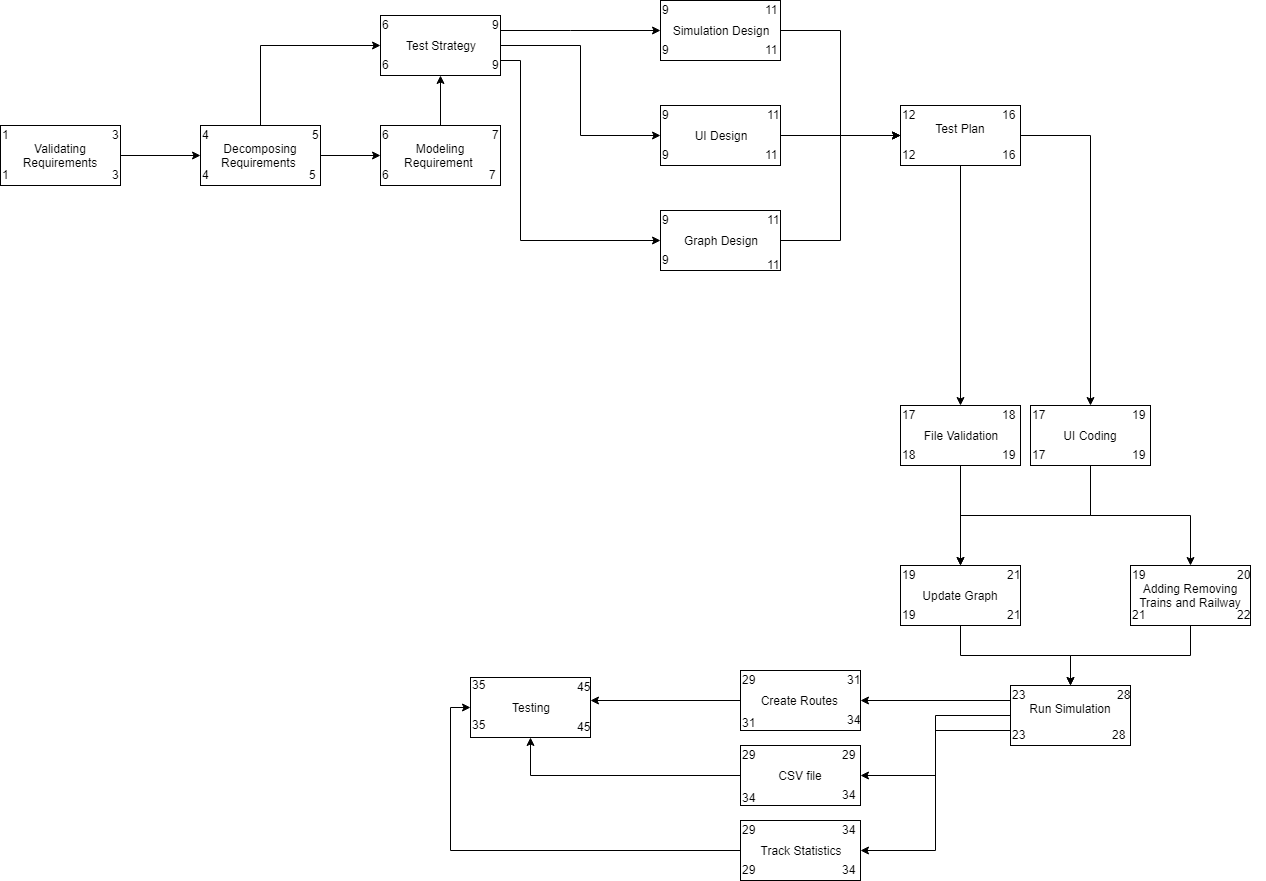
|  |  |  |  |
| --- | --- | --- | --- |
| Tasks | Duration (estimate days) | Predecessor Task | Successor Task |
| A | ~~1~~ 3 | - | B |
| B | 2 | A | C,D |
| C | ~~1~~ 4 | B, D | ~~D~~ E,F,G |
| D | ~~1~~ 2 | B | ~~E,G~~ C |
| E | 3 | ~~D~~ C | ~~F~~ H |
| F | 3 | ~~E,H~~ C | ~~I~~ H |
| G | ~~1~~ 3 | ~~D~~ | H |
| H | ~~2~~ 5 | G,E,F | I,J ~~Q~~ |
| I | 2 | ~~F~~ H | ~~J~~ M |
| J | 4 | ~~I~~ H | ~~K~~ L,M |
| K | ~~5~~ 6 | ~~J~~ L,M | L |
| L | ~~3~~ 2 | ~~K~~ J | ~~M~~ K |
| M | 3 | ~~L~~ I | ~~N~~ K |
| N | ~~2~~ 3 | M | ~~O~~ Q |
| O | ~~3~~ 1 | M | Q |
| P | ~~2~~ 6 | ~~P~~ M | * Q |
| Q | 10 | P, O, N | ~~F~~ - |

* 1. **Timeline Chart**

**OLD**



**NEW**



1. **Staff organization**
   1. **Team Structure**

* Zaid Alsafi
  + Check on team members
  + Create and lead meetings
  + Distribute work
  + Weekly Reports
  + Revising Doucentaion
* Alex Navarre
  + Meeting notes
  + Revising documentation
* Maxx Achtman
  + Meeting notes
  + Lead QA
  + Testing
* Orlando Murrell
  + I.T.
  + Testing
* Abbass Srour
  + Coding
* Brendan Batchelor
  + I.T.
  + Revising Documentation
* Radhika Moradiya
  + Approval and Denial Right
  + Overseer
  1. **Management reporting and communication**
* Discord
* Google Drive
* Whats App
* Skype
* Canvas
* In person
* GitHub

1. **Tracking and Control Mechanisms**
   1. **Quality Assurance and control**

* All classes will have test stub evaluations before implementation
* Each group member will carry out testing on both their own and others’ work
* A designated group member will conduct final testing
  1. **Change Management and Control**
* Project artifacts will be tracked and maintained using a github repository
* Proposed changes should be submitted with a brief but descriptive summary of what was changed, i.e. added, removed, or modified and explain why
* Changes to artifacts will be announced through the discord server and the group may vote to revert changes made
  1. **Tools Used**
* GitHub
* Discord
* Visual Studio
* Google Drive
* Draw.Io
* Problem Step Recorder

1. **Appendix**
   1. **ER Diagram**

