*Software Analysis Specification*

Table of Contents

[**1.0 Introduction**](#_topto9mj4df7) **4**

[1.1 Goals and objectives](#_ioi5dqygmxyp) 4

[1.2 Statement of scope](#_tmnruflgauw0) 4

[1.3 Software context](#_byaqa3g1kuhn) 4

[1.4 Major constraints](#_dhetyt1ydrei) 4

[**2.0 Usage Scenario**](#_bgsnom2ph36p) **5**

[2.1 User Profiles](#_xi5s9euc15ry) 5

[2.2 Major Software Functionality](#_p0i1559fgskc) 5

[2.3 Special Usage Considerations](#_pnldhac2vrye) 5

[**3.0 Data Model and Description**](#_hhgas2st9fm4) **6**

[3.1 Data Description](#_6efhzc60edp9) 6

[3.1.1 Entity-Relationship Diagram](#_2t3sf99n0tct) 6

[3.1.2 Data Flow Diagram](#_5qbhvqx8o39c) 6

[3.1.3 Object Relationships](#_jpceti5uuags) 8

[3.1.4 Complete Data Model](#_53sldzes6cf) 8

[3.1.5 Data Dictionary](#_1b2ym9tvm2ou) 8

[**4.0 Functional Model and Description**](#_wcm9iub7hmpq) **9**

[4.1 Use Cases](#_vitiqkcff9jo) 9

[4.2 Software Interface Description](#_3fxjr7rkjduu) 9

[4.2.1 External Machine Interfaces](#_au99hz1ci5r0) 9

[4.2.2 External System Interfaces](#_gu9yjmcjubwv) 9

[4.2.3 Human Interface](#_c1qv5ptuvfmv) 9

[4.2.3.1 User Screen Interface Layouts](#_wan81cs0ob4b) 9

[4.2.3.2 Report Layouts](#_shysv3m3trog) 9

[4.3 Sequence Diagrams](#_o1lrnp2qw0ds) 10

[4.4 Communication Diagrams](#_gm1gvecgrach) 11

[**5.0 Behavioral Model and Description**](#_tse45re7w8ik) **12**

[5.1 Description for Software Behaviour](#_zdknwrrkzzyr) 12

[5.1.1 Events](#_ygv4cwlb359q) 12

[5.1.2 States](#_b0af4p1pz2kr) 13

[5.2 State Transition Diagrams](#_fvztnzhfuyd3) 15

[**6.0 Restrictions, Limitations, and Constraints**](#_4pgmdfbxxjwd) **15**

[**7.0 Validation Criteria**](#_43bjvee5rpv1) **16**

[7.1 Classes of tests/Test Strategy](#_y5ol3kkw7v91) 16

[7.2 Expected Software Response](#_sgg5sst18vtn) 16

[7.3 Performance Bounds](#_drvfc5hsdtir) 16

[**8.0 Appendices**](#_n1ykqit3taes) **17**

[8.1 System Traceability Matrix](#_rgkxah4uhd6a) 17

[8.2 Product Strategies](#_xlwtuwv9bw50) 17

[8.3 Analysis Metrics to be used](#_ff1ymbg4fn98) 18

[8.4 Supplementary Information](#_ocdqfnogfb2b) 18

# **1.0 Introduction**

## 1.1 Goals and objectives

Reference to Version ~~2~~ 3 of the SPMP in 1.1 Problem statement

## 1.2 Statement of scope

Reference to Version ~~2~~ 3 of the SPMP in 1.2 Project Scope

## 1.3 Software context

Reference to Version ~~2~~ 3 of the SPMP in 1.3 Major Software Functions

## 1.4 Major constraints

Reference to Version ~~2~~ 3 of the SPMP in 1.4 Major Software Functions and Version 2 of the SPMP in 1.5

# 

# 2.0 Usage Scenario

## 2.1 User Profiles

|  |  |
| --- | --- |
| **User** | **Profile** |
| General | This user will not care about technicalities or money. This user will run the simulation without any additional parameters. |
| Business-Oriented | This user will look at the cost of the simulation and how to optimize it. This user will modify the simulation as the user would see fit to optimize the railway system with the least cost through various means such as adjusting railways, stations, hubs, and train amounts. |
| Technical-Oriented | This user will look at the functionality of the simulation through different scenarios to simulate real word conditions. This user will look for maximal efficiency in time of the railway system |

## 2.2 Major Software Functionality

Reference to Version ~~2~~ 3 of the SPMP in 1.3 Major Software Functions

## 2.3 Special Usage Considerations

N/A

# 

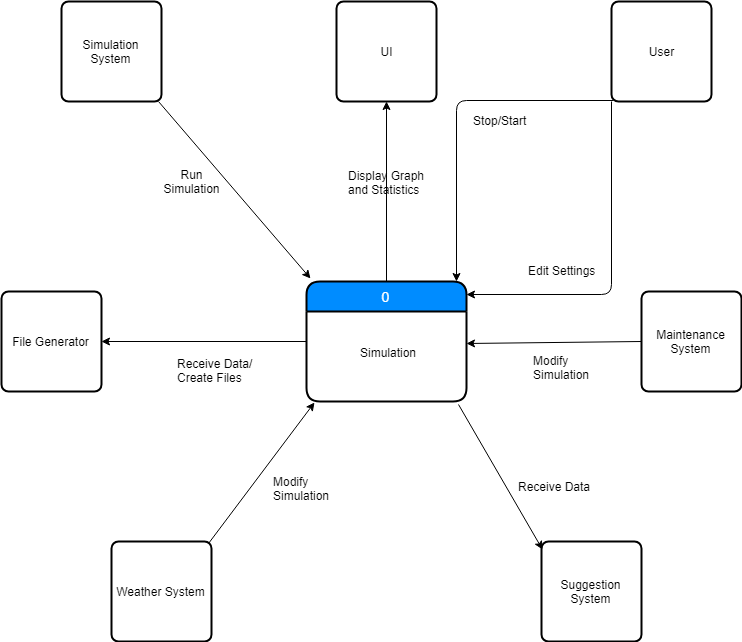
# 3.0 Data Model and Description

## 3.1 Data Description

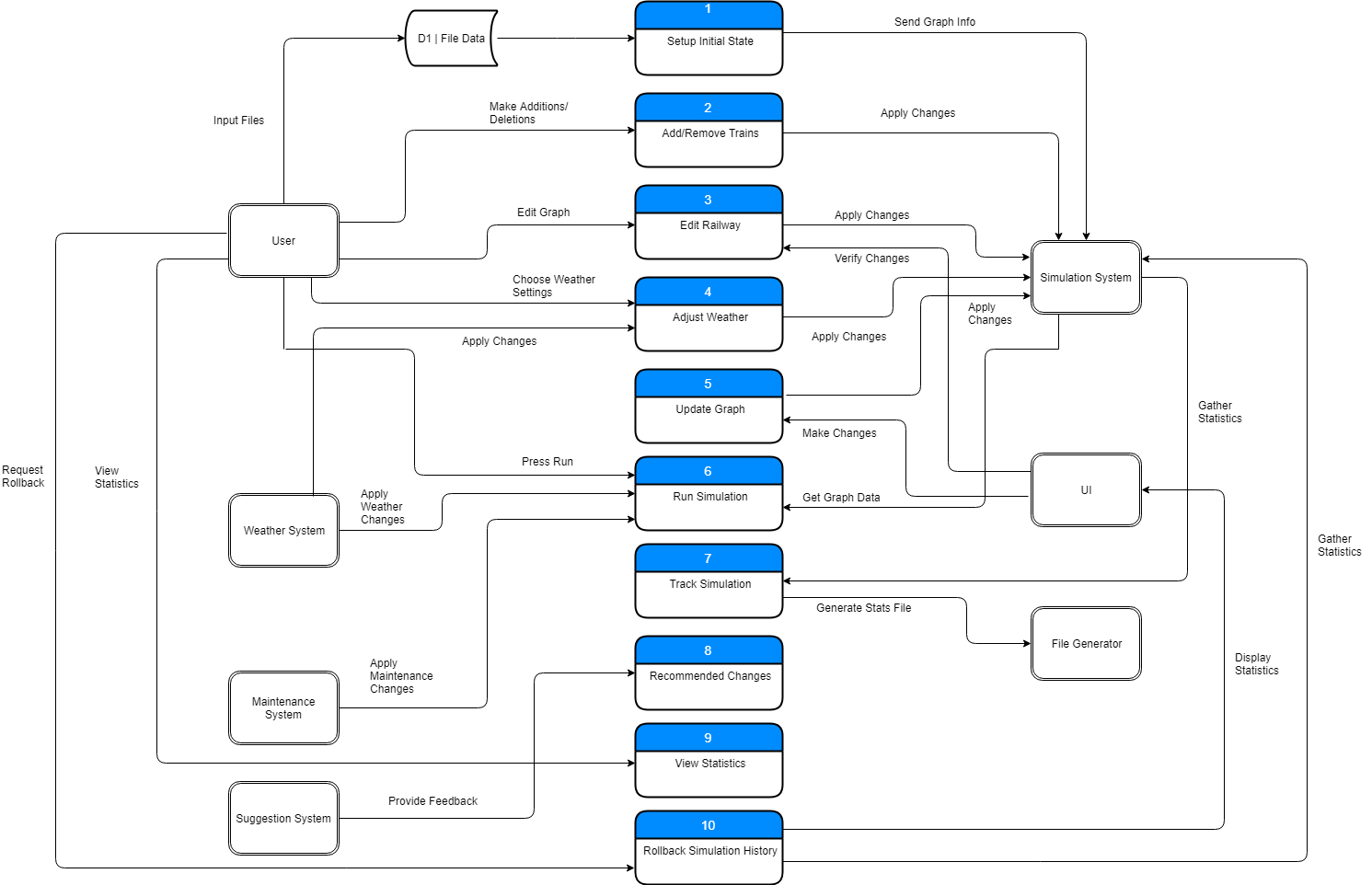
### 3.1.1 Entity-Relationship Diagram

Reference to Version ~~2~~ 3 of the SPMP in 7.1 in Appendix

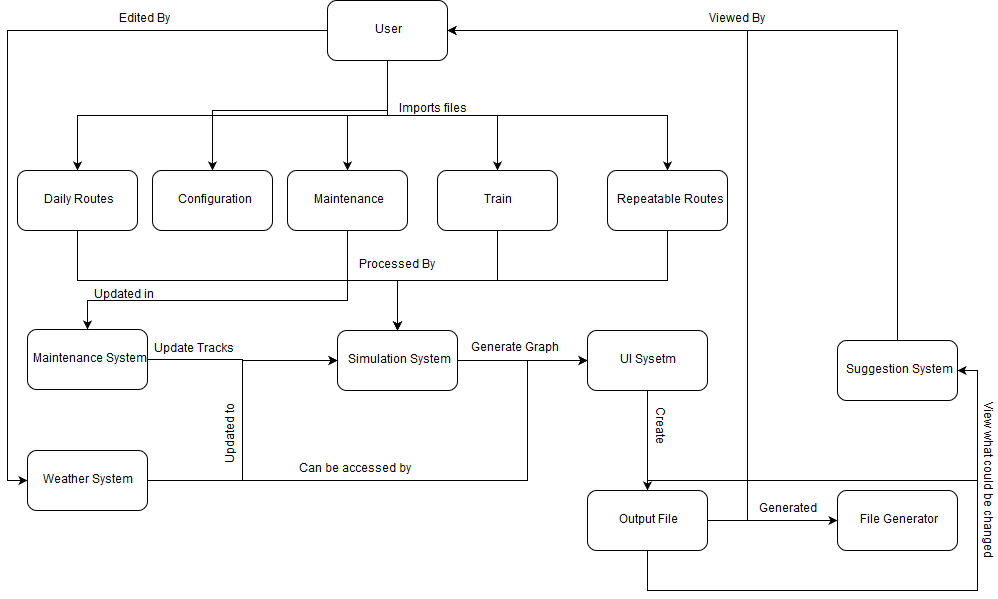
### 3.1.2 Data Flow Diagram

Context Diagram

Level 0 DFD



### 3.1.3 Object Relationships



### 3.1.4 Complete Data Model

### 3.1.5 Data Dictionary

Located [here on drive](https://docs.google.com/spreadsheets/d/1xX559lwj2DZIrDsf61azYBA9EZjOkoVIyPRsT8cU8qc/edit#gid=1842544997).

# 

# 4.0 Functional Model and Description

## 4.1 Use Cases

* See Train Simulator
  + Recommended Changes
  + ~~Edit track lines between nodes~~
  + Edit Railways
  + Track Statistics
  + Run Simulation
  + Add / remove trains ~~from the simulation~~
  + Setup the initial state ~~or update railroad~~
  + Adjust weather options
  + Rollback Simulation History
  + Update Graph
  + View statistics ~~about the simulation~~
  + ~~Train and Track Maintenance~~

## 4.2 Software Interface Description

### 4.2.1 External Machine Interfaces

* N/A

### 4.2.2 External System Interfaces

* Data modeling program such as Excel

### 4.2.3 Human Interface

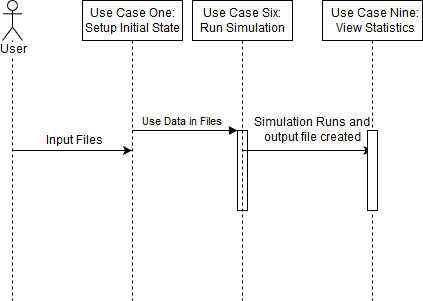
#### 4.2.3.1 User Screen Interface Layouts

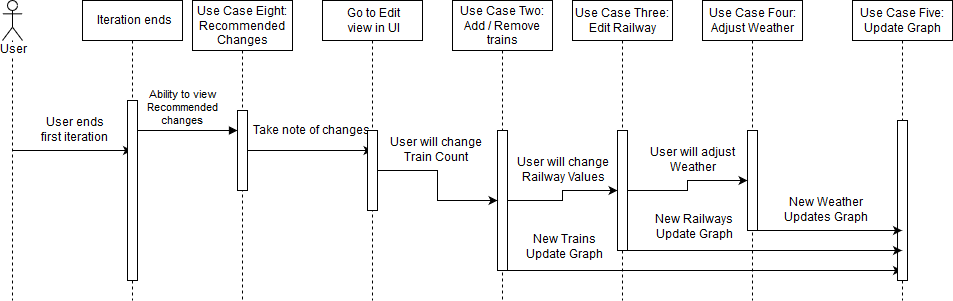
* Exception screen for wrong files uploaded
* Recommend changes view
* Settings options to edit railway information, edit weather settings
* Allow user to set number of days to run iteration

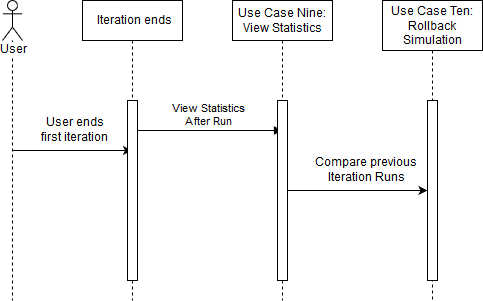
#### 4.2.3.2 Report Layouts

* .txt file for output data
* .csv file for exporting data for data modeling

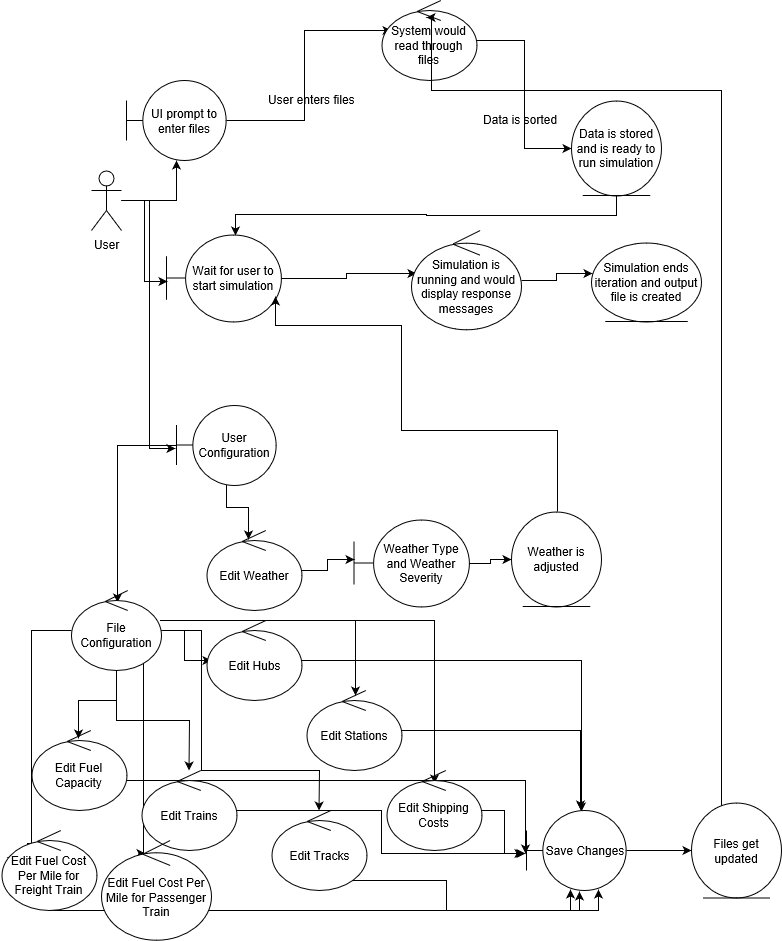
## 4.3 Sequence Diagrams







## 4.4 Communication Diagrams



# 5.0 Behavioral Model and Description

## 5.1 Description for Software Behaviour

### 5.1.1 Events

* User Exits: The User requests the application to close.
* Get File Inputs: The User requests the UI
* Start Simulation: Starts the Simulation. Before the Library call is made, must pass the Verification Stage.
* Pass Verification: Happens if the files are in the correct format, and if the configuration parameters are within bounds.
* Fail Verification: Happens if the inverse of *Passed Input Verification* is true.
* Outputs Simulation Results: Writes the simulation results to a file/directory. After this returns, the UI scans for the returned file/directory and formats its data to be visualized.
* User requests to Add/Remove Trains: Brings up an interface for modifying the existing trains in the network.
* User requests to Modify Railway: Brings up an interface for modifying the connections on the railway.
* Applying Changes: Applies any changes made to the railway.
* User Cancels Changes: returns to idle without applying any changes to the railway or configuration
* User requests to change config: Brings up an interface for modifying the configuration.
* Cancel: Closes any interface and returns to the current idle state.
* Apply: Closes any interface, and transfers to another state.
* Finishes: Used to signal that whatever state is finishes with its process and should return to the current idle state.
* User acknowledges error: Signals that the error has been acknowledged by the user, and returns to the current idle state.
* User Requests to view Recommended Changes: Brings up an interface showing recommended changes.
* User Requests CSV: Signals to start processing and formatting statistics into a Comma-Separated Value (CSV) File.
* User resets simulation: Clears the baseline and returns to Idle #1 State.

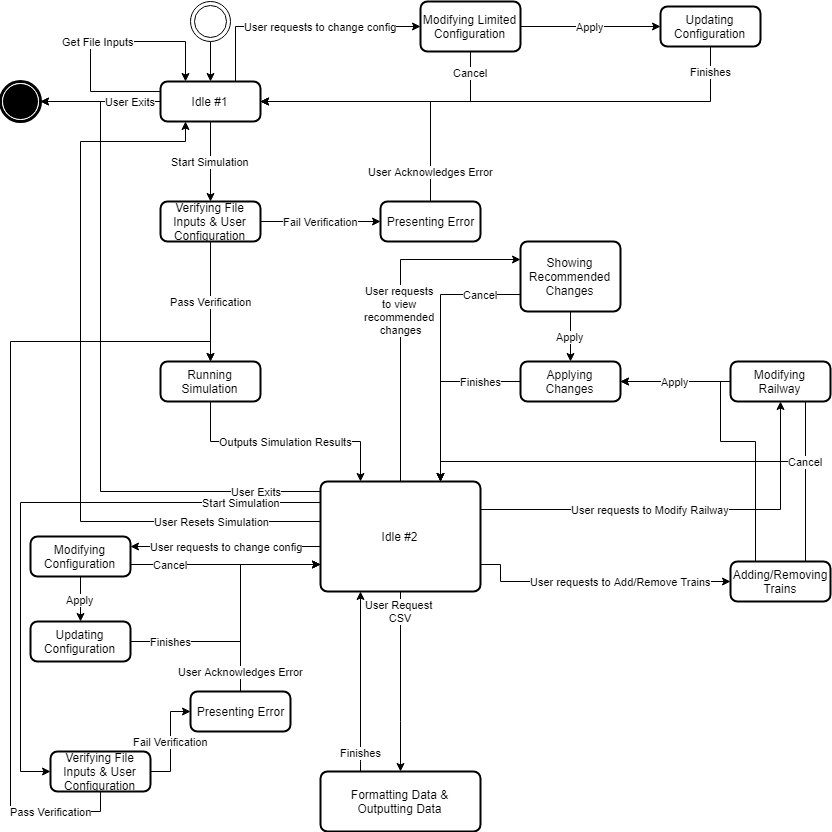
### 5.1.2 States

* Idle #1: The application’s first idle state. A lot of options are restricted until the first run of the simulation occurs so a baseline can be acquired.
* Verifying File Inputs & User Configuration: The Application verifies that the given files are valid for the simulation to run, and configuration is within bounds.
* Modifying Limited Configuration: The User is allowed to change configuration options including:
  + Duration: (Positive Integer (1 to 7))

This is a subset of the Modifying Configuration State, and a lot of options aren’t available.

* Presenting Error: The Application is informing the User that there is an error with the provided inputs and/or configuration.
* Running Simulation: The Simulation itself is running. The application may not be terminated until the simulation is complete.
* Idle #2: The Application’s main idle state. The user has access to modifying all options available. A graph of the last run of the simulation is also available.
* Adding/Removing Trains: The Application allows for the user to edit the current trains included in the simulation. The User can cancel or submit any of their changes.
* Modifying Railway: The Application allows for the user to change the connections between stations and hubs. The User can cancel or submit any of their changes.
* Applying Changes: Updates the railway system with the modified information.
* Showing Recommended Changes: The Application presents a brief summary of changes that should be made, and gives the user the option of applying them or not.
* Modifying Configuration: The User is allowed to change configuration options including:
  + Weather Severity (Range: 0-n)
  + Weather Type
  + Collision avoidance: (Toggle: On/Off)
  + How many iterations: (Positive Integer)
  + Duration: (Positive Integer (1 to 7))
  + Cargo transport cost per lb: (Positive Integer)
* Updating Configuration: Applies Changes to the configuration
* Formatting Data & Outputting data: Formats the data for external data modeling (.csv) and is writing files.

### 5.2 State Transition Diagrams



# 6.0 Restrictions, Limitations, and Constraints

N/A

# 7.0 Validation Criteria

## 7.1 Classes of tests/Test Strategy

* Valid Data
* Invalid Data
* File Handling
* UI Functionality
* Exporting data
* Output Display

## 7.2 Expected Software Response

* Program would identify valid data
* When invalid data is entered, error responses would occur
* Program would recognize file inputs and know how to organize in program
* User selections on buttons would return expected behavior
* Program would handle export of ~~specific~~ data amounts as defined in code for data modeling

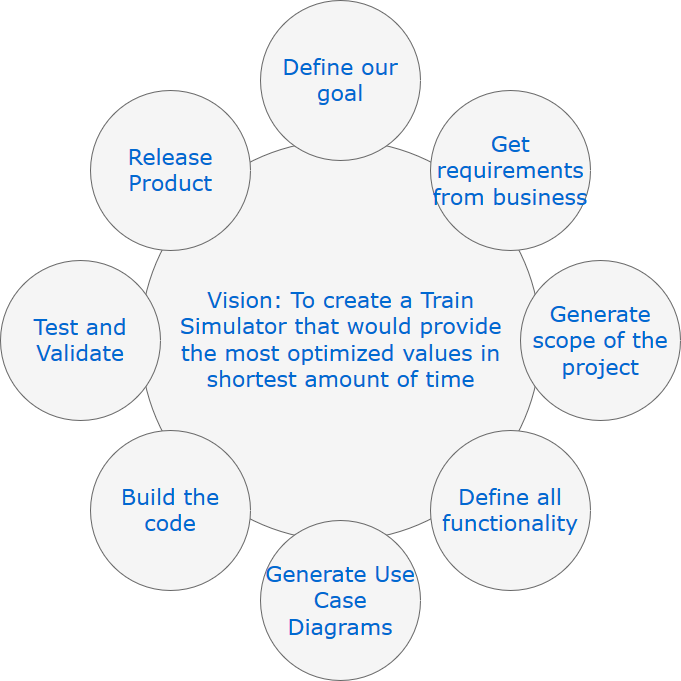
## 7.3 Performance Bounds

* N/A

# 8.0 Appendices

## 8.1 System Traceability Matrix

## 8.2 Product Strategies



## 8.3 Analysis Metrics to be used

* Distance traveled per train
* Name of train, station, hub, and track
* Type of train
* Collision occurrences per train
* Total amount of trains stopped at a station
* Amount of items picked up at a station
* Amount of visits at a station
* Amount of items dropped off at a station
* Amount of times a track is visited
* Amount of times a hub is visited

## 8.4 Supplementary Information