Software Requirements Specification MTOBridge

Team 15, Alpha Software Solutions
Badawy, Adham
Yazdinia, Pedram
Jandric, David
Vakili, Farzad
Vezina, Victor
Chiu, Darren

Contents

Table 1: Revision History

Date	Developer(s)	Change
October 4	David	Add context/partitioning of work
October 4	Adham	First Draft of FRs
October 4	Darren	Added some Non-functional Requirements
October 4	Victor	Project Issues
October 5	Victor	Added some Non-functional Requirements
October 5	Darren	Refined Non-functional Requirements
October 5	Pedram	Added Individual Product Usecase
October 5	Adham	Finished FRs
October 5	Adham	Added Likely Changes Table
October 5	Farzad	Completed parts of project drivers

List of Tables

List of Figures

This document describes the requirements for MTOBridge. The template for the Software Requirements Specification (SRS) is a subset of the Volere template Robertson and Robertson (2012). In addition, we have added a few sections not in the Volere template: Appendix (and its subsections), Phase in Plan.

1 Project Drivers

1.1 The Purpose of the Project

For years, Bridge Engineers in Ontario have based their bridge analysis on the Canadian Highway Bridge Design Code (CHBDC)(CSA S6-19) which typically features conservatism and adds excessive costs. This project aims to develop an application 'MTOBridge' that leverages prepared MATLAB engines (containing the logic for novel analysis methods) and better visualizes their outputs while enhancing usability. This enables the refined methods of analysis to become routine for bridge design and evaluation. Moreover, A unique feature of 'MTOBridge', which is otherwise unavailable in other bridge design/evaluation software, is that a direct comparison can be obtained between the results obtained from the CHBDC approximate/simplified approach and those from the refined analysis. This endeavor will facilitate confidence in using refined analysis for bridge evaluation and design.

1.2 The Stakeholders

1.2.1 The Client

Ontario Ministry of Transport (MTO) is the sponsor of this project and is making the investment to bring this program to fruition in partnership with the McMaster University Civil Engineering Department.

1.2.2 The Customers

Civil engineering firms in the bridge design industry. They will dictate whether or not they will adopt this new tool depending on whether it has helped them save costs for their clients and how easy it was for their engineers to use the tool.

1.2.3 Other Stakeholders

- Department of Civil Engineering, McMaster
  - The proposed program will be directly developed in collaboration with The Department of Civil Eningeering.
- Department of Computing and Software, McMaster
  - The proposed program will be directly developed by Engineers from The Department of Computing and Software.

1.3 Mandated Constraints

Const 1: The system shall be able to perform the calculations using the prepared MATLAB Engines.
Rationale: Dr.Yang and her team which are one of the main stakeholders are not going to implement their engines in another language again.
Fit Criterion: The system shall be able to call the appropriate functions within the MATLAB engine to perform the commanded analysis.

Const 2: The product shall operate using Windows 10.
Rationale: The client and majority of the costumers use Windows 10 and do not wish to change to a later version.
Fit Criterion: The product shall be approved as Windows 10 compliant based on our tests on a windows machine.

Const 3: The product shouldn't require MATLAB to be installed on the end user's machine to run the MATLAB engines.
Rationale: The majority of the Costumers do not have MATLAB licences and chances are they don't want to purchase them..
Fit Criterion: The product shall be able to use the logic written in MATLAB without needing to install MATLAB.

1.4 Naming Conventions and Terminology

Abutment:A substructure composed of stone, concrete, brick or timber supporting the ends of a single span bridge or the extreme end of a multi-span bridge. Usually, it also supports the approach embankment.
Dead load: The weight of the permanent, nonmoveable parts of a structure, such as the towers, cables, and roadway of a bridge.
Girder:A horizontal main structural member (as in a building or bridge) that supports vertical loads and that consists of a single piece or of more than one piece bound together.
Live load: Refers to traffic that moves across the bridge as well as normal environmental factors such as changes in temperature, precipitation, and winds.
Span: In engineering parlance means the gap between two supports.

1.5 Relevant Facts and Assumptions

User characteristics should go under assumptions.

2 Functional Requirements

2.1 The Scope of the Work

2.1.1 The Context of the Work

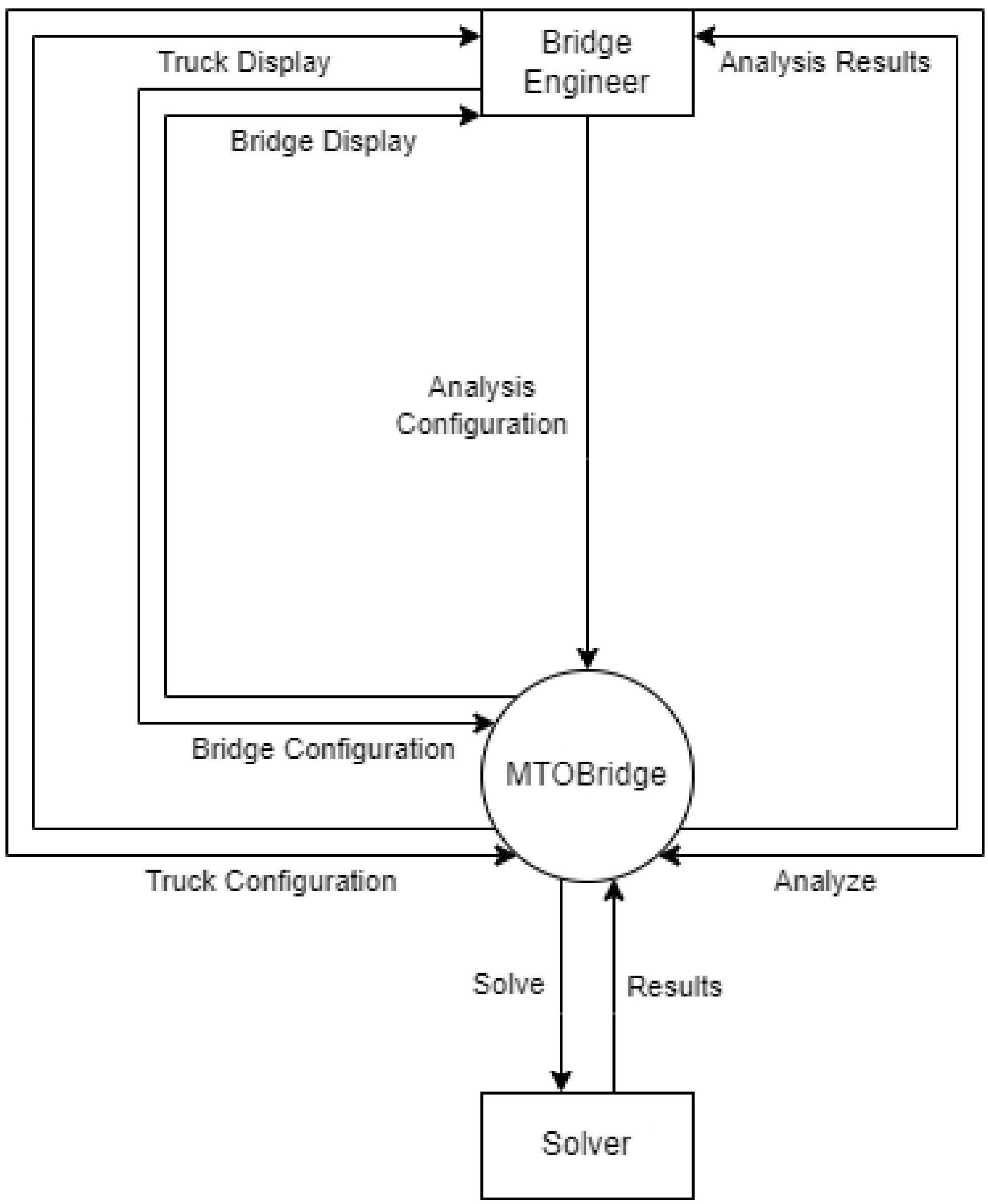

Figure 1: Context Diagram of MTOBridge

2.1.2 Work Partitioning

Table 2: Business Event List

Event Name	**Input/Output**	**Summary**
Engineer enters truck configuration	IN: Truck configuration, OUT: Truck display	Record truck configuration, show truck visualization
Engineer enters bridge configuration	IN: Bridge configuration, OUT: Bridge display	Record bridge configuration, show bridge visualization
Engineer enters analysis configuration	IN: Analysis configuration	Record analysis configuration
Engineer requests analysis	IN: Analyze request, OUT: Analysis results	Display analysis results
Time to solve forces	OUT: Solve request, IN: Solver results	Give configurations to solver to get results

2.2 The Scope of the Product

2.2.1 Product Boundary

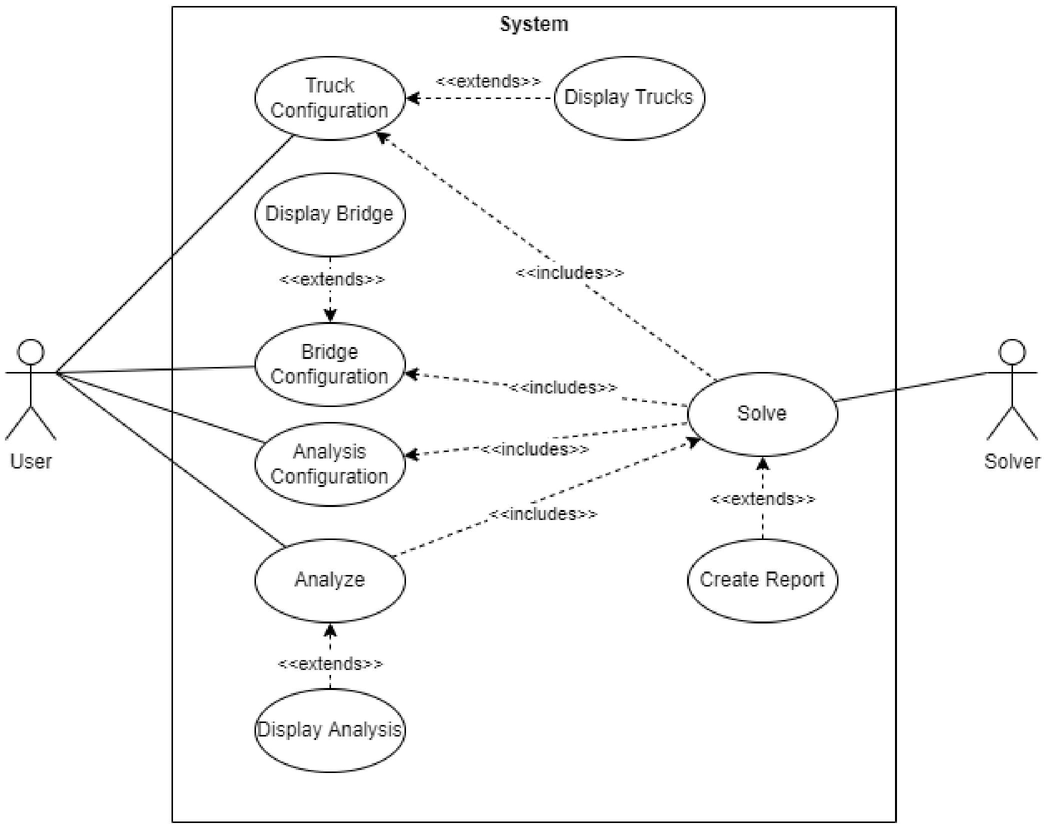

Figure 2: Use Case Diagram of MTOBridge

2.2.2 Individual Product Use Cases

User navigates to the Truck Configuration Tab
Primary Actors: Engineers
Pre Condition: None
Post Condition: Input values from the user conforms to the valid range required for the calculations.
Basic Flow:

- User is asked to enter the truck type (model) as well as the truck length.

- User is asked to enter the headway between the trucks as well as the number of trucks.
- User is asked to enter Axial loads and the distance between each other

Extension:

- User can view an animation the truck platoon representing the parameters inputted by the user.
- User can interact with the animation of the trucks to increase the headway or aixal loads distance.

User navigates to the Bridge Configuration Tab
Primary Actors: Engineers
Pre Condition: None
Post Condition: Input values from the user conforms to the valid range required for the calculations.
Basic Flow:

- User is asked to enterthe number of spans as well as the length of each.
- User is asked to enter the material used for the bridge. The system will have a series of preloaded information per material type.
- User will have the ability to configure any default information associated with the breidge material.

Extension:

- User can view an animation of the bridge highlighting the parameters inputted by the user.
- User can interact with the animation of the bridge to change the length of the spans as well as the mateiral type.

User navigates to the Analysis Configuration Tab
Primary Actors: Engineers
Pre Condition: None
Post Condition: Input values from the user conforms to the valid range required for the calculations.
Basic Flow:

- User is asked to enter the discretization length which will be directly used in the visualization process.
- User is asked to enter the concerned section which is the area of interest to user when the simulation goes through.
- User is asked to enter the type of force desired by the user. This can include Shear and Torsion.

User starts the Analyze process.
Primary Actors: Engineers
Pre Condition: None
Post Condition: None
Basic Flow:

- User selects the Analyze tab and confirms the input data from Truck, Bridge and Analysis configuration.
- The system checks for parameters once submitted.

  Alternative If the values are not correct, the user is alereted.
- If correct, parameters are sent to be solved.
- Text-based result in form of a report are retruned.

Extension:

- The recieved results are generated as a report including several graphs and charts.
- An animamated, interactable plot is genererated along with a visualization of the trucks on the bridge to dynamically highlight different values of force.

The solver solves the mathematical model using the given inputs from the system.
Primary Actors: Solver
Pre Condition: Truck, Bridge and Analysis Configuration are all valid.
Post Condition: Recieved results valid based on the testing spec.
Basic Flow:

- The system recieves the configuration data from each configurator.

- The system sends the data to the Solver where the equations are solved using the given parameters.
- Solver returns text-based results which are sent back into the system.

  Alternative The solver cannot solve the equations using the given input. The user is alerted to check their parameters.

2.3 Functional Requirements

FR.1 The Program should be able to call the backend MATLAB functions.
Rationale: To visually display the outputs corresponding to user input, the UI needs to first figure out what those outputs are via the backend MATLAB.
Fit Criterion: Testing will be performed to confirm whether or not the program can successfully call the MATLAB functions.

FR.2 The Program should allow the user to define the characteristics of the truck platoon, including truck configuration, number of trucks, headway, and travel speed.
Rationale: The goal of the program is to determine the forces exerted by a given platoon on a bridge, they can come in many different forms, so flexibility in the characteristics of the platoon are necessary for the relevance of the simulation.
Fit Criterion: At least one method of input(text based, dropdown list, etc) exists that allows users to correctly specify those 4 characteristics of the platoon.

FR.3 The Program should allow the user to visualize the effects of their truck platoon characteristic definitions on the final platoon.
Rationale: This is mainly to help the user verify that the inputs they put into the program correspond to the platoon they had in mind. As we are making a GUI, visual feedback is paramount to functionality.
Fit Criterion: There exists some visual representation of the truck platoon that changes to correctly reflect the impact of changes in user input.

FR.4 The Program should allow the user to save their current truck platoon configuration for later use.
Rationale: Users may be interested in testing the demands a particular truck platoon places on bridges disproportionately more often than others, in multiple situations, and over multiple sessions. Making it so they don't have to manually remake it every time is important for the efficient use of The Program.
Fit Criterion: The Program is capable of storing a truck platoon configuration in some form to be retrieved later.

FR.5 The Program should allow the user to load in their previously saved truck platoon configurations.
Rationale: A companion piece to FR.4, the ability to save a configuration is functionally useless if you can't pull it back out later.
Fit Criterion: The Program is capable of correctly recreating a truck platoon configuration from previously saved data.

FR.6 The Program should allow the user to define the characteristics of the bridge, inclduing what type of bridge it is and its length.
Rationale: Different bridges will react to the same truck platoon differently, therefore specifiying the relevant characteristics of the bridge is necessary for the relevance of the simulation.
Fit Criterion At least one method of input(text based, dropdown list, etc) exists that allows users to correctly specify those 2 characteristics of the bridge.

FR.7 The Program should allow the user to visualize the effects of their bridge characteristic definitions on the final bridge.
Rationale: This is mainly to help the user verify that the inputs they put into the program correspond to the bridge they had in mind. As we are making a GUI, visual feedback is paramount to functionality.
Fit Criterion: There exists some visual representation of the bridge that changes to correctly reflect the impact of changes in user input.

FR.8 The Program should allow the user to save their current bridge configuration for later use.
Rationale: Users may be interested in testing a particular bridge configuration disproportionately more often than others, in multiple situations and over multiple sessions. Making it so they don't have to manually remake it every time is important for the efficient use of The Program.
Fit Criterion: The Program is capable of storing a bridge configuration in some form to be retrieved later.

FR.9 The Program should allow the user to load in their previously saved bridge configurations.
Rationale: A companion piece to FR.8, the ability to save a configuration is functionally useless if you can't pull it back out later.
Fit Criterion: The Program is capable of correctly recreating a bridge configuration from previously saved data.

FR.10 The Program should allow the user to define which of the two solvers they are interested in using.
Rationale: as the MATLAB backend can solve for both the demand on a concerned section as the platoon drives along, as well as for which section has the highest maximum demand over the course of the whole trip, and these are very different pieces of info, allowing the user to determine which they are currently interested in is important.
Fit Criterion: At least one method of input(text based, dropdown list, etc) exists that allows the user to accurately choose which solver they are would like to use.

FR.11 The Program should allow the user to define a section of concern on the bridge.
Rationale: The first solver revolves around calculating the demand on a certain point along the bridge as the truck platoon drives over, specifying what point it is that we care about is necessray for this function.
Fit Criterion: At least one method of input(text based, dropdown list, etc) exists that allows the user to accurately determine a section of concern on the bridge.

FR.12 The Program should allow the user to define a discretization length for their bridge.
Rationale: The second solver finds which section has the maximum demand placed on it over the course of the platoon's trip. The discretization length determines how many sections the bridge is split up into, which is necessary for the functioning of the second solver.
Fit Criterion: At least one method of input(text based, dropdown list, etc) exists that allows the user to accurately define a discretization length for their bridge.

FR.13 The Program should allow the user to define which type of demand placed on the bridge they are interested in, between shear forces and positive/negative moment.
Rationale: There are a variety of different demands placed on the bridge as the platoon drives over, and the MATLAB backend contains calculations for all 3 of the above mentioned demands. Allowing the user to define which of the 3 they are interested in seeing is necessary for the functionality of the simulation.
Fit Criterion: At least one method of input(text based, dropdown list, etc) exists that allows the user to correctly define which of the 3 demands they are interested in simulating.

FR.14 The Program should be capable of visualizing the results of the concerned section calculation for the user.
Rationale: This is essentially the main purpose of the GUI; displaying the results of the MATLAB backend calculations visually to the user. This is one of the two main caluclations to be represented, so this functionality is very necessary.
Fit Criterion: There exists an accurate visualization of the mathematical results of the concerned section calculation.

FR.15 The Program should be capable of visualizing the results of the critical section calculation for the user.

Rationale: This is essentially the main purpose of the GUI; displaying the results of the MATLAB backend calculations visually to the user. This is one of the two main caluclations to be represented, so this functionality is very necessary.
Fit Criterion: There exists an accurate visualization of the mathematical results of the critical section calculation.

FR.16 The Program should be capable of outputting a report summarizing the results of its runtime.
Rationale: Having a long term representation of the data presented in the program after the current session of use is over is helpful for engineers comparing and contrasting diffrent simulations over time. Without this, the information would be lost as soon as the program was exited.
Fit Criterion: There exists an accurate report that cotains all the relevant data from simulations run over the course of the runtime in some output format.

2.4 Phase in Plan

3 Non-functional Requirements

3.1 Look and Feel Requirements

NFR.? The graphics will be informative.
Rationale: The user should gain value out of the presence of graphics.
Fit Criterion: Civil engineers who use the program will understand the different graphic elements used to represent bridge parts.
Traceability: ?.

3.2 Usability and Humanity Requirements

NFR.? The program will be intuitive to use.
Rationale: The program should be easy to use for its intended audience.

Fit Criterion: Civil engineers can use the program to generate a bridge system analysis within 5 minutes of introduction.
Traceability: ?.

NFR.? Program will have a user manual and (user-based) documentation provided.
Rationale: New users should be provided with resources to quickly start using the program.
Fit Criterion: User-based documentation covering usage and features of the program will be written.
Traceability: ?.

NFR.? The product will appear correctly on different display resolutions.
Rationale: Users of the product may wish to use it on displays of different resolutions.
Fit Criterion: The Program will be viewed and its appearance validated on displays of different resolutions.
Traceability: ?.

NFR.? The product will allow for the resizing of text.
Rationale: Users of the product may wish to increase text size to allow for easier reading of the text.
Fit Criterion: We will ensure that the text size within the program is resizable, and that the program still functions correctly when the text size is changed.
Traceability: ?.

NFR.? The program will be easy to install and run.
Rationale: The program cannot be difficult to install for the users of our product.
Fit Criterion: We will measure the time it takes to download, install, and run our program.
Traceability: ?.

NFR.? (Not sure if usability or look and feel) UI elements which provide similar functionality will have a similar look.
Rationale: Having UI elements with comparable functionality be visually consistent will help with usability.
Fit Criterion: We will classify UI elements into categories and ensure that all elements within each categories are visually similar.
Traceability: ?.

3.3 Performance Requirements

NFR.? The program will safely handle unusual user inputs.
Rationale: Program should be robust and not prone to failure due to common misinputs.
Fit Criterion: The program will not freeze or crash as a direct result of a user providing inputs to the system.
Traceability: ?.

NFR.? The program will be able to handle missing dependencies.
Rationale: The program should be able to handle and warn of absent files.
Fit Criterion: The program will produce an error message when MATLAB scripts are absent or unable to run.
Traceability: ?.

NFR.? UI elements will react promptly to user input.
Rationale: The users of the program will want the UI to react quickly to their input.
Fit Criterion: The UI will graphically update to indicate it has acknowledged user inputs within 100ms of the input.
Traceability: ?.

NFR.? UI will not be unreasonably slow.

Rationale: UI should not introduce substantial delay beyond what is needed to calculate results.
Fit Criterion: The program delay when calculating and displaying results will not exceed the underlying MATLAB script's execution time by 10%.
Traceability: ?.

NFR.? The program will be precise.
Rationale: The program must be reasonably precise to provide value for simulating bridges under load.
Fit Criterion: Calculations are accurate to within 1% relative error of similar bridge simulation engines.
Traceability: ?.

3.4 Operational and Environmental Requirements

NFR.? The program will run without slowdown on expected users' (MTO engineers) computers.
Rationale: The program must be able to run within requirements on the computers that it is intended to be used on.
Fit Criterion: Performance testing will be done on a computer with the same (or reasonably similar) hardware.
Traceability: ?.

3.5 Maintainability and Support Requirements

NFR.? The product shall be easily maintainable.
Rationale: The code must be easily maintainable to allow for future bug fixes and/or feature additions.
Fit Criterion: We will use file length, method length, and nesting depth as our primary indicators of code maintainability.
Traceability: ?.

3.6 Security Requirements

N/A; this project does not involve significant communication of user data.

3.7 Cultural Requirements

NFR.? The program should be able to be easily translated into other languages.
Rationale: People who don't speak English may wish to use the program, especially French speakers as Canada is a bilingual country.
Fit Criterion: Localization process to support French will not take over one week to complete.
Traceability: ?.

3.8 Legal Requirements

NFR.? Private MTOBridge assets will not be exposed for easy access to users.
Rationale: The client has expressed that their assets should be held confidential.
Fit Criterion: Received assets including MATLAB scripts will be excluded or compiled when in public repositories and distributed.
Traceability: ?.

3.9 Health and Safety Requirements

N/A; this project is only concerned with the graphical representation of the underlying bridge calculations.

4 Project Issues

4.1 Open Issues

N/A; there are currently no open issues for our project.

4.2 Off-the-Shelf Solutions

N/A; since we are using novel calculations designed specifically for this project, there are no comparable off-the-shelf solutions to our product.

4.3 New Problems

N/A; there are currently no new problems for our project

4.4 Tasks

N/A; there are currently no required tasks for our project.

4.5 Migration to the New Product

N/A; this product is novel and would not be replacing an existing product.

4.6 Risks

4.7 Costs

No monetary costs are involved in creating this product. The only costs in developing the product are the project team's time, and Dr. Yang's and their graduate students' time.

4.8 User Documentation and Training

An extensive user manual with case study examples will be produced alongside the product. This documentation will allow for bridge engineers to become familiar and comfortable using the software. We will also train Dr. Cancan Yang and their graduate students to use the product so that they can effectively present the product to the MTO.

4.9 Waiting Room

N/A; we don't currently have any requirements in the project waiting room.

4.10 Ideas for Solutions

N/A; we have not currently made any decisions about solution ideas.

References

James Robertson and Suzanne Robertson. *Volere Requirements Specification Template*. Atlantic Systems Guild Limited, 16 edition, 2012.

5 Appendix

This section has been added to the Volere template. This is where you can place additional information.

5.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance.

5.2 Likely Changes

Table 3: Likely Changes

Req.	Likelihood of Change	Rationale
FR.1	Nigh Impossible	This is a core feature.
FR.2	Very Unlikely	Testing different truck platoons is very important to the simulation's usefulness
FR.3	Very Unlikely	Visual Feedback is a core part of the usefulness of the UI
FR.4	Possible	This is a "nice to have" feature.
FR.5	Possible	Same Likelihood as FR.4, as if that changes, this changes
FR.6	Very Unlikely	Testing different types of bridges is very important to the simulation's usefulness
FR.7	Very Unlikely	Visual Feedback is a core part of the usefulness of the UI
FR.8	Possible	This is a "nice to have" feature.
FR.9	Possible	Same Likelihood as FR.8, as if that changes, this changes
FR.10	Nigh Impossible	This is a core feature.
FR.11	Nigh Impossible	This is a core feature.
FR.12	Nigh Impossible	This is a core feature.
FR.13	Very Unlikely	Bridge Engineers are very interested in seeing multiple different types of demands on their bridge, not just one
FR.14	Act of God	This is THE core feature.
FR.15	Act of God	This is THE core feature.
FR.16	Very Unlikely	A Report is important to preserve the simulation's findings

5.3 Traceability Matrix

5.4 Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?