Hazard Analysis
MTOBridge

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

Table 1: Revision History

Date	**Developer(s)**	**Change**
October 12 2022	Darren	Added System Boundaries & Components
October 19 2022	Adham	Added Adham/Victor/Farzads FMEA work into a latex table
X	Y	Z
...	...	...

Contents

[You are free to modify this template. —SS]

1 Introduction

[You can include your definition of what a hazard is here. —SS]

2 Scope and Purpose of Hazard Analysis

3 System Boundaries and Components

This hazard analysis addresses the system that consists of the following components:

1. UI Component, for providing a graphic display to the user and visualizing MATLAB results
2. Input Handler Component, for processing user inputs
3. MATLAB Interaction Component, for calling scripts and supplying specified arguments to them
4. MATLAB Engine Component, for performing bridge calculations
5. File Manager Component, for reading inputs from files and saving results in various formats

The system boundary includes these software components and any dependency files required for the application to operate. Although the MATLAB Engine Component is owned by the client and its exact contents verified independent of this project, this hazard analysis will address it due to being a crucial component of the system.

4 Critical Assumptions

We will be making the following assumptions about the system:

1. The MATLAB Engine Component will take less than 1 second to perform bridge calculations. This is because the equations that need to be solved have closed form solutions, and should be easy to compute.
2. The MATLAB Engine Component will be correct. We are assuming this because we do not have control over the MATLAB scripts that will be running.
3. The user of the system will have input (keyboard, mouse) and output (monitor/display) devices.

5 Failure Mode and Effect Analysis

Table 2: FMEA Analysis

Component	**Failure**	**Effect**	**Cause**	**Recommended Action**	**SR**	**Ref**
UI	UI Displays truck config incorrectly	User confusion / Misleading interface	**A.** Incorrect processing of regular user input— **B.** Runtime error in truck display module— **C.** Unexpected/boundary case user input	**A.** Thoroughly test the truck display module to avoid unexpected responses to input— **B.** Ensure truck display module has proper error handling to avoid catastrophic failure if an error is encountered, and perhaps instead prompt the user to try again, for example.— **C.** Design modules with separation of concerns in mind to limit complexity and increase program robustness.	None	HA-1
UI	UI does not update to match new truck config at all	User missing important information.	**A.** Failure to catch invalid user input.— **B.** Same as HA-1— **C.** Same as HA-1	**A.** Ensure truck display module has proper input bounds and safety nets to catch invalid user inputs, instead of just running with them.— **B.** Same as HA-1— **C.** Same as HA-1	None	HA-2
UI	UI Displays bridge config incorrectly.	Same as HA-1	**A.** Same as HA-1.— **B.** Same as HA-1.— **C.** Same as HA-1.	**A.** Same as HA-1— **B.** Same as HA-1.— **C.** Same as HA-1.	None	HA-3

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
UI	UI does not update to match new bridge config at all	Same as HA-2	**A.** Same as HA-2.— **B.** Same as HA-2.— **C.** Same as HA-2.	**A.** Same as HA-2.— **B.** Same as HA-2.— **C.** Same as HA-2.	None	HA-4
UI	UI attempts to display undesired calculation type	Display is worthless	**A.** Incorrect processing of regular user input.— **B.** Misleading or incorrect display of user solver selection.	**A.** Thoroughly test solver configuration module to incorrect processing of user input— **B.** Minimize complexity of input handler/solver selection display modules and the interaction between them to reduce chances of incorrect information passing and misleading or incorrect displays of input.	None	HA-5
UI	Truck platoon trip display does not match bridge load display.	Display is impossible to parse	**A.** Incorrect calculation display logic.— **B.** Unexpected bug or glitch is calculation display modules.	**A.** Thoroughly test calculation display module(s) to avoid unexpected behavior.— **B.** Include checks to determine if the two displays align and catch/correct it if they don't instead of just displaying it anyways.— **A/B.** Look into splitting the calculation display modules into two modules entirely, platoon trip and bridge load display, to simplify each part and reduce chance of logical errors.	SR-1	HA-6

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
UI	Platoon trip and Bridge load synch check(s) provides false positives or negatives	Deny a fine display or let through an erroneous one.	**A.** Incorrect synch check logic.— **B.** Unexpected bug or glitch in synch module.	**A.** Thoroughly test synch checks to avoid unexpected behavior.— **B.** Simplify synch check logic as much as is possible while maintaining accuracy to limit chance of incorrect logic programming	SR-1	HA-7
UI	UI incorrectly displays the concerned section	Display is worthless	**A.** unexpected bug/glitch in concerned section display module.— **B.** misleading or incorrect display of user concerned section selection.	**A.** Thoroughly test concerned section display module to incorrect processing of user input— **B.** Minimize complexity of input handler and concerned section display display modules and the interaction to reduce chances of incorrect information passing and misleading/incorrect displays of input.	None	HA-8
UI	UI incorrectly displays discretized bridge segments.	Same as HA-8	**A.** Same as HA-8.— **B.**Same as HA-8.	**A.** Same as HA-8.— **B.**Same as HA-8.	None	HA-9

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
UI	UI fails to display calculation results entirely.	Display is worthless.	**A.** Unexpected/boundary case user input.— **B.** Runtime error in calculation display modules.— **C.** Failure to catch invalid user input	**A.** Thoroughly test the calculation display modules to avoid unexpected responses to input— **B.** Ensure calculation display modules have proper error handling to avoid catastrophic failure if an error is encountered, and perhaps instead prompt the user to try again, for example.— **A/B.** Design modules with separation of concerns in mind to limit error propagation and increase program robustness.— **C.** Ensure calculation display modules have proper input bounds and safety nets to catch invalid user inputs, instead of just running with them.	None	HA-10
UI	UI stops reacting to user inputs	User locked out from using UI, program is worthless	**A.** Runtime error in calculation display modules.— **B.** Parallel computing issue such as deadlock that hangs the program.	**A.** Ensure all display modules have proper error handling to avoid catastrophic failure if an error is encountered, and perhaps instead prompt the user to try again, for example.— **B.** Implement proper thread safety measures to avoid deadlocks and other such issues.— **A/B.** Design all modules with separation of concerns in mind to limit error propagation and increase program robustness.	None	HA-11

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
UI	UI encounters parallel computing issue such as deadlock/race condition.	Incorrect results or unexpected program behavior	**A.** Multiple threads modifying the same values/waiting on each other.—	**A.** Implement proper thready safety measures to avoid deadlocks and other such issues	None	HA-12
Matlab Interaction	Data received is incorrectly formatted.	Program cannot function	**A.** One-time error in cross-program communication caused by outside factors (OS, hardware, etc.)— **B.** Bug or error in MATLAB engine	**A.** Try all calculations a second time when the first calculation fails— **B.** The MATLAB engine will be tested thoroughly to try to reduce the amount of bugs it has. The program will always log in-depth error information and display an error message to the user telling them to contact the developers when there is an issue with the MATLAB component.	SR-3	HB-1
Matlab Interaction	Unable to call engine.	Same as HB-1	**A.** One-time error in cross-program communication caused by outside factors (OS, hardware, etc.)— **B.** Engine not installed / installed improperly	**A.** Try all calculations a second time when the first calculation fails— **B.** The program will always display a message telling the user that they must install the MATLAB engine with a reference to the installation section of the user manual when the MATLAB engine is not detected	SR-3	HB-2

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
Matlab Engine	The engine crashes unexpectedly.	Same as HB-1	**A.** One-time crash caused by outside factors (OS, hardware, etc.)— **B.** Bug or error in MATLAB engine	Same as HB-1	None	HC-1
Matlab Engine	The engine calculations take more time than should be required (more than 1 second).	Program must wait for results	**A.** One-time error causing infinite looping caused by outside factors (OS, hardware, etc.)— **B.** Bug or error in MATLAB engine	Same as HB-1	None	HC-2
Matlab Engine	Data received from the engine is incorrect (as in physically impossible).	Same as HB-1	**A.** One-time calculation error caused by outside factors (OS, hardware, etc.)— **B.** Bug or error in MATLAB engine.	Same as HB-1	None	HC-3

Component	Failure	Effect	Cause	Recommended Action	SR	Ref
Input Handler	Handler passes inputs to other components that are too large or small.	Inaccurate analysis results	**A.** Accidental changes of input for example writing 100000 instead of 10000000	**A.** Validating numeric values are within an acceptable range	SR-4	HD-1
Input Handler	Handler invariant to type changes.	System Crash	**A.** Accidental mix and match of inputs for example inputting structure material inside load section	**A.** Validating input type before passing it on.	SR-5	HD-2
Input Handler	Handler passes on incomplete set of inputs.	Same as HD-2	**A.** Submitting before completing all input sections	**A.** Detecting if required inputs are missing from the model	None	HD-3
File Manager	File Manager loads corrupted configuration and saved files.	Inaccurate results or system crash	**A.** Process responsible for creating the file was interrupted.— **B.** files edited manually by power users	**A.** Have metrics that indicates file creation was completed and if not it is communicated to the user when loading.— **B.** Have metrics such as checksums to ensure the integrity of the files.	SR-6	HE-1
File Manager	File Manager partially saves file.	Data loss	**A.** Power outage— **B.** System crash	**A/B.** Automatically save to a file whenever user changes the configuration or at reasonable time intervals	SR-7	HE-2

6 Safety and Security Requirements

[Newly discovered requirements. These should also be added to the SRS. (A rationale design process how and why to fake it.) —SS]

7 Roadmap

[Which safety requirements will be implemented as part of the capstone timeline? Which requirements will be implemented in the future? —SS]