

Hazard Analysis

Bridging Gaps: AI for Diagram Accessibility

Team 22, Reading4All
Nawaal Fatima
Dhruv Sardana
Fiza Sehar
Moly Mikhail
Casey Francine Bulaclac

Table 1: Revision History

Date	Developer(s)	Change
Date1	Name(s)	Description of changes
Date2	Name(s)	Description of changes
...

Contents

1	Introduction	1
2	Scope and Purpose of Hazard Analysis	1
3	System Boundaries and Components	1
3.1	System Overview	1
3.2	System Boundaries	1
3.3	System Components	1
4	Critical Assumptions	2
5	Failure Mode and Effect Analysis	3
6	Safety and Security Requirements	3
7	Roadmap	3

[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

[You should say what **loss** could be incurred because of the hazards. —SS]

3 System Boundaries and Components

[Dividing the system into components will help you brainstorm the hazards. You shouldn't do a full design of the components, just get a feel for the major ones. For projects that involve hardware, the components will typically include each individual piece of hardware. If your software will have a database, or an important library, these are also potential components. —SS]

3.1 System Overview

The system is a web-based AI tool that generates alternative text (alt text) for uploaded images or figures and integrated with screen readers to improve accessibility for visually impaired users.

3.2 System Boundaries

- **Internal Components:** Alternative Text Generation Machine Learning (ML) Model, User Interface, Session History Manager
- **External Components:** McMaster Authentication System, External AI/ML Frameworks, Screen Reader Software

3.3 System Components

1. Alt Text Generation ML Model

- **Purpose:** To automatically generate accurate and descriptive alternative text for uploaded images using machine learning.
- **Key Functions:**
 - Process image inputs received from the backend and extract key visual features.
 - Generate contextually relevant text descriptions.
 - Return the generated alternative text to the backend to display to the user interface.

2. User Interface

- **Purpose:** To serve as the primary interaction point between the user and the system and allow users to upload images and view generated alt text.
- **Key Functions:**
 - Enable users to upload images through an accessible web interface.
 - Display generated alt text and allow users to edit, copy, and download the text.
 - Provide features that are accessible and complies with the Web Content Accessibility Guidelines (WCAG) 2.1 standards.
 - Communicates user requests and display outputs from the back-end.

3. Session History Manager

- **Purpose:** To ensure the continuity of the current session and manage user data during active use of the system.
- **Key Functions:**
 - Track unique user sessions throughout interaction with the web application.
 - Store previously uploaded images and generated alternative text for the current session to allow users to view history.

4 Critical Assumptions

This section documents the assumptions made during the hazard analysis of Bridging Gaps: AI for Diagram Accessibility (Reading4All). The number of assumptions is kept to a minimum to reduce the chance of overlooking potential hazards. Where assumptions are made, they set clear boundaries for the analysis and define the conditions under which the system is expected to operate safely.

- **Assumption 1: Input Integrity.** All image files provided to the system are assumed to be valid image formats and not corrupted or maliciously constructed to exploit parsing vulnerabilities.
- **Assumption 2: Standards Stability.** Accessibility guidelines (WCAG 2.1, AODA) are assumed to remain stable for the operational lifetime of the system.
- **Assumption 3: Model Performance.** The machine learning models used for visual recognition and natural language generation are assumed to operate within validated ranges of accuracy and reliability.

- **Assumption 4: Human Oversight.** It is assumed that alternative text generated by the system will undergo review by human instructors, teaching assistants, or accessibility specialists before being used in educational contexts.

Violations of these assumptions may introduce additional hazards outside the current scope of this analysis. Such cases would require re-evaluation of risks and system design updates.

5 Failure Mode and Effect Analysis

[Include your FMEA table here. This is the most important part of this document. —SS] [The safety requirements in the table do not have to have the prefix SR. The most important thing is to show traceability to your SRS. You might trace to requirements you have already written, or you might need to add new requirements. —SS] [If no safety requirement can be devised, other mitigation strategies can be entered in the table, including strategies involving providing additional documentation, and/or test cases. —SS]

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]

Appendix — Reflection

[Not required for CAS 741 —SS]

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?
2. What pain points did you experience during this deliverable, and how did you resolve them?
3. Which of your listed risks had your team thought of before this deliverable, and which did you think of while doing this deliverable? For the latter ones (ones you thought of while doing the Hazard Analysis), how did they come about?
4. Other than the risk of physical harm (some projects may not have any appreciable risks of this form), list at least 2 other types of risk in software products. Why are they important to consider?

Nawaal Fatima - Reflection

- 1. What went well while writing this deliverable?**
I think the easiest part was coming up with the assumptions themselves. From my summer work experience with Ms. Sui, I was already familiar with the main areas where Reading4All could run into issues (inputs, standards, model accuracy, and human review), it felt pretty natural to turn those into clear assumptions. It also helped that the section didn't need to be super long, so I could keep it focused and to the point.
- 2. What pain points did you experience during this deliverable, and how did you resolve them?**
The tricky part was figuring out how much detail to put in without over-complicating things. At first, I thought about writing assumptions that basically ruled out certain failures, but I realized that would go against the whole idea of hazard analysis. To fix that, I stuck to assumptions that made sense for defining boundaries without pretending hazards don't exist. Another small challenge was making the wording simple enough - so

I rewrote a couple of the assumptions to sound clearer and less “technical report” heavy.