Development Plan
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**
September 22, 2025	All	Initial version of document
Date2	Name(s)	Description of changes
...	...	...

[Put your introductory blurb here. Often the blurb is a brief roadmap of what is contained in the report. —SS]

[Additional information on the development plan can be found in the lecture slides. —SS]

1 Confidential Information?

[State whether your project has confidential information from industry, or not. If there is confidential information, point to the agreement you have in place. —SS]

[For most teams this section will just state that there is no confidential information to protect. —SS]

2 IP to Protect

[State whether there is IP to protect. If there is, point to the agreement. All students who are working on a project that requires an IP agreement are also required to sign the "Intellectual Property Guide Acknowledgement." —SS]

3 Copyright License

[What copyright license is your team adopting. Point to the license in your repo. —SS]

4 Team Meeting Plan

[How often will you meet? where? —SS]

[If the meeting is a physical location (not virtual), out of an abundance of caution for safety reasons you shouldn't put the location online —SS]

[How often will you meet with your industry advisor? when? where? —SS]

[Will meetings be virtual? At least some meetings should likely be in-person. —SS]

[How will the meetings be structured? There should be a chair for all meetings. There should be an agenda for all meetings. —SS]

5 Team Communication Plan

[Issues on GitHub should be part of your communication plan. —SS]

6 Team Member Roles

[You should identify the types of roles you anticipate, like notetaker, leader, meeting chair, reviewer. Assigning specific people to those roles is not necessary at this stage. In a student team the role of the individuals will likely change throughout the year. —SS]

7 Workflow Plan

- How will you be using git, including branches, pull request, etc.?
- How will you be managing issues, including template issues, issue classification, etc.?
- Use of CI/CD

8 Project Decomposition and Scheduling

- How will you be using GitHub projects?
- Include a link to your GitHub project

[How will the project be scheduled? This is the big picture schedule, not details. You will need to reproduce information that is in the course outline for deadlines. —SS]

9 Proof of Concept Demonstration Plan

Our Proof Of Concept(POC) will consist of roughly the following steps:

1. Identify the types of technical diagrams to target for alt text generation.
    - Such as but not limited to flowcharts, circuit diagrams, graphs, UML diagrams, and mathematical figures.
2. Determine the accessibility requirements and standards for alt text.
    - These requirements will be based on AODA and Web Content Accessibility Guidelines (WCAG) 2.1 guidelines, focusing on clarity, conciseness, and compatibility with screen readers.
3. Develop and evaluate the model's ability to generate alt text for a selected set of diagrams.
    - We will test outputs using a curated dataset of diagrams.
    - Evaluation will include both automated metrics (e.g., text length, keyword coverage) and qualitative user testing with screen reader compatibility checks.

4. Incorporate user feedback to refine the generated descriptions.
    - Feedback will be collected from users using metrics defined in the STEM Alt Text User Testing Project conducted by Ms. Jing in January 2025.
    - A human-in-the-loop process will validate and correct outputs to increase trustworthiness and usability.
5. Compare generated alt text against baseline approaches.
    - We will benchmark our tool's outputs against existing alt text practices (manual descriptions, generic auto-generation) and results from the STEM Alt Text User Testing Project.
    - The goal is to show measurable improvements in accessibility and user comprehension.
6. Demonstrate proof of integration with assistive technologies.
    - Final validation will involve testing with screen readers (e.g., NVDA, JAWS) to ensure proper reading order and interpretability.

The following is a brief list of primary risks to our success and how the POC results can mitigate them:

- **Risk:** The model may not be able to accurately generate alternative text that is considered accessible.
  Mitigation: The POC will include testing for the model on a diverse set of diagrams to evaluate its performance and regulate according to Accessibility for Ontarians with Disabilities Act (AODA) standards.
- **Risk:** The tool may not integrate smoothly with assistive technologies (e.g., screen readers).
  Mitigation: The POC will explicitly test generated alt text with standard screen readers to confirm compatibility and usability.
- **Risk:** End-users may perceive the generated alt text as inaccurate or misleading, reducing trust in the tool.
  Mitigation: The POC will incorporate a human-in-the-loop feedback mechanism, enabling iterative validation and correction of generated text.
- **Risk:** The project may face time constraints that could impact the depth of development.
  Mitigation: The POC will prioritize essential features and functionalities, allowing for a focused approach that can be expanded upon in future iterations.

10 Expected Technology

[What programming language or languages do you expect to use? What external libraries? What frameworks? What technologies. Are there major components of the implementation that you expect you will implement, despite the existence of libraries that provide the required functionality. For projects with machine learning, will you use pre-trained models, or be training your own model? —SS]

[The implementation decisions can, and likely will, change over the course of the project. The initial documentation should be written in an abstract way; it should be agnostic of the implementation choices, unless the implementation choices are project constraints. However, recording our initial thoughts on implementation helps understand the challenge level and feasibility of a project. It may also help with early identification of areas where project members will need to augment their training. —SS]

Topics to discuss include the following:

- Specific programming language
- Specific libraries
- Pre-trained models
- Specific linter tool (if appropriate)
- Specific unit testing framework
- Investigation of code coverage measuring tools
- Specific plans for Continuous Integration (CI), or an explanation that CI is not being done
- Specific performance measuring tools (like Valgrind), if appropriate
- Tools you will likely be using?

[git, GitHub and GitHub projects should be part of your technology. —SS]

11 Coding Standard

[What coding standard will you adopt? —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:

Nawaal Fatima - Development Plan Reflection

1. **Why is it important to create a development plan prior to starting the project?**
   Our project is very user-oriented and has more risks when it comes to reliability and functionality as Group 22 is designing for a demographic with which we have little/no working experience with. Knowing this, it is very important to have a blueprint of what we are bulding before we waste resources and cause our testers/end-users any unnecessary frustration. When we have a plan, we can also make sure that we are all working towards the same goal. It also helps us to identify potential challenges and risks early on, allowing us to develop strategies to mitigate them. The development plan asked a couple questions we did't consider, which helped us to think more critically about our project and how we can make it successful.

2. **In your opinion, what are the advantages and disadvantages of using CI/CD?**
   I think there are more advantages than disadvantages when it comes to CI/CD. The main advantage is that it allows for faster and more frequent releases, which can lead to quicker feedback from users and a more responsive development process. It also helps to catch bugs and issues early in the development process, which can save time and resources in the long run. However, one disadvantage is that it can be difficult to set up and maintain, especially for smaller teams or projects with limited resources. It also requires a certain level of discipline and commitment from the development team to ensure that code is properly tested and reviewed before being merged into the main branch. Overall, I think the benefits of CI/CD outweigh the challenges, and it is a valuable practice for modern software development.

3. **What disagreements did your group have in this deliverable, if**

any, and how did you resolve them?
We're in agreement about most aspects of the development plan. Most 'disagreements' we had were minor - such as naming conventiones to follow or what processes to esablish to ensure everything remains organized. We resolved these disagreements through open communication, making sure to listen to each other's perspectives and find solutions that worked for everyone. I imagine as we continue to work together, we may have more disagreements, but I am confident that we will be able to resolve them in a similar manner.

Appendix — Team Charter

[borrows from University of Portland Team Charter —SS]

External Goals

[What are your team's external goals for this project? These are not the goals related to the functionality or quality fo the project. These are the goals on what the team wishes to achieve with the project. Potential goals are to win a prize at the Capstone EXPO, or to have something to talk about in interviews, or to get an A+, etc. —SS]

Attendance

Expectations

[What are your team's expectations regarding meeting attendance (being on time, leaving early, missing meetings, etc.)? —SS]

Acceptable Excuse

[What constitutes an acceptable excuse for missing a meeting or a deadline? What types of excuses will not be considered acceptable? —SS]

In Case of Emergency

[What process will team members follow if they have an emergency and cannot attend a team meeting or complete their individual work promised for a team deliverable? —SS]

Accountability and Teamwork

Quality

[What are your team's expectations regarding the quality of team members' preparation for team meetings and the quality of the deliverables that members bring to the team? —SS]

Attitude

[What are your team's expectations regarding team members' ideas, interactions with the team, cooperation, attitudes, and anything else regarding team member contributions? Do you want to introduce a code of conduct? Do you want a conflict resolution plan? Can adopt existing codes of conduct. —SS]

Stay on Track

[What methods will be used to keep the team on track? How will your team ensure that members contribute as expected to the team and that the team performs as expected? How will your team reward members who do well and manage members whose performance is below expectations? What are the consequences for someone not contributing their fair share? —SS]

[You may wish to use the project management metrics collected for the TA and instructor for this. —SS]

[You can set target metrics for attendance, commits, etc. What are the consequences if someone doesn't hit their targets? Do they need to bring the coffee to the next team meeting? Does the team need to make an appointment with their TA, or the instructor? Are there incentives for reaching targets early? —SS]

Team Building

[How will you build team cohesion (fun time, group rituals, etc.)? —SS]

Decision Making

[How will you make decisions in your group? Consensus? Vote? How will you handle disagreements? —SS]