Development Plan
Computational Puzzle-solving the Ancient Textual History of Indian Buddhism

Team #, Team Name
Student 1 name
Student 2 name
Student 3 name
Student 4 name

Table 1: Revision History

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

This document presents the development plan for the capstone project *Computational Puzzle-solving the Ancient Textual History of Indian Buddhism*. The plan outlines the organizational and technical strategies our team will follow throughout the project. It includes details on confidentiality and intellectual property considerations, team meeting and communication plans, member roles, workflow, and project scheduling. In addition, it describes our proof-of-concept demonstration strategy, expected technologies, and coding standards. The appendix provides space for reflection and the team charter, which defines expectations, goals, and processes for effective collaboration. Together, these sections serve as a roadmap for ensuring the project is completed in a structured, transparent, and successful manner.

1 Confidential Information?

The image fragments that will be used to train the machine learning models in this project will be treated as confidential information. The team will be signing a confidentiality agreement to protect this data. At this stage, the specific agreement has not yet been provided. We are awaiting our next meeting with the industry advisor to receive clarification on the type of agreement that will be signed.

2 IP to Protect

Since the team will be using a General Public License (GPL), there will be no IP to protect in this project.

3 Copyright License

For this project, we will be using a General Public License (GPL) which will allow users to copy and modify the software. This will allow for greater collaboration and sharing of ideas within the open-source community.

4 Team Meeting Plan

The team will meet weekly in person on campus. If conflicts arise, the meeting will be held virtually. Meetings will be planned in advance by the organizer and led by the chair. During these meetings, the team will discuss progress, issues, and next steps.

The team will also meet with the industry advisor biweekly, in person, with the option to meet more frequently if needed. These meetings will last 30–60 minutes, be planned by the organizer, and led by the chair. During advisor meetings, the team will ask questions for clarification and receive feedback on project progress.

5 Team Communication Plan

The team will use Discord as the primary platform for communication, including day-to-day discussion, coordination, and sharing of resources. For urgent matters, team members will use SMS messaging to ensure timely responses. Communication with the industry advisor will be conducted through email to maintain a professional channel for updates, questions, and meeting arrangements. In addition, GitHub Issues will be used to track project-related tasks, bugs, and feature requests, ensuring transparency and accountability in the development process.

6 Team Member Roles

Our team will adopt five rotating roles to help structure collaboration and ensure accountability. While specific members will not be permanently assigned, these roles will be shared and will rotate on a biweekly basis to balance responsibilities throughout the project.

- **Leader** – Oversees the overall project, monitors progress against milestones, and ensures that all deliverables are submitted on time.

- **Organizer** – Prepares meeting agendas by identifying key discussion items and tasks that require team input.

- **Meeting Chair** – Facilitates meetings, keeping discussions on track and ensuring all agenda items are addressed.

- **Note Taker** – Records detailed notes during meetings, including decisions, action items, and deadlines, and circulates them to the team afterward.

- **Reviewer** – Reviews the team's work for accuracy, clarity, and completeness before submission or presentation.

These roles are intended to provide structure, improve efficiency, and distribute responsibility fairly among team members.

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

What is the main risk, or risks, for the success of your project? What will you demonstrate during your proof of concept demonstration to convince yourself that you will be able to overcome this risk?

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:

1. Why is it important to create a development plan prior to starting the project?

2. In your opinion, what are the advantages and disadvantages of using CI/CD?

3. What disagreements did your group have in this deliverable, if any, and how did you resolve them?

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]