CS 462: Senior Software Engineering Project Team Charter

Project Title: Prototype a web-based tool for creating and executing task-delineated, collaborative, AI-assisted assignments

Group 28

Project Deadline: End of Spring 2025

Team Roles

Name	ONID	Role
Oliver Zhou	zhouo	Project Manager
Trent Matsumura	matsumut	Developer - Backend
Ethan Lu	luet	Developer - AI Integration
Collin Kimball	kimbacol	Developer - Web UI
Sai Meenakshisundaram	meenkass	Documentation

Changelog

v1.0.0 (November 3rd, 2024): First submission for Canvas Assignment.

Author(s): Oliver, Sai, Team
Reason for Change: First submission Canvas.
Additions:

- Team Roles
- Background
- Agreement Summary
- Team Values and Priorities
- Document Outline
- Activities

v2.0.0 (February 2nd, 2025): Second submission for Canvas Assignment.

Author(s): Oliver, Sai
Reason for Change: Second submission Canvas to address new concerns for the winter term.
Additions:

- Outlining sprint meeting times with a mentor.
- Addresses needs discovered during winter term.
- Small changes to the communication process.

Introduction

The working agreement lays out the methods and expectations for guiding our team to work effectively together in a timely and successful manner, keeping us operating at a high standard. Our mission is to deliver a good final product for our project mentor and learn both technical and communication skills while working together as a team. The quality of our process determines much of the progress we can make in these 3 terms.

As the winter term has come around, this document has been updated to reflect the needed changes in communication that we feel are necessary to engage with alongside the project mentor to reinforce our workflow and maintain strong communication.

Working Agreement Summary

Role Responsibilities and Assignments

Team Lead - Oliver Zhou:

- **Direct the team**
- Keep up with TA and mentor communications
- Keep things running smooth
- Organize the project

Backend - Trent Matsumura:

- **Submit quality code**
- Meet product requirements

Frontend - Collin Kimball:

- **Create a usable interface**
- Meet product requirements

AI Integration - Ethan Lu:

- **Meet professor AI implementation requirements**
- Meet student AI implementation requirements

Documentation - Sai Anand:

- **Write readable documents**
- **Understand other developers roles and write effectively**
- Keep on top of tracking the team's progress

Mission

The goal of this project is to deliver a functional prototype of a task-delineated, AI-assisted, and collaborative homework assignment design tool for a project partner and professors/students in a university setting.

Scope

The scope of the project is highly configurable according to the project mentor, so the team should be ready at any time to refactor and refocus depending on new requirements.

Criteria/Objectives

Criteria is defined by suggestions from the 'Definition of Done' activity completed by one of the team members. This list defines the criteria for the completion of the project at a larger scale.

- Project requirements are met.
    - **Task delineation**
    - **AI assistance**
    - **Collaborative**
- Functional prototype of AI assisted learning tool usable by professors.
- All AI portions of the project are working and able to produce desired outputs.
    - **AI aggregation/summarization**
    - **AI helper bot (Renegotiation possible)**
- Code produces a tool that can be run off of a sharable link.
- Code has been reviewed and thoroughly tested via test cases.

This is according to the requirements of the project mentor.

Quality Standards:

- Bench marks should be that the tool runs smoothly with no faults. No crashes and run time should not be exceptionally long.
- Usability, the UI and layout should look neat and concise, with double checking from members.

Finishing and submitting work for the website's repo will require:

1. Review and approval from the project lead for pull requests.
2. Meets listed quality standards. Moves us closer to our project overview criteria.
3. Test cases (if applicable).
4. Neatly documented, then expanded upon by project documenter role (i.e. comments and commit messages, then fully documented into drive or whichever method Sai ultimately prefers).

Communication Methods

Main communication application: Discord
Alternative platforms: Email, phone numbers.

- **Team meetings** for important project details to be discussed.
- **Regular communication** (i.e. on completed tasks).
- **Frequent attendance** for meetings with project mentors and TA.
- **Problems and successes communicated** into the group Discord.

Time

Daily discussions and check-ins in the Discord. Stand-up during development weeks on an 'if-needed' basis, but otherwise messages are acceptable.

Weekly meeting: Fridays after Capstone or during the day.
Weekly TA meeting: Fridays at 12:00 pm.

Weekly Mentor meeting: Mondays at 10am.

Approach

Methodology: Agile

We agree to develop the project in an approach that focuses on our mentor's functional requirements and the usability, and efficiency/effectiveness of the tool. **We also commit to stay flexible when requirements change.**

Ex. We iterate weekly through the various progress points we want to make. We can start by prototyping functions one-by-one.

Organization

Project management tools:

- **GitHub for code, review, and task assignment.**
- **Google Drive for documentation.**

Agreed that tools are not fully settled as of the first version of the team charter for flexibility.

Recommendations/Considerations:

- Jira
- Confluence
- BitBucket

Accountability

- **Task ownership:**
  - Each member takes responsibility for their role.
  - Completes their role's tasks in time.
- **Communication of progress:**
  - Team members need to make clear where they are at.
  - Communicate when they need help.
  - Communicate blockers and collaborative moments.
- **Review and feedback:**
  - Feedback must be constructive.
  - Feedback is required.

Conflict Resolution

Conflicts need to be resolved and communicated honestly, respectfully, and constructively. Conflicts can and should also be addressed during review sessions during meetings or in the Discord. If things escalate, conflicts will be resolved with the TA or capstone instructors.

Ex. Disagreement about the structure or implementation of task-delineation could be resolved like:

- A Discord discussion about differences and communicating with the team lead.
- Come up with compromise and agreements
- Further escalation leads to resolution with a TA if nothing can be decided about the structure (i.e. One person wants a list of progress points vs. another person wants separate elements.)

Project Communication

Communication with the project mentor is necessary especially for the sprint process during development throughout the project life cycle. It is necessary for team members to communicate with the TA and mentor about progress. In order to do this, we agree to do the following:

- Post a bi-weekly sprint progress log will be posted on the communication channel.
- Speak with the mentor on mondays 10am-11am for post sprint bi-weekly progress review.
- Communicate with the TA before the end of each sprint.
- Document all processes.
-

Five Prioritized Items

This summary contains much of the important/relevant details to our team in this team charter agreement. We are committed to implementing these agreements collaboratively for the success of the team and the project.

We compiled a list of several potential focus areas below, which are reflected in the agreement summary:

Team Agreement Items:

- Organization
- Planning
- Communication
- Project Scope
- Meeting Mentor Project Requirements
- Timeliness
- Reviewing Code
- Agile Methods
- Norms and Team Culture
- Code Review and Standards
- Adaptability
- Conflict resolution
- Accountability
- Product Testing

Below we expand on five of our most important areas of focus from the team charter and discuss/justify why they are important to us. Much of these priorities focus on areas of improvement. We find that a lot of us are comfortable with the technical aspects of the project, so we prefer to focus on the project/group-oriented items.

Communication and Active Listening

We prioritize open communication and active listening because everyone's input matters. We commit to listening to each other's ideas and concerns, which develops an environment where everyone's voice matters at the table and is considered. Communicating in this fashion will help us work together better, and we all want to improve in this aspect.

Time – Punctuality

Keeping up on time is crucial for the success of our team because many of our tasks rely on each other keeping on top of things, especially in the code portions. One example of the importance of time

management is that staying on track keeps us in good standing on the class assignment, as well as the project timeline. Doing things early will help us catch problems and fix them on time. We are all college students who have struggled with procrastination and rushing in the past, so we decided this was important to improve on.

Organization – Planning

Taking time to plan out our work means we're clear about what needs to be done, which helps us stay efficient and organized. It also means we avoid unnecessary issues later down the road that could be solved by a little pre-planning. It's important for us because the project is limited in the allotted time and we want to be able to progress at a steady pace.

Approach: Flexibility and Adaptability

We decided to focus on flexibility and adaptability because our project mentor has communicated to our team that the project is highly flexible in its scope, and that we can discuss what can or cannot be done in these 3 terms. For this reason, we want everyone in the team to learn how to stay flexible. Additionally, this skill will help us assist each other whenever we have problems, even despite having different roles.

Accountability

Each team member is responsible for their contributions and holds themselves accountable to the team. Taking ownership of our tasks and being reliable means that we build trust among each other and we can rely on each other when we need to. For example, some tasks such as UI elements functionality from the frontend developer rely on the backend developer, so we need to keep each other accountable so that both roles can make progress.

Charter Improvements

As the team progresses through the term, the charter may need to be edited to be improved upon. The team agreement needs to be transparent, accessible, actionable, and most importantly, a living document that is revisited and worked on for the sake of collaboration.

The team has discussed various improvements as of now to further expand upon this document in the future. Many of these changes could be facilitated through team activities or individual activities as we revisit the document.

Below is a list including some improvements that could be made:

- Expand on previously defined criteria and agreements
- Rework elements of the charter that are no longer realistic
- Cut fat and focus on specific parts that need the most improvement
- Discover what parts of the team are struggling as workload increases and refocus
- Setting up more aspects of the team charter.
- Take the mentor's desires into more consideration when revising the charter
- Add a signature from each member of the team.

Team Agreement Activities

The team worked on a single teamwork activity as a whole together. Then, all members worked on their own individual activity for the planning activity requirement. Write-ups of all these activities are included below.

Teamwork Activity – Identify Individual Learning Objectives and Skills

Activity Summary

Each team member creates two lists:

- one for motivations and learning objectives, and
- another for current strengths and skills.

Discussion Summary

Our discussion around our lists concluded that we should put our focus in improving project skills. One thing that we discussed was how our lists were mostly focused around our technical skills that we gained throughout our degree. We all have a different subset of abilities despite working in the same field, which means that we are able to fill in gaps for each other in many areas, such as Trent's proficiencies in backend development, and Ethan's specialization in AI.

Despite being strong technically, that meant that our project management/soft skills had room for improvement. Since we have all had less experience in this regard, we decided to center our focus into these aspects. Our five focus areas as a team are centered around the team. Completing homework in our own times generally results in unorganized methods that don't reinforce good practices, especially with less accountability. So we decided that we wanted to develop communication skills for working as a team, keeping each other accountable, and working on-time consistently. Some of us felt more confident about these areas, but it was clear that we could all do more to work together more effectively.

Our learning objectives mainly center around developing our roles further while simultaneously working in a group setting. These learning objectives helped decide where to place our roles in the end (even though it was mostly settled beforehand). The place we ended after this activity complemented each other's strengths and weaknesses evenly and we are happy how the discussion went.

A formatted version of our lists are included below:

…

Project Manager: Oliver

Motivations and Learning Objectives

Motivations:

- Lead a project that meets the requirements of the project in an organized fashion.
- Create a functional tool that benefits education and applies artificial intelligence usefully.
- Makes the lives of professors and students easier.

Learning Objectives:

- Learning workplace management skills and tools that may be relevant in my future, regardless of whether I'm in a management position.
- Practice development processes like Agile.

Strengths and Skills

Strengths:

- Organized, and good at scheduling and coordinating tasks.
- Strong communication skills, and am able to coordinate and lead a team.

Skills:

- Communication skills.
- Taking the initiative and leading.
- Technical skills in coding to assist with development when needed.
- Organizational skills for structuring and facilitating development of the project.

Backend Developer: Trent

Motivations and Learning Objectives

Motivations:

- Make a final product backend for the site that allows assignments to be delineated and AI assisted.
- Deliver code that is quality and usable and performs well.

- Create a smooth experience for professors to make class assignments.

Learning Objectives:

- Learn to develop an artificial intelligence product.
- Improve on practices in software and web development.

Strengths and Skills

Strengths:

- Low level programming, and algorithms.
- Good problem solving skills.
- Web-development and web-design.

Skills:

- Python
- C/C++
- HTML, CSS, and JavaScript
- Various development methods like React.js and RestfulAPIs.

AI Developer: Ethan

Motivations and Learning Objectives

Motivations:

- Apply AI experience on the project to cover one of the primary requirements.
- Work on a real-world application of AI.
- Contribute to education.

Learning Objectives:

- Develop a stronger work ethic and time management skills.
- Direct experience with implementing AI models.

Strengths and Skills

Strengths:

- A background with statistics, math, AI and machine learning models.
- Working together with other students on projects.

Skills:

- Writing AI algorithms in Python.
- AI and Machine Learning.
- Statistics and data modeling.

Frontend Developer: Collin

Motivations and Learning Objectives

Motivations:

- Create a user interface that is easy for professors and students to use.
- Develop a web-based and AI project that is robust and functional.

Learning Objectives:

- Learn about how to improve web-design skills and user accessibility.
- Work together with a team to develop an artificial intelligence related project.
- Gain experience with web development frameworks.

Strengths and Skills:

Strengths:

- Problem solving.
- Graphic design and organization.
- Paying attention to details.

Skills:

- HTML, CSS, and JavaScript
- UI/UX principles from web-development projects.

Documenter: Sai

Motivations and Learning Objectives

Motivations:

- Complete a project that is useful in a real world situation.
- Further the development of AI applications in education.
- Ensure that the project's requirements are met for all web-based purposes.

Learning Objectives:

- Improve skills in project collaboration.
- Improve web design knowledge.
- Submitting clear and organized work.

Strengths and Skills

Strengths:

- Developing usable interfaces on websites.
- Attention to detail and ability to organize information.
- Communication skills.

Skills:

- Software development.
- Technical writing and formatting documents.
- C language.
- Assembly.
- Python, html, css, JavaScript.
- React.js

Individual Planning Activities

Oliver Zhou – Software Development Process

Individual Activity

Define and plan the software development process for your project to ensure clear workflow and task management.

Summary Write-Up:

I used this planning activity to decide much of the team agreement. It allowed me to make decisions and propose to the team about which methodology would work best for the team, break down the process for our sake, and assign roles for the team charter. Despite being an individual activity, I did collect feedback and opinions from the team for the team agreement here as well as the decisions I made for the activity as well as the portions of the team charter that were developed through this activity.

1. Choose a Development Methodology: Select an approach (e.g., Agile, Waterfall, SCRUM) that fits your project.

   I chose Agile as our primary development methodology. Agile is a process that we have all studied during CS 361 - Software Engineering, so I decided that Agile would be the best method to move forward for our team, since everyone had experience using it. It's also a process that is commonly in real companies, so practicing this methodology will help us stay organized and teach us a lot about working in a team.

2. Outline Development Phases:

   Phase 1: Disjoint and Waterfall

   This phase is for the start and middle of the fall term where the rigor of project development itself is light. Most things will simply be discussed in a more relaxed manner in the discord and worked on independently for project assignments. Things that require more effort (team charter, requirements, etc.) will be done in a waterfall-like manner if necessary.

   Phase 2: Agile

   This phase implements the Agile framework for our milestone progress. We will complete milestones with a single loop of Agile.

Phase 2.1: Agile Planning

The 1st real phase is when we implement Agile. We will start by planning out the requirements of the milestone.

Phase 2.2: Agile Design

We will document and design things outside of the code portion of the project using things such as chart creation tools. This will design the target for our milestone.

Phase 2.3: Agile Development

We will start developing the project according to each member's respective roles. This phase will most likely be the majority of our development process.

Phase 2.4: Agile Testing

We will write web-UI test cases for each milestone implemented in the project, and take account of everything working successfully and what needs improvement.

Phase 2.5: Deployment/Review

For our purposes, "deployment" is only amongst ourselves for the most part, or for showing the prototype to the project mentor for feedback. I'm combining this phase for brevity and because they should more or less be evaluated together in this shorter period. From this phase, we will loop back to phase 2 and re-analyze depending on the results of our review

Phase 3: Release

This is the final phase of the project near the end when things are completed. This phase will simply focus on presenting the results of the project.

3. Set Milestones: Establish key milestones and goals for each phase.

Our key milestones:

1. **Website Assignment Creation Basic Functions**
2. **Exportable Assignments**
3. **Usable Interface**
4. **Task-Delineation**
5. **AI-Summarization**
6. **AI-Helper Bots**
7. **Collaboration**

Phase 1 Goals:
Complete early on project requirements outside of the development process without the Agile structure.

Phase 2 Goals:
Incrementally develop each milestone/feature of the application until completion in an organized manner. Come out the other side with a working prototype.

Phase 3 Goals:
Deliver a functional prototype for the project mentor and a friendly and usable interface that professors can begin using immediately for their benefit.

4. Assign Roles: Clarify team roles and responsibilities at each stage.

I took the strengths and weakness from each of my team members and assigned these roles:

Oliver Zhou: Project Manager
Responsible for leading the team and keeping the project organized.

Trent Matsumura: Developer - Backend
Responsible for developing the functional backend of the web-tool.

Ethan Lu: Developer - AI Integration
Responsible for integrating AI models for the AI assisted tasks in the project.

Collin Kimball: Developer - Web UI
Responsible for developing a usable user-interface for the project.

Sai Meenakshisundaram - Documentation
Responsible for writing and documenting progress reports for the project and understanding how to communicate project details to other team members.

5. Team Agreement: Discuss the process with your team and ensure everyone agrees.

I discussed the process with my team and everybody had agreed on the Agile process as a whole. For this reason, I implemented this as the approach in the team charter document.

6. Provide a document outlining the chosen process, phases, milestones, and team roles.

This team charter's document contains my outline of these processes above.

Ethan Lu - Weekly Work Intensity

Week	Expected	Actual	Notes
1	Low	Low	
2	Low	Low	Operating Systems 2 Lab Week
3	Low	Low	
4	Medium	Low	Operating Systems 2 Lab Week
5	Medium	Low	
6	High	N/A	
7	Low	N/A	Operating Systems 2 Lab Week
8	High	N/A	
9	High	N/A	
10	Low	N/A	Dead Week
F	Low	N/A	Finals Week

Mid-Term Reflection:

While I did expect my actual effort to be low in the beginning weeks, I hoped to pick up my work ethic at about week 4 to begin work on the team project. However, I have repeatedly put off beginning work on it and have given only low effort attempts as of week 5 so far. I need to better manage my time by completing more of my work beforehand, specifically for some of the other online courses I am taking this term, so that I can more rigidly focus on the capstone project. I will attempt to do this by setting weekly alarms for myself so that I am always at least 3 days ahead of assignments for other classes (exception of unposted assignments).

Sairishabh Anand – Weekly Work Intensity Summary

Week	Expected Effort	Actual Outcome
Week 1	Medium	Medium
Week 2	Medium	Medium
Week 3	Low	Low
Week 4	Low	High
Week 5	Low	High

Reflection on Expectations vs. Experience

1. Week 1:

- Expected: Medium
- Outcome: Medium
- Reflection: With an expectation of medium effort, the need to finalize the project selection survey and initiate team discussions led to about the same expected workload.

2. Week 2:

- Expected: Medium
- Outcome: Medium
- Reflection: The team began forming and discussing project preferences, leading to a manageable workload and met with the project mentor, which went smoothly. I was not present because I had to go back to Porltand because of an green card final process, but did participate in the introduction document to introduce ourselves to the project mentor.

3. Week 3:

- Expected: Medium
- Outcome: Medium
- Reflection: Finalizing team roles and expectations required more about the same effort , since all the team members were on the same page and that led to a smooth and efficient use of time in dividing the team member roles and activities on the project.

4. Week 4:

- Expected: Low
- Outcome: Medium
- Reflection: Even with other assignments and midterms, the completion of the individual project abstract was definitely more work than imagined, but was still overall a medium workload effort since the abstract did not require high level effort.

5. Week 5:
- Expected: Low
- Outcome: High
- Reflection: The need to work on the team charter while managing other coursework led to an unexpectedly high workload.

Collin Kimball

Weekly Work Intensity

Week	Predicted Effort	Actual Effort	Other Commitments
0	Low	Low	Interviews
1	Low	Low	Interviews
2	Medium	Low	CS450 Project
3	Medium	Low	Interviews, CS450 Project
4	Low	Low	CS450 Project
5	High	Low	CS450 Midterm
6	Low		Interviews, CS450 Project
7	Medium		CS450 Project
8	Medium		
9	N/A		CS450 Project
10	Low		CS372, CS450 Final

Through week 5 I have been putting in relatively low effort on the capstone project compared to what I had expected. I expected my effort to be higher through weeks 2-5, however I have been focusing a majority of my time on other commitments. However, this next half of the term, I expect my effort to increase as I have less commitments that I need to focus on.

Trent Matsumura – Definition of Done:

1. List Completion Criteria: Define specific conditions that must be met for each task or feature to be marked as “done” (e.g., all tests passed, documentation updated, code reviewed).

Since the project is based on completing a project and tool, the completion criteria is based on how functional the final product is. The criteria for the final tool would be a working functional prototype application that can successfully give a user an AI generated delineated assignment. This assignment should have a chatbot assisting the user and allow a second user to see the assignment and receive feedback via AI while being able to manually input their feedback. In summary,

- All AI portions of the project are working and able to produce desired outputs.
- Code produces a tool that can be run off of a sharable link.
- Code has been reviewed and thoroughly tested via test cases.

2. Include Quality Standards: Ensure the criteria cover quality aspects like performance benchmarks, usability, or security.

Main criteria to cover is usability and performance. The app should be easy to use while meeting the heuristics and provide a smooth experience for the user without any hiccups. As for security, it's not that important in this project as the main focus doesn’t involve any use of personal information or vital info that would be harmful if leaked to the user. In summary,

- Bench marks should be that the tool runs smoothly with no faults. No crashes and run time should not be exceptionally long.
- Usability, the UI and layout should look very neat and concise, with double checking from all members.

3. Team Agreement: Discuss the definition with your team and ensure everyone agrees.

The team agreed on the standards presented in the criteria. This was then placed into the agreement summary in the team charter.

---

eod.