CS 461: Senior Software Engineering Project
Project Retrospective

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

Group 28

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

Contents

Contents **2**
1. Introduction **3**
Summary 3
Pre-retrospective 3
2. Team Retrospective **4**
Method – 4Ls Retrospective 4
Execution 5
1) Loved 5
2) Loathed 5
3) Longed for 6
4) Learned 7
3. Takeaways **8**
Successes 8
Challenges 8
Communication 8
Key Goals 8
Practices and Actionable Items 9
Conclusion 10
Appendices **10**
A1. Revisions and Review 10
A2. Version History 10
A3. Citations 10
Activities **11**
Team Activity – Regular Stand-Up Meetings 11
Individual Activities 12
Oliver Zhou – Read ACM Queue Articles 12
Trent Matsumura – Learning Journal 13
Ethan Lu – Read ACM Queue Articles 14
Sairishabh Anand – Book Summary: Software Engineering at Google 14
Colin Kimball –Analyze a FOSS project 15

1. Introduction

Summary

The team retrospective was conducted at the end of the fall term to reflect on the progress we made as a team this term. This activity was crucial in determining our results, successes and failures, and things to continue working on and improving in the coming terms. In this document, we will walk through the process of conducting this team retrospective, note down our process, and our findings. The end result will allow the team to collect our thoughts and strengthen our resolve to continue doing good work on this project.

Pre-retrospective

Before we conducted the retrospective, the team did our own activities to reflect on the past term. This was done firstly through our individual reflective activities, though only in part. The activities had a few of us review different articles and write summaries about how they could change our project, or reflect on how things went week-to-week. Alongside these activities, we had each team member think about a few questions as they prepared for our team retrospective, these being:

- What did you work on this term?
- How much have you completed, versus how much you intended to complete this term?
- How did your individual contributions assist the team for group assignments?
- What successes did you have?
- What challenges did you encounter?

These questions among many helped us prepare for our larger discussion at hand. These questions focus more individually as we wanted all of us to reflect on our own contributions at hand before we discussed how things went as a team. This gave us the time to collect our thoughts as well as the ability to analyze things at a smaller individual level before scaling things up.

---

2. Team Retrospective

Method – 4Ls Retrospective

The Canvas page for the team retrospective gave two suggestions for the framework of running the team retrospective. We decided to go with the 4Ls approach, and sandwiched it between our own analysis. The 4Ls approach is a reflection technique that is used in team retrospectives to evaluate performance and collaboration.

The reason we chose this approach is because this method of retrospective will help refine our workflow as we identify the gaps in our process, and further reinforce the parts that we thought were effective. We also learned that this method helps teams to be more responsive and put higher quality work together.

There are 4 keywords in the 4Ls approach that focus on different key areas. These being:

4Ls Retrospective terms and representations (Atlassian, n.d).

- Loved: What went well during the project? These are the things thatI team members appreciated, enjoyed, or found to be effective.
- Loathed: What made things worse? What was missing or needs improvement? This category identifies gaps, such as missing resources, skills, or support that would have made the sprint more successful.
- Longed for: What do team members wish they'd had during the sprint? This could be anything that would have made the work more efficient or enjoyable, like better tools, clearer requirements, or more time.
- Learned: What new knowledge or skills did the team gain? This could be technical knowledge, better communication methods, or insights into the project or process.

See the appendix A3 for the cited article.

To determine our answers for the retrospective, we conducted things alongside our weekly team meetings. Our weekly team meetings were a teamwork activity that we conducted 1-2 times a week to keep everyone on track. We additionally added the responsibilities of the team retrospective this time around.

Execution

As stated earlier, the team executed the team retrospective over Discord, our digital communication platform of choice. We started by answering the prompts of the team retrospective activity with their questions in mind. Our answers to the 4Ls are listed below:

1) Loved

We compiled a list of a few key points that we thought went well during the project:

- Completing activities punctually
    - The team worked on things before due dates, and everyone pulled their weight accordingly and on time.
- Responsiveness in communications
    - Team members communicated often and quickly when things were needed from each other.
- Planning and Organization
    - The process of documenting our progress for assignments and laying the groundwork for our project requirements, design, etc. were conducted in an organized fashion that had solid results. The team feels comfortable and prepared for the bulk of the development process coming next term.

Then we took a few inputs from specific team members who enjoyed or appreciated certain aspects.

Ethan – “I appreciated that the way we divided the team gave us defined roles and that we could rely on each other. I think that we did well working as a team.”

Sai – “The organization of the project was really good. The team contributed greatly to each portion that they were assigned.”

2) Loathed

The loathed section asks what parts of our project did not go well. Since a lot of the team was happy with how things went this term, there were no major problems that dragged the team down significantly. However, we still compiled a list of our issues to lay the groundwork for future improvements.

Challenges:

- Following agreed structures
    - While the team found its own way to operate, we sometimes deviated from our intentions and things became too free flowing, and while communications were still normal, the team structure itself could be disorganized.
    - This would mean people would be out of their role. It's possible that one of us would be unable to complete a certain task, and would need to be covered by another. That is not a problem by itself, but things got blurry sometimes.

Disorganization *as a team* was our biggest challenge, as things were sometimes not as formal as suggested by the structure of the capstone project process, alongside the activities conducted. That is most likely why we mostly did individual activities week to week, as we could then work on the project in our own times, but we may have missed out on the team cohesion that we could have had.

If things were more consistent and our formal specifications were followed more closely, then the sprint could have gone more smoothly, without any incidents. Our project mentor also has a similar free-flow approach to working on projects, so it may be a benefit for us to use this team structure as an advantage.

3) Longed for

We asked each team member to go around and say what they wanted from the sprint. We compiled our answers below:

Oliver: I wish that we had more experience working together as a team and a cleaner schedule that didn't have other classes in the way because then we could correspond together more often. That way, we could complete more team activities

Ethan: I wish that there was more information and guidance to come up with the designs for parts of our project such as the AI part. Designing the website without help is a big challenge for our team.

Trent: I wish that there were more tools for managing the team documents because Google Drive and Discord can be messy.

Sai: It would be a benefit to us if we had more clear requirements and time to work on things in some aspects because then we could refine our process and document better.

Colin: If we had more detail on assignment, then we would be able to write our documents with more direction. It can be tough writing these documents when things are vague and we are without much help other than our own guesses on how to structure things.

4) Learned

As a team, we learned a plethora of teamwork and individual skills that have been of great benefit to our project. I learned the importance of prioritizing communication and making sure that everyone was on the same page. We learned that things can get messy very quickly as we all work on things at a different pace and misunderstandings can occur. It's important that we resolve any of these obstacles whenever they get identified.

We also learned that having weekly team meetings and regular meetings with our TA and project partner have been great for keeping those communication lines in tact. It made it easier for us to coordinate and assign tasks, and adjust things based on feedback. Our team could be informal at times, but it gave flexibility when requirements changed early on. Embracing that flexibility can be key sometimes, since challenges can be resolved in so many different ways, and we want to make sure that we can move forward comfortably.

We've gained great knowledge in the process of working as a team in a longer project with a formal tone. It's not something that many of our team members have done before, and we find that this experience has been incredibly valuable.

And this concludes the summary of our team retrospective.

---

3. Takeaways

Successes

This term was marked by several different successes from our project. We successfully completed each of the documents on time with inputs from each team member, so the results of our labor were a success for the most part. Team collaboration had its ups and downs, but overall, the team was good at responsive communication. Things were productive and we now have a solid foundation of our project moving forward. We found that things such as role responsibilities and expectations were well established and balanced.

Challenges

There were a few challenges along the way. Through our retrospective, some of the members pointed out points where our process was unrefined or vague, and we highlighted the need for improving our alignment as a team to mitigate these challenges. Despite them, the team could deliver on the project, but there is room for improvement.

Communication

Our communication as stated earlier, was a mix of strengths and weaknesses. The team checkups kept the team to task and updated, but that communication did not always perfectly get through, and our team members sometimes had to be flexible and work on things a little off timeline/expectations. There were some changing priorities and disagreements while building the documents, and activities were not conducted as a team as often as we would have liked. This aspect is a key focus area. Ultimately, the team was able to come together during meetings to overcome any obstacles in our way, and things went mostly smoothly then.

Key Goals

Our key goals for next term is to maintain and reinforce our workflow as we begin proper development on the project. We want to fill in the gaps that appeared where things were messy or uncoordinated/disorganized. Then, we want to better utilize our ability to work flexibly and maximize its potential. Accomplishing these goals will strengthen us as a team and allow us to work as a well-oiled machine.

Practices and Actionable Items

Moving forward, we plan to put into practice a list of actionable items that we put together to help improve the team's performance. Then we will assign an item to each team member.

Item 1) Refine the Team Charter Agreement to Accommodate Workflow (Assignee: Oliver)

Description: We realize that the team charter was not followed as closely as we would have liked. The refinement and enforcement of an improved team charter will improve our communication and performance.

Item 2) Introduce More Comprehensive Check-ins (Assignee: Sai)

The current system of a weekly stand-up is good, but we want to further improve it so it works even better. We will check-in more frequently and compile lists of questions or goals to address in our team meetings so we better understand the direction of the project.

Item 3) Establish a Task Management System (Assignee: Trent)

The team has been looking at utilizing popular tools like Jira for managing tasks in our project. Currently, tasks are mostly assigned through communication, but it would be good to settle in proper storyboards and other management functions. This task involves setting up the environment for use.

Item 4) Schedule and Host Alloted Team Activity Sessions (Assignee: Colin)

Because we want to do more team activity options for assignments, we will make sure to dedicate one person to holding these specific meetings for strengthening our team workflow.

Item 5) Timeline Revisions and Improvements (Assignee: Ethan)

Our timeline needs to be improved because some of the items on the timeline are vague and were not followed by the team closely. There was also feedback given from the project mentor on it. The timeline will have concurrency periods added and refined for more realistic and smaller short-term goals.

Conclusion

With the conclusion of our project retrospective using the 4Ls method, we have identified key areas where we have succeeded on the project, such as delivering and setting up a strong and organized foundation for the proceeding term. We have also identified places where we need to improve, such as filling in communication gaps and revising our timelines and practices, which were not specified to our standards. In the future, we have set up several actionable items that are assigned to each team member to help improve the process of overcoming challenges that we uncovered through this reflection. This exercise was helpful and productive for all members of the team, and we are looking forward to continuing putting out quality work in winter term.

Appendices

A1. Revisions and Review

This document attempts to set us on the right path moving forward and reflecting on what we did well and bad in the fall term. This may be reviewed by our project partner. No revisions are intended to be done on this document in the future, as we will have a different document for the winter and spring retrospectives.

A2. Version History

v1.0.0 (December 1st, 2024) – Initial Draft for Canvas Submission.

- Retrospective Introduction
- Method – 4Ls
- Results and Takeaways

A3. Citations

Atlassian. (n.d.). The “4 Ls” Retrospectives. Atlassian.
https://www.atlassian.com/team-playbook/plays/4-ls-retrospective-technique

Used for understanding the 4Ls retrospective technique.

Activities

Team Activity – Regular Stand-Up Meetings

The team conducted weekly meetings (sometimes online correspondence more often) around the end of the week – usually on Fridays, to discuss various factors of the project. Here is a table of weekly meeting summaries which contributed to all of our working documents, meetings with our mentor, and success as a team.

Occasionally other meetings were conducted on weekends when further clarifications or communications were necessary (for example, for homework submissions that we wanted to talk about)

Week 1	Team initially met up, but no further correspondence was done this week, since things were starting out.
Week 2	Once again, the intention of team stand-up for working on the project was not set up yet. However, there was a meeting with the project mentor conducted this week however, where we got the gist of his intention with the project. We learned about the project deliverables and reworked some of the goals of the project.
Week 3	This week had our first stand-up meeting, though informally. We held a meeting to catch up on everyone's progress on the project abstract and team charter assignments. We also at this point started to conduct the first of bi-weekly meetings with our TA that also semi-serve the purpose of stand-up.
Week 4	We all talked about our progress on the team charter. The team charter was mostly written by our documentation and project lead, but everyone's inputs were taken in for the team agreement. We made sure to hear out everyone's thoughts and how we wanted to handle various things as a team while we progressed through the term.
Week 5	The team conducted a meeting to discuss the midterm week. Due to mostly everyone having obligations for other classes, the progress discussed in this week's meeting had little to show for, understandably as the team was busy. However, we still shortly discussed how to implement requirements specified from our project partner for the upcoming requirements document due date.
Week 6	This was another week of talking about the requirements document. We updated each other on our progress, which sections we were working on and assigned, and finished the document soon after. We also briefly discussed on-coming assignments again, that being the design document.

Week 7	This week, we focused on updating each other about the design document. Every developer role gave input on how they were thinking about implementing their specific portions of the project. We also did a check-in with each other again because we were getting far into the term, and wanted to make sure everyone was on-track.
Week 8	We discussed the terms of the upcoming demo and the project retrospective this week. The team updated their progress on the slides, and we came to the conclusion that while the presentation is finished, we still have work to do on the demo itself before we are happy with it. Similarly, the team also started to think about how the term went, as the project retrospective was coming up shortly.

Individual Activities

Oliver Zhou – Read ACM Queue Articles

I read the article: “Toward Effective AI Support for Developers” for this activity. This article was written by Mansi Khemka and Brian Houck, and the gist of the article is that it discusses how AI can best support software developers by addressing their needs and concerns. Based on a survey of nearly 800 developers, the study identifies key areas where AI can make a meaningful impact, as well as the primary reservations developers have about AI integration.

While our project doesn’t specifically cater to developers, as that isn’t our target audience, I think there is still a lot to be learned and applied to our project. Specifically, I think that our project similarly aims to tailor an AI product to assist a group of people – though for universities in our instance.

Developers in this article have laid out a few key uses that they want from AI, which is to automate routine tasks, streamline duties, and enhance their well-being. I think that if a similar study was conducted on other demographics such as university professors, they would come to similar conclusions. In our case, professors likely would appreciate it if getting summaries of their students’ performance was streamlined. For that reason, this article helps us determine a methodology for targeting our audiences needs.

Trent Matsumura – Learning Journal

Reflective Activity

Week 1: This week was the introduction to the project and nothing had really happened yet except forming teams. Challenges that were involved included getting to know my teammates and gathering contact information which wasn't something that was very new to me, so this week progressed with little to no challenge.

Week 2: We met with the project partner which really amplified the seriousness of this project in my eyes as it felt more like an actual project instead of a school assignment. The meeting involved the scope and task of the project as well as molding what requirements we thought were acceptable for this class. The challenges involved were definitely trying to understand what the project partner wanted exactly from us as well as integrating the project to fit the rubric of the class.

Week 3: This week was when the documents and planning officially started and when the meetings with the TA began. Our first meeting with the TA was pretty smooth and eventless, since then we have just been giving updates mainly through discord as that was the main way we were communicating. This was also the start of the team charter assignment that allowed us to solidify our roles on the project.

Week 4: This week was mainly focused on the team charter assignment as we worked through the document making sure to really think about what plan we had for the project.

Week 5: This week was a bit of a grace period where I put my attention to my other classes to finish assignments and midterms as the capstone load was light this week. We got ready to work on the requirements document and meet with our project mentor to review the document as well.

Week 6: The requirements document was completed and was a hurdle. It required a lot of typing and discussing the project. However despite the length of the assignment, it did not cause too many issues.

Week 7: This week was the design document and some troubles arose throughout the completion of the document.Some confusion around the design of the website and ui were present. We hadn't yet decided on a layout yet so the implementation of the design hadn't been completed yet. However after communicating a bit, we had resolved this issue and completed the requirements document.

Week 8: This week was when we worked on the demo slides. We haven't yet completed the demo slides with the amount of care and precision we hoped for, so in the future we will continue working on it until we have to present.

Ethan Lu – Read ACM Queue Articles

The article I read was “GPTS and Hallucination.” The primary topic covered in this article relates to the reason why LLMs occasionally seem to make up false information about certain topics. The primary reason for this has to do with crowdsourcing. Many LLMs are built on models that fit crowdsourcing, which is drawing common knowledge from many sources. Because of this, LLMs have an easy time correctly assessing common knowledge, but struggle when there aren’t clear answers within their training data. This is relevant to our topic as we are attempting to utilize LLMs within our project as an aid to students by supplementing them with academic guidance. However, due to the small nature of our project, it would likely be difficult to crowdsource enough accurate information for our LLM to be reliable on very specific academic topics, not to mention there might be topics covered within a teacher’s assignment our LLM would not be trained on. If we were to use an LLM without proper training, it would hallucinate answers and mislead students. As such, I think it is important for our LLM to be trained not necessarily to answer questions exactly, but rather guide students towards the proper areas and resources provided by the teachers. For instance, if a teacher’s assignment requires reading through a large body of text to solve a niche problem, rather than trying to train our LLM to solve it as well, we would want to train it to be able to identify where in the text to point a student towards.

Waldo, Jim, and Soline Boussard. “GPTs and Hallucination.” ACM Queue, queue.acm.org/detail.cfm?id=3688007. Accessed 25 Nov. 2024.

Sairishabh Anand – ***Book Summary: Software Engineering at Google***

Individual Activity

> *Software Engineering at Google* provides valuable insights into how to build and manage software projects effectively, focusing on scalability, collaboration, and long-term success. One of the key takeaways is the emphasis on fostering a strong team culture, where open communication and psychological safety allow everyone to contribute and address challenges effectively—something we can implement to improve teamwork in our project. The book also highlights the importance of code reviews, not just for catching bugs but for sharing knowledge and maintaining consistent quality, which we should adopt as a standard practice. Another critical point is Google’s focus on automated testing to save time and ensure stability as projects grow, which is directly applicable to our AI Canvas tool. Finally, their approach to scalable design, such as using modular and well-documented code, aligns perfectly with our need to build a system that can evolve easily. These lessons offer a clear roadmap for improving both the process and outcome of our project.

Colin Kimball –Analyze a FOSS project

Project:
https://github.com/spacedriveapp/spacedrive

Summary:

- The project utilizes releases with semantic versioning to report new features, updates, bug fixes etc. We can utilize this in our project to rollback the application to a previous stable version in case the current release introduces issues or fails to meet requirements.
- People are required to make pull requests before merging their code into the main branch. Additionally, reviewers must approve the code before it can be merged. We can use this in our project to ensure code quality and consistency, and prevent bugs from being introduced into the main branch.
- Code must pass all CI/CD tests to ensure reliability and functionality before being deployed. We can use this in our project to maintain application stability and prevent issues in production.
- Users/contributors can report issues/bugs.

---

eod.