CIS 4250 – Software Design V
Instructor: Prof. S. Scott

Project Post-Mortem

Ike Agbaje

Group #6

1125093

Sprint 4: Project Post-Mortem

Q1. What were the main technical or methodological knowledge, skills and/or abilities (KSAs) that were required to complete this team deliverable? What prior courses or experiences (e.g. co-op, group project, etc.) from your Software Engineering degree did you draw on for these KSAs? (bulleted list is preferred):
On the backend, I modified the MongoDB schema, drawing on database management skills from CIS*3530. This involved ensuring messages were correctly flagged for deletion and integrating Mongoose schema updates to support disappearing messages. On the frontend, I applied my knowledge in React, JSX, and TailwindCSS to structure and style components while ensuring seamless UI updates. I also worked with Socket.io to handle real-time message deletion, reinforcing my ability to use event-driven programming effectively. Additionally, I applied logic and conditional reasoning skills gained from CIS*1910, particularly in determining whether a message should be displayed or removed based on timers and state conditions. These logic-based decisions ensured that messages disappeared at the correct time.

Q2. What was your existing level of experience with these topics/skills before your team began working on this deliverable? (1-2 sentences):
I utilized my understanding of passing functions and parameters between JSX components, managing React state, and leveraging event listeners to handle real-time updates efficiently. For all 3 things I drew on my experience from sprint 3 but I didn't have much experience with them before that. I did have some beginner experience with databases before this project, and over the course of the first 3 sprints, gained more experience in modifying database schemas, particularly mongoose and MongoDB

Q3. Comment on your individual KSAs learning during this deliverable, and what additional learning may be needed to understand or be more competent with these topics / tasks in the future?

During this deliverable, I strengthened my skills in React state management, real-time event handling with Socket.io, and MongoDB schema modifications, allowing me to implement disappearing messages effectively. I also deepened my understanding of frontend-backend integration, ensuring that data updates smoothly across both ends. However, I would benefit from further learning in optimizing real-time performance and handling edge cases in event-driven applications, such as ensuring message deletions are always synchronized across all clients. In the future, improving my knowledge of advanced WebSockets, database indexing for efficiency, and React performance optimization would make me more competent in handling similar tasks.

Q4. What specific contributions did you make to this team deliverable? This should include technical or project management contributions.

- I implemented the disappearing messages feature by modifying the message schema in MongoDB and updating the backend logic to automatically delete messages after a set time.
- I updated the frontend UI to include a disappearing message toggle button, ensuring users can select whether a message should self-delete.
- I updated the Messages component to dynamically remove messages from the UI when they are deleted from the database.
- I modified the sendMessages function to include a flag for disappearing messages and ensured it was properly stored and retrieved.
- I tested the feature thoroughly, ensuring messages delete at the correct time and that UI updates happen in real time.

Q5. With whom did you collaborate for any of the above contributions (be specific – saying "all team members" is not sufficient. State which parts you worked on with whom)?

I collaborate with Naza on the disappearing messages feature; she worked on some of backend the stuff such as creating a JS function, then I took over to implement the rest of the backend and the frontend .

Q6. Comment on how well you managed your time over the time period allocated in the Course timetable to this team deliverable (i.e. the time between the prior team deliverable to this team deliverable).

We managed our time well overall, though not as efficiently as in previous sprints. Some tasks, like the disappearing messages feature, were left until the last minute. While we successfully completed all the user stories we started, we weren't able to finish all the remaining user stories we had left during this sprint due to time constraints.

Q7. Comment on what you believe to be your main contributions to the development of the project outcomes.

- One of my biggest contributions was the complete UI redesign and the reply-to-message feature. Since our updated application is meant to be a modern platform for globally distributed teams, the old UI didn't quite align with that vision. No matter how many features we added, the app wouldn't fully serve its purpose if it didn't look and feel the part. The reply-to-message feature was also a key addition - not only because it's an essential tool for clear communication, but also because it taught me many of the new skills I later applied in Sprint 4. Having used various messaging platforms myself, I know that the ability to reply directly to messages is one of the most effective features for maintaining clarity in conversations.
- The "disappearing message" feature I contributed to works well, but there's definitely room for improvement. One key enhancement would be giving users more control over how long their messages last, rather than having a fixed duration. Adding options like 5 minutes, 1 hour, 8 hours, or 24 hours would make the feature much more flexible and useful. Additionally, right now, messages only disappear after a page reload - ideally, they should vanish automatically as soon as the timer runs out for a smoother user experience.

Q8. Comment on your personal technical skill development throughout CIS 4250 that was beneficial to the team project.

- During this sprint, I strengthened my skills in React state management, event handling, and working with real-time updates. Implementing the disappearing message feature helped me improve my ability to manage state dynamically and execute tasks in parallel using event listeners. Additionally, I expanded my knowledge of MongoDB by modifying the message schema and handling updates efficiently on both the frontend and backend.
- At the start of the project, I had a basic understanding of React, JSX, and database management but had limited experience applying them in a real-world project. Now, I am much more comfortable with structuring React components, passing functions between JSX files, and using TailwindCSS to improve UI design. On the backend, I have improved my ability to modify schemas and handle real-time updates effectively.
- I am assessing my current skill level based on the complexity of tasks I can handle independently and the efficiency with which I complete them. At the beginning of

the project, tasks like implementing message replies or dynamically updating the UI would have taken much longer and required more troubleshooting. Now, I can complete these tasks with greater confidence and fewer errors, which indicates significant improvement.

Q9. Comment on your contributions as a member of the design team, in terms of your communication skills, time management skills, and leadership skills.

- For most of our written deliverables earlier in the semester, we divided the work by assigning each team member a section to complete. My contributions included writing my assigned portions and running a spell and grammar check on the final document to ensure clarity and professionalism. Additionally, I worked on the wiki page for some of the features I contributed to, providing written documentation to explain the recent updates and how they function.
- I helped make some of the slides for our sprint demo presentations, and most times tested the main branch the night before the demo to make sure all the new features for that sprint worked well together after merging.
- In all our meetings, I shared updates on the tasks I was working on, including my progress and what still needed to be completed. This helped maintain clear communication within the team, as we all kept each other informed. Additionally, I reached out to the team as needed, providing updates or asking questions whenever I encountered something that was blocking my progress.
- For most sprints (except sprint 4), I made sure to complete all my tasks before our final meeting, allowing enough time to merge our code, resolve conflicts, and test the new features. I managed my time well in this regard. In sprint 4, I ran into a persistent error that put me slightly behind by our last meeting. However, I was able to resolve it, merge my work, and test everything that night, ensuring it was ready in time for our demo.
- We each took turns being agile coach so everyone could contribute to the coordination of the project. I was agile coach for sprint 3, so I confirmed times from everyone for our meetings, let the meetings took notes during them, assigned issues on GitLab, worked on the wiki for that sprint, and coordinated putting together our demo PowerPoint for that sprint.