Report

Summary of Project Goals:

The goal was to create a service where people can message each other or groups effectively and securely. Users should be able to retrieve or initiate conversations easily, have information about their communications partners at their fingertips, find people, join groups, and create, update, and delete groups. The intended audience is individuals who work in a team environment or groups who need to manage projects, events, or work in a team environment. The messaging service will allow individuals to create an account and login into the system to securely message other users or groups using the platform. The objective of the software is to provide users another messaging alternative that focuses on privacy and security of instant messages. In all, the goal is to provide efficient and secure communication.

In order for us to achieve the goal, the service required us to focus on Users and Groups, Communication, and the Government. The User and Groups would allow for connections between users to users, groups to users, and users to groups. Security was important for service therefore, implementing message encryption was vital to the success of the project goals. For additional security in the application, Groups have the opportunity to enable secure password and hide themselves from the public eye. In others words, Users who were invited into groups would require a password to enter or engage in the group.

In regards to Communication, each channel was secured end-to-end. Users would be able to send a message in which in the process would be encrypted and later decrypted for the other USer to read and respond, if so chose to. Communication between parties did not only occur through direct messaging but also through invitation, notification, deletion/removal, and status updates. Each communication avenue was encrypted and then decrypted for the receiver.

Finally, in regards to the Government, though we strive for security, we kept concerned about abiding by FCC and Homeland Securities rules and regulations. In other words, we created an avenue for situations in which the Government may have a person of interest that is using our services. This avenue, channel, allows for the Government to watch their person of interest's activities and communication.

Overview of the Result:

The result of this project led to the completion of the services main functionality, or in other words, functional requirements. This project consisted of eighteen (18) functional requirements and thirty-two (32) non-functional requirements. The functional requirements were broken down into four (4) groups - Users/Groups, Communication, Government, and Other. Additionally, the requirements were prioritized using low, medium, high, and critical metrics. Each eighteen requirements were successfully implemented and tested which resulted in a 100% functional requirement backlog completion and coverage which includes statement, conditional, and branching reaching above ninety-percent (90%).

In regards to non-functional requirements, there is a remaining of fourteen (14) requirements in the non-functional requirement backlogs. With fourteen requirements remaining in the backlog, the backlog statics is approximately sixty-percent (60%) - seventy-percent (70%) complete. Within the non-functional requirements that have been completed, there has been coverage which includes statement, conditional, and branching above eighty-five-percent (85%).

Overall, the application has reached above ninety-percent (90%) in code coverage and the team has delivered approximately eighty-five-percent (85%) of what was promised. Additionally, the team was able to deliver extra features or stretch goals that makes the application easier to navigate and information to be better processed.

Team Development Process:

Sprint 01:

During Sprint 01, the team made many efforts in meeting to discuss team logistics such as communication platform, areas of programming strengths and weaknesses, and scheduling. Additionally, the team discussed the best tools and technologies that will allow for the success of the project. These meetings consisted of the deliverables of getting the team configured to the development process.

Towards the end of the Sprint, the team was able to finalized on a meeting schedule, a technology plan, Unified Modeling Language (UML) diagram about the project, and project requirements which included functional and non-functional requirements.

At the end of the Sprint, the team was able to complete all of their deliverables. During our retrospection, the team identified that in-person team meetings work best for us compared to virtual meetings. The team also identified the strengths and weaknesses of each member. This allowed for the team to plan and adjust accordingly. In contrast, the team identified what didn't work for them. Virtual meetings were not as effective in the project development process.

Sprint 02:

During Sprint 02, the team started off well in the development of the project but suffered mid-way and towards the end. Mid-way of the Sprint, the team went through a transition from in-person meetings to completely virtual meetings. This was due to external factors however, it has taken a critical impact towards the team dynamic and development process.

Nonetheless, during Sprint 02 and the transition, though the team adapted slowly to virtual meetings, the team was able to manage and consistently keep virtual meetings three to four times a week in order to advance the project. The basics of the project such as the login, register, account page were developed using basic HTML, CSS, and JavaScript. Information was temporarily stored within a Set instead of a database. Testing of the application services were done and reached above ninety-percent coverage. However, though the team made much effort to regain the lost time during the external event, the team suffered in failing to keep their deliverables for the Sprint.

During the Sprint, the team had many impediments and issues with communication which prohibited the ability to deliver the deliverables. The team did not have the database integrated, logging in and logging out of the service posed an issue for the user, and when it came time to push the code into master in GitHub, the team failed the Jenkins tests which took time from implementing additional deliverables and fixing other development issues.

In the end of the Sprint, the team decided to enhance their communication channels and prioritize writing clean code to pass the Jenkins tests. Moving forward, the team stressed on implementing and testing persistence and correcting the User Interface.

Sprint 03:

During Sprint 03, the team was in a better position than the previous Sprint. However, in this Sprint, the team encountered some external factors that led two of the team members unavailable to work on the project development. With that said, the other team members took up additional tasks and worked on completing them. Communication within the team has increased which allowed for adjustment when two team members were unavailable.

The team was able to deliver a majority of their deliverables. Deliverables that were not completed in the previous Sprint were completed and issues that prohibited a successful demo during last Sprint were fixed. However, one issue in which we exhausted our time was coverage. Due to a lack of coverage and poor time management, the team suffered in merging the code on time into master in Github.

As a result, the team decided on adding Jenkins test and requirement on another branch to lessen the chances of the team encountering such a problem again.

Sprint 04:

During Spring 04, the team made much progress compared to the last two Sprints. The team was able to complete all of its deliverables and pushed for stretch goals that focused on making the service better. The issue of getting all the code onto master in Github last Sprint was no longer an issue with the work around of adding Jenkins requirement onto another branch within our repo. Many user stories were implemented, persistence was functionable and tested, and coverage was above ninety percent (90%).

Overall, this Sprint was successful with minimal impediments around persistence within users, groups, and messages and the User Interface.

Retrospective of the Project:

Throughout the project, the team encountered many challenges that, at times, prevented the successful completion of a Sprint. Some of the challenges were due to external factors while others were due to lack of communications, management, and teamwork. However, during each Sprint, the team grew in confidence and learned from previous mistakes. As a result, the team identified what worked best and worse, what was liked or disliked, what was learned, and what needed to change in the course of the project to support a greater experience.

Throughout the project, the team learned that in-person meetings were more successful and meaningful compared to virtual meetings that took place on Google Meet or Slack messaging. However, due to external factors, the team learned how to adjust and made virtual meetings a better and more meaningful experience. The team liked focusing as a team best. In other words, the team was focused on working as a team and being flexible and agile for other members of the team. The team kept members accountable and, when needed, provided help to members that were stuck on a problem for several days. Each member of the team was eager to learn and were available to pick up additional tasks when others fell short. This act of working as a team took several weeks to do well. This was due because of lack of project management and team communication in the beginning of the project.

Nonetheless, though this was a struggle for the team to overcome, in due time, the team was able to improve on their communication and project management. Each Sprint provided an opportunity for the team to reflect on previous errors. Through those reflections, the team was able to excel through the following Sprints. In those reflections, the team learned that checking in on team members' progress is helpful for the member and project advancement. The team learned that emphasizing and placing strict measures for clean code is important for the project's success and team communication. Finally, the team learned that by taking additional time in the beginning to integrate tools and technology can help make project management better and engage team communication.

In the course, the team learned a lot about working as a team, working on a project, and the foundation of software engineering. The knowledge gained from the course has given each member of the team an opportunity to be a software engineer. Though many have had the opportunity to work as a

software engineer, this course gave to some individuals what companies sometimes fail to give due to time and other factors. With that said, the team believes that the course provides a great experience to working as a software engineer and, that if something must change, it would be having weekly check-ins with each team by the TAs to ensure that each team is on the path to success and is not waiting time with other factors that are not as important at that moment.