2207-BSE

Implementation – Self Reflection

CS106.2

Alexander Legner

Document Outline

Table of Contents

Table of Figures

No table of figures

Table of Tables

No table of figures

Intentionally Blank

Self-Reflection Report

Introduction

This project was for proposing and developing a ticketing system product. This was split into two parts: Project Proposal and Project Implementation.

The overall goal for this project was *"To create a system which allows users to register a ticket categorized as support or incident, then gather the details including but not limited to category, urgency, user details and tags. The system will then issue the ticket with a ticket number and forward all information to a designated group or agent for resolution. At any point a ticket should be updateable and allow users to track its status using the associated ticket number. Users should be able to search for specific incident details and provide formative feedback for the resolution of a submitted ticket."*

My role within the project was as a backend coordinator where I was responsible for the development of the back-end related code which include its design, documentation, and merge conflicts. My other primary role within the project was also as a product analyst where I was responsible for the research, user testing and feedback for the prototype and product. I was also responsible for ensuring the product is created with satisfaction of end users and clients that will be using the product through extensive research and user testing. Apart from the primary roles, I was also helping with many other shared tasks due to the small size of the team. The main responsibilities were there to help streamline who was responsible for what, however we all helped with parts of the project where needed.

Went Well

Things that went well in this project that I have identified are:

- Time management
- Prototype development
- Initial Start (back end)
- Collaboration
- Bug fixing

Time management – As a group we set out milestones at the beginning of the project. We met all the milestones and used out spare time to either start working ahead or further refining the product. We all stuck to the timeline and worked together to keep on track. Having a good team is particularly important and will make project run smoothly and I felt like the team worked together effectively and efficiently to do this.

Prototype development – The prototype development consisted of initial sketches and ideas, which fed into the LoFi wireframe and finally the HiFi prototype. The initial sketches worked well for the entire team as it allowed us to express our ideas to each other in a constructive way. We then all came together with our ideas and refined them into a single LoFi wireframe which I believe stopped any conflicts in design and helped the overall development of the prototype.

Initial start – When we first started the implementation, I created a main window which had the ability to display subpages in the main window. This followed our prototype design; however, it also created a solid foundation for all of us to start developing. Within hours of setting everyone up with this foundation, we were all working on separate parts of the project, and we were doing it effectively because of this initial foundation. Everyone followed the standards and layout set out in the foundation created which kept the project uniform and easy for everyone to follow.

Collaboration – Throughout the project we all worked exceptionally well together. From communication to getting work done, everyone was fun and good to work with. Having a good team is important, especially for large projects where you need to rely on another. We also had good collaboration tools which we dedicated time to setting up in the proposal phase of the project. These tools make working together much easier as they speed up workflows and keep everyone up to date and on track on the project.

Bug fixing – During the project we encountered bugs while trying to assess a feature we implemented during a development phase. We had a good system for dealing with these bugs where we would record the bug + details and then everybody would discuss why the issue was occurring and how to fix it. This approach worked well as usually it came back to me, and I had the experience to debug and solve the issue quickly while keeping on track of development. This was also a good thing as I was primarily responsible for most backend related code.

Not So Well

Challenges that did not go well in this project that I have identified are:

- XAML understanding
- SQL integration
- Initial GitHub setup
- Waiting

XAML understanding – None of us had enough experience when we started to understand XAML or WPF to create a working application. We had allocated time for learning this however some parts took longer than we expected. For example, we tried to re colour a dropdown box, however the syntax was complicated, and we underestimated how long it would take to do a simple task like this. This gave us setbacks as instead of 2-3 hours, it took 6+ hours to figure this out through research and trial and error. Once we had overcome the syntax understanding however, the project went right back on schedule as we had all worked together to keep us on track.

SQL integration – This was more of a personal experience where I could not get SQL working on my device at home. However, it would work on my device where we developed it together. This was due to the different OS and environment and so I was restricted to only develop the program when I was onsite. This did not become and issue and was more of a personal issue. All of us understood how the SQL integration worked, it just was not working for development outside of the intended devices.

Initial GitHub setup – When we first started, we were having issues where temporary files were uploaded, which created merge errors during every merge. We fixed this near the start of development by backing up and deleting the temporary files from the project which made merging in the future parts of development easier to do.

Waiting – Sometimes the time we were waiting on each other to complete a task so that we could start the next one. This was not due to planning but due to being ahead of schedule. For example, I would finish a task early and then having to wait for someone to create a button on the front end before I could start on the backend of that button. We decided to make sure we were not working on the same exact part of the project at the same time so that we did not create merge conflicts that were unnecessary and only required us to wait. This was not a huge issue, and we were on and ahead of schedule anyway.

Implementation

Lessons learned:

Key takeaways that I have learnt from the project experience are:

- Planning
- Team
- Initial Setup

Planning – The planning in this project is a large part as to why the project was completed on time and to a satisfactory level. I will take this away with me to future projects where planning ahead is key to a good project. This included detailed timelines, responsibilities, tools and most importantly goals. The planning allowed the team to work more effectively together as we knew what to do, when to do it by and what we can use to get it done.

Team – Having a good team is extremely important because it can really impact the projects quality, timeline, and collaboration. Although sometimes you do not get to pick your team, this project showed me how an excellent team progress through a project to together. We were constantly helping each other and working together for the end goal. I will take away key aspects of this teamwork including helping others out, solving problems collaboratively, and keeping the team informed as to what is currently happening so that you are not working on the same thing accidently.

Initial setup – We had a good foundation when we started the implementation of the project. This was due to starting with a simple workflow and style of the project. With the foundation, everyone was able to add and edit parts of the project with ease because the foundation was easy for everyone to follow. For my next project I will create a solid foundation before working on other parts of the project as the foundation will make any project better by improving workflow, collaboration, and uniformity for a project.

I also developed my programming skills for WPF/XAML and C# which the project was coded in.

Implementation

Conclusion:

Summarize your overall experience and key takeaways from the project.

Reflect on the significance of the project in relation to your personal and professional growth.

Discuss any future actions or improvements you plan to implement based on this experience.

Overall, I have key takeaways for what to do for future projects:

- Have a good plan
- Work as a team
- Create a solid foundation/start for the project
- Setup the tools correctly when starting the development
- Always keep to the timeline with your team
- Work together to create the prototype together
- Always identify bugs and discuss them as a team

This project has allowed me to develop my own teamwork and programming skills. It has also allowed me to understand how to take on large problems in software development by breaking them down into smaller ones that can be solved collaboratively with multiple ways to solve the same problem.

Next time I will find a superb team like the one I worked with for this project in my opinion the team is the most important part of the project and the project's success.