Reflection and Traceability Report on SyncMaster

Team 15, SyncMaster
Kyle D'Souza
Mitchell Hynes
Richard Fan
Akshit Gulia
Rafeed Iqbal

1 Changes in Response to Feedback

Throughout the capstone project, many items of feedback were received and use to improve the quality of the application. The tables in the following section itemize each item of feedback and describes the pull request or issue in which it was addressed. Github issues proved to be an immensly useful tool for tracking and responding to feedback here, along with enabling discussions within the team.

1.1 SRS and Hazard Analysis

TA	Task - TA SRS Feedback - Prioritize requirements	Provided a priority of requirements which was then used to guide the development as we first implemented the high priority requirements. Merged in PR #429	#217
TA	Task - TA SRS Feedback - Add assumption City practices don't change and add traceability visualization	Added traceability matrix as suggested in PR #284	#216
TA	Task - TA SRS Feedback - Elaborate on integrity in requirements	Elaborated on requirements such as integrity per feedback in PR #284	#215

TA	Task - TA SRS Feedback - Cite Volere template	Cited the Volere template in SRS per feedback in PR #284	#214
TA	Task - TA Problem Statement Feedback - Goal Content	Demoted goals as documented on github in PR #474	#156
Peer Review	Peer Review - Dates in Roadmap	Added a specific date for implementation in PR #429	#195
Peer Review	Peer Review - Security Requirement in FMEA Table	Added traceability to SR-AR1 for identified failure mode. Some functional requirements are there because they are traceable to the safety risk, they don't necessarily need to be in the safety requirement section. A new safety requirement was identified during the FMEA process and was added to the specific requirement section. Closed in PR #429	#194
Peer Review	Peer Review - FMEA Table Missing Notification Design function	The project requirements have since evolved and the notification system will be a front-end page which queries the logs gathered from the already identified design functions in the table. This particular module would not inhibit the use of the system, loss of data, unauthorized access, or harm to users, so will not include as a hazard of its own. Closed without PR	#192
Peer Review	Peer Review - Conflict between User Authentication and FMEA Table	Elaborated on how AWS Cognito is the application interfaced with in PR #429	#177
Peer Review	Peer Review - Scope not clearly defined	Added statement that the scope covers the outlined system components in PR #429	#175
Peer Review	Peer Review - Lack of Clarity For Potential Losses within Scope and Purpose	Added justification for why specific losses would occur in PR #429	#174

Peer Review	Peer Review - Unclear Goal Description	Immediately directing the user to the problem statement where the project is described in much more detail was intentional to minimize repeating of information in more than one document. It makes maintaining the documents easier as when an element of the project changes there are fewer places it will need updating. Closed wihout PR	#141
Peer Review	Peer Review - Rationale for Performance Requirements	Elaborated on choice of numbers per feedback in PR #284	#140
Peer Review	Peer Review - Missing Traceble Requirements	Duplicate feedback with issue #216, refer to that issue	#139
Peer Review	Peer Review - Longevity Requirements (What not How)	Included suggested change in PR #284	#138
Peer Review	Peer Review - Phase In Plan	Completing the functional requirements are the main objective, and user testing will help to establish the non-functional requirements. Meeting with the City to be arranged in January. Included another point in the Project Planning Subsection. Closed in PR #284	#137
Peer Review	Peer Review - Verifiability of Speed and Latency Requirements	PR-SL2 removed previously due to elimination of SharePoint integration from scope of work. Added verbiage to identify ordinary application use cases for SL1 and SL3. Closed in PR #284	#136

Peer Review	Peer Review - Missing Terms in Glossary	Added the suggested acronyms for GPS, MySDS, and Sharepoint. The various users are defined in detail in section 2 of the document so excluded that term from the glossary. Closed in PR #284	#135

1.2 Design and Design Documentation

Peer Review	Peer Review - [Design Doc] Module Uses	The team has decided not to put any AWS services/SDKs in the module guide as it was determined that it serves no purpose. The documentation for these services are freely available and it would probably be worse for us to write this documentation as we are not the designers and developers for these services. To address the issue, we have provided links in the appendix of the MIS to link the reader to such relevant documentation. Closed in PR #607	#358
Peer Review	Peer Review - [Design Doc] Lack of Cross-refrencing	Fixed by adding hyperlinks to module sections in PR #607	#359
Peer Review	Peer Review - [Design Doc] Diagram Clarity	Diagram has been reorganized in PR #607	#356
Peer Review	Peer Review - [Design Doc] Module Decomp Connections	The module decomposition is now organized in a hierarchical manner. Addressed this issue and the connection between modules is now clear pertaining to the different layers of abstractions in PR #607	#355

1.3 VnV Plan and Report

Peer Review	Peer Review - [VnV Plan] Inconsistency in Non-Functional Requirement TC-EU-2	Removed the undo statement in PR #474	#262
Peer Review	Peer Review - [VnV Plan] Explicitly State Individual Team Member(s) Roles	Outlining in the development plan, team member leading reviews and discussions varied depending on the subject at hand. Closed in PR #474	#261
Peer Review	Peer Review - [VnV Plan] More Diverse Team Roles	Elaborated on team roles in the development plan in PR #474	#260
Peer Review	Peer Review - [VnV Plan] Incorrect Initial states of several Non-Functional Requirements	Corrected the issue as described in PR #429	#259
Peer Review	Peer Review - [VnV Plan] Team role table is in the incorrect section	Corrected the location of the team role table in PR #429	#258
Peer Review	Peer Review - [VnV Plan] Design Verification Plan Inclusion	Added the noted idea into PR #429	#257
Peer Review	Peer Review - [VNV Report] Trace to Requirements	Trace to modules already exists, closed in PR #601	#487
Peer Review	Peer Review - [VNV Report] Trace to Requirement	Added links per suggestion in PR #601	#486
Peer Review	Peer Review - [VNV Report] Functional Requirement Evaluation	Added pass and fail details in PR #601	#485
Peer Review	Peer Review - [VNV Report] Usability	Phase 1 and Phase 2 are not two phases of the same test, rather the testing of each of the two portals. Closed in PR #601	#484

Peer Review	Peer Review - [VNV Report] Nonfunctional Requirement Evaluation	Trace to SRS provided in document, closed in PR #601	#483
Peer Review	Peer Review - [VNV Report] Non-Functional Tests	Added pass information in PR #601	#482

2 Challenge Level and Extras

2.1 Challenge Level

The challenge level for this project was a general challenge level.

2.2 Extras

The extras for this project are a user manual and a usability testing report. They can be found within the Extras folder in the GitHub repository here.

3 Design Iteration (LO11 (PrototypeIterate))

The design of the SyncMaster application evolved repeatedly over the course of its development. The primary driver of this evolution was how the initial requirements changed after meetings with the stakeholder and constraints and challenges that were not originally foreseen came to be known.

First, one of the original ideas in the project was to integrate the file syncing aspect of the application directly into the city's SharePoint. This would have had the benefit of enabling the stakeholder to use an existing tool which they were familiar with. However, discussing this with the stakeholder allowed for the discovery that it was against the City policy to connect any external application into the City network. This was an important discovery which shaped the first iteration of the design.

Another factor which evolved was changing from tracking jobs which are performed to only tracking specific site visits. This came about when demonstrating our rev0 prototype to the stakeholder, and discussing many complexities which arise from trying to track a job across many visits, and how this introduces feature creep as the project evolves into a work order system which was never the original intention of the application.

Finally, there was also discussion after rev0 and before rev1 about the idea

of there being a visitor account in order to reduce the administrative burden which comes with account management. We produced flowcharts illustrating the concept for the stakeholder, but came to realize that a visitor account would enable any member of the public access into the system. In discussion with them, we evolved the requirements to only allow authorized users into the system, ultimatly discarding the visitor account.

4 Design Decisions (LO12)

Limitations and Constraints:

One limitation we encountered was that we could not track the GPS location of contractors in real time, as they would not have their phone open to the app at all times when they are on site. This means, we could not automatically detect when they leave a site automatically. To get around this, our application records an entry time and exit time. The entry time is obtained during the QR code scan, and the exit time is indicated when the contractor uses the end visit button in the application.

Assumptions:

An assumption we are using is that this application will not be used frequently at night, as that is not normal working hours.
This allowed us not to be concerned about the light levels and how that affects the QR code, and also to use AWS Lambda because it can scale to 0 when not in use.

5 Economic Considerations (LO23)

This application is tailor made specifically for the City of Hamilton, with no intention on marketing it to other users at this time. There is certainly a demand for an application of this nature due to the current challenges faced without the existance of an application with the specific features included in SyncMaster. One attraction of this application is its low cost of maintenance. The below table outlines the projected cloud computing costs of maintaining the application.

File attachments and documents	5 GB free storage, then $0.023 per GB per month
Database	25 GB free storage, then $0.25 per GB per month
Compute/API	Can handle 1 million requests per month for free, not expected to exceed
Users	10000 monthly users free per month, not expected to exceed
Domain name	expected to be $20 per year for a domain name

If the City decides to use this application going forward, they would be guided by their internal procurement policies. Likely, this will involve the tendering of a service contract to procure services for the maintenance of the application. Bids for these contracts can vary widely in price, but estimating it to cost a few thousand dollars for occasional maintenance and upkeep is a safe budget.

6 Reflection on Project Management (LO24)

6.1 How Does Your Project Management Compare to Your Development Plan

In general, many of the items in the development plan stayed consistent throughout the project. We had frequent team meetings to keep on track of progress near deadlines and used the GitHub issue tracker to track items, so our team communication plan was effective. Team member roles were generally consistent, assigned to member based on their skillsets and the team had a good understanding of how tasks should best be delegated to maximize the teams productivity and ensure an even distribution of work.

6.2 What Went Well?

The use of the GitHub issue tracker and communication over discord was implemented successfully by our team to maintain regular updates and team communication. Items were delegated effectively to each team member based on their role and this was adhered to throughout the duration of the project.

6.3 What Went Wrong?

During some deliverables, we had a tendency to devote most of our development efforts close to the deadline, which didn't give us as much time to extensively test as we may have wanted.

6.4 What Would you Do Differently Next Time?

Next time, we would want to continuously improve on how proactive we are on some items, so that we a consistent distribution of work rather than a large number of hours closer to the deadline.

7 Reflection on Capstone

7.1 Which Courses Were Relevant

The following courses were relevant to the project:

- **SFWRENG 2AA4:** Software Design I - Introduction to Software Development
- **SFWRENG 2C03:** Data Structures and Algorithms
- **SFWRENG 2OP3:** Object-Oriented Programming
- **SFWRENG 2XC3:** Software Engineering Practice and Experience: Development Basics
- **SFWRENG 3A04:** Software Design III - Large System Design
- **SFWRENG 3DB3:** Databases
- **SFWRENG 3RA3:** Software Requirements and Security Considerations
- **SFWRENG 3S03:** Software Testing
- **SFWRENG 4HC3:** Human Computer Interfaces

7.2 Knowledge/Skills Outside of Courses

Some of us had experience with the following topics from co-ops or personal projects, but others had to learn them for the first time:

- **Amazon Web Services (AWS):** Needed to learn how to develop an AWS application (both forntend and backend) and about the various AWS services that exist.
- **Next.js:** Learned how to develop web applications using Next.js framework.
- **Material UI:** Developed an understanding of Material UI to build visually consistent and accessible user interfaces.
- **Figma:** Learned how to prototype and design intuitive user interfaces and interactions through the Figma design tool tp create mockups and wireframes.

- **GitHub Actions:** Needed to learn how to create github actions workflows for linting, building, testing, deploying, etc.
- **GitHub Issues:** Needed to learn how to use GitHub issues to effectively track work done.
- **Infrastructure as Code (IaC):** Needed to learn how to write SAM (Serverless Application Model) and CFN (CloudFormation) templates to make AWS deployment easy and replicable.
- **City of Hamilton Water Division:** Needed to understand the processes of the City of Hamilton Water Division to create project requirements and effectively come to a solution.