**CS 3343**

Software Engineering Practice

**Group Project – “Hot Meals”**

Group-1

**Self-Assessment Report**

**Group Members**

* Banbah, Kush: 55786740 (Programmer)
* Gupta, Aarnav: 55990960 (Scrum Master)
* Jain, Utkarsh: 55992915 (Assistant PM)
* Kasliwal, Aryan Girish: 55972222 (PM)
* Malhotra, Avi: 55773896 (Programmer)
* Rajagopalan, Pratul: 55858290 (Lead tester)

# Aryan Kasliwal

Software engineering practices is one course I’ve thoroughly enjoyed at the university. I really appreciate the project-based learning approach which gave us an opportunity to come up with a solution to a real-world problem and build it from ground up. This was also the first collaborative project most of us have worked on, so we learnt a lot of communication and collaboration skills required to build large scale projects. Personally, I was very intrigued when I took up the role of Project Manager. I always found coming up with innovative solutions enjoyable and getting a chance to develop software for our own solution is a unparalleled feeling.

When we came up with the idea around week 2, we were yet to learn most of the course content and that allowed us to be as imaginative as we could be. As a rule of thumb, I ensured we had a board and a marker in all of our meetings and anyone would randomly go up and draw and illustrate anything that came to their mind. As the project progressed, we started tailoring it according to the course’s requirements and learnings.

As the project manager, I ensured I stayed focused on the larger picture of the project to direct the project towards what we visioned it to be. I picked up many skills such as project scheduling and team management – deciding deliverables for everyone every week and systematically ensuring the project gets done within the timeline. Backlog management of tasks is one skill I can co-relate with my internship as well, where we all brainstorm and discuss ideas to clear backlogs and keep the project on schedule. Getting to work with my friends for this made it more exciting and fun.

As the project manager, I also decided to make a Kanban board for the team, and we would often bring it up to track progress of our work and schedule. A key task for project scheduling was to come up with a high-level step by step path and for this to happen we had to analyze the dependencies. We called each high-level task an epic and each of its sub-task a feature. It made logical sense to me to first analyze epic level dependencies and draw up a hollow project plan, then discuss this with the team and then get into the weeds of each epic. As we started building each epic, we learnt more about each feature and changed some high-level epics but being agile was the correct way to approach the planning.

Keeping aside my experience as a Project Manager, I also ensured I was able to fathom the CIOLs. Software engineering, applying design patterns learnt in Software design, making a scalable and testable solution, writing bug reports, refactoring code all of it was a part and parcel of all our experience and learnings.

# Avi Malhotra

While my official job role in this project was that of a “Programmer”, I got the opportunity to explore a multitude of job profiles and responsibilities over the course of 13 weeks. In fact, each of the team members delegated and shared responsibilities with each other not only according to our strengths, but also according to our areas of interest. Part of our “Hot Meal” project’s success can be attributed to our ability to work collaboratively and productively.

Most of my contributions in this project revolved around the project code itself. I coded multiple classes for all the three releases, and I was also primarily responsible for designing the class diagram before the team began coding. This step also included planning the software methodology, as well as the design patterns and principles our team planned on incorporating. My other responsibilities included working on the use case and sequence diagrams. However, I spent much of my time in the last few weeks working on the project documentation – namely the design & analysis report and the bug report.

Overall, this project proved to be a great learning experience for me. I got the opportunity to not only witness and apply the knowledge gained from this course, but also recall my conceptual understanding from some of my previous university courses (such as CS3342 – Software Design).

# Aarnav Gupta

I acted as the SCRUM Master for our group during the project, which meant that I made sure that our backlog was being completed on time, as per our planned timeline. It also meant that I also organized regular meetings for us to share our ideas and update us on their work.

But this is not all the work I did. I also helped in preparing the reports, which included working on the Gantt Chart, something that immediately helped me in my internship, as my supervisor asked me to prepare a timeline of all the projects, I’ll be working on till the end of my 9-month contract.

Moreover, this course helped me implement GitHub version control for the first time. I had done a Coursera course to learn how it works, but never got the opportunity to practically use it until now. GitHub conflict management was one feature I appreciated the most. Apart from that, it always weirded me out how multiple people can work on the same coding project together, even with GitHub. But, once we had a class diagram ready before jumping to code, it started to make sense. Things we learnt in the previous course now made way more sense too, of why we do them, and how they help in a real project.

# Pratul Rajagopalan

In this project for CS3343, I feel like I was able to gather a large amount of information about software development projects and then apply that knowledge effectively while working on this project.

As the lead tester, I learnt the intricacies of working on a large project like this and feel like I grew as an individual. Both my inter-team skills and my coding skills improved, as my role in this project meant that I had to coordinate with my other members to make sure that all possible areas were being covered without any overlap. I also learnt how to delegate and divide work effectively among members of the team to speed up the process.

I also learnt how to write effective junit test cases and how to use github for version control as I had never worked with either previously.

This project also helped me pace myself and helped me learn how to break down a daunting task into smaller bits that could be finished off quickly and easily thereby completing the whole task. In all, I learned a lot while doing this project and had a lot of fun along the way as well.

# Utkarsh Jain

I feel this project has made me learn more about how projects are carried out in teams. It’s not just about testing and design patterns but about communication, work distribution and dealing with problems that arise because we are working in a team.

Problems like prior commitments or people not responding on time or feeling annoyed by the project manager. All these things are new and give a better taste of how it feels to work in the industry. As far as the theoretical knowledge is concerned, I really liked the concept of testing and how easy it is to know how dependable one’s system is, of course the algorithms made me think, the testing part made me learn a lot, but the real learning was people management & conflict resolution,

It was a group of friends, but conflicts do still arise and to manage them is a different mental camp altogether. I feel like I did become better as a programmer and as a person.

# Kush Banbah

My official task was programmer but like all projects sometimes you must do a little bit of everything. My focus was of course programming and I’ve worked on proper projects which require step by step documentation, client meetings and more and this felt like another evolution of that.

Previously I used to work on projects by myself, so coding with others was a new experience. Working in a team structure where I couldn’t change code or design however I felt like was a little strange at times. But on the other hand, I was able to see different perspectives, new ideas and opinions of different people so that did help in improving my own skills by seeing other ways to code the same classes. Also using software development models for multiple releases, using different design patterns was a new experience and learning opportunity to use what I learned

I had a lot of fun and interesting discussions relating to that, which brought along another lesson. While working alone you don’t have to explain code to yourself or discuss it. When explaining my implementation to others and when trying to understand their point of views it was difficult at times due to lack of experience in talking in technical terms and there was a lot of miscommunications at times leading to wasted time and people getting annoyed unnecessarily.

Outside of coding, working with a team means everyone must pull their weight and be responsible for their own parts and be held accountable. This part was a little tricky, to tell people if they did something incorrectly, or to finish their own parts and managing the friendship dynamic and the professional work dynamic was a little difficult at times. However, I feel like it helped as a learning experience between managing relations and holding group mates accountable to their tasks. At times I felt like I had to carry other people’s weight to make sure the work got done. This isn’t entirely fair since I could have kept a better track of my work and check on them earlier instead of realizing it later when there wasn’t that much time to meet deadlines.

I think my key takeaways from this project would be the new coding styles, development planning styles I learned, also how to work in a technical project, communicating technical ideas and understanding other people’s ideas. Lastly, working with deadlines, internal and external in a group and with managing group dynamics.