**Data Engineering Final Project Reflection**

John Coogan

During this project, we composed a team from our Data Engineering course with a diverse set of backgrounds and skills to develop an end-to-end microservice for the Steam Platform. Steam is a video game platform which acts as a marketplace and forum for discussion regarding video games of all genres. The platform provides users and developers with active telemetry associated with specific titles to help users make informed decisions and developers monitor the reception of their titles within the community. We wanted to enhance this feedback by providing augmented functionality to both users and developers through a microservice. Specifically, we wanted to give platform users more informative and nuanced feedback on titles. To accomplish this, we scraped data directly from the platform, cleaned that data, and generated live telemetry including visualizations on reviews, playtime, and review clustering. The purpose of this report is to analyze and assess our groups process through the lens of Carl E. Larson and Frank M. J. LaFasto’s *Teamwork: What Must Go Right/What Can Go Wrong.*

I will structure this document around the core pillars of teamwork mentioned in this book: Clear Elevating Goal, Results-Driven Structure, Competent Team Members, Unified Commitment, Collaborative Climate, Standards of Excellence, External Support and Recognition, and Principled Leadership.

**Clear Elevating Goal:**

This principle of teamwork, and our team’s ability to adhere to it, is what made this project so exciting. The team members came together both because of our diverse skillset but also because of our passion for gaming. This gave our team a vested interest in providing a better experience for Steam users because we are all Steam users. In this way we were deeply emotionally invested in generating something that we would want to use. The goal was to provide an enhanced service to Steam users and its clarity was defined by the elements of our microservice which would enhance the existing Steam experience.

**Results-Driven Structure:**

This principle was one that essentially took care of itself. Our team found itself exploring techniques and methods which were new to us. Things like containerized applications, interfacing with network services, flask builds, and data pipelines were things we may have learned in isolation but getting them to work together was a challenge. The results which drove our team onward is that these individual elements interfaced appropriately and worked. At its core, we had a binary results-driven structure, either the application components worked, or they did not.

**Competent Team Members:**

This principle, which is not always under the team’s control, was an area where we got very lucky. Not only did our team have a high degree of competence prior to embarking on this endeavor, each team member repeatedly rose to the challenge of performing above and beyond their baseline capabilities. Learning new packages, methods, and development techniques, each team member was instrumental in elevating team performance.

**Unified Commitment:**

Our team’s unified commitment was apparent at every stage. Despite intense competing demands and the complexity of the task, the diligence which each team member demonstrated made clear that everyone was bought into our goal.

**Collaborative Climate/Standard of Excellence:**

This principle is where our team really excelled. Delicately balancing ambitious feature addition while managing scope and ensuring the microservice delivered on expectations without succumbing to external pressure and including partially completed elements to meet external guidelines. At no point was it inappropriate for team members to not only voice opinions but also for those opinions to be onboarded and alter the trajectory of the project. By employing sound development operations principles and continuous integration fundamentals we were able to manage features while maintaining a robust and automated codebase.

**External Support and Recognition/Principled Leadership:**

These are interesting principles since this was largely an academic project. Leadership manifested not as a single individual but as a group of peers. This peer leadership can be challenging but was effectively executed in this team. Individual accountability was not confrontational but rather self-generated through a desire to support the whole. External support came in the form of peers outside our group. Ideas and demos were passed between groups for feedback and that recognition mechanism, coupled with the friendly competition between teams to perform at high levels, made for a high achieving team.

**Peer Evaluation:**

**Kian Bagherlee:**

|  |  |
| --- | --- |
| **Sustain** | **Improve** |
| Strong drive to master new concepts. Where a lot of people get afraid of new things, Kian dives into them with considerable zeal. | Communication clarity is a subtle improvement. Kian’s thought process can sometimes exceed his verbalization of those thoughts. This can lead to confusion. |
| Simply getting something to work is not enough, it must work correctly. Best example is linking Azure and Flask, this could have been done a few different ways but Kian ensured that the methods employed were scalable and worked within our limitations. | (Applies to the whole Group)  More frequent and thorough feature demos. As we worked through the project there were areas that a single individual would work extensively. Being able to educate the rest of the team so the codebase is understood across the team is imperative for effective communication. |
| Easy and unemotional communicator, Kian is never averse to ensuring that problem statements are thoroughly explored and continually vetted. | (Applies to the whole Group)  Better asynchronous work. This project was developed largely in a traditional way, where the group members met in person, built features, and made progress. I believe that leveraging collaboration tools and becoming more skilled in doing this asynchronously will be a valuable skill professionally. |

**Katie Hucker:**

|  |  |
| --- | --- |
| **Sustain** | **Improve** |
| Ambitious and creative project vision. Worth being cognizant of and managing (see right) but worth sustaining as a trait. Much of the best parts of this project were a result of complete or partial feature recommendations from Katie. | Managing feature scope and perfection. Katie can get fixated on capturing the full breadth of a feature idea or reticent to move on until it meets here (understandably high expectations) but this can cause friction in team development |
| Excellent project vision. Katie is constantly aware of and using a larger picture vision to inform decision points. This is valuable since it can be all too easy to lose track of the desired end state during development. | (Applies to the whole Group)  More frequent and thorough feature demos. As we worked through the project there were areas that a single individual would work extensively. Being able to educate the rest of the team so the codebase is understood across the team is imperative for effective communication. |
| Katie delivers on complexity. Where many individuals may have their feature ambitions culminate at the nascent idea stage, Katie puts them into action. This is critical since ideas without action can cause intense friction but the bold contributions along ambitious feature avenues leads to opportunity. | (Applies to the whole Group)  Better asynchronous work. This project was developed largely in a traditional way, where the group members met in person, built features, and made progress. I believe that leveraging collaboration tools and becoming more skilled in doing this asynchronously will be a valuable skill professionally. |

**Suim Park**

|  |  |
| --- | --- |
| **Sustain** | **Improve** |
| Suim has good organizational skills. She demonstrated organized code as well as made sure that speaking points were documented to ensure a quality video demonstration. | Suim could improve on more assertive communication. Suim is technically sound and very capable but seems to be less comfortable communicating her ideas which can be a result of the team dynamic but is helpful professionally |
| Suim demonstrated exceptional flexibility and comfort with ambiguity during this project, quickly flexing from current implementations as required to rapidly integrate features as the code base changed. | (Applies to the whole Group)  More frequent and thorough feature demos. As we worked through the project there were areas that a single individual would work extensively. Being able to educate the rest of the team so the codebase is understood across the team is imperative for effective communication. |
| Suim demonstrated a strong ability to manage competing demands. I believe Suim had the most academic requirements this semester out of the group and was still able to contribute to the project which is a train that should be sustained. | (Applies to the whole Group)  Better asynchronous work. This project was developed largely in a traditional way, where the group members met in person, built features, and made progress. I believe that leveraging collaboration tools and becoming more skilled in doing this asynchronously will be a valuable skill professionally. |

**Ya Bei Zeng**

|  |  |
| --- | --- |
| **Sustain** | **Improve** |
| Ya Bei is highly efficient and produces code both quickly and with good fundamentals. Ya Bei has extremely good professional time management that results in very expeditious feature development. | Ya Bei is extremely intelligent but could improve her assertive communication to better message her ideas to the group. I feel like our group missed out on valuable insights on occasion. |
| Good technical skill. Ya Bei’s contributions to the overall microservice demonstrated high levels of proficiency while maintaining readability and reproducibility. | (Applies to the whole Group)  More frequent and thorough feature demos. As we worked through the project there were areas that a single individual would work extensively. Being able to educate the rest of the team so the codebase is understood across the team is imperative for effective communication. |
| Asynchronous talent. Ya Bei was by far the best in the group in her ability to incorporate tasking into her workload and accomplish that tasking efficiently and asynchronously. The overall group could benefit from better asynchronous communication but, out of all the group members, Ya Bei excelled at asynchronous workflows. | (Applies to the whole Group)  Better asynchronous work. This project was developed largely in a traditional way, where the group members met in person, built features, and made progress. I believe that leveraging collaboration tools and becoming more skilled in doing this asynchronously will be a valuable skill professionally. |

**Team After Action Meeting:**

When the group met to discuss our final project we covered topics such as effective tasking and dividing current endeavors in such a way that allows for gainful employment across the team asynchronously. We could also benefit from a more diligent usage of github for collaboration and feature management with clear and efficient issues as a mechanism to drive an asynchronous workflow.