SEPR 2019/20 Assessment 4

Team CheatCodez

Jonathan Grout, George Lesbirel, Cheuk Wang Wu, Lauren Quarshie, Muaz Atif, Lillian Coultas

Project Review Report

Our Approach to Team Management

By assessment 4, the team management had settled into more of a functional team structure [1], with separate teams for Game Development (George and Muaz), and Documentation (Lauren, Jonathan, Lillian, and Cheuk). This approach to management has worked well for the team as, of the major disadvantages of this style of team management, we were able to severely limit the effects of the major problems - as [1] says, functional teams tend to have difficulty communicating between the team split, but as we are such a small group, which definitely makes this structure more effective [2], and we kept up with weekly progress meetings during the course of the assessment (circumstances allowing), which helped in the mitigation of risk **R-04**, communication never became a big issue. This structure of splitting the group into sub-teams also helped to mitigate risk **R-02**, as team members who were particularly strong in either documenting or coding could use their skills in a specialised way.

Another common issue with functional management is that it's a very inflexible structure [1]. Again, we never found this to be a big issue, mostly due to the size of the team - we knew that with just 6 of us we had to be able to adjust roles as needed for the assessments - Jonathan and Cheuk have both switched to the Game Development team in previous assessments to help out with testing and assets, and reduce the workload for Muaz and George. This flexibility helped us to cope with any changes in requirements, or changes in workload between assessments, as we reassigned team members as necessary to different teams in order to not overwhelm either one with tasks.

The final disadvantage of our functional team structure is that it tends to push decision making further up the chain of command [1], [2]. Since we were only a team of 6 people, this is impractical. By assessment 4, we'd settled comfortably into a structure wherein Lauren managed the Documentation team, and George managed the Game Development team, with George as the overall manager for the project. We evolved this management structure very naturally between assessments 1 and 2, as we found managing both teams alone to be too much of a workload to fairly assign to George. We decided to keep this structure for assessments 3 and 4, which helped mitigate risk **R-04**, as we were all happy with this style of leadership and these leaders, and having smaller groups working under each manager allowed better monitoring of risk **R-03**, as difficulties with the tasks or availability of team members were quickly and easily identifiable.

Overall, implementing a functional team structure worked very well for us as a team, especially in conjunction with the development methodology we used, as it allowed us to specialise and focus on tasks, but not lose sight of the project as a whole. Potential improvements would have been to adopt this structure earlier, however it evolved as we began to understand how each team member worked, and who was both suited to being, and happy to be, a manager, as well as which development team people fit into best, which would have been impossible for us to know right at the start of the project, as this understanding came about as we worked together.

Development Methods and Tools

In assessment 1, we decided to use the Scrum methodology, over others like waterfall, mainly because it was a very flexible methodology, which allows requirement changes during development [3] (this helped mitigate risk **R-05** as well, as any smaller requirement changes didn't pose a large problem to development). We kept this methodology up the entire way through the project, meeting weekly to discuss progress made so far, and make sure that we were up to date on all aspects of the assignment despite all working on seperate parts. We haven't modified this method because it's worked very well for us in the previous assessments, allowing us to break down tasks and assign them, but quickly find any problems or confusion team members might have with their assigned tasks at the weekly meeting (helping to mitigate risk **R-03**). It also allowed us to quickly modify existing tasks, or create and assign new tasks, to suit any changes in requirements we had, helping to address risk **R-05**. It suited our needs so well in assessment 1, and was very helpful with keeping us on track, and making progress each week (thereby helping to mitigate risk **R-01**), so we decided not to evolve it for future assessments.

To keep track of our sprints we opted to use Jira as it is easy to use, helps organise work and allows assigning tasks directly to people working on the project. Using Jira helped us to mitigate risk **R-03**, as we could see if any team members were struggling to meet their goals, and reassign tasks as needed very quickly, so nobody fell behind.

We continued to use Github as a version control system and central code repository and Google Drive as a document sharing service - we didn't need to evolve these as the project continued because we picked them knowing they'd suit our purposes, as we were all familiar with them before beginning the project, knowing how best to use them, as well as the limits of their usefulness ahead of time. Due to self-isolation measures imposed during this assessment we were unable to conduct meetings in person anymore, so staying in contact through apps became more important than ever. For this reason, we used WhatsApp and Slack to communicate details of the project and Google Hangouts was used for video calls.

We also evolved the tools we used for development over the course of the project. During assessments 1 and 2, we had planned to use Box2D to handle collisions for the game, but due to picking another teams' project for assessments 3 and 4, we found this was no longer necessary, as the other teams had implemented the functionality Box2D offered, but without using it. We also decided during assessment 2 to change our IDE from Eclipse to IntelliJ. We decided to switch because of IntelliJ's integrated Github support, making version control much easier, but also the other features it offered over eclipse [5], such as making refactoring easier, which was very useful for assessments 3 and 4, where we were working with code from other groups. Once we better understood the scope of the future assessments, it seemed like the best choice. We started using LibGDX too, as it made game development much quicker and easier, allowing us to spend more time testing the game, and less time developing it. This helped to mitigate risk **R-01**, as using LibGDX sped up development, meaning that, if anything, we overestimated the time needed to develop software. From assessment 3 onwards we also needed to use Firebase as a tool to host documentation on the websites we inherited from Salt N Sepr in assessment 3, and Mozzarella Bytes in assessment 4.

References

[1] B. Moga, "Types of Teams [Advantages and Disadvantages], Activecollab Blog", *ActiveCollab*, 2020. [Online]. Available: https://activecollab.com/blog/collaboration/types-of-teams. [Accessed: 21st Apr 2020].

[2] Management Square, "Functional Organization: The Advantages And The Disadvantages", *Management Square*, 2020. [Online]. Available: https://www.management-square.com/functional-organization/. [Accessed: 21st Apr 2020].

[3] D. Uhlig, "Advantages and Disadvantages of the Scrum Project Management Methodology", *Smallbusiness.chron.com*, 2020. [Online]. Available: https://smallbusiness.chron.com/advantages-disadvantages-scrum-project-management-methodology-36099.html. [Accessed: 22nd Apr 2020].

[4] A. Yu, "8 REASONS YOU SHOULD USE JIRA TO MANAGE PROJECTS", *Linkedin.com*, 2020. [Online]. Available: https://www.linkedin.com/pulse/8-reasons-you-should-use-jira-manage-projects-alex-yu. [Accessed: 22nd Apr 2020].

[5] A. Cheptsov, "Enjoying Java and Being More Productive with IntelliJ IDEA | IntelliJ IDEA Blog", *Blog.jetbrains.com*, 2020. [Online]. Available: https://blog.jetbrains.com/idea/2016/03/enjoying-java-and-being-more-productive-with-intellij-idea/. [Accessed: 22nd Apr 2020].