Project Review Report

Thanks to our succinct planning and knowledge of our own, and each other's skills, we did not find it necessary to change anyone's roles in the group from the initial plan in assessment 2. Of course this was not our very first group structure as we lost Matthew during assessment 1, which in turn shifted Bradley into our designated Audio Designer. Apart from that minor spreading of responsibilities, the group organisation has never needed to change. It is important to note however that we have always had the opinion that the nature of the group and how everyone works is that no one is necessarily resigned to their duties, though they are accountable for them, and equally other members can assist them [1]. Throughout assessments 1 and 2, there was no indication that any member of the group felt uncomfortable or unable in their position so we never made any amendments to our initial lineup.

We have never felt the need to change our team structure. Since the beginning we have operated on a flat structure whereby everyone in the group is equal [2]. This is a particularly successful structure to use within a small programming group as it is very easy to keep all members informed on current changes and decisions, in one small Facebook Messenger chat. However, in case of disagreements we have a pre decided tall structure in place, only for when we believe we cannot work through a decision. In cases such as these, Alex is to have the final say as he nominated, and we voted for him, in one of our very first practicals. This helps keep our decision making concise.

Throughout our project, we continued to self analyse using CMM's Maturity Levels of Software Processes[3]. From the beginning, we were never in the *initial* level. Luckily we always had organised processes and prided ourselves in the repeatability of our work. For this reason, throughout Assessment 1, we were always at the *repeatable* level, meaning that basic project management techniques were established and success could be repeated. As we progressed through Assessment 2, we stayed at the *repeatable* level, however by the due date of that assessment, I am confident to say that we were at a *defined* level in our team, where organisation had developed its own standards as there was greater attention to team working and documentation. By the end of assessment 3, we had progressed yet another level to the *managed* level, we no longer needed to concentrate on organisation, instead it monitored and controlled its own processes through communication and the work that was getting done. As our final assessment draws to a close, we are getting nearer and nearer to our goal level of *optimizing*. Initially our goal was to reach the *managed* stage but we reached that easily in our previous assessment so we decided to stretch ourselves in our final piece of work and aim to consistently improve processes through monitoring feedback and trying to introduce more innovative processes.

Our group didn't stop evolving as the program progressed, a prime example of this would be currently as we are finishing Assessment 4, the group is continuing to divide into smaller subgroups for different documents. This proves to make the group far easier to manage and with that, comes far more progress on each piece of work we do nearer the deadline. We never intended for there to be these current subgroups in our initial plan, however they developed themselves due to the fact that more work needed to be done at once, still with different people collating at the same time. This even shows back to the fact that the group is in its *optimizing* level as new innovative processes have created themselves.

Software Development Tools

Tool	Why It Was Kept/Changed
LibGDX	After careful consideration our team chose to go with LibGDX as it is one of the most powerful game development frameworks in Java. With a plentiful supply of tutorials to help us get off the ground, we chose to stand by the engine as our knowledge and ability grew the more we used it.
Facebook Messenger	We decided to use Messenger as our primary communication tool very early on and stuck with it through the whole project as it allows swift responses between members through notifications and can be easily accessed from a wide variety of platforms (smartphones, laptops, PCs, etc).
Google Drive/Docs	We chose Google Drive as it allowed for team drives, which are stored on the cloud with automatic backup, meaning the risk of losing documents is extremely small. Also, the ability to create documents directly on Google Drive within a web browser using Google Docs was very useful as it was easy to access from any computer and allowed for much more useful version control and collaboration than, for example, Microsoft Word would have done.
GitHub	We chose to use GitHub as it gave us free access to private repositories, whilst giving us the ability to initialize different branches of the repositories. We chose to retain this method of version management not only because it fit perfectly with our Scrum approach, but also because as the project grew, we found it in our interests to retain all our work in the same place.
Trello	Though we did initially decide to use Trello for our task management, we later chose to abandon this tool, mainly due to the fact that the team began to designate tasks more fluidly and through communication lines, thus realising that to use this would create an extra, unnecessary workload.

Software Development Methods

At the beginning of the project, our team chose Agile [4] as the main software development principle and after careful consideration chose the Scrum [5] methodology. The main reason for this decision was that it is a rather flexible approach that works well with a small team like ours, where self-organising is easier and overheads for meetings and discussions are smaller. During the development process, this methodology remained as it has proved itself throughout the project to be valid due to the high marks we achieved over the phases as well as the popularity of our product within the cohort.

However, other adjustments have been made over the course of the project due to various reasons. Initially, motivation was high and our study workload was light which led to an overestimation of the time we could spend on the project, which was reflected by us allocating 3 meetings a week, one of which was with the client. However, as individual tasks became fewer but with larger workloads within them, the team was finding less time to devote to this. Due to this, we reduced the frequency of our meetings to once a week on Friday, which coincided with the easiest session for the client to attend. This was amended further in the final phase of development where we were extremely busy, leading to the decision to continue mainly through online meetings.

Additional adjustments made were with the product backlog. According to Scrum, the whole team should choose a task from the backlog and work on it within a Sprint. We generally kept to this principle, however during special occasions such as holidays when the team would temporarily split, we allocated different tasks to members so they would be able to work on it over the long break, promoting efficiency.

References

[1] Rear Admirals Assessment 1 Method Selection and Planning. [Online] Available:
https://taylorwillmott.com/SEPR/Assessment/1/Plan1.pdf
[Accessed 30 - April - 2019]

[2] Rear Admirals Assessment 2 Method Selection and Planning. [Online] Available:
https://taylorwillmott.com/SEPR/Assessment/2/Updates/Upd2Plan1.pdf
[Accessed 30 - April - 2019]

[3] Capability Maturity Model (CMM). [Online] Available:
https://searchsoftwarequality.techtarget.com/definition/Capability-Maturity-Model
[Accessed 30 - April - 2019]

[4] The Agile Manifesto. [Online] Available:
http://agilemanifesto.org/
[Accessed 30 - April - 2019]

[5] The Scrum Guide. [Online] Available:
https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf
[Accessed 30 - April - 2019]