Method Selection and Planning

Cohort 1 Group 7

Group Members:

Kyle Clifton (wnp512)

Bailey Eyers (rkh552)

Matt Hollyhead (fpk506)

William Martin (zjc524)

Max Pither (rkm538)

Enkhtuguldur Sarantsetseg (wbj512)

Ronny Watt (dvn513)

Software Engineering Methods

Our software engineering methods were based on an Agile approach. This involved us meeting each week to review our work and plan assignments for everyone for the next week. We found that this approach suited us because it gave us flexibility in our work, with us being able to continue older tasks or move on to new ones as we needed.

In-between meetings we each had a week-long sprint in which we would attempt to get as much of work for the week done, individually or as part of a group.

In addition, we maintained a flexible relationship with our customer where our game requirements were specific enough to meet the customer's needs but still gave us room to make our own design choices. Also, we were able to approach the customer whenever we had any questions. This suited us because it meant that we were always able to make sure that the customer was satisfied with our work.

Before we started coding our game, we made sure that we, and the customer, were happy with our game requirements. This was so we would have a clear vision of what we were making from the beginning and would reduce the chances of us stopping and changing our requirements in the middle of game development. We also conducted a risk assessment so we were aware of all the potential risks that could occur during game development and how they could be avoided.

Also we made sure to leave comments and docstrings throughout our code so anyone who read the code would be able to understand what it does.

Source Control

For source control, we decided to use Git and GitHub. GitHub is a free platform that allows clean collaboration on code and project files, allowing users to work on separate branches of the codebase in parallel using separate branches. It is quick and easy to learn, aligning with our development time. It keeps track of each person's changes using blame, helping us maintain an equitable contribution from each person. GitHub also provides a supported desktop application (GUI). This would provide a clean visualisation of our current branch and history, and integrate more cleanly with the service than a third-party GUI.

We considered Sourcehut, a reasonably popular alternative with around 25000 users. It also uses git under the hood, so we would be able to develop with similar benefits to Sourcehut. However, Sourcehut is not quite as mature as GitHub. Additionally, Sourcehut's website UI is more complicated, and we wanted something simple to make our website more quickly.

Collaborative Software

To collaborate on our work, we developed our deliverables on a shared Google Drive, to allow all of us to access each of our deliverables at any time we wanted and make any changes that we needed. In addition to Google Drive being free to use.

In addition, to keep track of which tasks we would be doing each week, we initially used notion to create tickets of each task we were doing. However, as we began to add coding tasks to GitHub as Issues, we started putting other tasks there to keep track of them.

To communicate with each other throughout the project we initially created a WhatsApp group chat. While we considered using Discord instead, to create a server which we could use to establish multiple group chats for certain groups and tasks, we decided to go for WhatsApp because it would be easier to manage. However when we started developing the code, we moved our group chat to a WhatsApp community, which we ended up adding 2 extra group chats to. One for coding and one for Delieverables. We also later added a group chat for the Maze Design

Summary

To support our Agile approach, these tools worked well because they allowed us to collaborate quickly and adapt our work as plans changed each week. GitHub's branching and issues helped us manage tasks during our weekly sprints and let us work without disrupting others. Google Drive supported the frequent updates to documents that Agile development naturally produces, while WhatsApp gave us a fast and flexible way to resolve questions between meetings. These tools matched our workflow and helped us keep progress throughout the project.

Team organisation

Throughout the project we would assign one more task to each team member each week. Then at the Tuesday practical sessions we would review our progress and in a separate meeting the next day we would identify what needed doing in the week and assign tasks to each person.

In the initial stage of the project, we worked on our tasks together, with the whole team completing tasks such as brainstorming game ideas, developing requirements and planning the customer interview.
However, we did not want everyone to be working on implementing the game at the same time as that could easily lead to errors through contradicting code, in addition to the game development becoming hard to organise. Also, some group members had prior experience working on our chosen game engine libGDX, while other group members had never used it before.

In addition, it would not be ideal for everyone in the group to be working on the documentation at once, as this could lead to many people in the group overriding each other's work and the documentation becoming disorganised.

Therefore we split into 2 groups. The programming group would develop the game code while the Documentation team would continue to work on the deliverables.

This allowed each group member to focus on one area of the project that their skill set was the most suited to. This would also make both the documentation and coding sections of the project easier to

manage as there were less people doing each.

To make sure that the documentation was as accurate as it could be, there was some crossover, with some Documentation team members doing some coding and some Coding group members doing some documentation work to make sure both groups fully understood the work of the other.

In addition this was beneficial to the coding team because they often had a lot of tasks to complete so any help the documentation team could give them was welcome, while the coding team doing some documentation was helpful for the documentation team because it allowed them to make sure their documentation was as accurate as possible.

Project plan

Throughout the project we would decide individual tasks for each person in our group meetings each week that would contribute to our larger goal of completing all the Assessment One deliverables to a high standard.

Initial goals

Our initial goals were:

1) Set up our collaborative software and website.
2) Think of ideas for our game and start working on documentation deliverables, including requirements and our customer interview questions.
3) Complete our customer interview and finalise our requirements.
4) Start developing the game and continue working on other deliverables.
5) Have our game ready to submit for assessment one and our Assessment 1 deliverables completed.

Plan evolution

Assessment 1:

Gantt chart showing overall group progress and workflow: Gantt Chart | ENG1-C1G11

In the beginning of the project, we worked on a lot of tasks as one team, such as getting initial requirements or brainstorming ideas for our game. This allowed us to all benefit from each other's ideas and made sure that between us, the work we created was to the highest possible standard. However, as the project progressed, we soon realised that there were too many tasks for us to complete as one group and we began splitting into sub-groups and creating plans for each individual subgroup to complete.

While in the beginning of the project our plans were quite linear, with us all doing one task which fed into the next one, as the project progressed we split into groups more and more often to get more

done. Also, as we began working on the game more, we began to work on multiple tasks at once to try and get everything completed. This increased towards the end of Assessment 1 as we realised we needed to get more done to get our work done on time.

Assessment 2: Gantt Chart

Gantt chart showing overall group progress and workflow:

When we first inherited the project, we worked as a team to ensure that all of the files from the previous team could be accessed and updated, and to come up with ideas for new features we could add to the game based on the new requirements in the updated product brief. We then distributed the tasks that needed to be completed for assessment 2. After the tasks had been distributed, we began working on them and all of these tasks began at the same time. For most of the project, we were able to focus on our individual deliverables, whilst also communicating with anyone else working on the same deliverable. Near the end of the project, we worked together to ensure that all deliverables had been completed so that the project could be submitted.

Weekly plans

Each week we had a practical session on Tuesday where we would meet and review our work from the previous week. Then in a separate meeting on the Wednesday of each week, we planned our tasks for the week that were due the following Tuesday, with group members communicating between meetings to discuss any issues. This allowed us to keep watch over what needed to be done each week and allowed us to plan accordingly.

Our plans for each week are stored on our website in the following link: weekly tasks | ENG1-C1G11

Assessment 1:

In the first week of the project we did not have the formal assessment document so our plans for the week were very basic. These were tasks to set up the project for development such as setting up our github repository and google drive and were completed quickly.

In the second week of the project we had the assessment document so we started to proceed with plans that would contribute to the project and assessment. We planned to start preparing for our customer interview, as well as completing some of our documentation deliverables to save time later in the project. In addition, we started to gather some of our game requirements from the brief that we could add to after the interview

In the third week of the project we had made a good start to our documentation. After interviewing our customer we were able to finalise our requirements and start developing our game. In addition, as we started adding entities, components and systems to our game, we also started documenting them in the architecture.

In the fourth week of our project, we started our game development and planned to add game development tasks to our github when they came up for anyone to take over if they wanted. Also, we split our group into documentation and programming, where we planned for each group to focus on

their individual tasks to optimise our group's efficiency. This involved the 2 sub-groups being divided into individuals with their own tasks

In the fifth week we knew we had consolidation week ahead, which would give us extra time to work on our project. Therefore we planned our weekly tasks around this, with each group member planning to work hard on their given tasks to make sure they were done over the week.

In consolidation week, we had an independent meeting on Tuesday 28th October. In this meeting we reviewed our work for week 5 and planned our work for week 6. We had to do both of these in one meeting because we were not all available for a second meeting throughout the week. In addition, we had a separate meeting prior to the main one, in which the coding and documentation teams reviewed the code and discussed how the architecture was made, to improve the development of the Architecture deliverable.

Week 6 was the final week of development, with the deadline for submission being the Monday of the following week. Therefore we worked hard to make sure all our deliverables were ready for submission, while making sure there was enough time for us to make a presentation for our game to the other groups in our cohort.

Assessment 2: Weekly Tasks

During week 7, each group presented their project to the rest of the cohort. We had to choose a different project to carry on from and the project we decided on was group 11's project. This was decided by a vote by our team and the majority of team members chose this project.

During week 8, we familiarized ourselves with the project that we had chosen and sent our decision to the cohort leader. During the practical session, we began to distribute tasks between us and come up with ideas for the additional events that needed to be implemented. We also discussed features that the project we chose was missing, such as a pause menu.

During week 9 practical session we further discussed how we needed to change the project and deliverables that we had inherited from group 11. On the wednesday we met up to begin making changes to the deliverables and reporting these changes in the change report. We also decided to have an additional team member working on the code to ensure that all requirements were completed on time.

During week 10, we continued to update the deliverables and document these changes in the change report. We created unit tests so that the different parts of the program can be tested as the code is written. We also began to implement some of these changes into the code so that the updated deliverables can all be added to the game before the deadline.

During week 11, we continued to implement the changes in the code and update the deliverables. We began working on the documents for the user evaluation so that we could effectively get feedback from other students when the user evaluation is carried out.

Over the winter break, we continued working on our deliverables for assessment 2 and updating the

code. We went over what we had done so that we could ensure that everything was completed by the deadline. As most of us had gone home, we communicated using our whatsapp group chat so that we could keep up to date with how each other were doing on our tasks.

During revision week, we met up one last time to finalise the deliverables and to ensure that everything was completed by the end of the meeting. Once everything had been finalised, we submitted our assessment 2 deliverables.