Method Selection and Planning

Cohort 1 Group 10

Cai Hughes<cabh500@york.ac.uk>
Ben Slater<bs1463@york.ac.uk>
Adeola Adeniji<aa3098@york.ac.uk>
Mathew Riedy<mr1723@york.ac.uk>
Riad Kasmi<rmk526@york.ac.uk>
Simon Konieczny<sk2144@york.ac.uk>

Methods

Our chosen method for this project was scrums: weekly informal meetings to discuss progress before giving out tasks to be done in sprints. The short sprints and weekly meetings help to decrease the bus factor and identify any problems within the team, such as non or low participation from any members or any issues in our work.
Scrums are useful as they only require a moderate amount of upfront scheduling. We have a small team and a relatively small project so a large amount of upfront scheduling is unnecessary; we don't want to spend so much time planning and organising that it affects our available time to implement. Another advantage of this agile method is identifying any short-sightedness in our plans (e.g. uneven distribution of resources or a team member getting stuck and needing assistance). This means we can replan when something eventually goes wrong (Murphy's law), as we can convene each week and assess not only our progress but how each member feels about the distribution of work.
At each weekly meeting we discussed the progress we had made during the week on our respective tasks, each member's current or anticipated problems and what their plan is for next week. These meetings allowed for us to make sure no member felt left behind or as if they were doing all the work as well as help us to stay on top of the project. At the end of each meeting we could use each individual's report to update our project schedule and reallocate resources where needed.
As a group it was decided it would be best to use IntelliJ as the standard IDE to streamline the workflow for everyone, as the learning curve for it is not steep at all coming from other IDEs, most of us were very familiar with VS Code. This proved to work well as no issues appeared related to IDE choice throughout the project.
For our version control software, we chose to use git, and github as our repository hosting software. It is widely used and therefore it will be easy to find tutorials. It is fairly easy to use, which is good as some members have not used version control software before. Ultimately, github is the most recommended and suitable for this kind of project. This worked very well for us; we were all able to learn to use the software quickly and make full use of the features available during development.
To maintain contact outside or practicals/group meetings we used discord. This is for very similar reasons as seen before: it has been used by group members before, it is widely used by other groups that require easy cross platform communication, and it has several features to make this communication even easier than standard messaging apps. With discord we can not only easily share files, but also have separate chats for separate parts of the project. We created individual channels for each deliverable for the assessments which clearly split the work and allowed for direct communication.
To collaborate on documentation we used Google docs: A free, incredibly easy to use word processor that allows for instant collaboration. One of the most widely used word processors, it made sense to use for this project. With it we can edit the same documents at the same time with no issues, which is essential for some of the larger documents that will require multiple people to work on for fair distribution of work. Using google docs also made it easy to keep all documentation in one place, in a google drive. Docs also allows for conversion of documents to pdf format which is the required format for submission.

Team Organisation

As a team we were made up from individualistic collaborators, this meant we naturally fell into a pattern of choosing responsibilities for ourselves and then working on them largely independently. We usually split a deliverable into 2-3 parts (depending on the marks allocated to it) and then various team members picked what they felt most interested in or comfortable doing. At the weekly meetings we would check up on progress and focus on cross-dependant elements of the work, such as making sure that the person doing a particular part of the implementation was implementing the appropriate requirements. We were also very flexible in that as various difficulties were encountered, tasks were reassigned

As a team we tried to delegate tasks from our plan at group meetings allowing those who felt most confident in certain areas of software development to complete those tasks. At the following week's meeting we would then catch up and discuss the progress we had made. From here we could then decide if any tasks needed help finishing and distribute our resources where they were required, before deciding what needed to be done next week. Naturally, our approach caused role subgroups to be formed as some members preferred to work on documentation, whereas others wanted to have a larger part in the implementation; we found that people tend to stick to work packages even if they chose different tasks each week. This went well for our team but might not work for others, e.g. a team may have a large number of members who want to work in one area. This issue was brought up, but we agreed that if we hold each other accountable and trust each other to do the right tasks correctly, the issue could be safely disregarded. In order to ensure we stayed on schedule, we regularly communicated and checked on each other’s progress. Unfortunately, our approach caused us to have to increase our workload in the second half of this project. However, we adapted our plan and by then we had entered the norming/performing stages of team performance, where each member had settled in and was able to work through the rest of the project effectively. Even though no direct leader was elected, natural leaders started to stand out as we entered different stages of team performance. This meant that the group was able to work in the areas they wanted, but when required there was someone to step up and allocate tasks that needed to be completed, also to follow up on anyone who struggled to meet the deadline for their assigned task. Another benefit of working without a leader is that we avoided problems such as members trying to take control and impose on others, taking over tasks and not giving people the opportunity to volunteer to work on any aspect of the project. Overall, this organisation was suitable for our team considering that this is the first time we have worked together on a project. It was important to allow flexibility while still being held to deadlines by the other group members. Everyone had a chance to pick areas that we wanted to develop and further explore.Then, as a group we mutually decided which member would fit a specific role the most. The organisation we chose was well suited to the project too, because the project itself was rather small. As a team we wanted to focus more on working out the material plans than assigning the leadership roles. While our team was successful in producing the project we set out to, the team organisation could have led to pitfalls, for example weak participation. The group, fortunately, managed to complete the project because we held each other and ourselves accountable.

Plan:

Week 1

For the first week we prioritised getting to know the software we were going to be using. We all followed tutorials on libGDX, creating our first game, to familiarise ourselves with the development framework and also spent time learning to use git/github. We looked at examples of games that we thought would relate to our own project, in order to develop an understanding of good game design and any features we would want to include. Finally if anyone had any spare time they could look at maps and sprites we could use for the game. The only meeting we had this week was our practical.

Week 2

We created a work breakdown (see week 2 plan screenshots documentation) listing the key activities we would need to complete. During the practical we also began discussing constraints, resources and a brief schedule and a brief outline of a schedule (also see week 2 plan screenshots documentation). During the week we collaborated on a customer meeting document, to help us present our ideas for the game to our customer. This included general questions to narrow down the key activities and clarify any points, and any ideas we had ourselves in order to make the game a reality.

Week 3

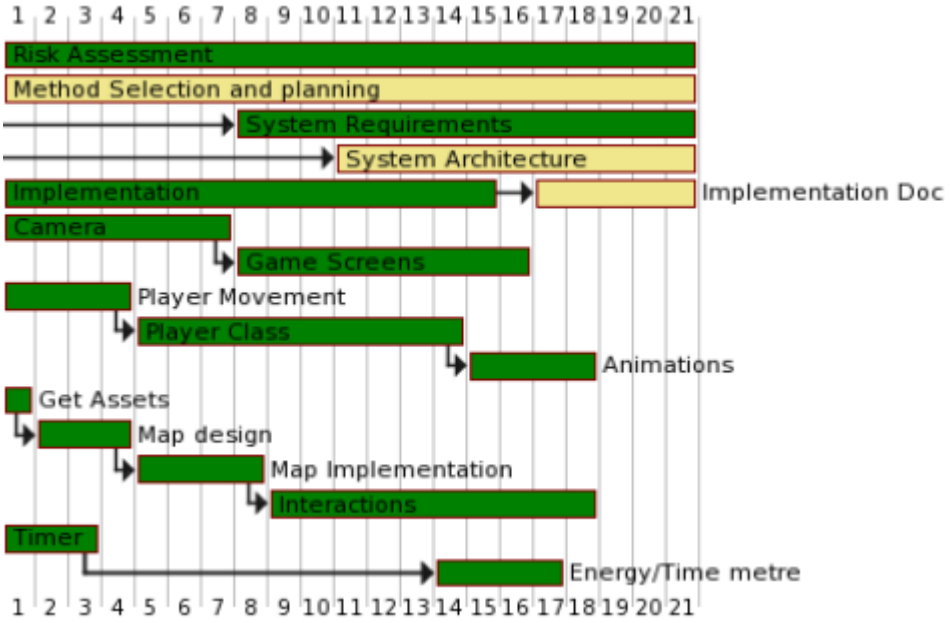
We had a customer meeting (26/02/24), where we were able to ask about the game to further narrow down a set of requirements and specific features that our customer would like to see. Following the customer meeting we had a brief team meeting where we discussed our next actions and from our first work breakdown and schedule, we created the first iteration of our gantt chart (see week 3 plan screenshots). At the practical we used this chart to decide who would do each part and agreed on a set of tasks for each person to complete that week: Damian starts the risk assessment document. Elijah starts the method selection and planning document . Dom and Tawfig start requirements document. Charlie, Yousef and Zubair begin work on implementation.

Week 4

This week we had an informal meeting on Monday (04/03/24) to discuss any issues we had come across; everything was fine and we agreed to discuss any changes to the plan at the practical. At the practical we reassessed progress and updated the gantt chart (see week 4 plan screenshots). We also decided to only meet at the practicals each week unless an extra meeting was needed. Finally, we set tasks for the following week: Zubair and Elijah create the map Zubair implements the map. Dom begins work on requirement documentation. Charlie finishes work on the camera. Yousef finishes the character class. Damian starts the game screen.

Week 5
This week at the practical we discussed our progress and any problems that had arisen during the week. We realised that we had fallen behind a little in the actual workload of the project and decided to increase the workload amongst team members. This was essential to make sure we had time to discuss any problems in week 6 and fix them before submission. The allocated workload was as follows: Zubair implements collisions. Dom starts work on the architecture document. Tawfig completes the requirements document. Charlie finishes work on the camera and starts work developing the energy metre. Damian finishes the game screens and begins work on code documentation. Yousef finishes animations and begins work on code documentation.

Week 6
Most of the work had been completed by week 6, so it was just the final parts. We wanted these completed before the practical so could just check and upload our deliverables. On monday (18/03/24), we had a short online meeting to discuss the final steps and allocated the following tasks: Elijah finishes method selection. Charlie and Yousef finish the final bits of in-game data tracking. Zubair creates the website. Dom finishes system architecture doc Anyone else works on the implementation document or helps anyone whole needs help. The final gantt chart for this week is as follows:



Final Week

As stated in the Team Organization section, the team functioned well as autonomous units, divided into focus groups by the part of the project they focused on. The work split for part 2 of the Assessment was as follows:

	A	B	C	D	E	F	G	H
1		website	change report	implementation	testing	user evaluation	continuous integration	
2	simon		14					14
3	mathew	3		8	3			14
4	riad		3		1	10		14
5	adeola		6	8				14
6	ben			7			8	15
7	cai				14			14
8		3	23	23	18	10	8	85

This assured an even work load, however, as the project progressed and some people finished ahead of others they picked up open tasks. The team met at least twice a week to catch up on running issues, especially within focus teams on one of the deliverables. No major issues or disagreements cropped up throughout the project and everyone collaborated smoothly. The Gantt chart for the last days of the project is as follows: