Method Selection and Planning

Team: No

Team Members:

- Jack Burman
- Sam Churchill
- C Lloyd
- Sam Ralph
- Rebecca Wardle
- Jiacheng Wu

Method Selection

Software Engineering Method

We have followed a SCRUM development process throughout the project. Meetings begin with each person saying what they've done since the last meeting as well as any problems they've faced so the team can discuss a possible solution. This keeps everyone up to date on the current progress made. Then everyone discusses the tasks they will work on until the next meeting and ask for help if they need it. After that we briefly go over potential risks to the project and how to mitigate them. This process keeps our meetings structured so each team member has a chance to participate and they clearly understand what they need to do next, which reduces lost time or redundant work due to confusion.

Tools used

We decided to store all documents on a shared Google Drive, as we are all familiar with its use so we didn't waste time finding and learning to use another tool. It allows us to organise our documents into folders to be less confusing, and has compatibility features such as file conversion which may help when handing over the project. Google Drive also lets us create documents within it which we can all collaborate on in real time. This is particularly helpful during meetings as we can edit/create documentation together from different systems for more clarity of direction for the project, which during the pandemic is useful.

We agreed to use Trello as our tool for project management, as it is an easy to learn free tool online. It helps with keeping track of various tasks that have/haven't been completed, as well as being a place where any member can post questions about the project so other members can help clarify what should be done. Trello also allows us to link files directly from Google Drive to specific posts, which is useful when the post in question involves a certain document, making an even better choice as our project management tool.

We chose Slack as our tool for group messaging. It's a free online tool making it accessible to us, and it has some useful features. Within it we can create separate message channels to organise conversations around specific topics for clarity. Trello and Github can also be linked to slack so that we get notifications when a change has been made within our space on those tools, which keeps us all up to date with the progress being made. Using slack we stay on the same page for the project, and without it our team would be unable to keep on track and manage time as efficiently.

GitHub has been used by our team to collaborate on the implementation of the game. It allows multiple people to work on different aspects of the game simultaneously so tasks can be split between team members, which works well with scrumming. This is also helpful during meetings as everyone can see the specific changes made to each person's version of the game which can be discussed separately before being merged with the main version.

The development framework we decided to use for our game is libGDX. It is free and compatible with Java which we are all familiar with. In addition it is a popular choice among our peers, which makes our project more appealing when deciding whether or not to take over it.

For graphics design we use Inkscape. Inkscape is a free tool that allows us to draw sprites at the exact dimensions we need for the game. This makes it more useful to us than software

like Paint because we need our images to be the right size for the tilesets we create in Tiled, and the characters in the game. It is also a beginner friendly tool, which is useful to the team because none of us have much graphics design experience.

To create a map for our game we used Tiled. Tiled allows us to create a tileset using separate images so that we can use them repeatedly on the map. It is easy for beginners to use and when changes are made to individual parts of the tileset the map is automatically changed which saves time. This is helpful as at the start of the project we will use placeholder art and later we will replace it with better looking art.

Team Organisation

Our team allocated roles for team members. We have a leader who organises team meetings, and generally steers the topic of conversation within them. This reduces the amount of time wasted in meetings as we are always discussing something relevant to the project. The leader also ensures that each person's progress is since the last meeting is tracked, which encourages team members to work on their parts of the project. In addition, this helps the team keep track of how much work has been done, and what work is still needs to be done, so we can more easily organise what people should do before the next meeting.

The secretary is responsible for taking concise notes on what has been discussed in meetings and storing them in the shared drive. These notes are important because they condense the important points of conversation into a document that team members can use to remind themselves what happened in the last meeting. This can be especially useful for the leader as they are able to see what has/hasn’t been discussed yet from the last meeting, so important topics of conversation are less likely to be missed, which can save time or improve the work being done on the project. It also provides a convenient way to catch up for anyone who couldn't attend a meeting, so they can stay on track with the rest of the team without another member having to recall the meeting after the fact, which reduces confusion and saves time.

Each team member has been assigned a deliverable that they are responsible for. This doesn’t mean they must do all the work for that deliverable, but rather they lead the work done on it. They must discuss the current progress on that part of the project in team meetings, and what must be done next. They also familiarise themselves, more so than other team members, with parts of the assessment brief relevant to their deliverable. In doing this, we ensure the team has a more structured time to discuss the specifics of a deliverable, so quality progress is constantly being made in all parts of the project.

Project Plan:
Critical Path:

T1.1 T1.2 T1.3 T1.4 T2.1 T2.5 T1.5 T3.1 T3.2 T3.3

Evolution:
Over the course of the project, we encountered several tasks that took longer than expected, and we had to adjust our timetable to suit.
For example in week 2, the space station features took slightly longer than expected to design, and as a result several other tasks had to be delayed.
Furthermore, as we were progressing throughout the iteration stages, we realised that we needed more time than we'd allocated, and had to extend the two stages (to 10 and 9 days respectively). Luckily we'd anticipated in advance that we may encounter delays, and had 3 spare days specifically for delays.
In addition we later realised that client feedback was not a worthwhile endeavour, and summarily removed them from our timetable.
Plan:

Task ID	Work Package	Description	Start/End Date	Dependencies
T1.1	Requirements	Solidify and ask questions about requirements to the client	22/10/2020 - 29/10/2020	
T1.2	Requirements	Create user requirements	22/10/2020 - 29/10/2020	T1.1
T1.3	Requirements	Create functional requirements	22/10/2020 - 29/10/2020	T1.2
T1.4	Requirements	Create non functional requirements	22/10/2020 - 29/10/2020	T1.3
T2.1	Design	Design basic features of space station Room Types/Looks, (infirmary) Which rooms have teleportation pads What systems are there? (Dependent on room types) Do systems provide any functionality, can they affect the game? I.e. Turn off lights/damage auber/etc	22/10/2020 - 29/10/2020	T1.3
T2.2	Design	Design basic features of infiltrators	22/10/2020 - 29/10/2020	T2.1

		How infiltrators will damage auber, (environment) How systems are damaged? Special ability (movement speed same/different as Auber?) Do infiltrators spawn at the same time? How many can exist at the same time? Can infiltrators damage multiple systems at the same time?		
T2.3	Design	Design basic auber features Heal Over Time/Heal Pickups/Go to infirmary Decide on arrest feature, beams, proximity, etc How should Auber be notified when systems are damaged? Audio/Visual notifications?	22/10/2020 - 29/10/20202 2/10/2020 - 29/10/2020	T2.1
T2.4	Design	Decide on art styles (Some form of 2D/top down style)	22/10/2020 - 29/10/2020	T2.2/T2.3
T2.5	Design	Create initial wireframe prototypes/ui?	22/10/2020 - 29/10/2020	T2.4
T2.6	Design	How does the demo player movement work? Is it recorded from an existing game?	22/10/2020 - 5/11/2020	T2.5
T2.7	Design	Risk Identification	22/10/2020 - 5/11/2020	T1.3
T2.8	Design	Risk Analysis	22/10/2020 - 5/11/2020	T2.7
T2.9	Design	Risk Planning	22/10/2020 - 5/11/2020	T2.8
T2.10	Design	Risk Management	22/10/2020 -	T2.9

			26/11/2020	
T1.5	Requirements Elicitation	Get feedback from customer on wireframes	22/10/2020 - 5/11/2020	T2.5
T3.1	Implementation	Create an initial simplistic working version?	22/10/2020 - 12/11/2020	Design
T3.2	Implementation	Improve/Design graphics	22/10/2020 - 26/11/2020	T3.1
T3.2	Implementation	Add missing features	22/10/2020 - 26/11/2020	T3.1
T3.2	Implementation	Finalises and ensure everything is documented	22/11/2020 - 26/11/2020	T3.1