Method Selection And Planning

Group 26 - Spice Traders (prior team 22)

Alice Cui
Robert Murphy
Dan Wade
James McNair
Charlie Crosley
Marc Perales Salomo

4.a Software Engineering Methods and Tools

For a software engineering method, we chose plan driven development (PDD). - We will use this method to plan and develop all features the customer will want in the final product.

- Having a plan based development is important, as it will hopefully ensure that we finish the assignment within the relatively short time frame. This makes it more suitable than another software engineering method such as agile, where the time it will take to complete a project is less clear.
    - Planning out the project also allows us to organise the team and give them specific roles.
- Of the plan driven development, we chose to use the waterfall model. This model uses clear linear stages (requirements, design, coding, testing, maintenance) of development to keep us on track
- Having clear stages will help us to plan our time on development. - The waterfall model is suitable, as the project is fairly static: the core requirements are unlikely to change and therefore we won't need to go back and change plans.

Development and Collaboration Tools

We wanted a method of keeping in contact with each other throughout the entirety of the project, so we created a Discord server in order to communicate with each other.

- Discord was the best choice for our group as all of us are Discord users, so familiar with the software. This also means that we will see messages quicker than other apps which we may not use as often.
- Discord also allows screen sharing and video calls for collaborative work. - The Discord server was split into separate channels, e.g. assets, scheduling, debugging, requirements. This is so we could keep the server organised and to stop information becoming lost in one big main channel, which would save on search time.

Alternatives discussed were google meet and skype. Google meet was considered as a choice as we chose to use Google drive and docs to store our project, however we found that discord had more functionality and we all used the software regularly, so it would be easier to keep in touch.

- Skype was rejected as a choice due to none of us having the software, and not having functionality such as channels that discord has.

As mentioned before, we used Google drive and docs as cloud storage for our deliverables and collaborative pieces of work.

- This means we are all working on the current and updated deliverables, and Google docs even allows us to work on the same document at the same time.
    - We chose Google docs over a local document editor such as Microsoft Word, so that we can collaborate easier on the project. Google docs also provides a timeline of edits so we can track the contributions of each team member.

In order to ensure that we had a level of version control for our code base, we agreed to use Git as version control for our project, and GitHub to store and update our code.

- GitHub allows us to easily collaborate and work together on the same code.
- GitHub allows us to easily maintain iterations of the code and allow us to revert changes in case of mistakes
- As it is a cloud platform, GitHub would allow us to work remotely
- In addition, Github is a widely used version control therefore we would be able to find resolutions for any issues we have online.

To track what work has been completed or is still outstanding, we decided to use a Trello workspace to log what work is still to be completed, and to track what we have already finished.

- Trello works well for plan driven development, as it means a Trello board can be set up at the start of development once plans have been created, and worked through to the end of the project. It can be used to judge our progress of creating the game.
- Following our analysis of the requirements and after creating a Gannt chart of work that was to be carried out, we created a basic Unified Modelling Language (UML) diagram using Lucidchart that represented the classes we thought we would need to implement to create this game using the requirements we elicited. We then used this diagram to cluster certain classes together - mostly via similarity, or if it was immediately obvious which classes would inherit from each other. The UML diagram ensured we had a quick and easy way to refer to the organisation of our game.

We decided to use LibGDX as the game engine for our game. This is because:

- LibGDX is one of the largest Java game engines, this has the advantage of greater availability of support if we encounter problems during development.

LibGDX is very clearly documented, making it easy to locate the necessary methods and variables to make progress on our game.

4.b Approach To Organisation

The general overview of our team's approach to organisation was to analyse the requirements in detail, discuss each member's skill level in regards to Java programming, and then split the requirements into workable 'chunks' in order to assign them to individuals to work on.

- As we are able to contact each team member via the Discord server, we could instantly notify the team of any queries or updates at any time of day. In addition, we could also use Discord to communicate via voice calls for anything urgent or requiring finer detail.
- Furthermore, we also discussed who was more proficient at programming and who was able to provide more of an administrative role within the team. This then allowed us to assign appropriate roles to the more challenging programming tasks to team members with the most relevant experience, and allowed those less able at programming to write up the documentation, oversee asset creation, keep documents/the website up-to-date, etc. this discussion lead to the assignment of roles within the group:

Role	What the role entails	Assignee
Lead Developer	Support the rest of the team and be the first point of contact for any programming related questions	James McNair
Secretary	Maintain structured meeting notes and ensure nothing is missed during meetings	Dan Wade
Librarian	Be the first point of contact for any enquiries relating to deliverables.	Marc Perales Salomo
Report Editor	Manage quality control for deliverables.	Robert Murphy

The remaining two members (Alice Cui and Charlie Crosley) , though not assigned a specific role, will support all other roles, take an active part in development and research.

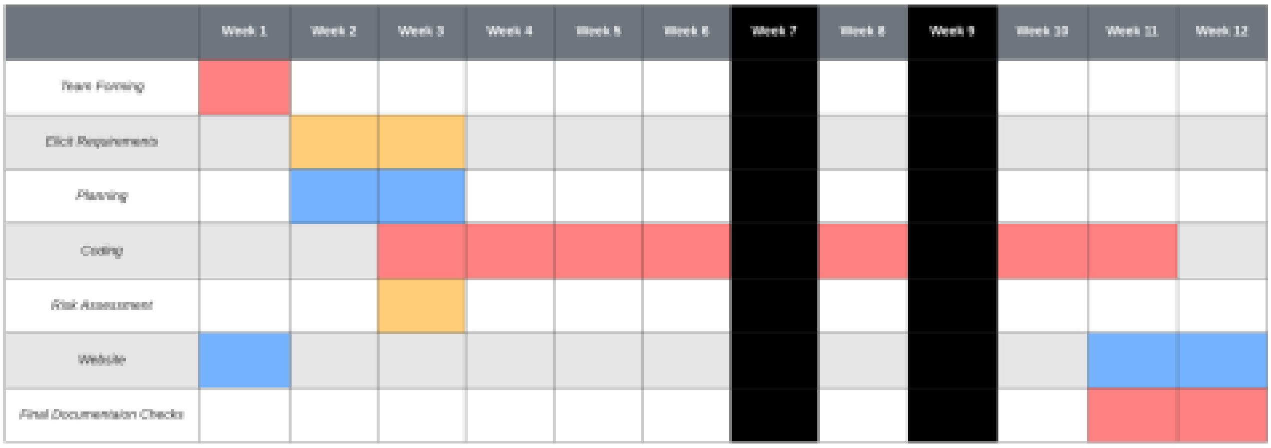
We also agreed that if anyone was able to make good progress or finish their section, then they could help out with other team members who needed assistance. This would help the project progress smoothly, without having to wait an extended period of time for someone to finish a task that is required for someone else's part of the project.

- Helping each other with outstanding work rather than moving onto the next planned piece of work for that person also helps us keep to the planned time frame for the project.

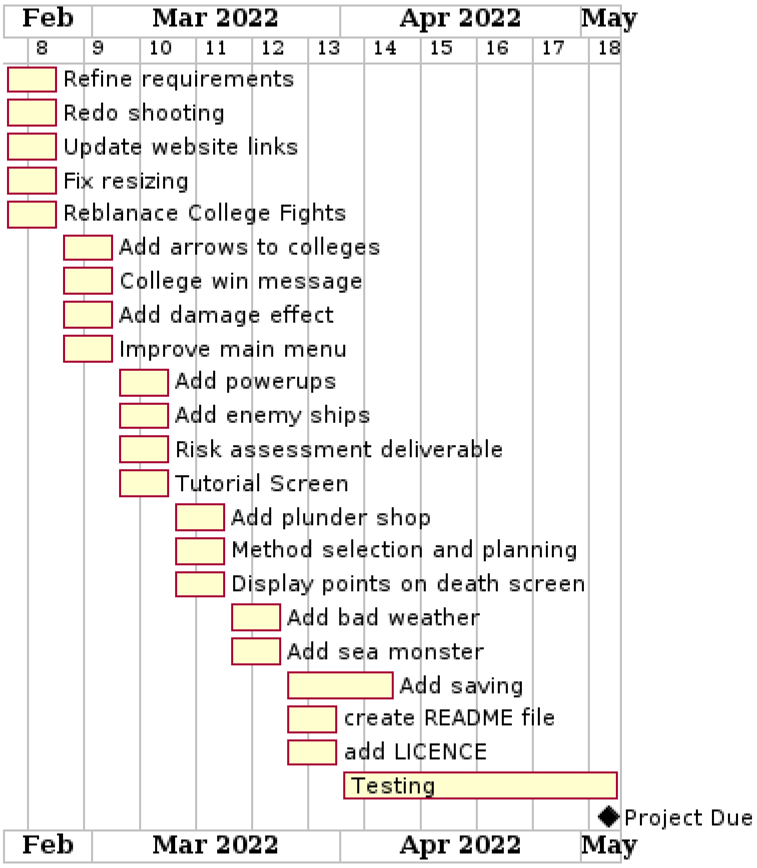
4.c Systematic plan

The plan for our project revolved around utilising our Gannt chart, UML diagram and our "coding chunks" assignment. Another integral part of our plan was also to use our team Discord server to have ongoing communication with each other.

Below is the Gannt chart we used to plan our project within the time constraints:
First assessment Gannt chart:



Second assessment Gannt chart:



These Gannt charts helped us ensure that our project remained on track. We started as soon as possible on assessment 2, firstly updating the game to fix bugs and add features that were required for assessment 1 but were missing. We then refined our requirements table to add the requirements for assessment 2.

- We realised that the programming section of the project would require the most time, so this task was allocated the largest amount of time. We wanted to have a finalised codebase by the end of the week prior to submission at the latest.

- Although we updated the risk assessment table in week 3, we all realised that this would need to be updated as we progressed, therefore the risk assessment was an on-going task.
- A copy of the website for assessment 2 was created in week 1, and contained a link to all of our resources and web pages that we would use throughout the project. The website was kept updated by Marc.
- In addition, we also allocated plenty of time after coding to ensure that we had buffer time in case we were delayed or behind so that we still had time to complete the assessment. This also allowed us to spend a lot of time on testing.