

Method Selection and Planning

Cohort 1 Team 5 /Bitcrushed Bob/

Maryam Mathews

Joseph Hinde

Jacob Mace

Will Aston

Zathia Jacquesson-Ahmad

Bulganchimeg Munkhjargal

Evan Weston

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.

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.

To support teamwork and organisation throughout the project, we used WhatsApp for day-to-day communication since it allowed the whole team to stay connected, share updates instantly, and discuss issues without delay.

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

The team structured its work by separating the project into two main sections technical and non-technical and then breaking down responsibilities within each section. We kept the setup from Assessment 1, which had originally been established through the ORSC 'Design Team Alliance' exercise, because it had been effective and enabled everyone to contribute in ways that made the best use of their individual strengths.

This way of organising the team proved successful as it aligned with agile principles, supporting independence, transparency, and ongoing collaboration. With responsibilities clearly defined, each member could focus on their individual tasks at times that suited them, which improved workflow efficiency.

To maintain coherence across the project, we met each week to ensure that all parts of the work integrated smoothly and that progress remained on schedule and plan the next tasks that must be done to keep the workflow as smooth as possible.

Project plan

For organising our weekly workload, we used a structured plan supported by Gantt charts. These charts helped us visualise the order of tasks and understand how each activity relied on the completion of others. By breaking the project into weekly goals, every team member could focus on manageable tasks directly connected to their assigned deliverables.

Our project plan divides the work between two main groups, each responsible for a different set of deliverables. This approach allowed us to progress on multiple areas at once while ensuring that everyone had a clear focus. The groups were formed collaboratively to ensure that all members were comfortable with their responsibilities and understood how their work contributed to the overall project. Before assigning roles, we reviewed the product brief to identify the key deliverables and determine which tasks needed collaboration between both groups.

A significant part of the project involved updating the Assessment 1 deliverables, so they aligned with the Assessment 2 requirements and reflected our group's direction. This responsibility was shared across the whole team, as any changes made to these documents also needed to be recorded in the change report with a clear justification. This ensured consistency, clarity, and transparency throughout the project.

The software implementation was led by the technical group, who were responsible for developing and improving the game. Although they handled the coding, their work relied on ongoing communication with the rest of the team to ensure all features matched the updated requirements

and design choices.

User evaluation took place once a stable version of the game was ready. This task was assigned to the non-technical group, who prepared the evaluation plan, created consent materials, and gathered feedback from participants. This stage required strong communication skills, as it involved working directly with users and ensuring ethical handling of participant information.

Continuous coordination between both groups was essential for smooth progress. Weekly meetings allowed everyone to stay aligned, share updates, and resolve any issues quickly. This consistent organisation helped maintain a steady workflow and supported collaboration across the entire team, making it easier to bring all components together into a cohesive final product.

Weekly plans

Weekly plans

Week 8 (Due 25 Nov)

Key tasks:

- Review the updated game code
- Record which requirements are currently fulfilled
- Identify requirements from the previous assessment that are still missing
- Review testing materials and select an appropriate testing approach
- Analyse the new product brief
- Draft initial ideas for achievements
- Begin development of the leaderboard feature
- Update the project requirements in line with the new product brief
- Begin drafting the change report and change log
- Research continuous integration tools and methods

Week 9 (Due 2 Dec)

Key tasks:

- Complete the leaderboard, including name input functionality
- Finalise the implementation of remaining game events
- Summarise the chosen testing strategy
- Begin creating tests for inherited and existing code
- Finalise the continuous integration deliverable
- Complete all requirement sections for the change report
- Complete the architecture section of the change report
- Complete the planning and method selection section
- Finalise the risk section of the change report
- Complete the written user evaluation plan
- Prepare participant information sheets and consent forms

Week 10 (Due 9 Dec – Final Meeting Before Winter Break)

Key tasks:

- Continue implementation tasks as needed
- Expand and refine testing materials and test coverage
- Finalise remaining documentation tasks
- Prepare for the next stage of the user evaluation process

Initial goals:

- Get editable versions of all deliverables from the previous team. To be completed by
- Make a working website that matches the previous team's site and shows all our project documents. To be completed by
- Update the Assessment 1 documents so they fit the Assessment 2 requirements. To be completed by
- Create a working version of the game for user testing. To be completed by
- Carry out user testing on the game. To be completed by
- Improve the game based on the feedback from testing. To be completed by
- Complete and submit all deliverables for Assessment 2. To be completed by

○

