Method Selection and Planning

Group 7

Charlie Meader, Sylvia Bulch, Emilija Dudko,
Sam Laljee,Eisvinas Daujotas,Max Sweetman

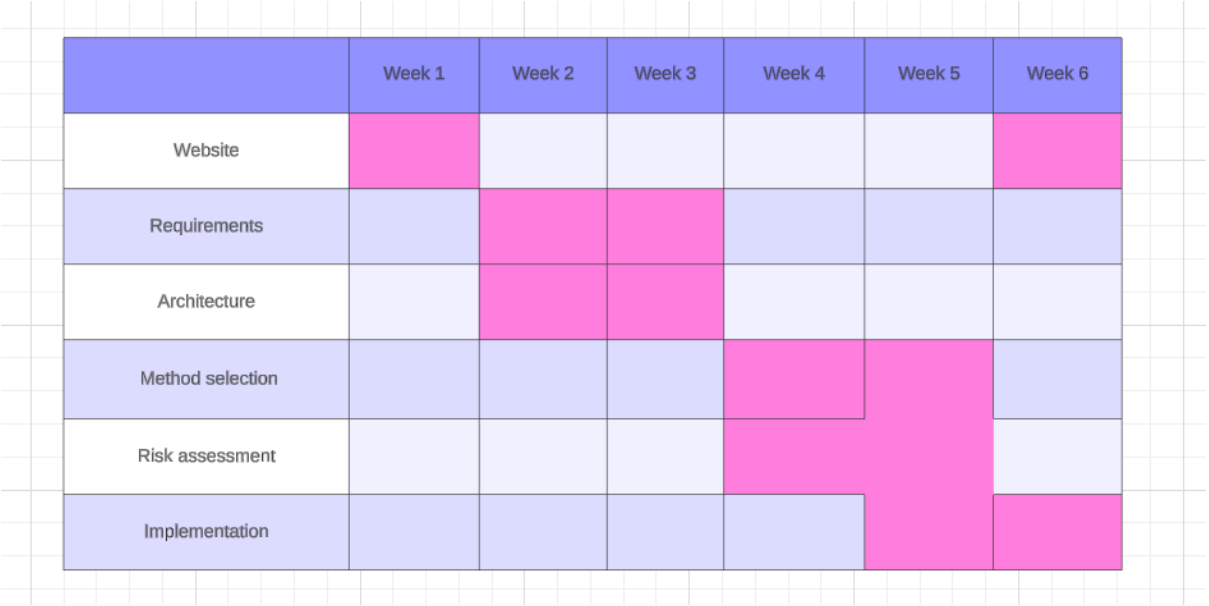
To support the project development, the team has utilised a range of collaboration tools for easy communication. GitHub acts as the primary face of the project. With it allowing for code to be shared easily, it is our primary resource for the implementation stage. Git allows for version control of our software meaning we can keep track of edits over time. Git also allows commit messages, which are useful to keep track of when certain edits are made, as well as who made the changes.

Additionally, this is where all work can be accessed once the development has concluded. This means the website will host links to all PDFs created, plus any diagrams that are mentioned within them. Furthermore, the group has committed to having all work shared within a Google Drive folder. This folder contains the current documents including links to the website that are being worked on.

For team organisation, we have searched for a model which includes collaborations between members and where the work can be split and assigned to people. Allowing for round-about flexibility and space to work on tasks everyone was comfortable with. As such, we have followed the SCRUM methodology. Our process is well reflected in the Gantt chart below and for the reason of its usage, the agile method works with our collaborative style and quick-paced style of working. We had the project broken into multiple sections for simplicity. The project being divided into phases means that the team can work simultaneously on multiple tasks.

While SCRUM was our choice for work, an alternative methodology considered was the XP. However, with the simplicity of our project, we decided against overcomplicating the process we would follow for the project to be designed.

Systematic plan:
The Gantt chart below provides a snapshot of the team's priorities every week up until the presentation date. The team has a rather flexible schedule regarding the work, meaning no fixed deadlines (excluding submission) and as such, this is reflected in the Gantt chart below, with tasks being expected to take more than one week to be finished.



We chose to set up the website in week 1, as GitHub allows setting up static websites with ease, but we still needed to edit the website in the final week to update it. We chose not to update it constantly, as we used other methods to coordinate the files mentioned earlier.

Requirements and architecture could only be done in week 2 at the earliest as this is when we conducted our customer meeting. We chose to split requirements and architecture into two groups of 3 as we felt that these tasks could be completed to a reasonable schedule alongside each other. After we finished this, we split into two groups of 3 again for method selection and risk assessment. These two tasks were started after the Requirements and Architecture were concluded as it allowed for a basic plan for the team to follow. This also allowed us to focus more on specific tasks rather than asking a group of 3 to work on two things at the same time.

In week 5, with both method selection and risk assessment being done, implementation of the project has started. The plan is for coding to take around two weeks up to the deadline giving the team plenty of time to complete the work. The days leading up to the submission will be focused on double-checking all completed tasks and code.

Throughout the project, the plan has rarely been altered. Only with exceptions where we return to previous works to finalise and add extra details. Other than this, the team has stuck to the plan shown by the Gantt chart quite closely.