Method Selection and Planning

Group Number: Cohort 2 Team 7

Group Name: pickNmix

Group Member Names:

David Lun

Harry Muir

Alex McBride

Phyo Lin

Sameer Minhas

Engineering Method:

The chosen methodology for developing the game is based off of the Rapid Application Development methodology or RAD. This is an agile development methodology which revolves around the development of prototypes without worrying about the end product, something that applies to us only needing to make a minimum viable product.

Planning:

When doing the planning it has mostly been through the meetings we have had which are mostly informal. Given each member has their own unique schedules, whenever we have had a meeting we usually have one or two members missing. So we post in Discord important information to the rest of the group and sometimes improvise what is being done during certain meetings when specific key members are missing.

Organisation of each member's tasks were done through an in person meeting, each individual doing one or two parts of the project and its deliverables, for some requiring two members.

The planning for the game project was done in one meeting and the rest of the game implementation was done through many prototypes before we eventually reached the minimum final product.

Collaboration Tools:

The bulk of our decision making and allocation of roles have been made in person. Whether that be designing as a group what the architecture should look like or what roles each of the members have. This has been pretty helpful as it allows for more information to be shared among the group more effectively, however in the case that one or more members are absent the productivity sometimes dwindles which results in time being wasted. This helps with the RAD methodology by allowing communication of more detailed information regarding the implementation and architecture of the game project.

When it comes to collaboration tools used, the main tool has been Git and Github to work on the project in a collaborative way. Git was chosen as it was deemed the best tool for the purpose of the game project. We use it to allow multiple people to work on the game project as well as allow multiple people to work on the website. Another use case is to share our planning snapshots where we have created a PlantUml gantt chart which visually represents each member's assigned tasks and the time frame of when they did them. Git has been very useful for the game aspect of the project as it allows for the separation of development into branches which avoids code clashing with each other, allowing for easy prototyping which suits the prototype heavy nature of RAD. However, this has only been useful within this scope of the project besides the website development as every other aspect of the project. Namely the documents needed are done in Google Drive which is better suited for this role. Another issue with Git that we have come across is the learning of how to use it and set it up at the beginning as a couple members of the group weren’t too familiar with it.

Discord is used as a means of communication between all the members whenever we have a scheduled meeting as well as communicating important information from our in person meetings to those who were missing. It was chosen primarily because of how many members have already set it up. During the implementation stage, it became a means of discussing technical details regarding what code would be best for the game and other things regarding the Github repository in a quick and convenient way that everyone can view at any given time. It has done its job well and there haven't been too many issues besides one member needing to just set it up. This helps with our chosen development methodology by allowing for almost immediate feedback between developers to allow quick iteration and changes to the project.

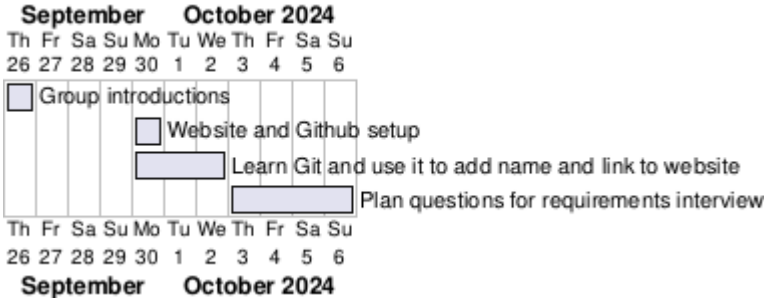
Google Drive is another tool we use to collaborate with each other by allowing the sharing of documents. This was chosen due to how familiar everyone is with using it, there wasn't any consideration for any other alternatives. This was helpful when we were making notes for the customer interview when we needed to share and then compile the notes for the requirements needed for the game. This is also the main tool used for writing up our deliverables for the project which are viewable and editable by other group members so the checking or improvement of documents from different members are easily done.

We used Lucidcharts to discuss and design the architecture, first as a group in person. We were deciding on what to use instead of Plantuml as we didn't know any collaborative tools that use it. This allowed multiple team members to view the chart and make their respective decisions off of that, mainly for the architecture document and the implementation of the classes for the game. This did the job for aiding the design of the first architecture diagram; it hasn't been used since and has been replaced with Plantuml for newer architecture diagrams as well as the Gantt chart for the planning snapshots and progress monitoring.

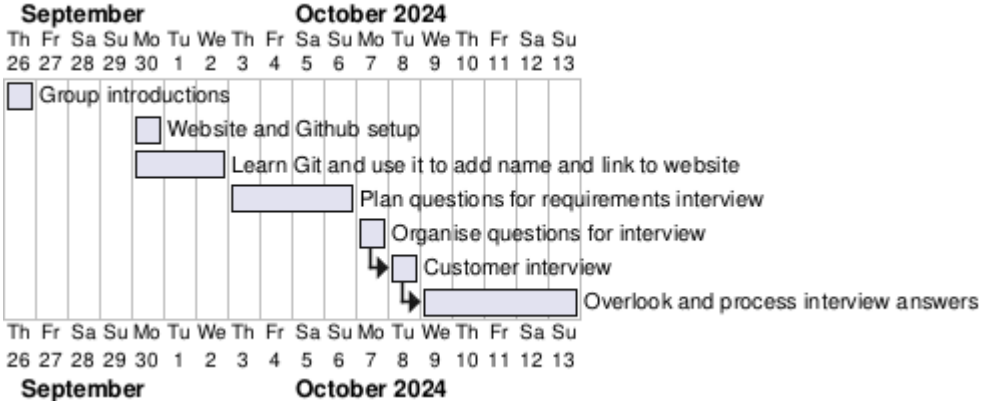
Plantuml, although not a tool in of itself, has been our current and primary tool for creating architectural designs and gantt charts for our planning needs. Its text based nature makes it easy to distribute through Git as well as its integration with Obsidian, a markdown editor, makes it ideal for both quick remodelling of the architecture as well as an easy way to represent our individual tasks.

Weekly Snapshots:

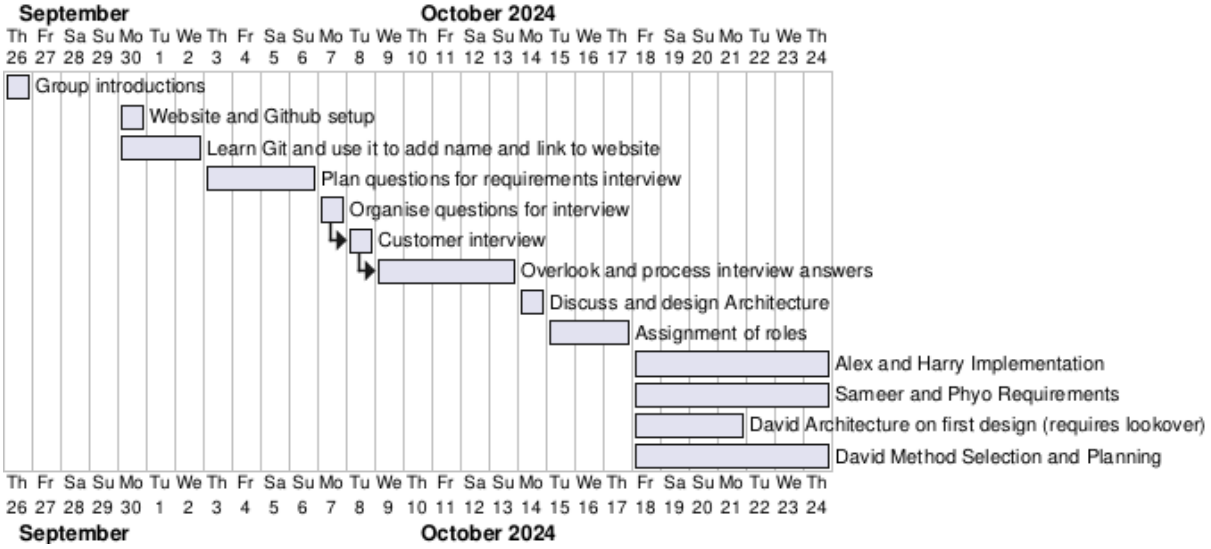
Snapshot for Week 2:



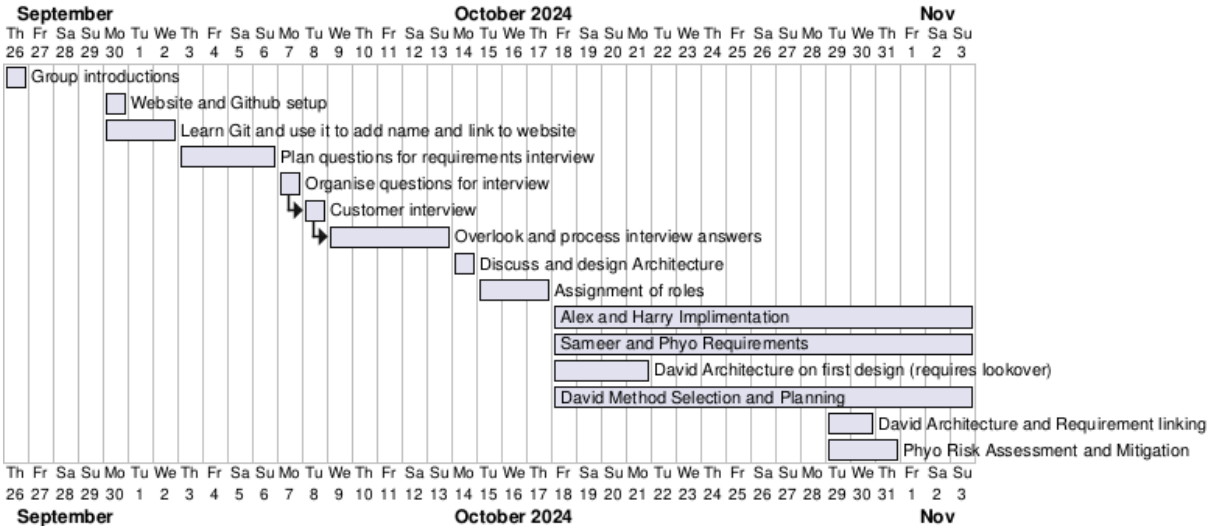
Snapshot for Week 3:



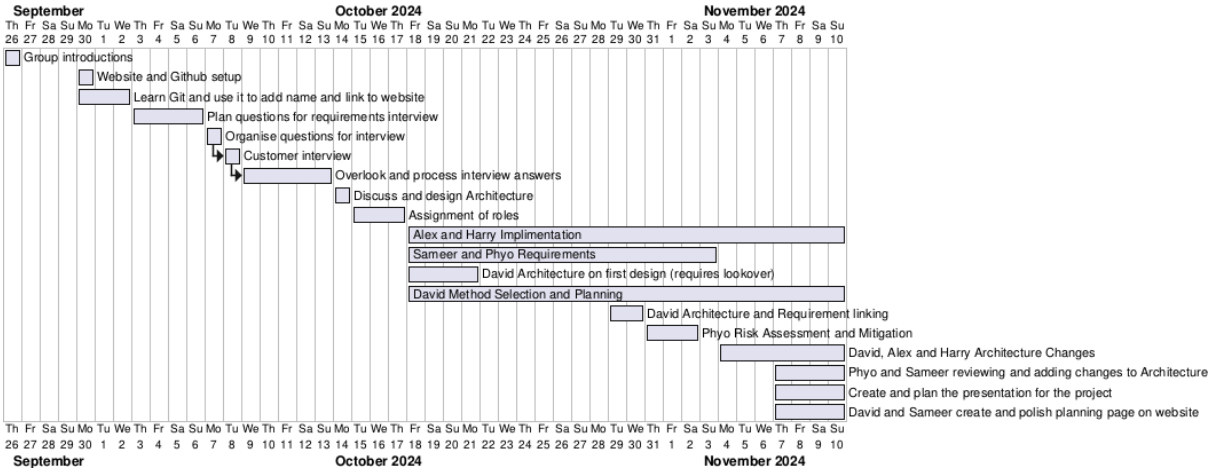
Snapshot for Week 4:



Snapshot for Week 5:



Snapshot for Week 6:


September
October 2024
November 2024
Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su
26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10
Group introductions
Website and Github setup
Learn Git and use it to add name and link to website
Plan questions for requirements interview
Organise questions for interview
Customer interview
Overlook and process interview answers
Discuss and design Architecture
Assignment of roles
Alex and Harry Implimentation
Sameer and Phyo Requirements
David Architecture on first design (requires lookover)
David Method Selection and Planning
David Architecture and Requirement linking
Phyo Risk Assessment and Mitigation
David, Alex and Harry Architecture Changes
Phyo and Sameer reviewing and adding changes to Architecture
Create and plan the presentation for the project
David and Sameer create and polish planning page on website
Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su
26 27 28 29 30 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10
September
October 2024
November 2024