Method Selection & Planning

Assessment 2

Group 3:

Ben Howard <bh1219@york.ac.uk>
Cai Hughes <cabh500@york.ac.uk>
Harry Richardson <hr1040@york.ac.uk>
Ivan Ndahiro <in597@york.ac.uk>
James Sutton <jis509@york.ac.uk>
Yuzhao Liu <yl5164@york.ac.uk>

Group 6:

Igor Smollinski <is942@york.ac.uk>
Pranshu Dhungana <pd861@york.ac.uk>
Zack Tyler-Kyle <ztk500@york.ac.uk>
Phoebe Russell <pbr508@york.ac.uk>
Sanjna Srinivasan <ss3264@york.ac.uk>
Sam Savery <sgs527@york.ac.uk>
Ewan Hutcheson <@eh1776york.ac.uk>

Methodology & Tools Phase 1

For this project, we shall be using the SCRUM methodology. The reason we chose this methodology was that it takes a short amount of time to complete the project, with a small number of people in the development team. The key is the customer's needs and this SCRUM enables us to adapt quickly to changes in requirements, and updating the customer on our current progress to provide a more satisfactory end product. Additionally, since the project development would happen over Term 1 and Term 2, resulting in a change of timetable, SCRUM allowed for the extra flexibility. Scrum is an agile methodology based on iterative sprints (short periods of time where we'd work on a set number of features). Each week, we would have a sprint and during the weekly meetings, we discuss our implementation and new goals for the next sprint. This made sure we kept on track of the core features and understood what features each member (those working on the implementation) would implement along with whether there were any problems during the previous sprint.

We used GitHub to host the project and Git for version control. These applications work hand in hand with Git allowing us to push project files to the online repository. This allowed those coding the project to remotely update the project and to revert changes / look back on previous versions if needed. Furthermore, using our chosen methodology was made very simple since we could all push and merge our respective implementations and organise required tasks using the GitHub Projects page.

Our team used a range of IDEs which included:

- ➔ Eclipse
- ➔ IntelliJ
- ➔ VSCode

Overall, we picked LibGDX as the development framework for the game. This is due to it being reliable with many resources available, use of Java which many members were familiar with and it was suitable for 2D games which we planned to make. To collaborate, we used Discord, an online instant-messaging program. This helped us communicate, organise and have meetings remotely. We initially chose WhatsApp but made the switch as we realised that Discord would provide a better experience for the team due to its file sharing features and how it's primarily used on PCs, Laptops and Mobile Phones while the majority of users only use the mobile version of WhatsApp.

For our website, we used GitHub Pages to host the site as well as using its domain name. This seemed like the most appropriate option since we could easily link our implementation and other documents using the GitHub repository. To store and collaborate on deliverables and any additional documents we used Google Drive and Doc. This allowed all of us to work on the documents simultaneously and at any time.

Method & Tools Phase 2 (Team 6)

We have kept the same Team Organisation Techniques above for Phase 2 of the project unless stated below:

We have decided to use GitHub to host the project and GitKraken for version control. This is because our team used GitKraken during Phase 1 of the project as we find it to have a better layout and easier to use than Github Desktop. It also integrates well with GitHub and we had no issues cloning the repository to GitKraken with all the changes from Team 3.

We also decided to use a Kanban board in GitHub to plan our project. This will help us with our Scrum Strategy.

Finally, we decided as a group to only use one IDE which was IntelliJ as multiple IDEs can cause confusion between the team and that some of the features available in one IDE might not be the same as the other IDE. This could potentially cause version conflicts while committing to GitHuB/GitKraken. Our team also used IntelliJ during Phase one of Development and we believed that IntelliJ was the best tool for the job building this game because of its features and integration with other softwares.

Team Organisation Phase 1

We have assigned different roles amongst the team members. We would arrange meetings twice a week on Mondays and Wednesdays. During these meetings, we would check the progress of each member's assigned task to make sure they were on track, and then when the task is completed, assign new tasks for the next prototype. We decided twice a week as we already had a set time to meet on Wednesday, however if any problems arose after, we could meet again on Monday and understand what will need to be done before the next Wednesday meeting.

If members weren't available for the meeting, we would relay information through our Discord group chat. We would also upload a brief summary of the meeting with tasks assigned. Additionally, during the implementation, we would often upload screenshots and discuss aspects of the features we were implementing. This is so all members of the team are up-to-date, whether they are present or not and can provide instant feedback if needed, as well as, allowing people to have something to look back to.

Our group chat was organised into chats for meetings, documents, images, websites, and general. We separated roles into implementation and documentation. For the most part, all of us worked on the implementation. This helped the project as all members of the team could give insight on what works, what doesn't and what needs changing, leading to a better understanding of the task as a team and a better product in the end.

Team Organisation Phase 2 (Team 6)

We have kept the same Team Organisation Techniques above for Phase 2 of the project unless stated below:

One change we have made from Phase 1 of Team 3 is that we are now implementing bi-weekly gantt charts as a part of the SCRUM strategy to help monitor progress and changes to the project overall. The gantt charts represent the 5 sprints we had each of which were two weeks long. This helped Team 3 during Phase 1, and likewise will help us to see how much is getting done each week, who is working on each section of the project and when deadlines for certain sections of the project are. This coincides with the weekly meetings where we discuss progress and allow us to allocate time segments for the gantt each week.

Furthermore, due to Team 3 struggling to keep consistent track of the above points during Phase One, we decided to use a Kanban board to help us organise the specific progression of a section and allow us to break down the whole project into more manageable chunks that a person/pair could complete in a short session without feeling overwhelmed. We have specifically assigned tasks to people according to strengths and hence know what each person/group is working on at any given time. We have given tasks priority and a size so it allows the group to know what to focus on and it allows us to track progress more precisely i.e. if the section is only partly done or completely finished. Essentially, it gives us more detail about what's currently going on as a group as just the gantt charts were too vague to convey overall progress of each section but allowed us to see how the project was progressing linearly. This will also act as our Planning Board for all tasks that

need to get done and the Sprint Review board which is all the objectives that are getting done in this current sprint in Scrum Strategy.

Lastly, we have made sure during Phase 2 that roles in the project are more defined and that we are working in small groups when completing sections so we can give each other feedback as a part of SCRUM as last time we worked independently mostly and there wasn't must feedback between team member which hindered progress. For example, Phoebe & Sanjna are working on Change Report and updating deliverables, Igor, Zack and Pranshu are working on Implementation/Continuous Integration, and Ewan and Sam are working on testing. We are now holding regular weekly meetings and engaging in two-week sprints as a part of SCRUM. This way each small group can work separately doing their own section independently but also check up on other sections during the sprint meeting such as the progress of the whole report, monitor change and to avoid confusion on different parts of the project. We also found that big group calls everytime we worked together except for progress meetings were a bit overwhelming as people were either talking over each other trying to work on different parts of the project or weren't able to concrete so we found holding smaller meetings with your group easier as you can discuss the current section you are working without interruption and then discuss these changes in the big sprint call. We have also designated Igor as Technical Lead who makes the final decisions of commits/merges of code, and Zack as Project Manager makes the final decisions of the overall project to avoid conflicts within the group about decisions as this was not mentioned during Team 3 report which may pose an issue when there a difficult decisions to make about the project.

Planning Phase 1

The plan evolved as people grasped the requirements of the project and as the project went on. Firstly, we had to plan out the roles people were given and what was needed earlier in the process. We outlined what everyone was doing from their strengths and weaknesses. Some team members had never created a game before and were unfamiliar with LibGDX, others were good at documentation and web development. Inevitably, everyone had an assigned strength that they were contributing in the making of this project.

An early task that needed to be completed was the meeting with our customer. This would help in narrowing down what was needed for the project. This took place on the 22nd of November and we decided to send 3 members of the team to this meeting as the task did not require a full audience present and only a few people to listen and note down their desires.

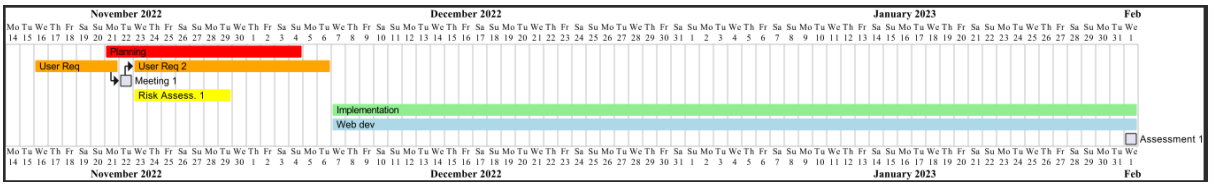
Next, we all contributed to the Risk Assessment in the two meetings on the 23rd & 28th of November. Due to the upcoming winter break, we set specific tasks that needed to be done and that we would begin the implementation stage and work on the Architecture and Method Selection & Planning documents. Each member of the team had an assigned task and we would meet once a week regularly, throughout December and early January, to provide our progress with one another.

After the break, we had meetings on the 16th & 18th of January to recap what we needed to do. Our plan had to evolve as certain features, such as the leaderboard and food stations, needed extra work before completion and documents needed finalising. This led to members working on specific parts of the project and allowed us to stay focused on what needed to be done.

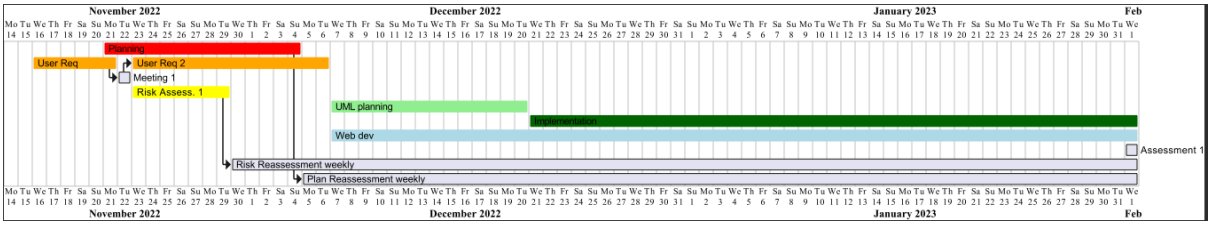
In the weeks of the 23rd & 30th of January, we completed these tasks and focused on last minute adjustments such as replacing any temporary assets and finalising the commenting of our code. To be safe, we increased the number of meetings we had which ensured we included everything from our brief and could bring up any issues easily with each other present

Below find the following gantt charts:

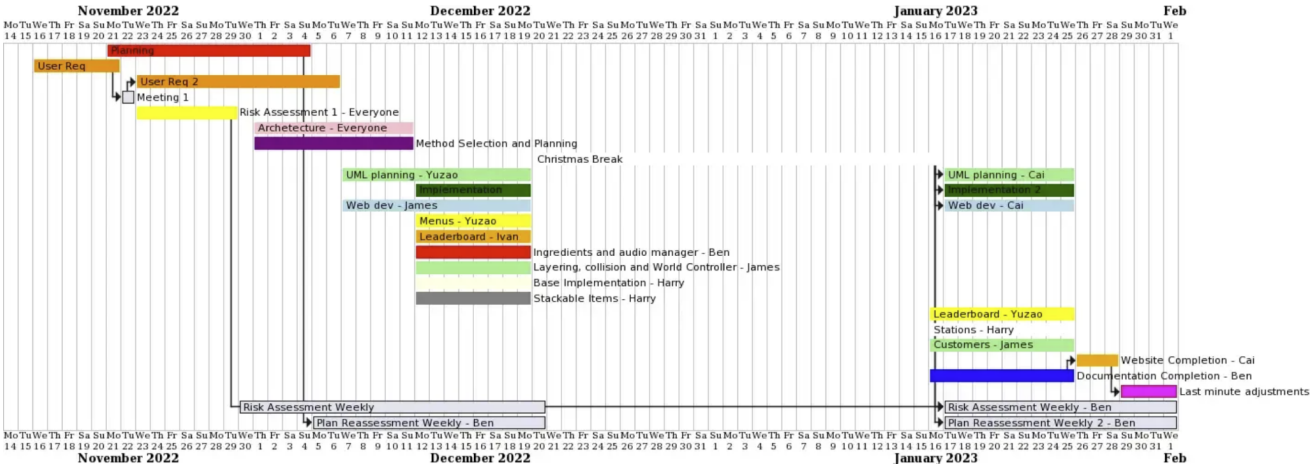
Initial gantt chart - 21/11/2022



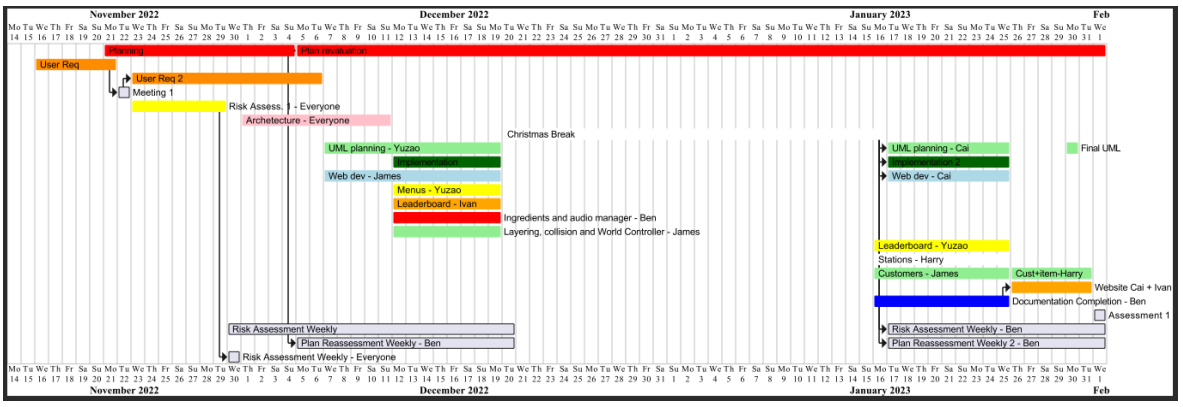
Second gantt chart - 3/12/2022



Third gantt chart 19/01/2023



Final gantt chart - What happened 31/02/2023



Planning Phase 2 (Team 6)
First sprint
Planning :
-Split up into groups depending on what section each person is the most comfortable with and occurrences to mark distribution.
-Briefly look over code and deliverables and see if there are any major errors or bugs to be looked at and record these.
-Do research on unit tests and start implementing simple unit tests to test the existing systems and their functionality.
-Read feedback from Team 3 to see what other changes we have to make to team 3 project better in accordance to the markers.
-Start making a draft of potential changes to deliverable for change report
-Start implementing new features that were mentioned in the brief for phase 2.
-Integrate team 3 repository into GitHub and GitKraken so we can start working on it.
-Set up workflow for continuous integration and automated testing.
Review:
-We split into different groups with Phoebe & Sanjna on Change Report and updating of previous deliverables, Zack and Pranshu on Implementation, Igor on Continuous integration/Implementation, and Sam and Ewan on Unit Testing.
-We found Memory leaks in the code that was caused by generating text in game and a bug where time did not stop during the mandatory animated tutorial. Also other smaller bugs were found during unit testing such as PlayerData Reading wrong and some GameScreen objects being persistent.
-Started the draft for changes to do with Requirements and Risk
-Started building and documenting another chef, ingredients for pizza, adding mixing and baking stations and adding related textures and graphics to the game.
-Team 3 repository has been integrated into GitHub/GitKraken so we can begin working on it.
-Set up workflow for continuous integration and automated testing
Retrospective:

- Good teamwork and communication, splitting up of tasks evenly and clearly to avoid confusions on task delegation.
- Test the code more thoroughly to identify all types of bugs, in this case, be able to identify memory leaks early on.

Second Sprint
Planning:
-Fix memory leaks and any other bugs affecting the project presently.
-Finish Unit testing the original system and start building new units for new features of the system.
-Finished draft for Requirements and Risk Change report and started working on the draft of the Methods and Architecture Changes.
-Continue to add features from the brief like the extra chef, ingredients and stations for the pizza recipe and related textured and graphics.
-Research into methods continuous integration and tweak workflow

Review:
-Major Memory Leaks have been fixed and old animated tutorial has been removed in game and replaced by a simple image depicting the controls, game systems and what to do. Done mainly to remove cumbersome 40+ second unskippable tutorial from every new game started, but handily this solved the timer issue too.
-Initial Unit tests have been completed and documented. Working on new unit test for each class.
-Requirements and Risk Change report draft has been completed. Work on Draft Method and Architure Change Report has been started.
-Finished building and documenting extra chef, ingredients and stations for the pizza recipe and related textured and graphics.
-Finished Research on Continuous integration and finished final changes to continuous integration.
Retrospective:

- Fixing major memory leaks is a good start. Finishing up parts of implementation and drafts of documentation is good news.
- Maybe doing more work on testing could have been helpful.

Third Sprint
Planning:
-Continue working on implementing unit test for each class and testing each class as we go
-Continue adding features from briefs like ingredients and stations for the jacket potatoes recipe and related textured and graphics. Also the addition of more customers/groups and start working on endless mode.
-Finish change report draft of Method and Architecture. Starting updating deliverables for requirements.
-Start writing up the report section for Continuous Integration
Review:
-Unexpected features were discovered by the team that need to be implemented all through the project including power-ups, saving and difficulty settings.
-Continued working on unit tests
-Completed draft of Methods and Architecture. Started updating deliverables for requirements
-Started write up of the Continuous Integration Report
-Finished working on Jacket Potato ingredients and workstations. Started work on more customers / groups and endless mode.
Retrospective:

- Finishing up the draft of the change report is useful.
- Frequently checking up on VLE could have helped us in noticing the additional set of user requirements needed to be implemented for phase 2 earlier on.

Holiday Sprint:
We have decided to work on sections of the project when we have time in the holidays to catch up on various sections before going into the next sprint.

Fourth Sprint:
Planning:
-Continue work on Continuous Integration document,
-Add new features to all areas of the project
-Continue work on unit tests and the write up
-Continue updating deliverables
-Finish the Continuous Integration Report
-Finish working on Difficulty Modes and Endless Mode. Start adding power ups, saving and difficulty settings.
Review:
-Our team missed a feature from the brief so we need to add it next sprint. This makes a chef and a couple counters inaccessible until you have gained enough money to buy stations so need to add this to all sections of the report.
-Continuous Integration draft document completed.
-Continue work on unit tests and write up
-Finish the deliverable for Requirements, Risk and Methods. Began work on architecture Deliverables.
-Finished draft of Contiguous Integration Report
-Finished more Customers/Groups and Endless mode have been added. Work has started on the power ups, saving and difficulty settings.
Retrospective:

- Finishing up on most deliverables, and implementation of endless mode has been beneficial.
- Thoroughly going over the product brief and making a proper checklist of all new user requirements needed for phase 2 would have helped us avoid missing the greyed out stations feature to be implemented in the game.

Fifth Sprint:
Planning:
-Add new features to all areas of the project including the buying or resources
-Finish writing and unit testing. Finish the report for the unit test.
-Add all features to the website including new deliverables, Update Deliverables and Gantt Charts and any other important links such as the kanban board.
-Finish updating deliverables
-Add any remaining points to the Continuous Integration report.
-Finish the implementation. Finish implementation powerups, saving , difficulty settings and the money station buying system.
-Check draft documents, Code and documentation for any remaining errors
-Submit project
-Work on Presentation
Review:
-Units tests are finished. Write up is also complete
-Website has been updated with everything
-Change Report and Deliverables are complete
-Continuous Integration Complete
-Implementation complete
-Checked documents for errors

-Submitted Project
-Completed presentation notes and slides

Gantt Charts

Sprint 1

ENG 1

Company name

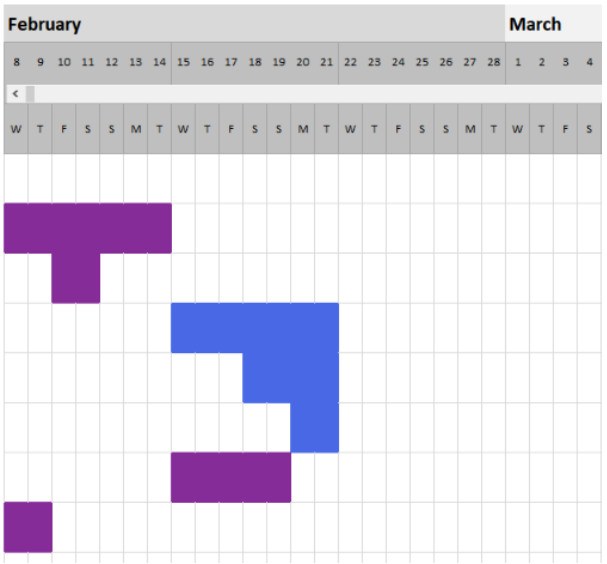
Project lead: Zack & Igor

Project Start Date: 08/02/2023

Scrolling Increment: 0

Legend: On track | Low risk | Med risk | High risk | Unassigned

Milestone description	Category	Assigned to	Progress	Start	Days
SPRINT 1					
Review Project and Deliverables	High Risk	Everyone	100%	08/02/2023	7
CI: Set Up GitHub Work Flow and Automated Tests	High Risk	Igor	30%	10/02/2023	2
Testing: Do initial Unit Basic Tests on the Project	Med Risk	Sam & Ewan	5%	15/02/2023	7
CR: Change Report drafts for Risk and Requirements	Med Risk	Phoebe & Sanjna	5%	18/02/2023	4
Imp: Add Chef, Pizza, Mixing Stations	Med Risk	Zack & Pranshu & Igor	10%	20/02/2023	2
Imp Fixes: Fix Memory Leak and Bugs	High Risk	Zack & Pranshu & Igor	50%	15/02/2023	5
Set up GitKraken and GitHub	High Risk	Everyone	100%	08/02/2023	2



Sprint 2

ENG 1

Company name

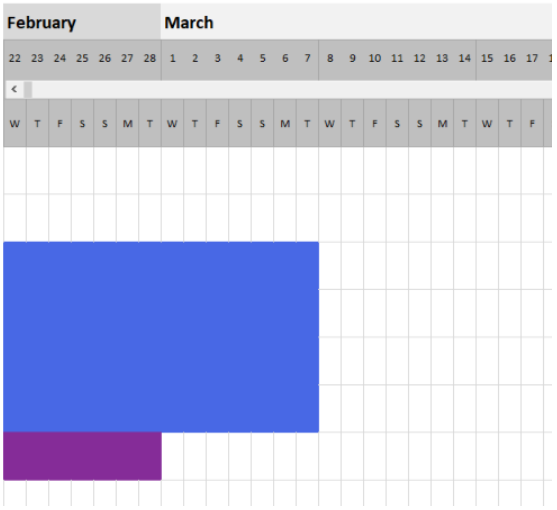
Project lead: Zack & Igor

Project Start Date: 22/02/2023

Scrolling Increment: 0

Legend: On track | Low risk | Med risk | High risk | Unassigned

Milestone description	Category	Assigned to	Progress	Start	Days
SPRINT 2					
CI: Finish Set Up GitHub Work Flow and Automated Tests. Research Methods for CI	Med Risk	Igor	50%	22/02/2023	14
Testing: Finish Unit tests on initial Project. Start Building Unit tests for each Class	Med Risk	Sam & Ewan	20%	22/02/2023	14
CR: Finish Change Report drafts for Risk and Requirements. Start Methods and Architecture Draft	Med Risk	Phoebe & Sanjna	20%	22/02/2023	14
Imp: Add Chef, Pizza, Mixing Stations	Med Risk	Zack & Pranshu & Igor	20%	22/02/2023	14
Imp Fixes: Fix Memory Leak and Bugs	High Risk	Zack & Pranshu & Igor	100%	22/02/2023	7



Sprint 3

ENG 1

Company name

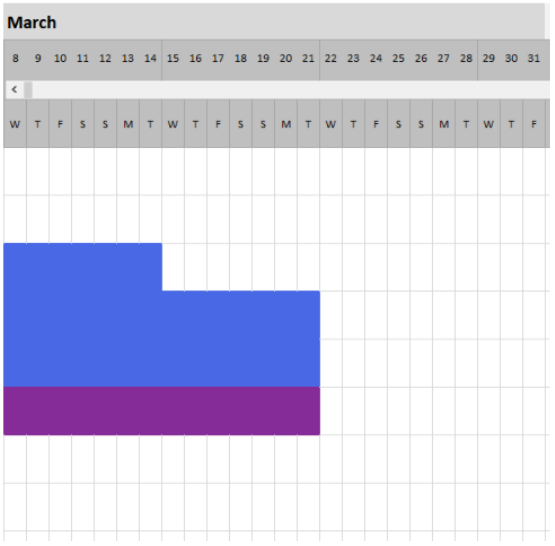
Project lead: Zack & Igor

Project Start Date: 08/03/2023

Scrolling Increment: 0

Legend: On track | Low risk | Med risk | High risk | Unassigned

Milestone description	Category	Assigned to	Progress	Start	Days
SPRINT 3					
CI: Start Write Up For Continious Integration	Med Risk	Igor	65%	08/03/2023	7
Testing: Countine building Unit Tests for each Class	Med Risk	Sam & Ewan	45%	08/03/2023	14
CR: Finish Methods and Architecture Draft. Start Updating Deliverables	Med Risk	Phoebe & Sanjna	50%	08/03/2023	14
Imp: Add Jacket Potato, Work Stations, Power Ups, Saving, Difficulty Settings	High Risk	Zack & Pranshu & Igor	45%	08/03/2023	14
		0			



Sprint 4

ENG 1

Company name

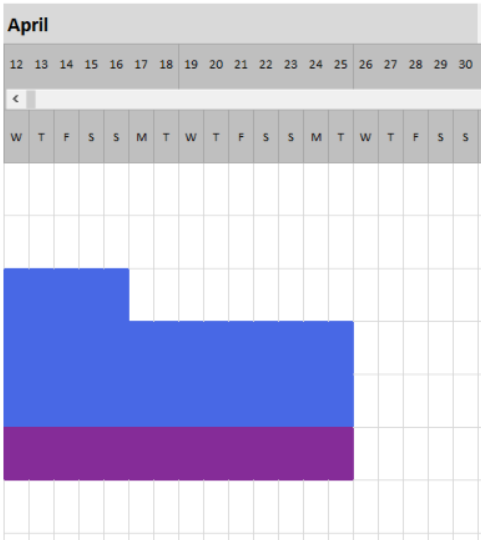
Project lead: Zack & Igor

Project Start Date: 12/04/2023

Scrolling Increment: 0

Legend: On track | Low risk | Med risk | High risk

Milestone description	Category	Assigned to	Progress	Start	Days
SPRINT 4					
CI: Finish Write Up For Continious Integration	Med Risk	Igor	95%	12/04/2023	5
Testing: Countine building Unit Tests for each Class and Write Up	Med Risk	Sam & Ewan	75%	12/04/2023	14
CR: Contuine Updating Deliverables	Med Risk	Phoebe & Sanjna	80%	12/04/2023	14
Imp: Add Power Ups, Saving, Difficulty Settings, Money System, Endless Mode and Groups	High Risk	Zack & Pranshu & Igor	80%	12/04/2023	14



Sprint 5

ENG 1

Company name

Project lead: Zack & Igor

Project Start Date: 26/04/2023

Scrolling Increment: 0

Legend: On track | Low risk | Med risk | High risk | Unassigned

Milestone description	Category	Assigned to	Progress	Start	Days
SPRINT 5					
CI: Finish Continious Integration	Med Risk	Igor	100%	26/04/2023	1
Testing: Finish Unit Tests and Write Up	Med Risk	Sam & Ewan	100%	26/04/2023	4
CR: Finish Change Report and Deliverables	Med Risk	Phoebe & Sanjna	100%	26/04/2023	4
Imp: Finish Implmentation	High Risk	Zack & Pranshu & Igor	100%	26/04/2023	5
Web: Update Website	High Risk	Zack	100%	30/04/2023	3
Check: Check and submit project	High Risk	Everyone	100%	30/04/2023	3
Pres: Presentation Prepration	Low Risk	Everyone	100%	30/04/2023	3