Method Selection and Planning

Assessment 2

Group 5 :

Callum Watson

Jack Guan

Craig Slomski

Chase Mo

Kamrul Islam

Yaseen Khan

4a. Engineering Methods & Tools:

Agile, SCRUM Method:

In Assessment 1, the previous team deduced a scrum methodology would be most appropriate for the circumstance as opposed to Waterfall. We decided to adopt the scrum methodology but adapt it with further detail on product backlog, a sprint backlog, sprint planning, sprint reviews, retrospectives.

A Scrum master was allocated (see 4b. team organisation) to ensure that the quality of each sprint goal is completed and the scrum artefacts are updated.

Change in Sprint Plan:
To detail sprint planning, we initially ordered the features to be implemented from what the developers wished to implement first. For instance, refactoring and implementing any features not implemented in assessment 1 then moving onto assessment 2. However a couple of weeks in, we retrospectively came to the conclusion that we were generating more work later on by pushing back the high risk requirements that were in assessment 2 till later on, which could cause more problems.
Therefore, we altered our strategy, keeping anything we have already implemented but from then on prioritising features of high risk (such as implementing save states) to reduce the later impact of further refactoring the game. The metric for identifying risk is purely from research and speculation from the development team and changes should be made if any future features have been. An alternative method would be using scorecards to measure the population, need, strategy, effort and confidence of each feature. However, this was deemed to be too time consuming for a small scale project and opted for a more simplistic method.
Furthermore, we decided to more clearly outline each weeklong sprint plan beforehand in the Gantt charts. As initially, each Gantt chart only showed a couple of weeks into the future which was not useful for the overall project.

Product Backlog:
The product log was formed from the User Requirements fueling Assessment 2. The first three rows were implemented before the SCRUM change but from there, following the SCRUM method we have prioritised the features that are deemed to be high risk or dependent on other features. They are in order from high risk to low risk in the table.

Feature and User Requirement:	**Note:**
As a player, I can see a leaderboard of local high scores from Scenario mode UR_LEADERBOARD_2	Implemented prior the SCRUM change, Assessment 1
As a player, I can hear background music whilst playing the game UR_SOUND	Implemented prior the SCRUM change, Assessment 1
As a player, I can unlock up to three chefs UR_CONTROL	Implemented prior the SCRUM change
As a player, I can save my progress and return later on an open day UR_CONTINUE_2	Deemed most high risk so prioritised first. Most potential future refactoring
As a player, my mistakes cost me reputation points UR_FAIL_STATE_2	
As a player, I can see fail states that cause me to lose reputation points UR_FAIL_STATE_2	Depends on having reputation points to lose
As a player, I should have access to both jacket potato and pizza recipes UR_RECIPE	
As a player, I should have access to powerups to help me complete scenarios UR_DIFFICULTY	Low priority/risk as it only alters Chef/Customer/Station base stats.
As a staff member, I should be able to configure difficulty for open day players UR_MODE_2	Low priority/risk as only alters Customer count. Should be done close to the end to allow for

	balancing.

Increment:
As each weekly sprint concludes, one or more increments are created demonstrating progress from each sprint and this will be followed by a sprint review. These are reflected in the 4c systematic plan where the progress towards the product backlog are summarised.

Tools:
During the handover to Assessment 2, Google Drive had further uses. It was used to store deliverables and documentation as it has a built-in Word processor as well as version control to allow team members to see contributions to the document which was immensely useful during this phase. During this phase we updated requirements from our customer and used that to update the system and user requirements and constraints. This then informed our updated design of the architecture and the UML diagrams we made such as class diagrams and sequence diagrams. To assist this process plantUML was used again as it provided tools to design UML diagrams, furthermore it allowed for the creation of gantt charts to help log our progress and illustrate the overall timeline of the project. We chose this over technology such as Draw.io, plantUML would ensure consistency as multiple team members would contribute to the architecture diagrams.

For the Implementation Phase GitHub was the main collaborative tool used. It allowed for easier project synchronisation with its features such as pull, push and branching. It also has conflict resolution support which Google Drive lacks to help support the development of the project. In addition, GitHub is useful in the Testing Phase as it allows testing to take place in dedicated testing branches and allows for automatic tests to take place in order During the actual implementation of the project we decided on using LibGdx, a game engine using Java. This was chosen over another engine in consideration Litiengine due to its more active community and more extensive documentation available.
Tiled was used for the tileset and collision of the map in the previous team's assessment and we have adapted that in order to expand the map to import into LibGDX

4b. Team Composition:

Keeping with the principles of the agile methodology to assign roles based on the core strengths of each of our team members discovered through Assessment 1. With half of the team (Chase, Jack, Yaseen) really familiar with coding and implementation with LibGDX and the remaining 3 members (Kamrul, Callum, Craig) of the team being able to support them through testing and continuous integration. In addition, having 2 members familiar with the documentation (Callum and Kamrul) who are able to lead the planning of the overall second stage of the project. Through a team meeting, everyone was comfortable with their roles and to proceed with the project.
Callum was assigned Scrum master and their roles and responsibilities are mentioned earlier in this document.

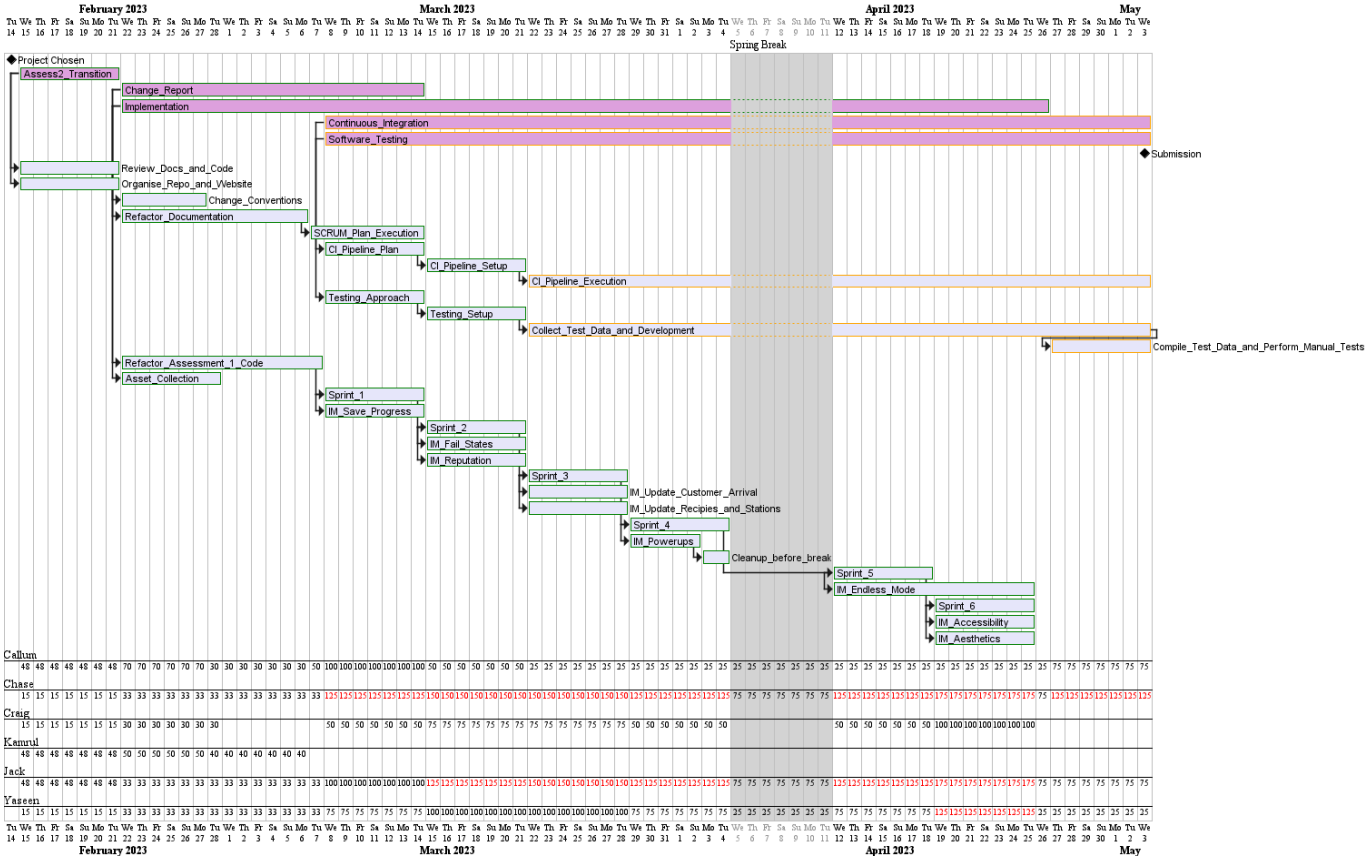
In order to set manageable tasks for each week, weekly online/in person meetings will take place where each of us will set ourselves goals for the week and will be replicated in a Gantt Chart (made in plantUML) with deadlines and team allocation.

To summarise, we have broken down the assessment into its constituent parts, divided our team according to proficiency in writing, testing, continuous development, implementation and project management, and distributed the workload according to these divisions. All of which have a supporting team member in the event a team member falls behind in their workload.
In addition to weekly meetings to track progress and visualising that progress and setting deadlines through the use of Gantt Charts.

4c. Systematic Plan:

All Gantt charts can be found on our website. The larger bars at the top summarise the key activities occurring in this sprint and the smaller bars beneath represent the individual tasks occurring and the time and members allocated to them.
Green outline : Completed task
Orange outline : Task in progress
Red outline : Task yet to begin

The following Gantt is the most recent iteration of the Gantt charts. The previous ones are available on our website.



Familiarisation of code, Starting: (14/2/2023)

The first week of the project was spent by the group becoming familiar with the code and file structure of the project, as well as making a rough plan of what needed to be done over the first 3 weeks. Ended: (20/2/2023)
Upon reflection, we should have spent some more time coming up with a plan that spans over the duration of the project, this is addressed in th

Change log and refactoring code, Starting: (22/2/2023)

In the next phase, the focus point was refactoring the code to implement missing assessment 1 deliverables and adapting the change log. At the time we believed it made sense to implement any missing features from assessment 1 first, however as the plan evolved these became lower priority tasks. The initial change log with the conventions of how to approach the changes were made and work

was done in order to adjust the requirements to suit the customers' new needs, risks were adapted to suit the current situation, considering more testing and implementation risks. Ended: (28/2/2023)

Change log and refactoring code, revisiting plan, Starting: (1/3/2023)

Building onto the previous phase, the change log continued to make progress. Such as: Creating architectures for new solution plans and revisiting the methodology and adding depth into our evolved SCRUM method. From now on there will be weekly sprints with specific and prioritised goals in mind based around risk, this was added to the change log. However, as the team were getting familiar with this strategy, work around Assessment 1 deliverables was wrapped up. Ended: (7/3/2023)

Planning Testing and Continuous Integration ; Sprint 1: IM_Saving_Progress ; Starting: (8/3/2023)

Sprint 1 addressed the features that posed the highest risk to the project according to our project backlog which was being able to save progress and return later.
Meanwhile the team members not currently implementing as this big change took place focused on writing up the planning behind Continuous Integration and Testing, allowing us to trail and set up these in the following week. Ended: (14/3/2023)

Sprint Review:

The previous three points in the product backlog have been addressed previously before key sprints were brought into this. However, in this sprint we have achieved the one goal we set ourselves. It was useful having one goal as this feature requires the most refactoring and would have been difficult implementing features in parallel to this one since it affects a significant portion of the game. This was successfully implemented over the week giving us an opportunity to focus on planning consistent testing and continuous integration for the rest of the project.
Gantt charts have been updated to reflect the whole project plan.

CI and Testing Setup ; Sprint 2: IM_Fail_States/Reputation ; Starting: (15/3/2023)

Sprint 2 addressed the other high risk features addressed in the product backlog due to them affecting a significant portion of the game (chef interactions and customers). This was implementing fail states for existing ingredients and deducting reputation points upon failing to serve a customer. These alter different parts of the game, making it easy to implement in parallel.
Setting up GitHub Actions and formatting manual and automatic testing was a task for anyone not implementing. Ended: (21/3/2023)

Sprint Review:

As predicted, the fail states and reputation worked well with being implemented in parallel and met the goal to be able to challenge the user with failed states of meal preparation and causing the user to end the game when meals were not served.

Sprint 3: IM_Update_Customer_Arrival & IM_Update_Recipies&Stations ; Starting: (22/3/2023)

Sprint 3 addressed some important features from the stakeholder requirements for assessment 2 that were prioritised as they were a greater necessity to the customer. Since customers and the kitchen are separate both physically in the game and at a coding level, both of these features could be implemented in parallel with little collisions on GitHub. These were to as a player, having customers arrive over time and in groups instead of all at once and to give the players access to a greater range of ingredients and stations to cook new recipes.
Following the continuous integration strategies and testing methods set out in the previous two weeks, these methods will be continuously used in the background to the sprints and testing documentation will

be updated to display this. Ended: (28/3/2023)
Sprint Review:
For the most part implementing these features worked in parallel, with the exception to changing customer orders to accommodate these new recipes. However, since this was a small job, this was held back to next week. Meanwhile all other features were implemented, completing the next part of the product backlog.

Sprint 4: IM_Powerups ; Starting: (29/3/2023)

Sprint 4 addressed further important features from the stakeholder requirements from assessment 2 that required customers and stations to be in working order beforehand, or else they would both need to be refactored later as the implementation of IM_Powerups to assist player's experience change how fast stations work and customer patience. Due to the diversity of powerups and it being close to a scheduled break the next week. Implementation was not given too much so everything would be wrapped up nicely for everyone returning. Ended: (4/4/23)
Sprint Review:
With numerous powerups being added including (regaining reputation, pause customer timers, speed up station timers, cash boost and reputation reduce), it left some additional time which was spent ensuring there were no test cases failing or any loose ends that had not been addressed previously.

Spring Break, Starting: (5/4/2023)

A week is set aside for the team to relax and take a break from the engineering project in order to have a more productive start, returning to it over the next couple of weeks. Ended: (11/4/2023)

Sprint 5: IM_Endless_Mode ; Starting: (12/4/2023)

Sprint 5 aims to implement functionality to support an endless mode, which requires more fine tuning, balancing and pulling the rest of the game together. This also is a good opportunity to update the website with data collected from testing and continuous integration. Ended: (18/4/2023)
Sprint Review:
Unfortunately, the endless mode was not finished and sufficiently tested at this stage so it has been passed on to the future sprint for fine tuning. More of the team will be working on the future sprint to accommodate for this (the back up implementation team mentioned in the team organisation section) Gantts have been updated to reflect the extension of this component.

Sprint 6: IM_Accesibility/Aesthetics, Starting: (19/4/2023)

Sprint 6 aims to clear up the previous sprint along with the least risk requirements with minimal code change including Accessibility and Aesthetical/UI changes. This was saved to last as it allowed us time to finish any unfinished code chunks and focus on manual tests and improve the visuals for the customer, ready for presenting. Ended: (25/4/2023)
Sprint Review:
At this stage, the implementation is ready for final testing and compiling test data and moving documents and test data onto the website.

Final Week, Submitting, Website Updates and Presenting ; Starting: (26/4/2023)

The final week's goal is to proofread prepared documentation and upload to the website.
Zip the latest tested version of the game and post it and it's evidence of testing online.
Finally prepare a presentation to our customers due to take place on the day of the deadline. Ended: (2/5/2023)

Submission and Deadline (3/5/2023)

A presentation was prepared to ensure as accurate a representation of our final product as possible. It includes video footage from the game itself as well as places to discuss features to the audience. Everything ready for submission by this date. Ended: (3/5/2023)