UNIVERSITY OF YORK
DEPARTMENT OF COMPUTER SCIENCE

Method Selection & Planning
Engineering 1 - Assessment 2

Group 2, Cohort 1 ("*The JVMs*")

Ben Hatch
Charlotte Sharp
Dan Nicholson
Ethan Andres
Freddie Higham
Joe Silva
Rosie Kern

In this project, we are choosing to use agile methods. There are many reasons agile methods fit this project best. Firstly, agile methods produce working software frequently while adapting to changing requirements [1]. We will be producing the game with reduced requirements for assessment 1, and then adapting to another team's game with the whole requirements scope for assessment 2. These requirements will fit in much better with agile methods than plan-driven methods, as plan-driven methods do not adapt to change nearly as well. This project does not require the benefits of plan-driven methods, like their use in managing large teams and increased focus on documentation. Documentation is important for this project, especially when transferring projects between teams, but it is not worth the overhead to focus on it as much as plan-driven methods require. Our team is of a small size, so the benefit of a central plan is lost as communication is much simpler. Additionally, agile methods allow us to engage with the customer better, as we can produce a functional game for them to provide feedback on. This is particularly useful as we will get many opportunities to communicate with our customer in labs, or meetings when necessary.

We will be using scrum as our agile method of choice. The high-level nature of scrum allows us to tailor it to fit our specific needs, and therefore work most effectively [2]. This advantage over other methods such as extreme programming made selecting scrum the best choice for us.

We will have week-long scrum sprints with a meeting before and after each, where we will discuss plans for the sprint before, and progress after. Focus will be placed on changes which will need to be made for the next sprint if not enough tasks were completed. The daily scrum can be discussed between collaborators on specific tasks to denote how their work is progressing when necessary, but is not strictly required due to the collaboration tools we are using to support the project.

The two main tools utilised for collaboration on this project are Google Workspace for working on documents and GitHub as our frontend for git for version control.

Google Workspace includes Google Drive and Docs, which are both vital components. Drive hosts all of our files relating to documents, mainly Docs files for the assessment and other files including notes and links. It was chosen due to its tight integration with our already existing university emails, so everyone already could easily access and utilise it. Additionally, it provides convenient integration with Docs, which is important due to how much it is used for the project. We chose Docs for several reasons: it is available in the web browser which makes it accessible for everyone, it allows simultaneous editing of documents by multiple people, and it has a detailed version history showing all changes made to the document. This last feature is particularly useful in the case that evidence is required. We considered whether we should use git to version control our whole project, using LaTeX. However, we decided Google Docs was much simpler to use, especially due to higher experience with it in the group compared to LaTeX. For version control in the actual development of the software, git is the software we selected. Git has considerably more advantages when used for software development version control compared to if we used it for the documents; to name some: allowing pull requests, forks, cloning to the local machine to be edited with an IDE and executed, advanced merging, and continuous integration. Therefore, something like Google Workspace was not considered for this part of the project. We looked at other version control software briefly before settling on using git. Git's decentralised nature compared to other

version control like SVN is beneficial when developing. This as well as more experience in the group meant using git was the logical choice for us. Git will be very useful during scrum sprints, as it allows for easy modification of the software by multiple developers.

To work with git, we are using GitHub for our central repository to allow for better management of continuous integration, documentation, tracking of issues, and pull requests. We selected GitHub, as it has many very useful features for collaboration, is simple to get started with, many group members have prior experience with it, and it provides the ability to easily configure a website using GitHub Pages. The tracking of changes on GitHub will be important during scrum sprints, as they allow people to view the daily changes in a much more concise and simple manner. Additionally, continuous integration will allow for the easy deployment of functioning software at the end of each sprint, letting the executable be accessed easily by everyone and shown to the customer.

For communication, we are using Discord primarily, with email as a backup or if otherwise more useful for specific purposes. Discord is free, easy to use and set up, cross platform, and every team member has experience using it. In case there are issues with Discord, everyone has access to email each other using our university emails. We considered using Slack instead of Discord, but the listed benefits of Discord led us to choosing it, especially considering we do not find any of the benefits of Slack to be important enough.

Our team's approach to team organisation closely aligns with agile methodologies that are widely used nowadays. In particular, for our project we have chosen to use Scrum methodology as it is well-suited for both team and the project because of its emphasis on collaboration, adaptability and iterative development.

One of the key aspects of our team organisation is fostering collaboration to utilise experience, knowledge and skills of each member to implement the game and all deliverables. From the beginning, we started building trust and respect among team members, creating an environment that ensures that everyone feels comfortable sharing any ideas and potential solutions. This approach created an atmosphere that encourages brainstorming and allows us to view the project from multiple perspectives, which is crucial in the development of our project. Thus, enhancing creativity and problem solving, we easily prepare questions for the customer interview which are a fundamental part of defining user requirements. Next approach that ensures that all team members are kept informed and involved in the development process is open communication. This is achieved through utilising Discord as our primary communication method that is familiar to all team members. This enables us to keep everyone up to date with any progress and challenges that might arise, enhancing cohesion within the team.

Another cornerstone of our team organisation is task allocation based on individual skills, preferences, and knowledge. Given the project's time constraints, this approach was a strategic decision crucial for saving time and ensuring the proper execution of each development step, leveraging the expertise and expressiveness of each team member.

We took a more flexible approach to team organisation as we did not assign fixed roles to members however we did assign tasks to each member based on their own capabilities. We did this by discussing what part each of us felt most confident working on then assigning tasks best suited to each member. We also monitored the distribution of tasks to ensure an equitable breakdown of work among team members. As a team, our goal is to optimise productivity and prevent burnout by ensuring that each member is engaged in tasks aligned with their strengths and interests.

The last but not least important approach is feedback. As we utilise agile methodologies for our project delivery, we adopt an iterative approach to development, making feedback from other team members is crucial. Typically, it is achieved by either informing everyone during each team meeting or, if it is urgent, by posting a message in Discord, which serves as our primary communication channel.

We aim to deliver a high quality product that meets the requirements of product brief while maximising team efficiency and satisfaction. Therefore, our team's approach to organisation aims to be well-suited to both the team dynamics and project requirements. This is achieved through qualities such as collaboration, open communication, adaptability, effective task management and feedback. Additionally, our team places a strong emphasis on continuous improvement and learning, striving to adapt to changing circumstances and refine our processes as needed.

The project follows a scrum-based approach, with weekly splints, defined key tasks, starting and finishing dates, task priorities, and dependencies. Additionally, for better visualisation of how our plan evolved, we have created Gantt charts which were updated as our project progressed. The weekly snapshots of them can be found here.

Following the scrum based procedure we had opted for we split the work down for assessment 2 into work packages using a gantt chart and put this data into a table with a unique ID and description of the task. We then split these down further into individual tasks which were put into a table with a unique ID, description, start date, end date and any dependencies that exist around the task. This was appropriate and essential as it created an outline for an order of when tasks should be completed in line with any existing dependencies. For example we could complete testing for collisions until the final changes to the map had been made.

Work breakdown structure:

ID	Name	Description
WP1	Change report docs	Make suitable changes to the previous teams Requirements, Risks, Plan, Architecture and Implementation documents
WP2	Change report	Update the Change report on all changes made and justify them suitably
WP3	Continuous Integration	Set up the framework and outline the teams procedure and the outcome
WP4	Implementation	Implement the final version of the game and update the document with any further addition made
WP5	User Evaluation	Conduct 7 user evaluations and compile the data, making appropriate recommendations
WP6	Software Testing	Write both automated and manual tests for the system and report the findings
WP7	Website	Edit the website so that it contains updated versions of the documents, game and repository.

Dependencies:

ID	Name	Description	Start date	End date	Dependencies
T1.1	Requirements	Update the requirements document	22/04/24	29/04/24	None

T1.2	Risk Mitigation	Update the risks document	22/04/24	29/04/24	None
T1.3	Method Selection and Planning	Update the Method Selection and Planning document	22/04/24	19/05/24	None
T1.4	Architecture	Update the Architecture document	25/04/24	02/05/24	None
T2.1	Change Report	Keep a record of all changes made to the documents and justify changes made	22/04/24	04/05/24	T1.1 T1.2 T1.3 T1.4
T3.1	CI framework	Set up the framework to enable CI practices	22/04/24	25/04/24	None
T3.2	CI document	Write the CI deliverable document commenting on the plan and the outcome	07/05/24	10/05/24	T3.1 T4
T4.1	Map Changes	Add all locations and interactions to the map	25/04/24	04/05/24	T1.4
T4.2	Score Function	Implement a function to calculate the ending score	25/04/24	02/05/24	T1.4
T4.3	Streaks/ achievements	Implement a way for a user to achieve 3 possible streaks	30/04/24	04/05/24	T1.4 T4.2
T4.4	Player name input	Implement a way for the user to enter their name	25/04/24	28/04/24	T1.4
T4.5	Leaderboard	Implement a leaderboard to show top 10 scores	30/04/24	05/05/24	T1.4 T4.4 T4.2
T5.1	Plan user evaluations	Recruit participants, prepare observations templates and environment	02/05/24	04/05/24	T4
T5.2	Conduct user evaluations	Conduct 7 user evaluations and record the data	04/05/24	10/05/24	T5.1
T5.3	Compile user evaluation data	Using the data observed compile the data to form recommendations for edits	10/05/24	14/05/24	T5.2
T5.4	User evaluation doc	Complete the user evaluation outlining the procedure and findings	10/05/24	17/05/24	T5.3 T5.2

T6.1	Write tests	Create automated tests and manual tests for the implementation	25/04/24	06/05/24	T4
T6.2	Conduct the tests	Run the tests and report the findings	30/04/24	08/05/24	T4 T6.1
T6.3	Test report	Report the findings of the final tests in the document	08/05/24	12/05/24	T6.2
T7.1	Website	Edit and update the website with updated documents	14/05/24	18/05/24	T1 T2 T3 T4 T5 T6

Week 1

Beginning from the first team meeting during practical sessions, we actively engaged in team forming, examined the brief of the product, and defined the potential shape of our project. Starting from the first practical session when we met each other, we began developing a team identity, which helped us to familiarise with each other, thus facilitating how our team will operate for the rest of the project. Another important aspect of our meeting was the identification of the software that we would be using during the project development. We decided to mainly focus on using Google Workspace to collaboratively work on our project deliverables whilst using GitHub as version control tool. Besides that, we ensured that all team members agreed on how we would keep in touch and discussed potential approaches to team meetings and tasks. In the case of our team, Discord became the main tool for communication that everyone is comfortable with. The next activity that we undertook was an online treasure hunt where the members made a start at looking things up and started to think about who might be responsible for each subtask and we also started thinking about tasks itself. That activity enhanced our vision of the resources that would be needed in order to complete this project. Additionally, we established the initial project structure and implemented a simple CI.

Week 2

The second sprint of this project, which spanned from 22nd to February 28th began with planning of the workload for the incoming week and defining any obstacles that may occur along the way. Luckily, at this early stage, we did not accumulate any backlog that might need to be dealt with. One of the most important aspects of the development process of any project is establishing outstanding communication with the stakeholders. That is crucial to ensure that their feedback has been heard, helps identify and mitigate risks effectively and contributes to the decision-making process.
Having examined the product brief once again, we started to identify potential questions that might arise during the implementation of the project and could be answered by the product stakeholders. We compiled them all together and discussed the availability of each team

member to arrange the meeting with the stakeholders. Besides that, we started planning the project by splitting the work into smaller chunks and primarily assigned team members to certain tasks, which was supported by the desire and experience that each team member had towards the task.

Week 3

During our third sprint, which started on February 29th and finished on March 6th, we conducted an interview with a customer that provided us with valuable insights and answers to our questions. We then analysed the interview and ensured that all data was nicely stored on our team's Google Drive. This interview analysis also served as a great addition to the brief for defining both system and user requirements.
The next crucial step that we undertook was splitting the work into smaller tasks that could be assigned to the team members. In our team organisation, we did not adhere strictly to role assignments; instead, we assigned tasks based on deadlines. Additionally, during that week, two members of the team identified risks and iteratively kept monitoring and updating them over the next two weeks.
From the implementation part of the project, the team member responsible for it created a main menu and developed player movement examples. They also focused more on exploring the LibGDX game engine, as these steps did not depend on user requirements. Meanwhile, the other two team members began developing them. Another team member started to work on the website while concurrently collaborating on the architecture with another team member.

Week 4

During the week from March 7th to March 13th, our team focused on reviewing all the tasks required for the project and assigning team members to work on deliverable documents. With an emphasis on producing documentation deliverables, two team members continued working on the User Requirements document, while another two started working on the Risk Assessment and Planning document. Additionally, we employed a Responsibility-Driven-Design approach to determine the main themes and components of our game. This involved creating the first prototype of our architecture using Class-Responsibility-Collaborator cards, utilising defined Candidates. This allowed us to lay the groundwork for proper implementation starting from the next sprint. However, during this sprint, we encountered a setback when our initial attempt to load a tile map failed, resulting in a backlog for the following week. Despite this, the issue was not critical and did not significantly slow down our progress. On another positive note, significant progress was made on our project website, with it nearing completion by the end of the sprint.

Week 5

During the final sprint, which started on March 14th and finished on March 21st, we focused on finalising all deliverables and preparing everything for submission. This sprint was arguably the most intensive one, as we aimed to complete all remaining tasks. The main focus of this week was on game implementation, which included making player movement work, fixing problems with the tile map, adding all interaction parts, and ensuring the collision

system was functioning correctly. Additionally, we worked on implementing the end screen and score calculation. We also finalised our User Requirements and Risk Assessment documents and focused more on finishing the Architecture and Planning documents. Despite the intensity of this sprint, our team remained committed to delivering a high-quality product within the project's time constraints.

Overall, our project progressed very smoothly without any major issues with task completion. This was largely due to the effective team organisation and the responsibility of each team member towards their assigned work.

Week 9

This week was treated as the starting point for assessment 2 as we had just inherited the project from the previous team. We held a meeting at the beginning of the week where the primary focus was to understand what needed to be done and design a time frame for the tasks, creating a rough master plan which would be the influence for creating our gantt chart. The work breakdown for this week was to: update the document deliverables, update the change report, set up continuous integration workflow, begin to write tests for the existing implementation and expand the map so it included all of our final locations. While this was a lot of work to complete we were able to delegate tasks effectively throughout the week and collaborate with implementation ideas during the practical sessions. Even though we fell short of a team member due to illness, progress was still made this week without any major issues because when delegating tasks we considered the bus factor, meaning a different member of the team was able to take on extra tasks without overworking.

Week 10

This week our aim was to complete our implementation so that we could start writing tests and have time to make any necessary changes. We held a meeting at the beginning of the week where we discussed our visions for implementing the leaderboard and the streaks, to ensure we were all on the same page about how the score function would be affected by the changes. We came to the conclusion of having 3 streaks for movie watching, sports and going into town all with comical names linking to the marketing user requirement. This led to a discussion on any changes we wanted to make to the design of the game and then changed the sprite of the player as we felt it was not very suited to the market of students. During the rest of the week we split down into teams to focus on different parts of testing; automated, manual and user. Freddie, Joe and Dan worked on creating automated tests; Ethan, Ben and Rosie worked on creating and running manual tests; and Charlotte did the planning and preparation for the user evaluation. The aim after this was complete was to start conducting the user evaluation, giving time to compile and make recommendations next week. During the course of the week we kept on track with the plan and made good progress with the implementation, testing and kept up to date with the change report.

Week 11

The main focus of this week was completing user evaluations and beginning conducting the automated and manual tests on our system. During the meeting at the beginning of the week we conducted a test run of the user evaluation to ensure it was working then completed 2/7

of the actual user evaluations. Upon completing these we were able to already identify common usability problems with the system to do with the visibility of features on the map so started to plan changes we wanted to make and were able to implement small ones before conducting the remainder of user evaluations. Once these were complete we were able to compile the data and make any final recommendations for changes to the current system. While 3 members of the team made these changes the remaining 4 members started work on the deliverable documents which were the user evaluation report, the testing report and the contiguous integration report. Whilst making progress on these documents we were also able to finalise the change report and all the deliverables within the report as well meaning that we could start to make our submission folder. During this week we completed all the necessary steps for this project and the only remaining tasks were to complete the various reports and publish our website.

Week 12

During the beginning of this week our focus was finalising any small minor changes that needed to be made before submission. This included checking all the change report deliverables and ensuring that they were consistent with the change report itself, and making finishing touches to the testing, continuous integration, user evaluation and implementation documents. This process was managed by 2 members of the team as not many changes were needed and it allowed for others to make progress elsewhere. As well as making final changes to the documents, some small additions were added to the implementation, which was completed by 2 members of the team, and the final branch of “eval changes” was merged to the main branch which completed our implementation. The remaining 3 members created a plan for the presentation and started to draft out scripts and slides to be reviewed later on in the weekly meeting. During this meeting at the end of the week we proofread all the deliverables and finalised any changes we wanted to make, then these were uploaded to the website. Once this was completed our focus was on the presentation, where the 3 members who constructed it rehearsed in front of the group and any final changes were made. Our work this week was sufficient in that we felt submission ready and had time to prepare for the presentation.

Week 13

As this week was submission and presentation week, the focus was on rehearsing our presentation. We planned to submit early as we felt confident in our deliverables and wanted time to focus on preparing for any questions that may be asked of us during the presentation.

References

[1] I. Sommerville, “Agile planning” in *Software Engineering.* Harlow: Pearson, 2016, pp. 680-682.

[2] K. Schwaber and J. Sutherland (2020, Nov.). *Scrum Guide.* Scrum Guides [Online]. Available at: https://scrumguides.org/scrum-guide.html [Accessed: 10 March 2024].