UNIVERSITY OF YORK

DEPARTMENT OF COMPUTER SCIENCE

Method Selection and Planning
Cohort 2 - Group 16 (Skloch)

Group Members:

Charlotte MacDonald
Hollie Shackley
Luis Benito
Kaustav Das
Sam Hartley
Owen Gilmore

Originally by: Noah Forryan, Lewis Morgan, Naufal Tun Thamanian, Tom Owen, Dan Manby, José Fernandes of Group 19

a)

Team 19 decided to use the game engine LibGDX for several reasons such as it allows them to write game code and it has a great number of features for the development of the game like graphic rendering, physics simulation and much more. It also integrates well with the chosen map editor (Tiled Map Editor) which makes incorporation of external resources into the game easier and faster. It's known for its performance optimisation that allows the game to run smoothly and efficiently. It's an excellent choice for developing Java-based games. Team 19 considered a few other game engines: LWJGL and Slick2D. LWJGL has good support and an active community but a very steep learning curve so was more suitable for more experienced programmers. LibGDX also has these advantages and more so team 19 chose that instead. Slick2D had the benefit of simplicity and a focused 2D development experience but its development has not been as active as LibGDX and therefore team 19 decided to choose LibGDX. As we, team 16, had also used LibGDX, we continued to use this.

As we had used IntelliJ in assessment 1, we decided to continue with this for assessment 2 as all team members had become familiar with it. IntelliJ was originally chosen as it interacts well with Gradle which LibGDX relies heavily on. VSCode was also considered but it didn't interact well with Gradle.

To collaborate on the code, we used Github. Team 19 had been using Github so it made sense to continue this and we had also used Github for assessment 1 so all team members were familiar with it. Alternatives, such as Apache Subversion, were considered at the start of assessment 1 but no members had experience with tools other than Github. Team 19 had hosted their website on Github as had we so we continued to host the website we inherited for assessment 2 on Github.

Team 19 used Gantt charts to visualise the project timeline and task dependencies. This was useful to see the tasks' start and finish dates. We used PlantUML for these Gantt charts as we had used it in assessment 1 and it integrates well with Google Docs. PlantUML was also used for architecture diagrams and the work breakdown structure so using it for Gantt charts reduced the bus factor as a large proportion of the team are familiar with it. Alternatives, including Mermaid and Graphviz, were considered in assessment 1 but these were decided against.

For the design of the map, team 19 decided to use 'Tiled Map Editor' because it offers several advantages over other popular map editors that they looked into. It's easy to use and allows developers of all skill levels to create maps rapidly and efficiently and it also has a wide range of customization options which permitted team 19 to create the map following the specific requirements. It supports multiple layers for organising all the map elements (background layers, object layers, tile layers, etc) which allowed team 19 to enhance the details of the map. Finally, it integrated with team 19's chosen game engine (LibGDX) without any need for additional conversion. As we had also used Tiled Map Editor for assessment 1, we continued to use this.

For easy communication, team 19 decided to use Discord given that it has a comprehensive set of features that helped maintain organisation within the team and facilitated communication amongst the team members. Team 19 also used WhatsApp due to its convenience and ease of communication. We had also used both Discord and Whatsapp in assessment 1 which worked well so we continued with this.

For the other deliverables, team 19 had been using Google Drive and we had used Google Drive for assessment 1. Continuing to use Google Drive reduced the risk of formatting issues and allowed for easy tracking of changes made to deliverables. This made writing up the change report easier as we could see exactly what had been added, changed or removed.

In assessment 1 we had decided to follow an agile approach. Reasons for this still applied to assessment 2 and we were experienced with this so it was efficient and easy to stay the same. As in assessment 1, multiple stages of the process were being completed at the same time; plans were expected to change significantly as there were extra stages, such as continuous integration and testing, that a lack of experience made hard to predict the timing of; and there were flexible interactions with the customer. Assessment 2 also continued with a short time frame, face-to-face conversations and regular reflections which are all preferred by agile approaches [1].

We had taken inspiration from scrum in assessment 1 due to already being committed to weekly meetings. These weekly meetings continued into assessment 2 so weekly sprints were still a logical choice. At the start of each meeting, we had a review and retrospective to reflect on what did and didn't happen and any issues that came up. We then used this to plan the next week. This had worked well in assessment 1 and kept us on track allowing us to completely finish on time. It was therefore chosen to continue with this.

In assessment 1 we also drew a small amount of inspiration from the spiral lifecycle. This was because the assessment placed importance on good documentation control and a large amount of documentation [2]. Although there was slightly less of a focus on documentation in assessment 2, we still used this when it came to the change report and reporting on continuous integration, testing and user evaluation.

b)

Each week we had one face-to-face 2 hour meeting which fitted well to agile and scrum. Within these meetings we reflected on the previous week, planned for the following week and discussed the deliverables. Brief notes were taken detailing our progress so we wouldn't forget the work that each group member was doing and as we had one team member who wasn't able to join face-to-face meetings for a few weeks. For the team, this was beneficial as it held all members accountable for their workload so if someone was underperforming the group would find out and be able to rectify the situation which certainly increased the group's effort.

When we wrote the risk assessment at the beginning of the assessment, we decided to always have a bus factor of greater than one. This meant that if there were any circumstances where members of the group couldn't complete their tasks, there would be another group member to take over their workload. The exceptions to this were the website and assessed presentation due to the small number of marks allocated. For the website, change report and implementation, sections were allocated to team members with experience of these from assessment 1 to speed up the process. Equitable work allocation was followed so that each team member was assigned to 15 marks and at least one deliverable leadership role. The deliverables were assigned as follows:

Deliverable	**Team member(s) assigned**	**Proportion(s)**	**Leader**
Website	Sam	100%	Sam
Requirements change	Luis	100%	Hollie
Architecture change	Charlotte	100%	
Planning change	Hollie	100%	
Risks change	Kastauv	100%	
Change summary	Hollie	100%	
Implementation	Sam Owen	43% 57%	Owen
Testing	Charlotte Kastauv	33% 33%	Kastauv

	Sam Owen Hollie	11% 11% 11%	
User evaluation	Hollie Luis	50% 50%	Hollie
Continuous integration	Charlotte Kastauv	50% 50%	Charlotte
Assessed presentation	Luis	100%	Luis

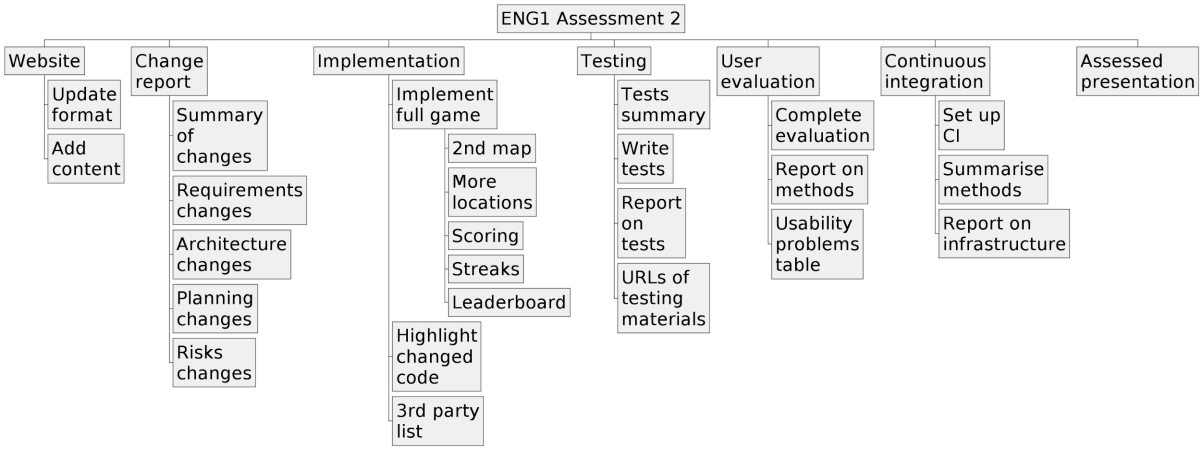
After writing an initial risk assessment, at every meeting, we reviewed the risk assessment and updated any changes. We did this for each meeting with the whole group which allowed everyone to contribute allowing all the potential hazards to be discussed. This made us more organised because the whole group was aware of these risks. In assessment 1, we assigned risk management roles. Owen and Sam were product owners, Charlotte and Kaustav were team leaders and Hollie and Luis were project managers. As we were experienced with these roles and felt that we knew what we were doing with them, we continued with this organisation.

As in assessment 1, the customer provided the role of upper management in case of issues that couldn't be resolved within the team but this was not needed. Communication with the stakeholder in assessment 1 began with an in-person client interview followed by smaller conversations in weekly practicals. In assessment 2, we had a video client interview to discuss the assessment 2 requirements and then had smaller conversations in weekly practicals to update the customer on our progress and to ask any questions that had come up regarding the deliverables and the requirements.

Decision making was primarily done through a democratic process based on the overall opinion of the team. Opinions were usually unanimous before discussion but where they weren't each opinion was explained by the team members who held it to allow other points of view to be considered. This then resulted in team members coming to the same opinions or finding a compromise. If this hadn't happened, the deliverable leader would have been given the final decision and upper management would have been consulted if needed.

c)

Work Breakdown Structure



Assessment 2 Deliverables Table

ID	Title	Due date	Description	Visibility	Relevant tasks
D1	url2.txt	23/5	Website	Shared	T1

D2.1	Change2.pdf	23/5	Change report	Shared	T2
D2.2	Questions for client	19/4	Preparation of client questions	Internal	T2.1
D3	Impl2.pdf	23/5	Licencing information	Shared	T3.6
D3.2	Code	23/5	Implementation	Shared	T3
D3.3	Executable JAR	23/5	Finished game	Shared	T3.5
D4	Test2.pdf	23/5	Testing	Shared	T4
D5	Eval2.pdf	23/5	User evaluation	Shared	T5
D6	CI2.pdf	23/5	Continuous integration	Shared	T6
D7	Assessed presentation	23/5	Presentation to client	Shared	T7

Assessment 2 Tasks Table

Task ID	Description	Start date	End date	Dependencies	Priority
T1	Add all assessment 2 content/links to website	17/4	22/5	T2-6	High
T2.1	Prepare for and have client meeting	17/4	19/4		High
T2.2	Change requirements documentation	17/4	1/5	T2.1	High
T2.3	Change risks documentation	17/4	22/4		High
T2.4	Change planning documentation	17/4	22/5	T2.3	High
T2.5	Change architecture documentation	17/4	8/5	T2.2	High
T2.6	Summary of changes	1/5	8/5	T2.2-2.5	High
T3.1	Implement at least 2 more recreational locations	17/4	8/5		High
T3.2	Implement extra buildings and features	17/4	8/5		Medium
T3.3	Implement scoring system including streaks	17/4	8/5		High
T3.4	Implement leaderboard	17/4	8/5	T3.3	High
T3.5	Document code and create JAR	17/4	8/5	T3.1-3.4	High
T3.6	List 3rd party libraries and assets with licences	1/5	22/5	T3.1-3.4	High
T4.1	Summaries tests methods and approaches	24/4	8/5		High
T4.2	Write tests for the code	8/5	22/5	T3.1-3.4	High
T4.3	Write up tests report	8/5	22/5	T4.2	High
T4.4	Testing materials	1/4	22/5	T4.2	High
T5.1	Report on user evaluation methods	8/5	15/5		High
T5.2	Complete user evaluations			T3.5	High
T5.3	Table of usability problems and the severities	8/5	15/5	T5.2	High
T6.1	Summarise continuous integration methods	17/4	8/5		High
T6.2	Set up continuous integration	17/4	8/5		High
T6.3	Report on continuous integration infrastructure	17/4	8/5	T6.2	High
T7	Create presentation for customers	8/5	20/5	T3	High

Plan Evolution

Throughout assessment 1, team 19 updated the Kanban board and Gantt chart during their meetings to show the work they had been doing or would do. This meant they had clear goals and knew which areas each team member should focus on - allowing them to evenly distribute the workload between members throughout the project.

There were a few reasons for the Gantt chart and project schedule changed. Firstly, team 19 initially underestimated how much work the whole project would take. As you can see in the earlier Gantt charts, there were overly optimistic plans for the completion of the project. For example, the first draft of the requirements' time frame changed, as one member realised that a different approach was needed, and needed to start again. Secondly, they realised they needed to do in-depth reviews of each other's work increasing the quality of the work. These delays were not impactful as they had already given themselves an extra day before the deadline giving them some extra time in case of issues.

For assessment 2, we didn't use Kanban boards as this wasn't our established way of working. Gantt charts were still updated each week. After the first week of assessment 2, the change deliverables were behind where we wanted them to be due to confusion over if it was just a change report or if the original deliverables had to be changed to. This was established in the meeting that week to enable progress to continue. Luis had also decided that he no longer wanted to work on implementation so his implementation marks were swapped with Owen's user evaluation marks to continue equitable work allocation. After the second week, a decision was taken to move the completion of all deliverables to Monday 20/5. This was due to an exam on Tuesday 21/5 and keeping Wednesday 22/5 for practice of the assessed presentation on Thursday 23/5.

After the third week, the progress with the requirements change was not as much as had been hoped. This was holding back the architecture change and testing. Implementation progress was also less than hoped so the 3rd party list had not been completed to prioritise the implementation itself and as we were still finding more assets to use. The progress with the change summary was not as much as hoped due to some changes not being complete. Usability problems had to be moved later by one week due to a misunderstanding with when the user evaluations would be completed.

After the fourth week, the progress with the requirements change was not as much as had been hoped. This was continuing to hold back architecture and testing. Implementation was still running behind but was nearly finished. The 3rd party list still needed to be completed due to implementation running behind. The usability problems table had not been completed and the planned progress for the assessed presentation had not been made.

After the fifth and final week, all deliverables had been completed ready for submission. The exception to this was the assessed presentation for which progress was not as much as had been expected. However, this was not a problem as the assessed presentation was not happening until later in the week and therefore there were an extra 2 days for working on this.

References

[1] K. Beck, et al. (2001). Principles behind the Agile Manifesto. Manifesto for Agile Software Development. [Online]. Available: https://agilemanifesto.org/principles.html [Accessed: 13 March 2024].

[2] A. Garg, R. K. Kaliyar, and A. Goswami (2022). PDRSD-A systematic review on plan-driven SDLC models for software development. 8th International Conference on Advanced Computing and Communication Systems, Coimbatore, India, Mar. 25-26, 2022, IEEE, 2022