UNIVERSITY OF YORK DEPARTMENT OF COMPUTER SCIENCE

Change Report

Group 14

Tecch Titans:

Bradley Mitchell

Daniz Hajizada

Ellie Gent

Joel Crann

Keela Ta

Leo Crawford

Lukas Angelidis

Introduction:

After the project progressed through its revision process, it went through a series of enhancements aimed at improving user experience and gameplay mechanics, alongside with addressing any issues faced during the testing stage of the development. This report provides all of the modifications made to the game by the team, highlighting the key aspects of the project, such as Requirements, Method selection and planning, Risk assessment and mitigation and Architecture. The key aspects of the document will be described in greater detail as the document progresses, but for the purposes of this introduction, some of the related terms will be used.

The overall structure of the implementation remained similar to the assessment 1 counterpart, however some changes were necessary in order to implement the new features in the full project brief. This included adding additional events to the EventManager, along with making modifications to the accomEvent by changing when the time is passed, as in the original project the final day was not included in the score; this had to be fixed. The brief states there should not be a hard time limit on the game, so we decided to disable automatic time progression to prevent this being the case. We also added a DailyActivities class so that we could have more intimate detail of what the player did on each day, which is crucial information when calculating the score and streaks. We also refactored the code to allow us to implement unit testing and followed best Java practises to make continuous integration easy to implement with Gradle, and make sure refactor allows Gradle to find dependencies required to add the GdxTestRunner.

Our team used an approach where every task was documented in a local logbook, accessible by all the team members. This logbook detailed who was responsible for each task and the current stage of each task, such as whether it is open, closed, or still in process. This method allowed the team to effectively plan and keep track of the progress and changes made to any aspects of the project, including deliverables. Gantt charts were also used and played a crucial role in representing the schedule and dependencies of the tasks, making it easier to allocate time and resources efficiently.

Major alterations were applied to the requirements section, which had to align with the new criteria set for Assessment 2. Crucial changes include new functionalities and leaderboard.

Architectural changes were necessary to support the new functionalities introduced by us. The updates were made to optimise performance and ensure the safe integration of both existing and new features. We also made sure to follow the same design process that the original team had initially followed when creating the architecture.

The planning and method selection suffered massive changes to manage our team and deliverables. The title page was updated, along with the team roles and responsibilities. The previous Scrum-based planning method was completely abandoned as it did not suit our team's dynamics. Tools like GitHub, for version control, Instagram and Discord, for communication, remained from Assessment 1 for team familiarity.

Risk assessment and mitigation were also hugely updated. The main updates were connected to the risk assessment table, which now includes new risks associated with an additional column.

Throughout the project, we were constantly motivated to improve, driven by the user feedback and continuous requirements changes. The processes, ideas, tools, and conventions applied, ensured a structured and effective improvement of the project. Each section of the report offers insights into the project's progression and decisions behind each change, highlighting our collaborative effort to deliver a high-quality user experience.

Requirements:

This section provides a brief explanation and justification into the changes made to Requirements. The main objective for the changes to this section was to accurately reflect the new requirements released for assessment 2 regarding implementation. Specific modifications implemented are laid out in order of the layout of the original document. All changes made to the original document are displayed in bold for easy referencing.

Some changes were made to the introduction section of the document, such as clarifying that the cohort were only customers for assessment 1, as this was in reference to the project presentation in April. The deadline date for assessment 2 was also added to the end of the introduction to reflect the fact that these requirements apply to both assessment criteria. The last change to the introduction was the mention that a new product brief was released for assessment 2. This provided an explanation for the added requirements in later sections as a reference to where they came from. Most of the content within the introduction was not changed as we did not see this as necessary. Although most of the introduction is focused around assessment 1's customer meeting, this was integral as this was the basis on which most specific requirements were made. A second customer meeting was not held after the extended product brief was released, and therefore there was no need to amend or remove this section.

Within the 'user requirements' section, two new requirements were added, based on the extended product brief. These were UR-MEMORY and UR-STREAK to add the new streak and leaderboard functionalities to the game. With this addition, the user requirements table was re-ranked in order of 'shall', 'should' and 'may'. This was needed in order to maintain readability and order of importance for implementation.

Within the 'functional system requirements' section, four new requirements were added, based on the extended product brief. These included FR-LEADERBOARD and FR-SCORE for the new leaderboard functionality, and FR-STREAK-VISIBILITY and FR-STREAK for the new streak functionality. These were then referenced to the newly implemented user requirements. FR-GAMEPLAY 2-4 were also updated to reflect the assessment 2 requirements by including more than one place for the actions specified.

No changes were made to the 'non-functional requirements' section as we believed that our game met all the criteria for each of these requirements. During user evaluation, we assessed each non-functional requirement and each one was met; the users did not encounter any issues directly related to any of these requirements. No additional requirements were added for this section as we believed that the ones provided by the previous team were still satisfactory for our product.

To view all changes, the link to the both previous groups' deliverables and the updated deliverables can be seen through [this link].

Architecture:

This section provides a brief explanation and justification into the changes made to Architecture. The main objective for the changes to this section was **to properly implement the new requirements added by expanding upon the architecture that had already been designed, following the same methods that they had used to create the initial architecture**. Specific modifications implemented are laid out in order of the layout of the original document. All changes made to the original document are displayed in bold for easy referencing.

No changes were necessary to the architecture design process, as the fundamental design of the architecture remained the same. We built upon the architecture which was derived from the RDD ideology. The tools used section also remains unchanged, as we used the same tools and process for updating the diagrams later in the document.

As there were requirements to be implemented in comparison to assessment 1, the architecture had to change accordingly. The first change was made in the structural diagrams section, describing the additional classes and screens that would need to be added in order to fully implement the product brief as required for assessment 2. A class called DailyActivities was added to the Environment package, which is required for implementing the streaks functionality, and is very useful for calculating a score at the end of the game. Additionally, LeaderboardScreen was added to the UserInterface package, as there has to be a screen to display the top 10 player's scores, as denoted in the product brief. Adding these additional classes to the diagrams proved to make the formatting quite cluttered, so I used a technique in PlantUML which was not used by the previous team. I directed where the arrows and classes should be displayed on the diagram myself in code, rather than just letting PlantUML automatically generate the image, which can sometimes result in text and classes overlapping and becoming unreadable.

The new classes' purpose was then described in the document. Since there had been a change in the structure of the architecture, the final diagrams had to be amended accordingly. Firstly, the final class diagram was updated with the additional LeaderboardScreen in the UserInterface and the DailyActivities in the Environment package respectively. This was done by modifying the initial PlantUML code of the original diagram, then placing the new classes in the appropriate packages. In addition to updating the class diagram, the extended package class diagrams were also updated. The new Environment and UserInterface packages have details about the fields and methods inside each of the new classes, just like how all the initial classes were portrayed in the original deliverable. Underneath these new diagrams are some brief descriptions of the purpose and functionality of these new classes. I have made sure to make a distinction in the document of which are the final diagrams for the assessment 1 requirements, and which are the diagrams for the assessment 2 implementation of the full product brief.

Since there is now another screen in the architecture, the behavioural diagrams must also be amended, specifically the screen state diagrams, to show how this new screen interacts with the other screens in the architecture. A new behavioural diagram is added with LeaderboardScreen included, showing the interaction between the leaderboard screen and main menu; it shows how the leaderboard can be entered via the main menu, and exiting the leaderboard will return you to the main menu.

The final Component-Entity-System diagram also had to be updated with the new leaderboard stage in mind, however the rest of the diagrams were fine to remain as they were. This is because we did not change the way the GameScreen was rendered, nor did we change how a Player interacts with an Object. The layout of the leaderboard screen is similar to the credit screen, however instead of displaying text in a scrollable window, it displays a number of rows of disabled TextButtons, with each row having 3 buttons respectively. The contents of these buttons are obtained from the 2d leaderboard array, which is part of the referenced HustleGame; in each row, the first button represents the ranking, the second button the name, and the third represents the score.

Finally, each of the new requirements (highlighted in bold) were added to the table at the bottom of the page, that relates elicited requirements to the architecture itself. Specifically speaking, UR-MEMORY and UR-STREAK were the additional user requirements added, and FR-LEADERBOARD, FR-STREAK-VISIBILITY, FR-SCORE, and FR-STREAK were the additional functional requirements added. Each of these requirements specifically references which parts of the architecture are used to fulfil the requirements. A minor change was made to FR-GAME-PLAY1-4 in the table, as it seemed like some text had been cut off in the original submission, so I completed the description to make it properly describe the requirement.

We did not need to modify the overall architecture in order to add new maps and buildings, as it was simply a case of creating a new map asset/modifying the campus east map, then changing the current map in the GameScreen. The renderer handled rendering the new map without any modifications needed.

To view all changes, the link to the both previous groups' deliverables and the updated deliverables can be seen through [this link].

Method selection and planning:

This section provides a brief explanation and justification into the changes made to Method Selection and Planning. The main objective for the changes to this section was to accurately reflect the new deliverables presented to us in assessment 2. Specific modifications implemented are outlined in order of the layout of the original document. All changes to the original document are displayed in bold for easy referencing.

The first change implemented was updating the title page to reflect the inclusion of the new team members. Next, a paragraph was removed in the 'software engineering methods' section, as this was a reference and explanation of the use of Scrum for planning, and within our group, this was something we were not familiar with nor used during planning.

In the 'collaboration tools' section, a few changes were made. The previous team stated that they used Google Slides for Scrum reviews in their meetings, and this was removed because, as previously mentioned, Scrum was not used within our planning methods. There was also a comparison of Github to Google Apache Subversion, and this was removed as only Github was considered in our group for the collaborative version control of this project. We chose Github without considering many alternatives as most team members were familiar with it or could quickly learn the UI for our project. Whatsapp was also mentioned as an alternative to using Discord for communication, but this was changed to Instagram for our report, as Whatsapp was never a consideration for our group and Instagram was an app that all group members already had an account with.

Next, in the 'team organisation' section, there were several changes. We had at least 3 team members on each deliverable instead of 2 as we had more members than the previous group, so this was updated accordingly. It was also added to this section that the members of our group that were on Implementation in assessment 1, continued in the same role for this assessment as they felt the most comfortable and experienced with picking up the previous teams' code. Lastly, the deliverable percentage breakdowns were updated to reflect the new team member names, the new deliverables and the percentages to represent each members' contribution.

A new work breakdown structure was created to replace the previous teams' one, as all the deliverables and tasks for these deliverables were vastly different than for assessment 1. The deliverables table was also updated for this same reason, as well as the due dates and relevant tasks.

For the tasks table, around 60% of the original tasks were kept, but with a change in wording to reflect the change report, using words like ‘Amend’ and ‘Update’. This was because the tasks were still relevant for assessment 2 to some degree. For the implementation tasks, the assessment 2 requirements were added, including the updated ones as these were extra components and our main focus on implementation this time around. Sections such as Testing, User Evaluation and Continuous integration added completely new tasks to the table as these are new deliverables for this assessment only.

Lastly, the ‘discussion of plan evolution’ section revolved around the Gantt charts created each week by the team based on their meetings. Therefore, completely new Gantt charts were created to reflect our teams’ names, deliverables and time-frame. The new Gantt charts can be viewed [here] or on the updated version of the Method Selection and Planning document. It was also added that logbooks were used to track our development lifecycle and assess how our planned schedule needed to be amended, so we added this to this section, instead of the previous mention of Scrum. Based on the updated Gantt Charts, the previous groups’ explanation paragraph was removed and rewritten to reflect our performance and scheduling based off of our own charts.

To view all changes, the link to the both previous groups’ deliverables and the updated deliverables can be seen through [this link].

Risk assessment and mitigation:

This section provides a brief explanation and justification into the changes made to Risk Assessment and Mitigation. The main objective for the changes to this section was to ensure that the risk assessment reflected both our team and assessment two, in order to allow easy mitigation of risks. Specific modifications implemented are laid out in order of the layout of the original document. All changes made to the original document are displayed in bold for easy referencing.

The first change to the risk assessment was removing the way they assign roles: they followed a structure that relied on having team leaders/project managers, and risk owners were allocated accordingly. Our group has a much more flat team structure so this did not apply to us. As a result, we then had to modify the entirety of the ownership column to reflect our team’s structure, with owners being based around the teams working on each section as opposed to specific managers.

For clarity, we added an extra column to their table detailing which assessment the risks applied to (one/two/both). This made it easier for us to know which mitigation strategies we needed to keep up with and which were no longer applicable, without deleting prior risks.

We also updated their table with new risks that applied to us after taking over the other team’s deliverables. The new risk 17 has a mitigation strategy for if the code taken over features a lack of modularisation, and risk 19 details mitigation for if the documentation is potentially difficult to understand.

The priority categorisation method used by the previous team (a matrix multiplying the potential impact by the probability of occurrence) was well structured and clear to understand, and therefore it did not seem useful to modify this, and so risks added also followed this structure.

To view all changes, the link to the both previous groups’ deliverables and the updated deliverables can be seen through [this link].