Assessment 2: Brownfield Development - Method Selection and Planning

Cohort 1 Team 2

Team Members:

Trace Chinelle
Vidhi Chohan
Apollo Cowan
Siyuan Liu
Aryaman Marathe
Charlie Mason

Part A

The process we used to effectively track the changes made to the Assessment 1 deliverables, was to have an initial plan of how we wanted to change and adapt the game to follow the updated product brief for Assessment 2, combined with weekly planning, ongoing agendas and communicating with one another helped to ensure that we maintained focus on the key requirements.

Our group had a weekly agenda that was updated frequently on Google Docs, with outlined goals, internal deadlines and a weekly record of our progress. At every meeting, we would address the agenda and review the tasks that had been completed or in some cases, tasks that weren't completed due to a lack of understanding. Any major changes to the initial deliverables were discussed as a group to ensure that everyone had a good sense of what was going on.

For communication when off campus, we have a WhatsApp group chat where we post polls that can allow us to vote about a design aspect. Regular feedback was shared to allow us to identify any major issues early on.

We relied on GitHub as our primary version control platform, we could easily track the changes being made, look back at previous versions, and resolve conflicts. Branches helped allow our group to work on different aspects of the code simultaneously, as well as ensuring a previous version in case any major bug arose. Branches would be checked systematically, and the code would be reviewed before being merged to the main branch to help have a maintained codebase. Having a main codebase allowed everyone to stay updated on changes being made and the workflow of the group as a whole.

The conventions for the code remained consistent matching previous variables and the naming style. The documentation for the code was concise which allows us to have a clear idea of what is going on. To track the changes made for the deliverables we had a Google Docs that was dedicated to the task, anything that was being removed, added or updated was listed in bullet point format allowing for easy referencing.

Requirements

Req1.pdf **Req2.pdf**

In Req1.pdf ‘Introduction to Requirements’, team 9 described their approach to requirements during phase 1 of the project. They describe how they elicited their requirements from the client including a Single Statement of Need. They describe how they organised their newly elicited requirements into requirements tables which are also in Req1.pdf, explaining the meaning of each column in each table.

In Req2.pdf, I split the first section into ‘Introduction to Requirements - Greenfield Development’ - which contains all of the introduction from Req1.pdf - and ‘Introduction to Requirements - Brownfield Development’. In this second section, I describe the team’s approach to requirements during the second phase. I explain how we created some new user requirements and functional requirements, based on the updated product brief, and added them to the user requirement table and the functional requirements table.

I also explain how we assessed the existing requirements and removed some which we didn’t believe we would be able to implement easily and edited the descriptions of others to make them more achievable. We also changed the priority of some of the existing user requirements from ‘may’ to ‘shall’.

After the Introduction sections there are updated versions of each requirement table.

In User Requirements:

We added UR_LEADERBOARD because the product brief introduced a leaderboard as a new major element of the game. We added UR_ACHIEVEMENTS because the product brief introduced achievements as a new major element of the game.

We removed UR_DIFFICULTY_SETTINGS because it seemed like it would be too difficult to implement and the game would already be interesting enough with all of its other elements.

We changed the description of UR_STUDENT_SATISFACTION to be more specific about how student satisfaction could be potentially calculated. We changed the description of UR_BUILDING_VARIETY to reflect the change in the product brief that there must now be 2 recreational buildings as opposed to 1. We changed the description of UR_EVENTS to remove mention of the difficulty settings and to be more specific about the potential effects of an event. We changed the description of UR_GAME_PROGRESS to specify that the game will span 5 years over 5 minutes, as opposed to 3, because it would be simpler for the user to understand. We changed the description of UR_INTUITION to have less clunky wording but not changing its overall meaning.

We changed the priority of UR_MONEY from ‘may’ to ‘shall’ because we decided we definitely wanted to include a money system in the game. We changed the priority of UR_STUDENT_SATISFACTION from ‘may’ to ‘shall’ because while satisfaction did not need

to be implemented in phase 1, it does need to be in phase 2. We changed the priority of UR_EVENTS from ‘may’ to ‘shall’ because while satisfaction did not need to be implemented in phase 1, it does need to be in phase 2.

In Functional Requirements:

We added FR_LEADERBOARD_MENU because a leaderboard menu would be needed as part of the new brief. We added FR_NEW_HIGH_SCORE because the game needs to keep track of whether someone’s score is high enough to go on the leaderboard. We added FR_ACHIEVEMENT_SCORE because unlocking new achievements should affect the score of the player. We added FR_ACHIEVEMENT_MENU because the player would need to be able to access their achievements.

We removed FR_DIFFICULTY_SELECTION because we already removed difficulty from the user requirements. We removed FR_DIFFICULTY_EFFECTS because we already removed difficulty from the user requirements.

We changed the description of FR_TIME_LIMIT to be more specific about what could potentially happen after the time limit runs out. We changed the description of FR_BUILDING to remove reference to difficulty or upgrading buildings because neither of these were going to be implemented. We changed the description of FR_BUILDING_TYPES to reflect the change in the product brief that there must now be 2 recreational buildings as opposed to 1. We changed the name and description of FR_EVENT_TYPES to FR_EVENT_EFFECTS to be more specific of what the effects the event could potentially have on the player. We changed the description of FR_EVENT_VARIETY to be more specific about when events would occur. We changed the description of FR_TIMER to be more specific about what information the timer shows. We changed the description of FR_MONEY to remove reference to difficulty because it was not going to be implemented. We changed the description of FR_BUYING to remove reference to upgrading buildings because it was not going to be implemented.

We changed the user requirements associated with FR_TIME_LIMIT to include UR_ACHIEVEMENTS and UR_LEADERBOARD. We changed the user requirements associated with FR_BUILDING to remove UR_DIFFICULTY_SETTINGS . We changed the user requirements associated with FR_EVENT_VARIETY to remove UR_DIFFICULTY_SETTINGS. We changed the user requirements associated with FR_MONEY to remove UR_DIFFICULTY_SETTINGS.

Architecture

Arch1.pdf **Arch2.pdf**

After inheriting the game from Team 9, we looked over their approach for the architecture and implementation. The initial design process aligned with what we were originally planning, a modular-based code system as well as using a tile-based map system to allow for easy collision detection for building placement. We continued using their state, sequence and overall class diagrams and updating them in further detail. For the design aspect of the architecture writeup, since our goals aligned, no major changes were made in the thought process. The overall structure of the game remains the same, the organisation of the initial code aligned well and was a core reason for us choosing it. Following on we had a more event focussed gameplay, as every 30 seconds a game-changing event would happen, and rather than just affecting the score we thought it would be appropriate to add a financial aspect to building houses to make it feel more realistic and add a sense of uncertainty to the game.

Based on the next requirements, we decided that due to the scale of the project and the core features that still needed to be added having a settings menu and having sound being implemented wasn't essential and was removed from our plan. After reading the updated customer requirements our main focus was creating an economical aspect to the game as well as focussing on the achievements, events and leaderboard. We have updated the state diagram to reflect how we envision the new menus at a high level.

Achievements and Leaderboard lists were added to fit the pre-existing architecture design and are updated to be included in the diagrams. An additional thing we added was implementing money into the game. Originally players could place as many buildings as they would like down with little to no repercussions, now the implementation of having a player balance can allow players to strategically place buildings in a way that can maximise the highest satisfaction score as well as being able to earn some hidden achievements.

Originally the timer system counted down from 5 minutes, now we have included a game over screen. The game over screen is a core menu for the game as it displays the score, and achievements you have unlocked as well as showing the current leaderboard. The timer also has been updated to allow events to be run periodically ensuring smooth and consistent gameplay.

The overall top-down structure of the game remained somewhat the same with the removal of the settings menu and the addition of the new game-over menu and leaderboard menu. The sequence diagrams have now been updated to include all the new relevant classes which satisfy the requirements of assessment 2, it allows the person viewing the diagram to understand the flow of the game much easier now. The overall class diagrams have changed and been updated quite a lot as the evolution and the design of the game have gone in a slightly different direction than what the original team had in mind. The updated class diagram shows in detail the new changes and classes that have been incorporated as well as how they function.

Methods

Plan1.pdf **Plan2.pdf**

In Plan1.pdf the document is split into ‘Engineering Methods’, ‘Team Organisation’, and ‘Plan for Key Tasks and Deadlines’. In Plan2.pdf I included all of Plan1.pdf’s sections - listed as Greenfield Development - and also included a second version of each of those sections as related to phase 2 - listed as Brownfield Development.

‘Engineering Methods - Greenfield Development’ in Plan2.pdf shows the same information as ‘Engineering Methods’ in Plan1.pdf.

In ‘Engineering Methods - Brownfield Development’ in Plan2.pdf, we discussed which methodology we planned to use in phase 2, including the strengths and weaknesses of agile - the approach we decided to use. We also justified the continued usage of GitHub and LibGDX from the project’s phase 1 to phase 2 and we discussed which IDE our team decided to use and why. Finally we discussed and justified our use of Google Drive and Whatsapp as our collaboration and communication tools for the second phase of the project.

‘Team Organisation - Greenfield Development’ in Plan2.pdf shows the same information as ‘Team Organisation’ in Plan1.pdf.

In ‘Team Organisation - Brownfield Development’ we explained our team structure and division of work, justification of our team’s organisational approach, our coordination and communication practices, and the effectiveness of this approach overall. This section went into detail about the team approach in the second phase of the project, giving reasoning behind each decision.

‘Plan for Key Tasks and Deadlines’ in Plan2.pdf shows the same information as ‘Plan for Key Tasks and Deadlines’ in Plan1.pdf.

In ‘Plan for Key Tasks and Deadlines - Brownfield Development’ we presented our project schedule for phase 2. We presented each task of the project with a short description of its importance, as well as its priority level and the person - or people - carrying out the task. We used a Gantt chart to display this information visually, as well as providing a link to see Gantt charts of a week by week plan. Finally we discussed a number of ways in which the project evolved, from the start of the second phase until the end.

Risk

Risk1.pdf **Risk2.pdf**

After our team took over the chosen project from Team 9, we adjusted our approach to risk management to align with the previous uncompleted project for a continuation. Some risks were removed and changed to suit the new requirements, while making the corresponding strategy of new risks which could potentially contain a certain affect to the future development. In terms of the continuation of the former project, our first step was to carefully review the assessment brief for the second phase and determine the areas where adjustments were needed. Regarding each of the group members, they were reassigned the work responsibilities to ensure clarity in ownership. Next, we brainstormed additional risks specific to the second phase of the project, recording and categorizing them in a new risk table. We excluded risks that were considered highly improbable or irrelevant to the current project scope. For the remaining risks, we created clear and concise descriptions, ensuring that these could be easily referenced and understood during future assessments. By categorizing risks into distinct types, we enabled the team to quickly identify their relevance and address them effectively during specific stages of the project.

In respect of new risks, they were added to our risk table that differed with previous table, here are these risks:
Testing work of specific code is not running as intended (R13): It was added due to the reason that during the second phase, we identified that implementation and testing could be the main core of the group project; any failures in testing specific functions will delay the whole project timeline.

Tests are irrelevant to the requirements (R14): This risk was written in table to ensure that all testing efforts remained aligned with project's requirements and objectives, avoiding wasted time and resources on irrelevant tests.

Insufficient amount of users to evaluate the game (R15): As user evaluation is an essential part of the second phase, we recognized the potential risk of not recruiting enough participants, which could impact the quality of feedback we collected.

Participants may not receive sufficiently clear information about the process and objectives (R16): This was included to handle the concerns that unclear instructions could affect the accuracy and reliability of the evaluation results.

With reference to the removed risks in our latest risk table, they were no longer relevant or important for the current stage:
Use of poor-quality libraries (R2 in team 9's table, same as below): This was removed because, at this phase, the majority of libraries and tools have already been chosen by the previous team. The likelihood of this risk occurring has become insignificant.

The customer changes the requirements, causing us to have to rewrite large parts of the game, possibly under limited time (R6): The customer requirements have already been finalized and agreed upon, making this risk really unlikely to occur.

Changes of likelihood and severity to Risks:
Before stating the changes, R11, R13 and R14 are old IDs in the previous table, which correspond to R9, R11 and R12 respectively.

R11 (Lack of communication within the team causes issues):

- **Likelihood**: High → Medium
- **Severity**: Medium → High
  The likelihood decreased due to the highly frequent and effective communication among our group members. However, severity increased as the team recognized the impact of miscommunication could seriously affect the quality of the project.

R13 (Scope of project increases and results in some requirements not being met in time):

- **Likelihood**: Medium → Low
- **Severity**: High → Medium
  The likelihood decreased as the project scope was better controlled through a clear schedule and task prioritization. Additionally, the severity was reduced due to the improved management of deliverables.

R14(Parts of the writeup are poorly formatted or hard to understand):

- **Likelihood**: High → Low
- **Severity**: Low → Medium
  The likelihood decreased as each group member completed their written work, they also reviewed their paperwork and allowed members to have a quick scan of it to ensure that content was clear and easy to understand. But, regarding the increment of severity, the team considered the poorly formatted could influence the assessment score.

The responsibilities for each risk were reassigned because this project is based on the different working scale. Unlike the previous assessment, the tasks for the new phase were divided into smaller, distinct parts to reflect the new requirements at the second phase. After calculating the workload, the team determined that each member's fair share was approximately 15 points, which were distributed by combining sections of different complexity and score values for balancing workload of each group member. Team members were given the right to choose their respective parts, allowing them to align their responsibilities with their personal preferences, skills, and expertise. This approach ensured an equitable distribution of work while also enabling each member to contribute effectively to the project.