Change Report

Group 26 - Spice Traders (prior team 22)

James McNair
Alice Cui
Robert Murphy
Charlie Crosley
Dan Wade
Marc Perales Salomo

A. The first step we took towards change management is to perform a walkthrough review of the existing code and documentation in order to obtain a better understanding of the chosen project. Once the new requirements were published we compared them with the current version of the game and elicited the changes that were necessary to satisfy the requirements for assessment 2 along with any improvements we could make to more closely match the requirements of assessment 1.

   We reviewed the deliverables we were handed as a team and listed additions that needed to be made in order to perform the update without exceeding the page limit. We also brainstormed as a team on what we can improve from the old deliverables, mainly commenting on the formatting and the content. Then we condensed the old versions of the deliverables by deleting irrelevant information for assessment 2. We also changed the formatting of some of the deliverables to improve readability and to make space. Lastly we updated the content using new information about the project.

   After reviewing the code as a group we decided to reorganise the code into packages in order to facilitate our future coding tasks and achieve a more logical architecture. After this we took a similar approach towards making any modifications to the code. First we decide on what needs to be added during meetings, then we discuss what needs to be changed to allow these additions, lastly we check if there's anything we can improve to achieve a better game ensuring the whole time we keep the stakeholder involved with the process.

I. Requirements

https://booksaw.github.io/spice_traders/old/Home/Req2.pdf
The requirements deliverable we took on was clearly formatted into three distinct tables: the user requirements table, the system requirements table and the non-functional requirements table. We decided to continue with this structure in our updates as the tables are good for organising information. However the tables did not show any priority order of the listed requirements. This makes it hard for us to prioritise tasks so we decided to add a colour coding system to the tables based on each requirement's priority level.

After carefully reading the table's contents we found that some of the requirements are not very specific and the same idea is repeated elsewhere. Thus we decided to merge some of the existing requirements together to better organise them and make room for additional requirements. Furthermore we agreed to modify some of the wording of the requirements to better reflect the new sets of requirements given in assessment 2:

- UR_PLAYER_COLLECT into UR_PLAYER_PLUNDER
- FR_PLAYER_COLLECT into FR_PLAYER_PLUNDER

As most of the requirements for assessment 1 are still valid for assessment 2, we kept most of them as they were. A new set of user requirements and system requirements was added according to the new lists of requirements:

- UR_PLAYER_POWERUP added: Player should be able to pick up five unique power ups
- UR/FR_DIFFICULTY added: User should be able to select different levels of difficulty
- UR/FR_SAVE added: User should be able to save the game at any time
- UR/FR_OBSTACLES added: Player should encounter obstacles/enemy ships throughout the game
- UR_FINISH added: The game should end either on a game over or victory
- FR_PLAYER_SHOP added: Players should be able to spend their plunder
- FR_ENEMY_SHIPS added: Enemy ships should attack when in range of the player
- FR_PLAYER_POWERUP added: 5 unique power ups should spawn across the map that can be collected
- FR_PLAYER_WIN added: When a player defeats all enemy colleges they should win the game
- FR_PLAYER_DEFEAT added: When the player runs out of health or their own college dies, they should lose

Lastly we updated the introduction to the requirements deliverable to reflect the new version of the document.

II. Abstract and Concrete Architecture

https://booksaw.github.io/spice_traders/old/Home/Arch2.pdf

Section 3.a changes:

Within the abstract architecture portion of the document, we made modifications to reflect the changes of the requirements and architecture for assessment 2.

When reading through the architecture document as we received it, we found the structure of the document questionable. As bullet points were used throughout the document, it was hard to distinguish the key ideas from the minor details, making the document tedious to read. So we reformatted the document by using paragraphs where detailed explanation is needed and bullet points to highlight main ideas.

We had trouble reading through the UML diagram as the font size was very small, making it hard to decipher each component in the diagram. The updated version of the UML diagram has a larger font size and verbs have been added on the arrow between classes to describe their relationship.

We also took some measures to keep the document under the page limit with the second assessment changes:

- Shortened existing paragraphs where appropriate without removing the main ideas.
- Deleted the development view image of the project for assessment 1 along with any reference made to it, as this was no longer relevant for the abstract architecture for assessment 2.
- Changed the mentioned tool from LucidChart to PlantUML as we remade all the diagrams to reflect the changes required for assessment 2.
  Removed detail in component breakdown as excessive detail was provided causing the document to go over the page limit.
- Updated references to requirements that did not need to be included in assessment 2 as they are no longer relevant.

We decided against making any major changes to the actual abstract architecture itself as we found the abstract architecture made by the previous team appropriate and easy to build upon when making the concrete architecture. Furthermore the abstract architecture we inherited was able to accomodate for the implementation of the new requirements for

assessment 2. We simply added the new classes of Tornado, SeaMonster and Powerup to the Basic Entities category.
Within the Concrete architecture portion of the document, changes were made to reflect the additional infrastructure required to implement the new set of requirements:

- Removed the link to the external UML diagram and added an image of the concrete architecture to display the newly added sections of the UML. The website provided image of UML included all methods and variables making the image very bloated and difficult to understand, the new image within the document helps break down the important portions of the code so that the diagrams are easier to understand.
- Included the SeaMonster class to the concrete architecture to implement the SeaMonster entity from the abstract architecture.
- Added Powerup abstract class extended from Entity class to the concrete architecture to accommodate for the creations of individual power ups.
- 5 Powerup classes created and added: FreezeEnemy, FasterShooting, CoinMagnet, SpeedBoost, AbsorptionHeart.
- Reformatted some paragraphs to match the style guide provided within the assessment.
- Updated reference from LucidChart to PlantUML as different tools were used for assessment 2.
- Rephrased some sentences to make their intentions clearer without changing the meaning of them.

<u>Section 3.b changes:</u>

In section b, the relationship between the concrete architecture and the requirements is described in the form of a table. It maps the requirements ID to the entity in the first two columns and includes a description of their relationship in the third column. Our team decided to keep this format as it is more readable than big paragraphs and contains all the key information. So we updated the requirements table to be in line with the modified requirements deliverable and added additional information provided for assessment 2 (for example the points shop). The descriptions have also been updated appropriately to provide justifications for the newly added infrastructures.

We also found a few inaccuracies in this section, such as the mention of the Map class being an abstract class when it isn't within the actual implementation. So any inaccurate referencing has been removed.

III. Method Selection and Planning

The following URL links to the updated version of the Method Selection and Planning document, which contains the Gannt chart plan for assessment 2:
https://booksaw.github.io/spice_traders/old/Home/Plan2.pdf
In this document, we removed a lot of bullet points as bullet points should only be used to list or highlight key points, not for every sentence. We reformatted the layout to improve readability and improve use of space. We also reworded some sentences to make the meaning clearer.

In section 4a, we removed the paragraph which said the software engineering method was suitable because “requirements will not change during the development of the game” (third bullet point of original document). Whilst the requirements were determined at the start of the project, it was not until development of the game that it was realised that some requirements were not clear or needed further clarification, and as a result needed changing. Therefore we decided that it was not a valid justification for the plan driven development development method. A similar idea was used to justify the use of the waterfall method (last bullet point of first section on original document). So we changed this to say that whilst the requirements may change, it won’t require us to go back and change our plans. We still used a waterfall method for the approach to our project.

In the section about development and collaboration tools, we felt that there was too much detail when describing what Discord is. So we removed what we thought was unnecessary, such as the sentence “Discord allows us to contact each other with any questions or to liaise about sections of the project, while allowing input from all other members of the group”, as the core idea of what Discord is was already described in the first bullet point.

Furthermore, when discussing tools for managing the codebase, occasionally the word Git was used instead of Github. As Git is the method for managing code and Github a cloud platform which uses Git, we made sure to clarify this to prevent any confusion.

Another change we made with regards to the explanation of the use of Github, was that remote working and covid-19 was the main reason for using Github. We decided to change this as firstly, we would very likely use Github regardless of the pandemic as it is a widely used and reliable platform, and secondly, the main reason for using Github would be to maintain code history and allow easy collaboration on the code.

There was also a sentence describing how work was allocated over the Christmas break (this was in the section describing the use of a Gantt Chart and UML diagram). This was not relevant to us so we removed it. Additionally, in this section the phrase “which

components of the application would need to inherit from each other" was used twice so we removed it.

One section we thought that was missing was the section which justified which game library was used. Whilst we do not know the original team's reason for choosing LibGDX, we assumed they had similar reasoning to us, and so wrote a couple of sentences similar to what we wrote in assessment 1 to justify the use of LibGDX

In regards to section 4b, one thing we thought was missing was describing the team structure and roles. So we included a small table outlining team roles.

Additionally in section 4b, there was a section about working over the Christmas break. As this described how the previous team worked over Christmas (and also the fact it's not Christmas anymore) we decided to remove this.

With section 4c, we added our Gantt chart to outline the project plan for assessment two. We also removed the table outlining tasks and how they were assigned (found at the end of the original document), as this was only relevant to assessment 1. Although we have kept a copy available on our website.

Lastly, we updated the section describing the Gantt chart to be relevant to the Gantt chart for assessment 2.

iv. Risk Assessment

https://booksaw.github.io/spice_traders/old/Home/Risk2.pdf
We made changes to the risk assessment to improve readability and improve the level of detail of risks.

On the introduction to the risk assessment page, one issue we had was the formatting. Every sentence was bullet pointed; this is not how bullet points are supposed to be used so we introduced some paragraphs but kept some bullet points to highlight key information.

We thought the level of detail section was quite poor, it didn't really explain the level of detail used in the risk assessment. There were also some sentences that didn't really make sense, for example the first bullet point of the original document under the level of detail section which said "Overall, the level of detail used per risk was moderate, as we felt that designating more time to anticipating and preventing risks would be more beneficial in the future when we encounter them." We decided to rewrite the level of detail section as we felt it didn't answer what was asked.

In terms of changes to the risk table, the first thing we did was re-assign the owners of risks from members of the old team to the new team. We then systematically went through each risk to determine whether it was relevant, needed more detail or was OK and didn't need modifying.

One major problem we found with the risks was that the prevention of a risk occurring was discussed more than how to mitigate the effect of a risk should that risk occur. We had to update the risk table to make sure we discussed mitigations rather than avoidance of risk.

We found there were many risks that were a bit silly, or just repetitions of similar risks. For example, in the original risk table, risks R1, R2, and R3 were discussing Github, Google Drive and Discord going offline, so we decided to combine these risks as a general risk covering if an internet tool goes offline. This avoids duplication of risks.

One risk we thought was a bit unfair was risk R5, this risk covered illness of a team member. In the mitigation and avoidance column, it was described that if this risk occurred the team member would have to join meetings through discord. We thought this was unfair because if a team member was ill, they shouldn't be expected to turn up to meetings. So we changed the risk mitigation so that the ill team member should inform the group of illness and then work should be redistributed to other team members if necessary and ask the ill member to catch up on things missed using meeting notes.

One risk that we were unsure of the meaning of was R7, this described a “double booking”. We weren’t entirely sure what sort of double booking the previous team was trying to describe, so we decided to remove it from the risks table.

Another risk we changed was R10, this described the event of a power cut. As this was a quite unlikely event, and in most circumstances would only last for a few hours, we decided this was not necessary to include on our risk assessment. Furthermore, this risk would partially be covered under the unavailability of online tools risk which we added (it may be possible for a bit of local development but that can be tricky without the internet).

A risk we improved the level of detail on was R11. This risk covered the possibility of a late requirement change. The phrase “buffer time” for mitigation was not clear what it meant, hence we have changed it to “Discuss effect on schedule, use spare week to ensure work is completed”. We are assuming that “Buffer time” meant to allow for spare time in the schedule in case of delays or changes to the project so made this clearer.

Another thing we changed was the categorisation of risks. Some risks were classed as “group member” risks, but again we weren’t really sure what this meant so we changed it to “project” as these were risks that we thought were likely to affect the deliverability of the project.

Furthermore, we have removed quite a few risks from the original risk assessment, these are R14, R20, and R21. These risks seemed a bit silly, such as “wifi going down” (which would be covered under the new online tool unavailability risk) or a risk that described “using the same class names”, hence we have removed them.

Another risk we removed was R18. This risk described issues with the game library, libGDX, not offering the features the team wanted. We decided to remove this risk because the game is too far into development and the mitigation to this risk, which was changing the game library, would likely require the whole game being redeveloped. This is simply not feasible at this stage, and would defeat the purpose of assessment two.

Lastly, we added some of our own risks to the table. We felt some important risks were missing from the risk assessment, so we copied some risks from our table from assessment 1 and added these to the table. These are risks R22 - R25.

In terms of the risks encountered table, we removed risks encountered from the previous team as those team members are no longer part of the team and so are not relevant. We found that we did not encounter many risks during the completion of our project, but we have added and discussed the risks that we did encounter to the risks encountered table.