Change Report

Cohort 1 Team 9

Dominic Hall
Firas Marzouk
Ben Morrison
Harry Whittaker
Amelia Wigglesworth
Zehang Li

2. (a)

Within the assessment, our team came together in order to make changes to deliverables we inherited from the previous team. With the documents requiring changes, we split the work for the people most accustomed to the specific documents based on the role they have played in the wider scope of the assessment.

- For deliverables, allowing Harry to work on any changes within architecture was efficient, as he had produced our diagram for Assessment 1.

- Requirements were easier documents to update, as the previous team had included most Assessment 2 requirements, but left them as not implemented within their requirements table.

- Updating the method selection and planning documents was completed by Firas, as he worked on this document during Assessment 1 and so knew what topics to expand upon.

- Risk assessment changes were fairly simple to implement, as all that was required was an updating of owners of risks, and we simply had to make changes to what systems we used in comparison to the team we picked up deliverables from.

- With Harry heading software development, we delegated tasks between most of the members of our group on producing all the updated code required for the second Assessment. Majorly worked on by Harry and Amelia, the work on code was separated per task, with new items being implemented weekly. Updating from what the previous team had left behind was an efficient process as their code was well structured and commented. To manage the changes to codes we recorded the changes as a custom Github Issue where we would document the changes that were made to the code and the justification for doing so.

2. (b)

(i) Changes to Requirements - https://assessment2.yorkpirates.uk/pdfs/Req2.pdf

- Requirements were in need of some changes, with the addition of new requirements specifically for Assessment 2.

- Based on our initial round of testing, we ensured that user requirements were updated, to ensure all the user needs were met for the game. Making sure the game was able to be completed within a 5 minute timespan was a requirement to be met. With the added implementation of engaging in combat with hostile buildings, we met the user requirement for this specific encounter.

- Additionally, the requirement for combat with other ships was added, meeting all encounter requirements. This was followed by the requirements for encountering obstacles whilst sailing and bad weather encounters within the game. Within the earnables category, we also had to add the requirement of spending money earned, which was not required for Assessment 1.

- As none of the group members had access to a Mac, we were unable to test whether the requirement for having the game run on Mac was met, so we had to ensure it was put down as incomplete. We also could not test its functionality on Linux based systems . On the other hand, many items the previous game put down as not implemented were able to be implemented in our game. Implementations such as accepting keyboard input for menu navigation, allowing restarting of play from initial configuration, keeping track of player's points, giving points for time survived and other items including gaining XP and showing game stats upon defeat.

- One of the most important requirements we had to add was collision between different ships, as this ensured entities wouldn't be able to go through each other. Another was ensuring the game finished within around 5 minutes for a win or loss for the player was also important, to ensure the game didn't drag on for too long.

- By including requirements not implemented, such as interacting with friendly buildings and encountering friendly ships, we establish some clarity on what the group working on coding were unable to complete. With requirements such as a final boss upon completion of the last objective, we did not work to establish any such concepts within our final game and so their priorities will remain as 'shall' within the functional requirements table.

(ii) Changes to Architecture - https://assessment2.yorkpirates.uk/pdfs/Arch2.pdf

Overall we did not make any major changes to the architecture retaining the abstract and most of the inherited code with minor alterations within it. We did however make some sizable additions but these kept within the framework building upon lower level components and interacting with higher ones and hence keeping to the previous team's vision for the architecture.

- **Additions to Architecture**
  - One of the major additions to the architecture was implementing the new screens for the shop and to quit the game and the logic behind the screens which allows them to work with the game.

  - Other additions to the architecture include the addition of Monster, Weather and CollegeCannonball entities. Monster and Weather are both entities designed to damage the player. CollegeCannonBall is cannonball but refactored to be fired from colleges instead of ships.

  - We also added the utility to Save the game by accessing data from managers and writing it to an XML file. This also helped us to implement the restart feature.

  - Some other screens were also added for shops as well as a quit confirmation screen.

- **Changes to Architecture**
  - Primarily the changes we made to architecture were modifications to the classes we inherited. The most major of these changes was changing some variables from static. Upon inheriting the code and making changes to it we discovered that certain variables were static which led to issues like health for example being static. Changing it to dynamic allowed us to implement requirements primarily being combat.

  - Other changes include changing some fields from private to public. This was done to access the values either for testing purposes or functionality. Getters and setters may also have been used.

  - Extra functionality that could have been contained in other utility classes or managers was alternatively placed in the UI screens. This was done to save time and only done in cases where the functionality was only required with respect to that screen such as with the shop screen.

  - Some classes were given extra methods to improve their functionality and better manipulation of themselves as well as other classes. Such as adding death functions to entities such as Colleges to help implement saving and loading.

(iii) Changes to Methods & Plans -
https://assessment2.yorkpirates.uk/pdfs/Plan2.pdf

- **Tools used should be updated to reflect the use of Jira instead of Trello**
  - We changed the project management software from Trello to Jira. We made this switch because the team was used to the Jira ecosystem and hence changing to a new system would have been detrimental to the project and hindered progress as we would have had to spend more time learning a new software system.

- **Website hosting will be Cloudflare Pages instead of GitHub pages**
  - Our web development lead had already set up a website for the project, and migrating the website to this system would be easy, allow for easier modification and would have no noticeable difference to users who use it. It benefited our group as we had a process in place for updating and managing the website already, that would streamline progress with the project if we continued to use it.

- **Continuing with IntelliJ**
  - Our predecessors used IntelliJ as their integrated development environment which helps smoothen the transition as we used it in our previous project. Thus we continued to develop the project with the software, as well as using its features to help generate Javadocs & reports, and its built in unit testing system to efficiently develop and test our code.
  - They also created their architecture diagrams using PlantUMl, for which IntelliJ has good extensions and hence we used it to update their diagrams. See architecture document

- **Team roles will be updated to include our own team as well as existing and new roles**
  - One of the main changes made to the method selection and planning document was updating the roles within the team, as we had different people working within the same roles from the previous team. By updating the meeting chair to Dom, librarian to Firas and report editor to Ben, we've ensured that the members of our team leading such roles were the ones written into the document.

- **Task breakdown table should reflect Assessment 2**
  - Within part C of the method selection and planning document, we updated the table to match our planned route to completion of the second assessment. By changing all the lengths of assignments and what time period they took place in, we allow our updated documents to be accurate to our progression. With this updated version of method selection and planning, it is also possible to produce an updated gantt chart.

- **Gantt chart should reflect our own plan**
  - In part C, we also updated the Gantt chart to cover our own plan for the project throughout assessment 2. This provides a visual aid to our written plan to see when work should be completed.

(iv) Changes to Risk Assessment & Mitigation -
https://assessment2.yorkpirates.uk/pdfs/Risk2.pdf

As we are not changing the format of the acquired risk assessment, nor the risk analysis techniques, there were no changes required to the introduction or explanation of the risk assessment. We needed to update the owners of each risk to assign each risk to the members of our team. Monitoring was not listed or complete by the previous team for the majority of the existing risks, and most were listed as “not currently happening”. We chose to update this within the risk register to ensure we track and mitigate the risks constantly as we progressed with the project. Relevant mitigations will be updated to reflect the different tools being used by our team, such as the switch from Trello to Jira for task tracking. A new risk assessment has been carried out to ensure there are no missed risks with the new group and environment ahead of us carrying out the project and any risks found appended to the existing risk assessment.