Change Report

Module	SEPR
Year	2019/20
Assessment	3
Team	York Fire Marshalls
Members	Samuel Whitehead, Andrew Connor, Alex Dawson, Finn Jackson, Fred Dodd, Phuong Kha
Deliverable	Change Report

Change Management Summary

After choosing another team's product to continue work on, we have had to make some changes to their deliverables, documentation, and code. We found it necessary to make these changes before progressing the project as it will ensure we work effectively as a team and change management of the project properly without causing confusion or conflict.

Deliverables, Documentation and Code

To begin with, all of our group members read through the project's documentation individually before evaluating any changes as a group. We decided to do this as each group member may provide different ideas and this will ensure we can make necessary changes and everyone understands the project thoroughly from the start.

We examined their deliverables thoroughly so that we can adapt and make any appropriate changes for our project. We had a look at their requirements to see what has to be done to complete the project and their planning to make sure we would keep our project on track. We also considered how they organise their team, so that we can perhaps adjust our team’s roles if it is necessary to do so. Method selection is one of the most important things for us to read carefully in their documentation as it is where we can see how the previous team approached the problem. We can either continue their approach, or take inspiration from how we approached our previous project and adapt it to the project as it is now. We must also read through the method selection and planning report as it details which tools we must familiarise ourselves with to complete the project. It is important to understand the tools they used as although we will change some of them in favour of our own preference in tools, we will have to continue using their tools to prevent confusion and inefficient code. We also had a closer look at their testing since that would be an opportunity for us to see how well the software works and if there were any errors that we need to fix. We went through risks of the previous team to understand their risks and added any missing ones that we found after finishing assessment 2.

After acquiring the team's code, we cloned their website so that all the past documentation for the game was together. We spoke to group members who previously worked on this project, they gave us a further understanding of the game and gave warnings about the known bugs. This was an important part of changing management as it increased our knowledge of the project massively and meant we could progress to extending the code and documentation quickly.

To be able to implement the previous team‘s code, we have familiarised ourselves with their classes and methods by looking through their code and their comments to ensure we fully understand the code, so that we can extend the code to complete the game. We can also ask for further explanations about any unclear code to clarify any misunderstanding. We also tried to learn the tools and libraries they used in order not to alter the code too much from the previous version.

Testing

We had made many changes to the Testing Report as our team had decided that the previous report was inadequate and to update the program we had to increase its standard. We decided to completely remake the traceability matrix as we found the previous one did not serve its purpose, and removed and added many tests to fit the purpose of the tests.

Method and Approach

We decided to keep the method of testing the same as the previous group, we also had trouble creating JUNIT tests as libGDX makes it difficult to do this. We are using the same approach as the previous group of completing the tests on a weekly basis to allow us to keep track of which requirements have been completed and to make sure we have not broken other requirements in the process.

Testing Plan

We decided to make many changes to the testing plan as we feel that it did not suit the requirements documentation. Firstly, we added a column for requirements which would match each test with the requirements it was testing had been completed, this also caused us to remove many of the previous tests as we found they had no purpose without matching with any of the functional requirements. We then added tests that would be necessary after we implemented the requirements that had yet to be implemented. We aimed to have every functional requirement to have at least one test, with some having up to four tests.

Traceability Matrix

We decided to completely recreate the traceability matrix as we felt that it did not fit its purpose, as it did not give a clear view of which requirements fit with each test. We recreated it so that clearly each test and requirements link is clearly displayed using their unique identifiers, and each requirement has a number associated with how many tests are linked with it. This was done so that it is clear when each requirement has a suitable amount of testing for full coverage

Method Selection and Planning

We made a few changes to the method selection and planning report, but most of the document remains the same. We found it necessary to make the changes as our team works slightly differently to how the project was previously run, and we use different tools to perform the same tasks. It is important that we continue to use the same communication and collaboration tools as our team previously has, this is because changing these would add unnecessary confusion to the process of changing management and developing the game. We understand however that the team who has developed this project so far has done so effectively, so we will take notes and adapt our development methods appropriately.

Software Development Methods

We have not made any changes to the development methodology used in this project. The previous team used a very similar method to us and we find that it is appropriate for us to continue using their method precisely as it will ensure that everything is completed efficiently and on time. The only changes to this part of the document are to further explain a point and make it clearer.

Tools

Our team has decided to keep the majority of the tools used in the project the same as the previous team used. We will continue to use the development tools that the old team used, this is because the game was written and designed with these tools and changing them would complicate the transition and future of the project unnecessarily.

Despite this, we will discard some of the tools the previous team used in favour of our team's preferred tools. In particular we will continue using our old communication tools and collaboration tools. We have no experience using Azure DevOps and find it unnecessary for us as we have not required it previously. We will make use of Messenger instead of Discord simply because we already have that organised and they both provide very similar features. The previous team also made use of LibGDX for the game but didn't document it, so we have ensured that this is now documented fully.

Team Roles

One part of changing management of the project is that we have to assign new team roles appropriately and according to the project itself. We decided that we will keep the team roles the same as they were in our team's previous project as that has worked effectively for us so far. Changing it would create confusion and possibly reduce efficiency of the group. We considered changing roles to the roles the previous team used, but decided that due to the similarities between this project and our previous one then our old structure will be more than suitable.

Future Plan

After evaluating the documented future plan, we determined that we would not make any changes. The current plan will ensure that everything is complete in time and in a comfortable manner, meaning all code and documentation will be given the time needed to complete it to the highest quality. We have updated the plan to the current stage of the project and added extra tasks it seemed the previous team had emitted from the gantt chart. This is essential as if we miss out tasks from our documentation then we may miss them while developing the project. We mustn’t miss tasks as it may mean that our project isn’t developed fully at the time of the deadline.