

Change Report

Group Name: Team siKz

Group Number: 6

Ryan Bulman

Frederick Clarke

Jack Ellis

Yuhao Hu

Thomas Nicholson

James Pursglove

a.

Process

The team's formal approach to change management is loosely based on research of Ian Sommerville's description of change management systems^[1]. While the method described there would require a formal system of submission and review before changes can be made, it was decided that this would not be as appropriate for our situation. Such a format would require constant oversight and would divert time away from working on new deliverables, as well as slowing down the process of making the changes.

Report methodology

Instead what we did take was the use of a form to report each change as a way of auditing the potential changes. This would include:

- Change ID
 - The given identifier for each change to enable traceability and referencing
- Submitter
 - The person making the change to ensure accountability
- Description
 - A brief description of the change - what is removed/replaced/added
- Justification
 - The reason given for major change

This is repeated for each of the four required deliverables, and a similar method is used within the implementation section to describe changes made to the code. Each section will link to the amended version of each deliverable, and beneath the table there will be additional space for comments on each change - should this be necessary.

The Change ID follows a standard format of XXX00, as this allows us to remain consistent across each deliverable and maintains traceability with any potential references to the changes.

The submitter is not necessarily the only person to have influence over whether a change is made or not. However, they are the ones who assume responsibility for that change if another group member wants to challenge said change.

The URL containing the deliverables - both updated and original - is linked above the first change form.

b.

url - <https://tomnicho.github.io/yorkpirates/deliverables>

Requirements

Change ID	Submitter	Description	Justification
REQ01	Fred	Merged FR.START.START and FR.START.EXIT into one requirement	Frees up space and the distinction was unnecessary
REQ02	Fred	FR.ATTACKCURSOR changed to allow additional user input	The game did not play well with mouse aim on laptop trackpads, so an additional input option is required
REQ03	Fred	FR.GAME.SOUND removed	Sound was not fully implemented originally and is not necessary to implement now
REQ04	Fred	Added user and functional requirements based on assessment 2	Assessment 2 states that additional features must be added and these requirements represent them
REQ05	Fred	Removed CR.DEADLINE	The original deadline does not apply to our group and is self-explanatory regardless
REQ06	Fred	Removed UR.FINISH.GAME	Unnecessary end condition that was not implemented and does not fit our groups interpretation of the requirements
REQ07	Fred	Removed clarification in FR.CLG_ATTACK	The clarification was regarding a late addition which was unnecessary from our perspective and took up space in the table.
REQ08	Fred	Updated the definitions for the priorities	These better represent the priority workflow and their importance
REQ09	Fred	Added discussion on the difficulty	This was lacking from the original document and a description of how we used the column was fitting.

REQ04 -

The new user requirements have the following IDs:

UR.AVOID.OBS,
UR.SPEND_LOOT,
UR.FIGHT_SHIPS,
UR.POWER_UP,
UR.CHOOOSE_DIFF,
UR.SAVE_LOAD

The new functional requirements have the following IDs:

FR.GEN_OBS,
FR.OBS_IMPACT,
FR.SHOP,

FR.UPGRADES,
FR.SHIP_AI,
FR.TMP_UPGRD,
FR.CHANGE_DIFF,
FR.SAVE,
FR.LOAD

Additional notes

A lot of the original document has been kept as it was found. Most of the requirements for assessment 1 are still requirements for assessment 2, so a lot of the original requirements are still relevant to our updated version. We did not generate a massive list of new requirements either, so the need to make space was also limited.

The framework used was also relatively similar to our original one, so we felt comfortable adapting to an existing model and conforming new requirements to that, rather than vice versa.

Architecture

Change ID	Submitter	Description	Justification
ARC01	James	Replaced concrete architecture diagram with a new diagram.	We left their existing abstract architecture diagram as is since we only began planning after the concrete architecture was complete. The concrete architecture diagram we have replaced with an updated version that included classes and functions we created during out portion of the assessment.
ARC02	James	Replaced inheritance diagram.	We updated the inheritance diagram to include new classes from new features. We also updated it to include the new XML file we created.
ARC03	James	Updated bibliography to link to new versions of diagrams.	We had new diagrams that needed linking,
ARC04	James	Added justification for XML file.	We had been given a new requirement of being able to load and save the game. The previous team had hard coded all the game objects and loaded the map from a .tmx. This was fine when the game only needed to be loaded once but did not give us many options for saving and loading so to accommodate we added an xml file to save and load the object data to and from.
ARC05	James	Removed some detail from concrete architecture justification.	There was not enough space to justify classes we needed to add to the architecture so we shorted some existing content to make room.
ARC06	James	Added justification for creating obstacle class and inheriting classes.	Due to the new requirement to implement obstacles and powerups we needed to create new classes and abstract classes and justify these in the concrete justification section.
ARC07	James	Added justification for creating ShopUI and Shop classes.	Due to the requirement to implement loot spending we needed to create new classes for implementing this.
ARC08	James	Used plantUML gizmo for new uml diagrams.	To coordinate editing the UML diagrams we used a google docs plugin

Methodology and Planning

Change ID	Submitter	Description	Justification
MET01	Jack	Changed from waterfall method to agile	We changed the methodology from waterfall to agile to make it easier for the team to complete the project. This is because the agile technique splits the project up into stages and allows members to review and then update them accordingly.
MET02	Jack	Changed from IntelliJ to VSCode	This is because all of the team members felt that they were more confident working in VSCode as they wouldn't have to learn new software such as IntelliJ.
MET03	Jack	Didn't use Trello	We found that Google Docs was a lot easier and more efficient to use than Trello. We would all have to sign up to Trello and learn how to use it, so we just used Google Docs to decide what each person was doing and in person meetings to make sure they were on track.
MET04	Jack	Didn't have a team leader	Instead of having a team leader, we split each section evenly and each member made sure everyone else had everything they needed and were on track to complete their tasks. We did have some stronger programmers than others so those were tasked to help out as much as possible, where necessary.

Risk assessment and mitigation

Change ID	Submitter	Description	Justification
RAM01	Ryan	Removed risk 000, 'Poor selection of platforms and engines'.	Selection was made at the beginning of the project and we have been using them without issues since then so it is no longer a risk.
RAM02	Ryan	Removed risk 002 'Inconsistent coding practices throughout'.	We can now follow pre-existing code to use the same coding practices throughout so it is no longer a risk.
RAM03	Ryan	Removed risk 016, 'Member is not suited to a role or task'.	Team members are working on similar areas of the project to before the first assessment and have had no issues with not being suited for a role/task so it is no longer a risk.
RAM04	Ryan	Risks added, updated, merged and colour coded.	Justifications for specific risks can be found in the notes column of the table. Colour coding was added to make identifying probability and consequence categories easier.
RAM05	Ryan	Some risks were kept and some of those which were kept were modified or expanded upon.	Justifications for risks and changes to risks can be found in the notes column of the table.

[1] I. Sommerville, *Software Engineering*, 10th ed, Harlow, Pearson Education, 2016