

Change Management Log

ID	Owner	Document/Code Changed	Date	Description	Impact
1	Ed Gould	SailingScreen Class - variables, constants and methods	9/2/19	Before updating the SailingScreen class from the previous project, the variables and constants associated with the map were fixed (coded as raw numbers) meaning they had to be changed manually. I've updated the code to use features in the LibGDX library to read this data directly from the map, allowing the map to be changed easier and keep consistency with the map design against the code.	The code is now more dynamic than before as the values are not hard coded. This allows for easier changes to be made to the map and any values associated with the said map. This does, however, add new risk
2	Ed Gould	Added a test map	9/2/19	Created a map to test the methods added to the SailingScreen class as well as testing the modifications to constants and variables.	This allows me to make changes and modifications to the current map knowing the new methods has been implemented reliably.
3	Ed Gould	Implemented the first version of the new map, making various tweaks	10/2/19	The previous map was not fit for the addition of two new colleges and a new department. Because of this, I've decided to make an entirely new map fit for five colleges and three departments. I have also left spaces in the map to allow for the expansion of other events/features in the future more easily. The data for the map file was also cleaned up and simplified, making changing the map a lot easier.	This gives us a map that was not only more fit to final requirements, but it also gives us a good environment for testing out game features. The data for the map is now simplified to future changes, if needed, will be much easier.

4	Ed Gould	Visual improvements	11/2/19	Visually improved map file to make colleges/departments distinct from each other, and to look better	
5	Luke Richardson	Modified the Requirements document changing 'reference column' to 'ID' column for clarity and efficient use of space	12/2/19	The previous requirements document had a 'reference' column which took up unnecessary column width as the references were at maximum 6 characters long and could take up multiple lines, for example, NF2(R). Because of this I have changed the column label to 'ID' so it creates more space in the table for requirement description and fit criterion.	This allows us to fit more requirements on each page as well as making the layout of the table less vertical (more horizontal width for requirements description and fit criterion is easier to read).
6	Luke Richardson	Added Requirements	12/2/19	Taken the requirements from the previous group and added new features which we hope to implement this assessment.	This makes the implementation clearer to the development team and allows them to plan implementation around this.
7	Ed Gould	Expanded Department and DepartmentScreen classes to allow expanding new combat options easy in the future	14/2/19	Expanded Department and DepartmentScreen classes to allow expanding new combat options easy in the future	

8	Luke Richardson	Updated Method Selection and Planning document	14/2/19	Updated Method Selection and Planning document	
9	Ed Gould	Expanded College and CollegeScreen classes to easily add weapon changing functionality in the future	14/2/19	Expanded College and CollegeScreen classes to easily add weapon changing functionality in the future	
10	Ed Gould	Weapon and combat reworking	16/2/19	<ul style="list-style-type: none">• Reworked damage calculation for attacks• Gave weapons less ambiguous descriptions (specific damage amount and accuracy percentage)• Removed redundant variables and methods from Attack class• Added two new weapons, Explosive Shell and Double Shot (in respective classes ExplosiveShell and DoubleShot), inheriting from Attack class• Reworked accuracy mechanic to be more clear to the user how likely attacks are to hit• Changed CollegeScreen, adding the ability to change equipped weapons through the college interface• Changed Department and DepartmentScreen classes to allow the player to buy weapons	

				<ul style="list-style-type: none">○ Added a cost to each weapon	
11	Umar Farooqi	DepartmentTest.java written	16/2/19	<ul style="list-style-type: none">● buyWeaponTest<ul style="list-style-type: none">○● buyUpgradeTest● getUpgradeCost	
12	Umar Farooqi	Player.java written	16/2/19	<ul style="list-style-type: none">● payGoldTest<ul style="list-style-type: none">○ payGoldFailedTest○ payGoldSuccessfulTest	
13	Umar Farooqi	ShipTest.java	16/2/19	<ul style="list-style-type: none">● damageTest<ul style="list-style-type: none">○ damageBroadsideTest○ damageDoubleShotTest○ damageExplosiveShellTest○ damageGrapeShotTest○ damageRamTest○ damageBoardTest○ damageSwivelTest○ healSailTest○ healHullTest	
14	Ed Gould	Balance changes	17/2/19	Balanced combat, prices of weapons and upgrades, and how the player gains points	

15	Shubei Quan	Implemented minigame	17/2/19	Minigame Implemented	
16	Luke Richardson	Black Box testing and evidence	17/2/19	Ran 29 black box tests on the final iteration of the game for this assessment	
17	Umar Farooqi	Unit Testing testing and evidence	17/2/19	Ran 29 unit tests and evidenced them in a table in the Tess3 document available on the website	