Comp2050 Software Maintenance

Coursework2 Issue Sheet

Session	18-19	Semester	Autumn
Module Name	Software Maintenance	**Code**	COMP2050
Module Convenor(s) (CW Convenor in Bold)	**Nasser Mustafa**. JianJun Chen		

Coursework Name	Maintaining and evolving existing software	**Weight**	50%
Deliverable	1- The source code of the SpaceInvaders game including all the changes and additions that you make to the source code 2- The source code for the Sprite Editor. 3- A Read-me file that describes how to run your code in (1 & 2) 4- A video showing the sprite editor functionality (<2 minutes, no audio is necessary).		
Format	The submitted source code and other required files should have the following formats: 1- A zipped folder submitted to Moodle contains the SpaceInvaders source code, Sprite Editor source code, and the Read-me file. 2- The source code for the SpaceInvaders, the Sprite Editor source code, the Read-me file, and the video for the Sprite Editor functionality should also be stored on Git. 3- The Sprite Editor solution shall also be a standalone **.jar** file stored on a separate folder **(\bin folder from the project root)** 4- The template for the Moodle zipped folder name is (*Your ID-Your name*)		

Issue Date	Issued w/b 3rd December, 2018
Submission Date	Deadline Wed 2nd January, 2019 at 16:00
Submission Mechanism	On both Moodle and Git (same code submitted on both places)
Late Policy (University of Nottingham default will apply, if blank)	Standard policy
Feedback Date	Tuesday 15th January, 2019
Feedback Mechanism	Individual comments via grading system on Moodle.

Instructions	
	1.Setup There are different components to this which together will make up 50% of the module mark. The breakdown is as follows: **20% Use of Git**. Including, successful forking/cloning of the original code. Regular commits to the server, and meaningful commit messages. **20% Adding the new level code** **20% Adding the new weapon code** **40% The implementation and design of the standalone Sprite Editor application** (and associated GUI) written using JavaFX. In all cases good programming practice will gain higher marks. Nicely presented and easy to use interfaces will be rewarded. A proportion of the marks will depend on you supplying a working JAR file to run your Sprite Editor, and submitting a video of it in use. (See Format box above). 2. **Requirements**: **See the separate Software Maintenance spec sheet for details.** Use Git to version your changes regularly, using suitable commit messages, etc. Note that you are not required to fix any other bugs you may find in the game. Your only job is to make the required changes.