

REQUIREMENTS OF THE ASSIGNMENT OF SPRINT 2

SPRINT 2

by

A. Smesseim, S. Baraç, A. el Khalki, A.W.L. Oostmeyer, M. Maton

in partial fulfillment of the requirements for the degree of

Bachelor of Science
in Computer Science

at the Delft University of Technology,

to be presented on 25 September 2015.

Supervisor: Dr. A. Bacchelli

CONTENTS

1	Testing	1
1.1	Requirements	1
1.1.1	Must Haves	1
1.1.2	Should Haves	1
1.1.3	Could haves	1
1.1.4	Would/won't Haves	1
2	Barricades and sound effects	3
2.1	Functional Requirements	3
2.1.1	Must Haves	3
2.1.2	Should Haves	3
2.1.3	Could Haves	3
2.2	Non-functional requirements	3
	Bibliography	5

1

TESTING

1.1. REQUIREMENTS

For testing the game Space Invaders, four categories can be identified from the requirements using the MoSCoW[1] model for prioritizing requirements.

1.1.1. MUST HAVES

1. The code must have at least 60% line coverage in the non-GUI components.
2. The tests must be meaningful, in the sense that the test methods do not only invoke the methods for increasing test coverage, but also verify the behaviour of the tested classes.
3. The tests must not take longer than 10 minutes on a new laptop provided by TU Delft.

1.1.2. SHOULD HAVES

4. The code should have at least 70% line coverage in the non-GUI components.

1.1.3. COULD HAVES

5. The code could have at least 80% line coverage in the non-GUI components.

1.1.4. WOULD/WON'T HAVES

6. The code won't have 100% line coverage.

2

BARRICADES AND SOUND EFFECTS

2.1. FUNCTIONAL REQUIREMENTS

For barricades and sound effects of the game Space Invaders, the requirements are divided into four categories using the MoSCoW[1] model for prioritizing requirements.

2.1.1. MUST HAVES

1. Four barricades must be placed between the enemies and the player, equally spaces horizontally over the playing field.
2. The barricades must not let bullets pass through them.
3. The barricades must be colored green.
4. 'Ave Maria' must be played as background music.
5. The size of each barricade must be 200x100 pixels.

2.1.2. SHOULD HAVES

6. If a bullet hits the barricade, then only the point of intersection of the barricade should be damaged (i.e. destroyed).

2.1.3. COULD HAVES

7. A sound effect could be played when a player fires a bullet or a bomb.

2.2. NON-FUNCTIONAL REQUIREMENTS

In addition to the functional requirements specified in chapter 2, the game should also adhere to several non-functional requirements, outlined in this chapter. These requirements will not affect the functionality of the game, but will ensure that the game is gradable by the instructors of the course TI2206. The non-functional requirements of the game are:

8. The game must be playable on Windows (7 or higher), Mac OS X (10.8 and higher) and Linux.
9. The game must be implemented in Java.
10. Scrum methodology must be applied
11. The source code must be hosted on GitHub as a public repository.
12. The version control used for developing this game must be Git.
13. The tests must be automated using JUnit.
14. This project must use Maven for build automation.

15. This project must use Travis CI for continuous integration.
16. This project must employ the following static analysis tools: Checkstyle, PMD and FindBugs.

BIBLIOGRAPHY

- [1] S. Ash, *Moscow prioritisation briefing paper*, DSDM Consortium (2007).