

Requirements Fishy Logger

by

Group #6

*Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science.
Mekelweg 4,
Delft*

Name	StudentId	Email
Michiel Doesburg	4343875	M.S.Doesburg@student.tudelft.nl
Matthijs Halvemaan	4353803	M.J.W.Halvemaan@student.tudelft.nl
Dmitry Malarev	4345274	D.R.Malarev@student.tudelft.nl
Sunwei Wang	4345697	S.Wang-11@student.tudelft.nl
Clinton Cao	4349024	C.S.Cao@student.tudelft.nl

TA:

Valentine Mairet

V.A.P.Mairet@student.tudelft.nl

Course: *Software Engineering Methods (TI2206)*

Table of Contents

1. Functional Requirements	3
1.1. Must Haves	3
1.2. Should Haves	3
1.3. Could Haves	3
1.4. Won't Haves	3
2. Non-Functional Requirements	4

1. FUNCTIONAL REQUIREMENTS

The functional requirements of the logger of the game “Fishy” are described below. These requirements are divided according to the MoSCoW methods.

1.1 Must Haves

- The logger must record when the game is launched and exited.
- The logger must reset when the game is exited.
- The logger must log the score gained when the player fish eats a smaller enemy fish.
- The logger must log the following actions:
 - Buttons pressed by the player in each screen. These are the buttons that the player uses to control the fish (WASD keys) and the buttons of the graphical user interface (Play button, Quit etc.).
 - Collisions between the player fish and the enemy fish.
- The logger must log the following information of the game:
 - Information regarding if the player has won/lost the game.
 - Information regarding switching of the screens (graphical user interface).

1.2 Should Haves

- The logger should print out the collisions of fish with the screen edge.
- The logger should print in which direction the player is moving.

1.3 Could Haves

- The logger could save the log in a text file.

1.4 Would/Won't Haves

- The logger should not print the exact coordinates of the player fish. This is because the console will be filled with a lot of information (of just movement) and finding the log for buttons or other actions might be hard.

2. NON-FUNCTIONAL REQUIREMENTS

The functionality, services and design constraints that need to be included in the requirements specification are provided below. These requirements do not indicate what the system should do, but instead indicate the constraints that apply to the system or the development process of the system.

- The logging has to be implemented from scratch without using any existing logging library.
- The logging has to be accessible via console while the game is running.
- All the logging must be printed out in a natural language. With this it is meant that someone who is not a developer can understand the logs.