

Requirements specification document

Bubble shooter

By

Multi Culti Soup

Kabilan Gnanavarotheyan
Jurrien Theunisz
Timotei Jugariu
Henk Janse
Yifei Liu

Table of contents

Functional Requirements	3
Must haves	3
Should have	3
Could have	3
Would/Won't haves	4
Non-functional Requirements	5

1. Functional Requirements

In this section the requirements regarding functionality and service are listed. The requirements are described using the MoSCoW model, grouped in four categories: functionalities that the system must have, should have, could have and will not have.

Must have

- The game starts with an empty board with a start button
- If the start button is pressed, the game shall start with a hexagon consisting of 91 balls
- Around the hexagon there will be 4 walls at a distance equivalent to the width of 19 balls put next to each other
- The balls have one of the 6 different colors: blue, green, orange, purple, red, yellow
- The player will have a ball at the top of the screen which he can shoot in a certain direction
- The player will shoot by pressing the left mouse button, the ball going towards the position the cursor had at that time
- The ball the player shot can bounce off the walls
- The ball the player shoots will be a one of the 6 colors, randomly chosen
- If at least two adjacent balls with the same color as the ball the player shot, the balls will disappear (including the shot ball) and the player receives a point for every disappeared ball
- In the hexagon are also special balls that have a slightly different behavior: they double the score the player gets in that round if they are removed
- The special balls are removed
- If the player doesn't score, the ball will attach to where it hit
- The ball in the middle has none of the 6 colors and can not disappear
- The whole hexagon of balls will rotate around the middle ball every time the shot ball hits the hexagon
- The player shall be able to change the dimension of the board and the number of balls the initial hexagon consists of.

Should have

- The hexagon will rotate depending on the distance traveled by the ball fired by the player
- If the player doesn't score a point after 5 shots a random number of additional balls will be added to the hexagon
- The number of additional balls that will be added shall variate between 5 and 15.
- A ball that has hit the wall more than 4 times shall disappear, the shot counting as a miss
- If any ball of the hexagon touches the wall the game is lost and a window will pop up with the score that the player has achieved. The window will also include a restart button and a home button.

Could have

- The player shall be able to pause the game by pressing the Escape button. A window will pop up containing a restart button and a home button.
- The game shall play music in the background

- A sound will be played when the player scores a point
- The player shall be able to turn the background music off
- The game shall reward the player bonus points when a certain amount of points have been made in one shot
- The game will show a pointer arrow at the direction the player wants to shoot the ball

Would/Won't have

- The player is able to change the board background
- The game shall have the option to share the score on social networks
- The player is able to change the ball speed
- The player is able to disable wall bounces

2. Non-functional Requirements

In this section design constraints are described. These constraints have no effect regarding what the system does, but on the development process of the system.

- The game shall be playable on Windows (7 or higher)
- The game shall be implemented in Java
- For the iterations after the delivery of the first fully working version, the Scrum methodology shall be applied
- For the implementation of the game the following software and frameworks will be used: git, Maven, PMD, GitHub, FindBugs, Travis CI, JUnit
- The implementation of the game shall have at least 75% of the lines covered by automated tests