Requirements Bejeweled

by

Irene van der Blij, Mayke Kloppenburg, Samuel Sital,

Kiran Kaur, Takang Kajikaw Etta Tabe

**TI2206 Software Engineering Methods**

Content

[1. Functional requirements 3](#_Toc428877101)

[1.1 Must haves 3](#_Toc428877102)

[1.2 Should haves 3](#_Toc428877103)

[1.3 Could haves 3](#_Toc428877104)

[1.4 Won’t haves 3](#_Toc428877105)

[2. Non functional requirements 4](#_Toc428877106)

# 1. Functional requirements

For the game Bejeweled, the requirements regarding functionality and service are grouped under the functional requirements. Here the MoSCoW model is used to further work out these requirements, considering the importance of the requirements.

## Must haves

* The entire grid is filled with jewels, each cell has to contain a jewel.
* The board has to consist of a grid of 8 by 8 cells.
* The player must be able to switch two jewels only if it results a combination of 3 or more matching jewels in a row or column.
* If the player gets a combination of 3 or more matching jewels in a row or column, then those particular jewels vanish from the board.
* If a combination of 4 or more jewels is made, a special jewel appears in one of the cells where the combination was made.
* When there is an empty cell, the jewel above this cell will fall down to fill the empty cell.
* When there is an empty cell on the topmost row, new random jewels will appear here.
* If the player gets a sequence of three or more matching jewels, then the player gets a certain amount of points.
* The game should end when there is no possible move left.
* When a combination with a special jewel is made, all jewels directly around this jewel (also diagonal) will disappear from the grid.
* Level? A level should end when the amount of points to finish this level is reached.
* How many points for each combination?

## Should haves

* The game should show the player’s score.
* The game should have multiple levels, and the player starts at level number 1 when the game starts.
* The game should reset the player’s score and the level number when a game ends.

## Could haves

* The game could have the possibility to give a hint to the player when he asks for it.
* The game could show the player his final score and level number when the game has ended.

## 1.4 Won’t haves

# 2. Non functional requirements