

'Balls of Steel'

Group 4

John Laurin

Joakim Åkerlund

Milan Ivanovic

Daniel Öberg

Christoffer Lundell Johansson

Functional Requirements:

1. The system will provide a menu-screen on start-up, allowing the user to select to start a new game, to continue one that is in progress, to view the high-score, to make general settings, to view instructions and to exit the program.
2. The system will allow users to turn of sound and/or music via the settings-option on the menu-screen.
3. The user will be able to gain an understanding about how the game is played via the instructions-option on the menu-screen.
4. The user will be able to see the 10 highest times, as well as the signature of the user who achieved it via the high-score option on the menu-screen.
5. Upon selecting the new-game option, the game will display the main game-area, with the game in paused mode. The game will display a number counting down by seconds from 5. Reaching zero, the game will commence, and the user will be able to exercise control over the ball.
6. The user will be able to return to the main menu at any time during play by pressing the 'return' key, as defined by the manufacturer of the phone.
7. The user will be able to control the motion of the ball by using the 'left' and 'right' keys, as defined by the manufacturer of the phone.
8. The game will provide a number of obstacles to be overcome, including walls, mines, magnets and wormholes.
9. Mines shall, upon coming in contact with the ball, return the ball to the levels starting point and deduct a preset amount of time from the global countdown.
10. Mines can either be static or dynamic. Dynamic mines will move in a predetermined pattern, static mines will remain at their original position at all times.
11. Magnets shall exercise a force upon the ball if it is within a given distance and alter the trajectory of the ball.
12. Wormholes shall, upon contact with the ball, transfer the ball to a corresponding wormhole in another location on the level.
13. Walls are the standard impediment. Upon contact with the ball, the ball bounces off at a trajectory determined by the angle of the wall and the balls original vector.
14. The user will be able to collect keys throughout the levels. Upon collecting all the keys, the exit point for that level will be unlocked.
15. Upon reaching the exit point for a level, the next level will begin. Players will receive additional time for the global countdown, determined by which level they just completed.
16. Upon reaching a new level, the game will be paused and a countdown will commence. The global countdown will be suspended at this time. When the countdown reaches zero, the level will begin and the global countdown will begin again.
17. The player will be able to collect power-ups throughout the level. These will include time-increases, low gravity, reverse gravity and freeze.
18. Time Increase power-ups will give the player additional time to the global countdown.
19. Low Gravity power-ups will lower the gravitational acceleration of the ball, resulting in changes to how the ball behaves when bouncing. This effect ceases after a given time.
20. Reverse Gravity power-ups will change the direction of the gravitational force, in effect turning the ceiling into the floor and vice versa. This effect ceases after a given time.
21. Freeze power-ups will stop the motion of all dynamic mines throughout the level. This effect ceases after a given time.
22. The game will provide a number of levels. If the player completes all of these, the player will return to the first level and will be subject to increased difficulty. This will take the shape of increased penalties for hitting mines, less time given upon level completion and less power-ups given across the level. This effect is cumulative until the game ends.
23. When the global countdown reaches zero, the game will display a Game Over screen, on which will be displayed the total time played. It will also display whether or not the time was sufficiently high to be displayed on the High score.
24. If the player time was sufficient to be displayed on the High Score, a High Score screen will be displayed, indicating the place that was earned, and allowing the player to input a signature of no more than 6 characters.
25. If the players time was not sufficient to merit a mention on the High Score, the High Score will be displayed.
26. At the High Score screen, the player will be able to return to the Menu Screen. The player will not be able to continue on the game that was completed.

27. Upon selecting the Exit option, the game will shut down.
28. If the phone receives a call during play, the game will pause, and the phone will use its standard procedures for handling incoming calls. Upon terminating the call, the game will reassert itself.
29. If the phone receives an sms, mms or some other form of message, the game will be paused.
30. If the phone issues a warning or another kind of system message, the game will be paused.

Non-Functional Requirements:

1. The memory usage of the software should not exceed the limit of 1MB during run-time at all times.
2. The response time between choosing a menu option and the screen for that option appearing shall not exceed 10sec.
3. The memory size of the save file shall not exceed 50KB.
4. The memory requirements for the level-files must not exceed 10 KB.
5. The framerate during gameplay shall not drop below 2 frames per second (fps) during gameplay at all times.
6. The response time between user input and in-game change should not exceed 0.5 sec.

Use Cases:

UC1: Gameplay

Primary Actor: Player

Stakeholders and Interests:

- Player: Wants entertainment and to be able to get a high score.

Minimal guarantee:

The game will perform as specified.

Success guarantee:

The player accumulates a time that merits a place on the High Score. The Game records the new entry on the high score.

Main Success Scenario:

1. The player starts the Game.
2. The chooses the 'New Game' option.
3. The level loads and the game commences.
4. Loop until global time is 0:
 1. The player begins to move the ball around, attempting to overcome the stated obstacles.
 2. The player manages to collect the keys for the level.
 3. The player reaches the exit point.
 4. The next level is loaded and additional time is awarded. However, there is a net loss of time
5. End Loop.
6. The Game is ended and the total time is displayed.
7. The Game informs the player that his time is sufficient to get a place at the High Score.
8. The player presses the 'OK' button and is navigated to the High Score screen.
9. The Game displays his allotted place in a different color and with a default name.
10. The player edits the name to his own and presses the 'OK' option.
11. The Game records the name and the time to the High Score and returns the player to the Main Menu.
12. The Player selects 'Exit'
13. The Game is shut down.

Extensions:

- *a. The game crashes during play or load times.

1. The player restarts the game. All previous game data is lost and the game begins anew.
- *b. The phone receives a call during play.
 1. The game minimizes and pauses until the player decides to continue.
- 2a. There is a save game available.
 1. The user chooses the "Continue Game" option in order to continue a previously played game.
- 7a. The game informs the player that his time is insufficient to get a place at the High Score.
 1. The player starts a new game and tries again.
- 7b. The player doesn't press the "OK" button.
 1. The score is lost and the player is redirected to the main menu where he can try playing the game again.
- 10.a: The player does not edit the default name.
 1. The Player selects the 'OK' option without editing the default name.
 2. The Game records the default-name as the name for the High Score.

Special Requirements:

- 1a. Mobile phone with a color screen and Java ME support.

Technology and Data Variations List:

*c. All movement and selections are handled by user input with the default movement keys as defined by the mobile phone manufacturer.

Frequency of Occurrence:

Could be nearly continuous, however it will probably occur in short periods of time.

UC2: Changing the Settings

Primary Actor: Player

Stakeholders and Interests:

- Player: Wants to turn off all sounds.

Minimal guarantee:

The user can check what sound options are on/off.

Success guarantee:

All sound in the game is off.

Main Success Scenario:

1. The player loads up the game.
2. The player chooses the settings option, and the game loads the settings menu
3. The player is presented with the sound options which he turns off.
4. The player returns to the main menu and can now start a game without sound.

Extensions:

*a. The game crashes during play or load times.

1. The player restarts the game. All previous game data is lost and the game begins anew.

*b. The phone receives a call during play.

1. The game minimizes and pauses until the player decides to continue.

1.a. The player is playing the game before deciding to turn off the sound.

1. The player presses the 'return'-button.
2. The game pauses and the player is returned to the main menu.
3. The player chooses 'settings'
4. The player is presented with the sound options which he turns off.
5. The player returns to the main menu and selects 'continue'
6. The play area is displayed and the game is still paused.
7. The player presses a key and the game begins again.

Special Requirements:

1a. Mobile phone with a color screen and Java ME support.

Technology and Data Variations List:

*c. All movement and selections are handled by user input with the default movement keys as defined by the mobile phone manufacturer.