

Team 15 High Order Testing for MOBA “Overleven”

Project Name: Overleven

Team : Brendan Raftery, Benjamin Loisch, Mangkorn Yuan, Conner Isaacs, Nada Alnoory

Severity Breakdown

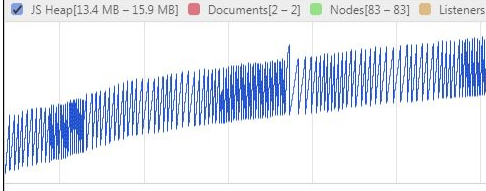
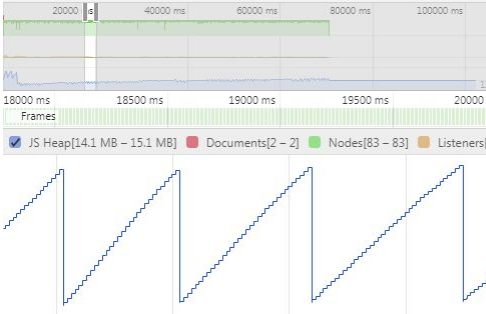
- 1 : **Critical** (Test cases critical to the success of software)
- 2 : **Important** (Test cases encountered on day to day functional tasks)
- 3 : **Workaround** (Test cases for which the software could run even with the defect)

Type Breakdown

- 1 : **Usability** (Poor UX or UI design)
- 2 : **Security** (Improper access to game mechanics or information that should not be allowed)
- 3 : **Stress** (Issues with higher amounts of volume within a shorter period of time)
- 4 : **Volume** (Issues with volume regardless of amount of time taken)

For all requirements, the game can be found at <https://overlevengame.firebaseio.com/>.

Defect No.	Defect Description	Defect Severity	Defect Type	Test Case No. or Description
1	Tier 1 representation layer bug (Usability), as leaderboard fills with scores since no scrolling box is present the text may appear off screen.	Workaround	Usability	5
2	Tier 1 representation layer bug (app focus) (Usability), when selecting another application window while the game is running, if the monsters are touching the player the HP goes down but no sound is emitted.	Workaround	Usability	11
3	Potential memory leak error (performance issues), using Google Chrome dev tools performance analyzer, we can analyze the JS	Workaround	Volume	The JS Heap increases by barely a MB/minute. This won't pose any

	<p>Heap over time. Gameplay performance tested over two minutes reveals a steady increase in the JS Heap:</p>  <p>And those jagged peaks up close over 2 seconds:</p> 			<p>serious problem because we have a lot of MB mem to spare, but something we just wanted to point out. It is difficult to pin down what may be doing this. Looking closely at the jigsaw pattern, it isn't consistent enough to be some sort of game-timer variable related mem-leak, so we can only guess it is some sort of deep rendering leak bug, potentially creating memory leaks over time by drawing the images to the buffer. In the second image we count from 18k ms to 19k ms about 60 steps and the FPS count that google dev tools says for the app is about 60 FPS so this confirms it could be an issue with maybe their rendering library phaser.io</p>
4	<p>Cool downs do not tell you how much time is left</p>	Workaround	Usability	<p>In games like this one it is usually very important to know when an ability will be off cooldown for the player to use. Right now there is no way for the player to know</p>
5	<p>Lack of information on player abilities</p>	Workaround	Usability	<p>No information exists within the</p>

				application that explains how the abilities are supposed to work to the player. Some icons for abilities aid in explaining but are not clear
6	Art on slime sprites have improper settings	Workaround	Usability	The way the sprites for slimes are rendered causes the top of the sprite to continue over to the bottom, resulting in black lines below all of the slimes on some frames