Outcomes to Date

The tutorial level is the introductory area where the player gets introduced to the mechanics of the game. To meet this goal, we gave on-screen instructions that show the player the controls. We only implemented this for jumping however and not moving, health and death, traps or enemies. This was due to communication issues within the team slowing progress down.

<insert picture of whole tutorial level>

<insert picture of on-screen instructions>

The playable character is what the player controls during the game. He can run and jump in order to get around obstacles in their way. The player does not animate during these actions, but animation was more a stretch goal.

<insert picture of player movement>

The traps in the game are one of the obstacles in the game that obstructs the player’s path. The trap that was implemented was the spike trap/pit. This trap is stationary and when the player touches them, the player dies. Other more advanced traps were to be implemented during later levels in the game after the player understands the initial trap concept.

<insert picture of spike pit>

The enemies in the game are the other obstacle in the game that obstructs the player’s path. The enemies are more dynamic than traps in that they interact with the world more. The enemy we implemented was an orc, a large and muscly humanoid. Due to implementation issues and time crunch, the orc only stands in the way and doesn’t interact otherwise.

<insert picture of orc>

The health and death mechanic are directly tied to the playable character. The character has a set amount of health and lose health when running into traps or enemies. Once the player loses all their health, they die and shows the death screen. The death screen allows the player to either respawn or exit the game.

<insert picture of health change>

<insert picture of dying>

Testing

What testing has been performed? Did this result in any changes? Have any of your plans for testing changed during the project? What would you do differently if you had your time again?

The project has only been internally tested within our team. We were mainly looking for glaring bugs within the game rather than testing whether the project was fun to play. While making changes, we would run through and see if the changes were what we anticipated and wanted. This allowed us to catch numerous bugs within the code. We used the validation criteria for each MVF where applicable, but we made some changes to the features and made some invalid. If we started this project over again, we would try and finish the prototype earlier, allowing us to get some outside testers to try the prototype. This would have given us an outside perspective and change aspects based on feedback.