

An Text-Based Role-Playing Game (RPG), based off of Avatar: The Last Airbender

Jad Mukbel - <https://github.com/jmukboo1>

Pranesh Naagamuthu - <https://github.com/Twigs-io>

Aaron Lim - <https://github.com/DoubleAPowerBox>

Donovan Pilcher - <https://github.com/Groovyfaun>

Overview of what we are trying to achieve -

A user will select single player mode or two-player mode.

From there, they will enter their player name, and be asked where both their parents are from, and whether their parents were benders, and if so, what kind.

Using the information entered above, their character's power will be generated based on the hereditary probability of them being a bender or not. For example, if their mother is a water-bender, and their dad is a fire-bender, there is a 50% chance of their character being either bender. However if their mom is a water-bender, and dad a non-bender, they would have a 50% chance of being a non-bender.

These steps will be repeated if the user initially selects two-player. If the user selects single-player, they may pick an automatically generated character from the members of Team Avatar, listed below. This will also be the character's stats if the user enters single mode, and enters any of the names from Team Avatar.

Aang - Air Bender

Katara - Water Bender

Toph - Earth Bender

Zuko - Fire Bender

Sokka - Non-Bender

From here. The user will face either Character II or the CPU in a match. It will be similar to rock paper scissors, in the sense that one move will be stronger than another, but can be countered by another.

We will take more time planning to discuss if sub-bending would take too much time for this project. For example, implementing the ability to utilize ice with water-bending, lava for earth-bending, etc.

For the CPU we can try and make it so that there are presets to select from such as the people from Avatar, and make their choice selection randomized

Tools and Technologies -

We plan to use C++ as our primary coding language and we have not decided on having any tools implemented into it

Input/Output -

The input would be as mentioned before text based, and all of the code will be shown as a text based design