The Legume Game

Your main character is a bean, named Garbanzo. You occasionally have a sidekick: Pinto (multi-player).

Storyline: Garbanzo and his friends, raw beans, are dumped into a heated crockpot! Oh No! If they are in the boiling water too long, they will turn to mush and get eaten by a human.

Gameplay: Garbanzo can ride bubbles to the surface to escape, but must avoid falling bean carnage and the dreadful ladle-boss. Additionally, bubbles can pop. Bubbles generate on the floor of the pot and rise to the surface.

You can score points in form of bubble-coin and every time you escape you move on to the next level , which is more difficult and more sparse in bubble-count, and has a different background design.

Sinking to the bottom of the crockpot doesn’t hurt Garbanzo, but the mush-timer is omnipresent and falling will hurt your chance of escaping the pot before mush mania (final stage, when timer runs out, final big boss).

Different bean characters: You start as Garbanzo, and eventually can earn Lima (no popping bubbles), Black (a bean dressed as a ninja), Navy (can swim, but must endure unboiled-pasta-missiles), and the top tier, Red Bean, surrounded by a mochi health-sphere.

Experimental feature: the game plays music sometimes and gives you hints in exchange for bubble-coin.

Technical Features/Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Requirement | Difficulties | Function |
| Main Character | Needs to be controllable in both X and Y axes  Can interact with bubbles  Can fall down  Should be multiple characters | Making sure collisions and physics work  Making different models for each character | jump()  moveLeft()  moveRight() |
| Bubbles | Appears from the bottom of the crockpot and they move up. Should be able to physically interact with main character. They pop after some time | Move up at consistent rate and making sure they pop | floatUp()  pop() |
| Background | Should be changing as levels change | Making sure that each level has a different background |  |
| Special Ability | Should be character specific and allow them to have a bunch of utilities | Making sure that each ability is assigned to each character | activate() |
| Multiplayer | Should allow two characters on the screen controlled by separate keys | Making sure each character is able to move and interact with objects independently | jump() |
| Coins | Displayed at the top left of Screen | Making sure that it remains constant |  |
| Mush-Timer | Timer is displayed in the middle of the top of the screen | Making sure that the timer is accurate |  |
| Ladle Boss | Ladle boss pushes you down by hitting you. The player must avoid the ladle to reach the top. | Having the ladle interact with the character in an accurate and fair way | randomHit() |
|  |  |  |  |

Dependencies: Basic JavaScript and Canvas libraries

Design Assumptions: Moving background, start screen, game over screen

Priority List:

1. Main Character - Necessary
2. Bubbles - Necessary
3. Moving Background - Necessary
4. Mush Timer - Necessary
5. Ladle Boss - Not Necessary but adds extra fun
6. Multiplayer - Not Necessary but adds extra fun
7. Special Ability - Not Necessary but adds extra fun
8. Coins - Not Necessary but adds extra goal
9. Music - Not Necessary but adds ease of playability

Feasibility:

Most difficult items: Main Character, Bubbles, Ladle Boss

Time: Probably several hours

Technical Issue: It is unclear how you are able to win the game

Variables:

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Structure | Global or Local | Description |
| bubbleLocation | Array of ints | Global | Array that is comprised of the x and y location of the bubble |
| Coins | int | Local | Keeps track of the coin count for the player |
| Timer | int | Local | Keeps track of how much time the player has remaining |
| playerLocation | Array of Ints | Global | Array that is comprised of the x and y location of the player |
| ladleLocation | Array of Ints | Global | Array that is comprised of the x and y location of the ladle |

No variable should have considerable impact on processor speed or use too much memory. Depending on what the background is it may or may not take up a lot of memory

None of the functions should really be dependant on any other function although when the game starts play() shouldn't be active and the player location should be somewhere random on the bottom of the map.

Player control should be with the arrow keys for player 1 and WASD for player 2. This means that moveLeft(), moveRight(), and jump() will take these keys as inputs.

\*Deleted Music component

Day 1: Code Primary Bean

Day 2: Start Coding Bubbles

Day 3: Finish Bubbles

Day 4: Code Bean-Bubble interaction

Day 5: Start Ladle

Day 6: Finish Ladle

Day 7: Create Timer

Day 8: Create Secondary Bean

Day 9: Create Tertiary Bean

Day 10: Create Quaternary Bean

Day 11: Create Multiplayer

Day 12: Implement Coins and polish Start and End Screens