Terminal Velocity (prototype)

Short Plot Summary

You stand atop a cliff. The only way you can go is down. Will you?

Required Mechanics

- Walk
- Jump
- Gravity
- Screen size modification to pair with velocity from falling
- Squash and stretch on landing
- Text system
- Pause menu

Controls and Default Keybinds

The below control scheme assumes an Xbox style controller, but it should be noted that Playstation and Switch style controllers are also out there and ought to have different icons for their controls. Controls for keyboard are based on Hollow Knight and Spelunky 2's default keybinds

Mechanic	**Keyboard Scheme**	**Controller Scheme**	**Alt Controller Scheme**	**Notes**
Interact	X	A		Distance for interaction
Walk	A, D	L Joystick x-axis	D-Pad x-axis	
Jump	Z	A		
Open Pause	esc	Y/Start Button		
Scroll	W, A, S, D	L Joystick	D-Pad	
Select	X	A		
Close Menu	esc	Y/Start Button		

Aesthetic Plans

The backgrounds and characters will be designed with a 60x45 viewport size in mind which will be upscaled to the correct size later. The top of the camera view requires a mimic of a header including a label. This should be made using a sprite that rests at the top of the camera's viewport. While the game is being developed, this can be ignored, but it does mean some of the final number might need a bit of modification to include the size changes that would occur with the added space. The window will be locked to the center of a screen unless something in the settings is modified.

The world itself is monochromatic. I may add a single color to emphasize certain locations and make it a bit more magical. I want the vibe at the end of the level to be like the Depths in Rain World. It needs to feel more etherial and a bit less drab that what is above. The game ends with the player looping like Saint entering Rubicon. This is because I am very normal about Rain World.

Resolution

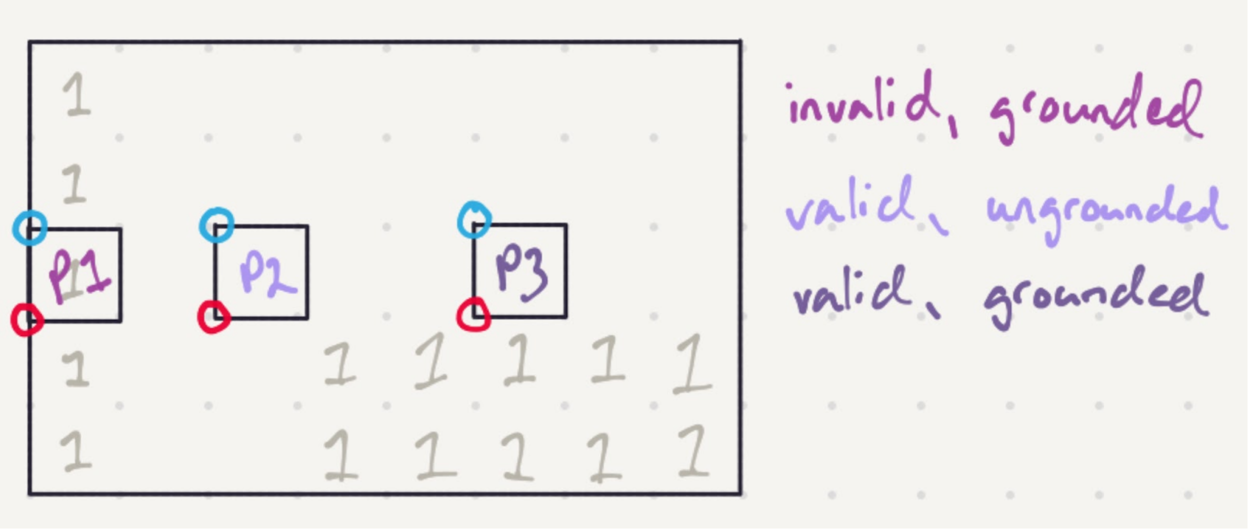
Originally, the design resolution was planned to be 40x30 pixels, but upon testing, I found a slightly larger resolution to be less jarring when the player was moving. The window resolution scales based on the player's downward velocity and clamps the window to a predetermined minimum size.

Location

- The player starts at the top of a cliff with nowhere to go but down
- There are cave systems that the player jumps a bit through, but they cannot get up once they go too far down
- Either way, it's a short romp that gets darker the further you get down until, suddenly, there is something colorful at the end.

Collision Detection

- To make the collisions work, there is a collision vector which is to be made from a 2d image
- The player has both a position vector and integer x and y values, where the ints are the floor of the position vector
- Collision detection and rendering, then, use x and y to place the background texture and compare the player's position to the collision mask



- As shown in this image, there are two checks for valid and grounded per frame drawn
- The position vector is an x and y float within the drawn P1, P2, and P3 pixels
- Last valid position is calculated by reversing the position using the velocity vector for the player until the position is valid
    - LastValidPosition does not need to return a grounded position
    - It also resets velocity so that the calculations don't go beyond what is allowed due to large y velocity
-

Physics

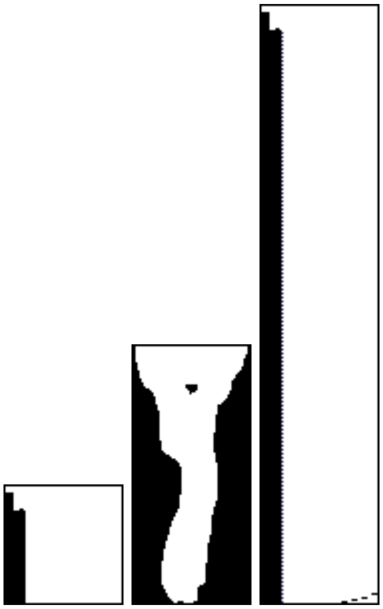
- Player can control x midair, but their x velocity will be halved at least until better variables are found
- Per where the player is ungrounded, the y velocity takes the previous y velocity and adds gravitational constant G times delta time to the velocity
- This new velocity is then used to displace the player's position vector
-

Rendering

- The player is kept stationary in the center of the monitor while the background texture is the part that is moved around
- This prevents any render texture tearing that can happen when using two cameras
- After, the

Level Design Mockups

- Test environments



Sound

- A long ~~droning~~ piece of piano music to emphasize the desolate world.
- Usage of repeating arpegio in the background potentially

- Inspiration:
    - OMORI
        - Lost Library
        - Orchard
    - AI: The Somnium Files (AITSF and AINI)
        - Snowy Nonlogical
        - Psyncin' in the Mountain
    - Spelunky 2
        - Bottom of the Well
    - Little Goody Two Shoes
        - Night Prayer
        - Whispers of the Moon - Overworld by Nighttime
    - Super Paper Mario
        - Memory 1
        - Memory 2
        - Memory 3
        - Memory 4

Concept Art

More finalized concept for the aesthetic