

Process & Decisions Documentation

Goatman Walking Thing Side Quest

Name: Charlie Cilla

Assignment: Side Quest – Multi-Scene Narrative Experience

Date: February 3, 2026

Role: Individual Designer & Developer

Project Overview

This Side Quest is a short, multi-scene narrative experience built in p5.js. The project explores atmosphere, scale, and player choice across four distinct environments. Rather than focusing on mechanical complexity, the goal was to guide the player through a controlled sequence of spaces where interaction feels deliberate and, at times, unsafe.

Each scene introduces a different visual and emotional tone, using image processing, video backgrounds, and gated interactions to move the player toward a final moment that resolves the experience rather than looping endlessly.

Design Intent

From the beginning, I wanted the experience to feel paced and intentional. Movement is simple, but progression is not free. The player is always moving forward, but only when the game allows it.

The environments are meant to feel slightly off. Characters change scale, saturation, and contrast between scenes. Interactions are not always rewarding. One choice in particular leads to immediate failure, reinforcing the idea that curiosity has consequences.

Major Design Decisions

Pixel-Level Image Processing

Early on, I moved away from p5.js filter functions and instead implemented manual, pixel-level image processing for all character sprites. Filters proved unreliable and caused performance issues when applied in real time. Processing each pixel directly gave me consistent results and full control over contrast, saturation, brightness, and blackness.

This decision allowed each scene to have its own visual identity without relying on heavy animation or effects. Emotion is communicated through the character's appearance rather than their movement.

Scene Structure and Gated Progression

Rather than allowing open navigation between areas, each scene is gated through interaction prompts and fade transitions. Players cannot skip scenes or move backward freely. This ensures that the narrative unfolds in sequence and that key moments are experienced as intended.

Fade-to-black transitions act as both pacing tools and psychological resets. Each new space feels separate, even when visual elements like the character persist.

Interaction With Consequences

The interaction system evolved into a two-choice dialogue mechanic during the “Walking Thing” encounter. One option immediately ends the game, while the other advances the story.

This moment was intentionally binary. There is no hidden third option or neutral outcome. The choice reinforces the idea that not everything in the world exists for the player’s benefit.

Process Summary

Development happened through continuous iteration, building the experience scene by scene in a single extended session. I began with Scene 1, focusing on character

movement and processed visuals. Once that foundation was stable, I expanded outward.

Scene 2 introduced video backgrounds, reduced character scale, and the first major interaction with consequences, scene 3 layered multiple interaction types, including a low-stakes “pet” action alongside a progression gate. Scene 4 was designed as a conclusion, using a white flash effect and a non-looping video to signal finality before returning to the title screen.

Throughout development, I frequently adjusted visual processing values per scene to maintain contrast between environments while keeping the character recognizable.

Visual Consistency & Layering

Maintaining readability became a recurring issue as video, foreground elements, and UI stacked on top of each other. Instruction text and prompts were often hidden behind visual layers.

The solution was consistent draw-order control. All UI elements were moved to the end of each scene’s draw cycle, ensuring they always appear above backgrounds, characters, and video layers. This approach was applied across every scene for consistency.

Use of GenAI Tools

Tool Used: GitHub Copilot (VS Code Chat Interface)

Date: February 3, 2026

Purpose: Code assistance, debugging support, and implementation suggestions

Copilot was used as a support tool during development. It assisted with scene routing, video integration, interaction detection, fade systems, and pixel-processing logic. All core design decisions, narrative structure, and visual direction were human-led.

Human Decision Making

I regularly rejected or redirected Copilot suggestions when they conflicted with performance goals or design intent. Decisions such as using manual pixel

manipulation, separating scenes into individual files, implementing flag-based video playback control, and limiting player freedom were made independently.

Narrative choices, including the Walking Thing interaction and the automatic ending, were designed to emphasize tension and finality rather than replayability.

Verification and Testing

All GenAI-assisted code was tested directly in p5.js. Performance issues, particularly with filters and dithering, were identified through live testing and ultimately removed. Video playback, audio behavior, and interaction triggers were repeatedly tested to ensure consistent behavior across scenes.

Limitations & Reflection

Some visual moments could benefit from additional polish, such as particle effects or more expressive transitions. A dithering effect was explored but removed due to performance costs and visual noise that distracted from the experience.

Scene 3's "pet" interaction is intentionally minimal. While it could be expanded, its current form supports the pacing and tone without overstaying its welcome.

Overall, the project demonstrates intentional scene control, visual storytelling through processing rather than animation, and careful use of GenAI tools as support rather than authorship.

Appendix

Appendix

Complete Development Dialogue Transcript

THE GOATMAN – Interactive Multi-Scene Narrative
Charlie Cilla

A. Assignment Context

Session 1: Project Concept and Design Discussion

User Prompt 1.1

This is our task for this week, it is based off of the template repository we have here in ccilla_sidequest_W3: Build an interactive story that unfolds through multiple game states and files, branching like a small decision tree. Bonus: Track a player stat (e.g., trust, health, karma) across scenes and unlock endings based on it

User Prompt 1.2

Yes, lets slow down a bit and figure out what we want to do for this game, design wise, atmosphere and visuals

Assistant Response 1.1-1.2

Discussed project scope and gathered design requirements:

Agent asked key questions about:

- Theme & Genre (fantasy, mystery, horror, etc.)
- Visual Atmosphere and aesthetic preferences
- Story Structure and number of scenes
- Player Stat System options
- Number and types of endings

Status: Established need to clarify visual assets before proceeding with implementation

Session 2: Character Animation System Setup

User Prompt 2.1

If i was to input images of my art pieces would you be able to use them and if i have a character with different animation states how would i implement them into the code?

User Prompt 2.2

Ok, I have a character, the character has in essence three states with a full (18 frames) walking animation (going left- I imagine we can simply flip the image for going right) the first is labeled simply with a (number).png. This is the character with its face straight towards the player. The second is the characters face half turned and its labeled (number)turn1.png. Finally, the third is with the characters head fully turned away and it is labeled (number)turn2.png.

User Prompt 2.3

I have added the images

Assistant Response 2.1-2.3

Implemented complete character animation system:

Created frame loading system in main.js:

- Loaded all animation frames (facing, turn1, turn2 states)
- Organized into characterFrames object
- Handled frame numbering variations (1.png, 2turn1.png, etc.)

Created character state machine in game.js:

- Character object with position, direction, scale, speed
- Animation states: idle, transition_start, walking, transition_end
- Frame cycling with controlled animation speed
- Sprite flipping for directional movement
- A/D key controls for left/right movement
- Smooth transitions between idle and walking

Status: Full character animation working with smooth transitions

Session 3: Manual Pixel Processing Implementation

User Prompt 3.1

can we try the filters again. Im not sure why it didnt work

User Prompt 3.2

I think I would like to use the filters to increase the saturation and contrast

User Prompt 3.3

Is this a possible way to do it? [provided manual pixel processing example code]

User Prompt 3.4

Yes, lets try it

Assistant Response 3.1-3.4

Implemented manual pixel-level processing system:

Abandoned p5.js built-in filter() functions due to performance issues.

Created applyContrastAndSaturation() function:

- Uses img.loadPixels() for direct pixel access
- Applies contrast formula: $(259 * (\text{contrast} + 255)) / (255 * (259 - \text{contrast}))$
- Implements saturation adjustment using grayscale calculation
- Added brightness multiplier
- Added blackness parameter for darkening
- Processes all pixels via nested loop
- Calls img.updatePixels() to apply changes

Applied processing in setup() after images fully loaded:

- Created processedCharacterFrames array
- Applied filters with initial values: contrast 3.0, saturation 0.9, brightness 1.0, blackness 0.6

- Stored processed versions separately from originals

Status: Character visual processing working without performance degradation

Session 4: Visual Processing Refinement

User Prompt 4.1

Ok, now increase the contrast

User Prompt 4.2

Decrease the saturation a little bit and Increase the blackness a bit

User Prompt 4.3

increase blackness a bit

Assistant Response 4.1-4.3

Refined visual processing parameters:

Iteratively adjusted processing values:

- Contrast: 1.3 → 3.0 → final optimized value
- Saturation: 1.4 → 0.9 → 0.57 (final)
- Brightness: 1.0 → 0.9 (final)
- Blackness: 0.0 → 0.6 → 0.85 → 0.45 (final)

Created dark, desaturated, high-contrast aesthetic matching the mysterious forest atmosphere.

Status: Scene 1 character processing finalized

Session 5: Canvas Layout and Centering

User Prompt 5.1

Ok, now make the canvas centered in the browser and make it fill the immediate area so you dont have to scroll down and have the outside edges on the left and right black

Assistant Response 5.1

Implemented responsive canvas centering:

Modified style.css:

- Set html/body to display: flex, justify-content: center, align-items: center
- Added background: #000 for black letterbox
- Set canvas max-width: 90vw, max-height: 100vh
- Hidden header/footer elements
- Removed all margin/padding causing scroll

Status: Canvas centered with black edges, no scrolling

Session 6: Title Screen Video Implementation

User Prompt 6.1

Ok, i have added two new files to images, the first is Goatman Title Screen.mp4. This will be the new title screen it will be frozen on the first frame until a start button is clicked, it will be on the left side of the screen. It will then play the video and once it is done we will enter into the first section of the game using background section1.jpg with the character on the farthest right side of the screen with its size also reduced by 50%.

Assistant Response 6.1

Replaced static title screen with video:

Implemented video title screen in start.js:

- Loaded Goatman Title Screen.mp4 using createVideo()
- Set video to pause on first frame
- Positioned START button on left side (x: 200)

- Added `video.play()` on button click
- Implemented `onended` callback to transition to game screen
- Added skip button that fades in during playback

Loaded Scene 1 assets in main.js:

- Background Section 1.jpg as `section1Background`
- fore.png as `section1Foreground`
- Repositioned character to x: 700 (far right)
- Reduced character scale from 0.5 to 0.25

Status: Cinematic title screen with automatic transition to game

Session 7: Scene 1 Composition and Layering

User Prompt 7.1

I just got this screen: [reference image showing desired character position]

User Prompt 7.2

Also if you can understand this image i am providing please position the character where they are in this image and is there any way to add contrast to better match the colours of the image?

User Prompt 7.3

Make the character x2 smaller

User Prompt 7.4

Now 100 pixels lower on the screen and starting to the farthest right side of the frame

User Prompt 7.5

Can you stretch the fore.png a little bit and move it down slightly

User Prompt 7.6

Can you slightly stretch the character vertically

Assistant Response 7.1-7.6

Refined Scene 1 composition and positioning:

Character adjustments:

- Positioned at x: 750 (far right), y: 600 (100px lower)
- Scale reduced to 0.2
- Added vertical stretch factor of 1.1
- Applied color tinting for atmospheric matching

Foreground adjustments:

- Stretched fore.png to 110% width, 90% height
- Moved down 40 pixels
- Ensures proper layering: background → character → foreground

Status: Scene 1 composition finalized with proper depth

Session 8: Scene Transition System – Enter Prompt

User Prompt 8.1

So if you walk to the farthest right side, you get prompted with the button E to enter the next area

User Prompt 8.2

Actually can we move that to the left side

User Prompt 8.3

Can we finally move the prompt to go to scene two to the left of the main section and have the character enter on that side?

Assistant Response 8.1-8.3

Implemented scene transition prompt system:

Created enter prompt logic in game.js:

- **Added showEnterPrompt flag**
- **Trigger condition: character.x < 120 (left edge)**
- **drawEnterPrompt() renders styled prompt box**
- **E key or click initiates transition**

Created fade transition system in main.js:

- **Added fadeAlpha, isFading, fadingTo variables**
- **startFade(targetScreen) begins transition**
- **updateFade() handles alpha blending (0→255→0)**
- **Screen switches at alpha 127 (halfway point)**
- **Black fade overlay renders centered on canvas**

Status: Smooth fade transitions working between scenes

Session 9: Scene 2 – Video Background and Character

User Prompt 9.1

Its a video i want to loop. and when the character enters this area their size is reduced by 2/3rds

User Prompt 9.2

the background is called GROUND.mp4

Assistant Response 9.1-9.2

Created Scene 2 with video background:

Video implementation in main.js:

- Loaded GROUND.mp4 using `createVideo()`
- Set to loop with volume enabled
- Hidden DOM element
- Conditional playback check in draw loop
- Full canvas rendering

Created scene2.js structure:

- Character2 object with reduced scale (0.15)
- Entry position x: 50 (left side)
- Reduced movement speed (0.5)
- Animation state machine matching Scene 1
- Initially used unprocessed characterFrames

Updated index.html with scene2.js script tag.

Status: Scene 2 functional with looping video background

Session 10: Scene 2 – Walking Thing Interaction

User Prompt 10.1

add an option to interact with the Walking Thing in the middle of scene2, the player has two options that they can click with their mouse. Option one Grab Their LEG. If this option is chosen the game ends with a black screen fade in, "They didn't like that, you were squashed" This will take you back to the title. The other option "Will you be my mount, Walking Thing?". This will take you to scene3

Assistant Response 10.1

Implemented branching dialogue system:

Created interaction detection:

- **interactionAvailable** when $\text{abs}(\text{character2.x} - \text{width}/2) < 80$
- **showInteractionOptions** boolean controls visibility
- **drawInteractionHint()** shows "Click to interact"

Two-choice dialogue system:

- **drawInteractionOptions()** renders both clickable buttons
- Centered positioning at width/2
- **isHover()** helper function for mouse detection

Option 1: Grab Their LEG

- Sets **endingActive = true**
- **drawEndingFade()** creates black fade overlay
- Shows message: They didn't like that, you were SQUASHED
- 120-frame timer then calls **startFade("start")**

Option 2: Will you be my mount, Walking Thing?

- Calls **startFade("scene3")**
- Progresses narrative

Added back button:

- **showBackPrompt** when $\text{character2.x} < 120$
- **drawBackPrompt()** at left edge
- E key or click returns to Scene 1

Status: Branching narrative with consequences implemented

Session 11: Scene 2 – Character Saturation

User Prompt 11.1

increase saturation of character in scene2

Assistant Response 11.1

Created scene-specific character processing for Scene 2:

Added processedCharacterFramesScene2 array in main.js:

- **Applied processing: contrast 1.2, saturation 1.8, brightness 1.0, blackness 0.0**
- **Much higher saturation than Scene 1 or Scene 3**
- **Creates vibrant appearance matching video background**

Updated scene2.js to use processed frames:

- **Modified drawCharacter2() to reference processedCharacterFramesScene2**

Status: Scene 2 character highly saturated and vibrant

Session 12: Scene 3 – Video Background and Pet Interaction

User Prompt 12.1

Ok i have input the video for scene3. Its called savanaScene.mp4

User Prompt 12.2

Ok now have the character appear on the right side of the screen. and when they move a little bit forward they get prompted to "Pet", this will do nothing but say "They liked that". If the character walks about 100px from the left they will be prompted to interact. This will trigger the next scene to play.

User Prompt 12.3

can we widen scene 3 so it fills the frame and also increase the player size slightly and move them up 200px

Assistant Response 12.1-12.3

Implemented Scene 3 with pet mechanic:

Video loading:

- **Loaded savanaScene.mp4**
- **Set to loop with volume enabled**
- **Aspect-ratio-aware scaling to fill canvas**

Character3 object:

- **Spawns at x: 750, y: 550**
- **Scale: 0.08**
- **Direction: -1**
- **Uses scene-specific processed frames**
- **Full animation state machine**

Pet interaction system:

- **showPetPrompt when character3.x < width-200 and character3.x > 300**
- **drawPetPrompt() displays Click to Pet**
- **Click sets petCompleted = true**
- **drawPetMessage() shows They liked that for 120 frames**
- **One-time interaction**

Progression system:

- **showNextScenePrompt when character3.x < 100**
- **drawNextScenePrompt() at left edge**
- **E key or click sets section4ShouldPlay = true**
- **Triggers startFade("scene4")**

Status: Scene 3 complete with pet interaction and progression gate

Session 13: Scene 3 – Visual Refinements

User Prompt 13.1

can we move the background to the left more and reduce the saturation of the character

User Prompt 13.2

reduce bounds of pet prompt to less on the left side and after it is clicked it wont prompt again, increase blackness of character by 0.15 and contrast by a little bit

Assistant Response 13.1-13.2

Refined Scene 3 visuals and interaction:

Background adjustment:

- Added -100px left offset to video rendering

Created processedCharacterFramesScene3 array:

- Applied processing: contrast 1.1, saturation 1.0, brightness 1.0, blackness 0.15

Pet prompt refinement:

- Changed left bound from $x > 200$ to $x > 300$
- petCompleted flag prevents repeated prompts

Status: Scene 3 visuals and mechanics polished

Session 14: Scene 4 – Flash Effect and Ending

User Prompt 14.1

increase saturation to 1 and i have added the scene that plays upon interaction, when it is interacted with flash a bright white screen that fades

in quick and then disappears slowly. The scene is called: Savana scene trigger.mp4

User Prompt 14.2

Finally when the scene ends have it fade to black and return to title screen instead of repeating

Assistant Response 14.1-14.2

Implemented Scene 4 with flash effect and ending:

Scene 4 video loading:

- **Loaded Savana scene trigger.mp4**
- **Set to play once**
- **Initially paused**

Flash effect system:

- **Alpha increases quickly, fades out slowly**
- **White overlay drawn over canvas**

Playback control:

- **section4ShouldPlay flag gates playback**
- **Prevents premature triggering**

Ending callback:

- **Video onended resets state**
- **startFade("start") returns to title**

Status: Complete narrative loop with dramatic ending

Session 15: UI Layering Fixes

User Prompt 15.1

**Ok now, I just need the text instructions that display within the game to appear above every other layer, just so they can be displayed to the player.
the prompt for a/d to walk are hidden behind**

User Prompt 15.2

Now im just getting a black screen

User Prompt 15.3

Still behind

User Prompt 15.4

this too

Assistant Response 15.1-15.4

Fixed UI rendering order across all scenes:

Instructions and prompts moved to render last.

Scene 1:

- background → character → foreground → prompts → instructions**

Scene 2:

- video → character → prompts → ending → instructions**

Scene 3:

- video → character → prompts → interactions → instructions**

Title screen verified unchanged.

Status: All UI text visible above all layers

Development Date: February 1-3, 2026

