# Pondering

Course: GAME 360 - Development with Game Engines

Semester: Fall 2025

Student Name:Gabriel Lugo-Maldonado

Student ID: 01102327

Submission Date: 11/7/2025

# 1. GAME OVERVIEW (1 page)

## Game Title

Pondering

## Genre

Top-down Puzzle Adventure

## Target Audience

Casual players around 10-30 who are familiar with puzzle elements and enjoy simple, low-stakes combat

## Estimated Playtime

10-20 minutes

## Elevator Pitch

Explore a disjointed enigmatic world of dark space in search of a way home. But be wary of the sentry guards who don't take kindly to strangers in their realm. Solve puzzles and fight to secure a path forward.

## Core Gameplay Loop

The core loop of the game involves the player encountering a series of interlocking puzzles in the game world. The player must solve each puzzle one at a time to unlock the key to solving the next puzzle. Meanwhile, enemies will attempt to slow the player down by attacking them.

- Immediate loop: Player explores the world, attacks enemies, collects coins.
- Session loop: Player solves a puzzle, finds new sections of the world
- Long-term loop: Player solves many puzzles, unlocks new abilities

## Unique Selling Points

• Locked abilities behind progression

• Minimalistic art style

• Dark atmosphere

# 2. GAMEPLAY DESIGN (1 page)

Content Guidance: This section details the actual mechanics of how your game is played. Be concrete—describe exact controls, specific mechanics, and clear win/lose conditions. Use examples from your Task 3 project where applicable.

## Core Mechanics

**Primary Mechanic 1:** World Interaction

- **Description:** The player will be able to interact with doors, signs, and items in the world.
- **Activation:** Activated by pressing Left-click.
- **Duration:** Instantaneous

- **Cooldown:** None
- **Visual Feedback:** UI will display the option to interact with the object the player is hovering their mouse over.
- **Audio Feedback:** None
- **Purpose:** The player must interact with objects to solve puzzles and collect information.

**Primary Mechanic 2:** Attacking

- **Description:** The player will be able to attack an enemy in front of them.
- **Activation:** Activated by pressing Left-click.
- **Duration:** 0.5s
- **Cooldown:** 1s
- **Visual Feedback:** The player will display a “swoosh” in front of them.
- **Audio Feedback:** An air swoosh sound effect will be played.
- **Purpose:** The player must defend themselves from enemies in order to pass through certain sections of the world.
- **Balance:** There are no upgrades to attack, so the damage done vs health of enemies must be carefully adjusted.

**Primary Mechanic 3:** Leaping

- **Description**: The player will be able to leap small distances.
- **Activation**: Activated by pressing the Spacebar.
- **Duration**: 1.5s
- **Cooldown**: None
- **Visual Feedback**: The player will leap into the air.
- **Audio Feedback**: A boing sound will be played
- **Purpose**: The player must leap across gaps to reach certain areas of the map.

## Player Controls

List all player inputs and their corresponding actions:

• Movement: WASD / Arrow Keys - Move character in 4 directions

• Leap: Spacebar - Jump forwards across small gaps

• Primary Action: Left Click - Interact/Attack

• Pause: ESC - Pause game

## Win Condition

The player must simply solve all puzzles and reach the end of the map to win. Point score is a secondary optional objective to encourage exploration.

## Lose Condition

There is no typical “game over”. If the player dies, they respawn after 0.5 seconds, but they lose half of their collected points.

## Progression & Difficulty

• **Early Game:** The player must solve puzzles in the early stages of the map to find their sword. They cannot attack, so they must avoid enemies.

• **Mid Game:** Players can attack, so they can trek through more portions of the map. They must find their spring boots to be able to leap across gaps.

- **Late Game:** Players must defeat a strong final puzzle with multiple enemies by utilizing their sword and jet boots.

# 3. TECHNICAL PLAN (1 page)

## Pattern Integration & Expansion

**Singleton Pattern Usage:**

**GameManager:**

- **Current (Task 3):** Manages game states: Menu, Playing
- **Expansion (Final):** Add score tracking, level progression, and Pause state.

**AudioManager:**

- **Current (Task 3):** Plays basic sound effects
- **Expansion (Final):** Add volume controls and new sound effects for abilities

**Observer Pattern Events:**

**Current Events (Task 3):**

- OnCoinCollected - Updates player score & plays audio cue
- OnScoreUpdate - Updates player score UI
- OnLevelCompleted - Reveals victory message & plays audio cue

**New Events Needed:**

- OnEnemyDefeated - Triggers score increase and particle effects
- OnPlayerHurt - Deals damage to player
- OnPlayerDied - Triggers respawn process and subtracts from score
- OnInteract - Triggers interact on object hovered
- OnAttack - Plays audio cue & deals damage to enemy
- OnLeap - Plays audio cue & triggers player leap action
- OnItemPickup - Plays audio cue
- OnPuzzleSolve - Plays audio cue
- OnPause - Pauses the game

**State Machine Implementation:**

**Player State Machine:**

- **Current States (Task 3):** Idle, Moving, Won
- **New States (Final):** Attacking, Dead, Interacting, Paused

**Enemy State Machine:**

- **Current States (Task 3):** N/A
- **New States (Final):** Patrol, Chase, Death

**Game State Machine:**

- **States:** Menu, Playing, Paused, Victory, Death, Settings

## Features to Add (Weeks 11-16)

(Development will follow the GDD template as it already fits quite well)

**Week 11: Player Systems & Core Mechanics**

□ Implement complete player controller with all states

□ Add combat/interaction system (Pattern: State Machine + Observer)

**Week 12: AI Systems & Game Logic**

□ Complete enemy AI with full FSM (Pattern: State Machine)

□ Implement scoring and win/lose detection (Pattern: Singleton + Observer)

**Week 13: Level & Effects**

□ Build complete level with all mechanics

□ Add new SFX to audio system

□ Add particle effects and visual feedback

**Week 14: Polish & Testing**

□ Playtest and balance difficulty

□ Bug fixes and optimization

□ Complete technical documentation

## 4. SCOPE & TIMELINE (1 page)

Content Guidance: This is YOUR reality check. Be brutally honest about what's achievable. It's better to have a polished, complete small game than an incomplete ambitious one. Explicitly state what you WON'T do.

### Core Features (Must Have)

These are essential features that MUST be completed for a functional game.

□ Feature 1: Player movement and basic controls

□ Feature 2: Enemy with patrol/chase

□ Feature 3: Win/lose conditions

□ Feature 4: UI (health, score)

□ Feature 5: Audio system (SFX)

### Stretch Goals (If Time Permits)

These enhance the game but aren't critical for completion.

□ Goal 1: Enemies variants with ranged attacks

□ Goal 2: Save system

□ Goal 3: Background music

### Will NOT Include

❌ Multiple levels - focusing on one complete, polished level

❌ Animation - animation is time consuming and requires fine-tuning that is not affordable

❌ Textures - art, like animation, is extremely time consuming and not important for a prototype

❌ Cutscenes or narrative sequences - would require additional art/animation

❌ Inventory system - adds complexity beyond scope

❌ Boss fights - focusing on core enemy AI first

# 5. ARTISTIC STYLE AND DESIGN CHOICES (1 page)

## Visual Style & Inspiration

**Art Direction:**

Simple, minimalist, low-poly 3D objects. Nothing much more than simple geometric shapes with a mix of low-poly models.

**Inspiration Examples:**

- Game 1: Geometry Dash
- Game 2: American Arcadia
- Art Style: Why - Ease & speed of development

**Color Palette:**

- Primary Colors: Black (#0E0E0E), White (#E0E0E0),
- Accent Colors: Gold (#E0B300) for collectibles, Red (#E00000) for enemies
- Purpose: High contrast for readability.

**Asset Sources:**

□ Unity Asset Store: Low Poly Ultimate Pack, FREE Casual Game SFX Pack

□ Free Resources: OpenGameArt.org (effects)

□ Original Art: Simple shapes in Unity, basic particle effects

## User Interface (UI) Design

**UI Elements Needed:**

- Main Menu: Title, Play Button, Settings Button, Quit Button
- HUD (In-Game): Health (screen vignette), Score (top-left), Controls (bottom-left)
- Pause Menu: Resume, Settings, Main Menu, Quit
- Death Screen: You Died message, Respawn timer
- Victory Screen: Congratulations, Final stats, Main Menu

**UI Mockup/Layout:**

[Include simple ASCII diagram or description]

Example HUD Layout:

```
┌──────────────────────────────────────────┐
│ Score: 1250                              │
│                                          │
│  [GAME AREA]                             │
│                                          │
│  Controls                                │
└──────────────────────────────────────────┘
```

## Sound Design

**Music:** None

**Sound Effects Priority:**

- High Priority: Win, Die, Collect,
- Medium Priority: Attack, Damage, Enemy attacks
- Low Priority: Leap

List specific sound sources: FreeSound.org, Unity Asset Store (FREE Casual Game SFX Pack)

## Accessibility Considerations

□ Color-blind friendly palette (test with colorblind simulator)

□ Readable text size (minimum 16pt for UI)

□ Clear audio cues (don't rely solely on visuals)

□ Adjustable volume controls

# APPENDIX & RESOURCES

## References & Inspiration

- **Tutorial/Resource: Brackeys - https://www.youtube.com/@Brackeys**

## GDD Examples & Templates Used

Professional GDD examples reviewed:

- Diablo Pitch: http://www.graybeardgames.com/download/diablo_pitch.pdf
- BioShock resources: gamedocs.org/documents/
- GDD Template Collection: github.com/mikewesthad/Game-Design-Document-Resources
- Unity GDD Template: Unity Connect Documentation

## Version History

- Version 1.0 | 11/7/25 | Initial GDD submission for Project Setup

---

**Total Page Count:** ~9 pages of core content

**Word Count: ~**1,500 words