**Murder Mystery**

*Game Design Document*

**Copyright notice / author information / boring legal stuff nobody likes**

*Index*

**Index**

1. Index
2. Game Design
   1. Summary
   2. Gameplay
   3. Mindset
3. Technical
   1. Screens
   2. Controls
   3. Mechanics
4. Level Design
   1. Themes
      1. Ambience
      2. Objects
         1. Ambient
         2. Interactive
      3. Challenges
   2. Game Flow
5. Development
   1. Abstract Classes
   2. Derived Classes
6. Graphics
   1. Style Attributes
   2. Graphics Needed
7. Sounds/Music
   1. Style Attributes
   2. Sounds Needed
   3. Music Needed
8. Schedule

*Game Design*

**Summary**

You’ve been invited to a party. When a murder occurs, can you figure out who did it before they kill again?

**Gameplay**

Gameplay will be investigative in nature. The goal is to identify the murderer, however, you know little of the other guests. The player must interrogate the other party guests to come to the ultimate conclusion.

**Mindset**

I would like the player to be suspicious of the other players until they are, without a doubt, not guilty. Withholding information should cause this.

*Technical*

**Screens**

1. Title Screen
   1. Options
   2. Play
      1. Choose player
   3. Create Party
      1. Venue Select
      2. Theme Select
      3. Pets?
   4. Join Party
2. Game
   1. Journal
3. Credits

**Controls**

Player can use a keyboard or controller. The murder triggers after some amount of time. Player can interact with people via button-press, and the player will select the murderer from a list of names in their journal.

**Mechanics**

Are there any interesting mechanics? If so, how are you going to accomplish them? Physics, algorithms, etc.

*Level Design*

**Themes**

1. House
   1. Mood
      1. Cheerful, bright, jovial (before murder)
      2. Dark, foreboding (after murder)
   2. Objects
      1. *Ambient*
         1. Furniture
      2. *Interactive*
         1. Furniture
         2. Food
         3. Guests
         4. Pets

**Game Flow**

1. Player begins by arriving to the party
2. Host will make announcements when everyone arrives.
3. Player will mingle with other characters and can interact with them.
4. Murder occurs!
5. Player must now investigate and find the murder.
6. After collecting enough evidence, the player accuses someone.
7. If right, player wins, if wrong…

*Development*

**Abstract Classes / Components**

1. BasePhysics
   1. BasePlayer
   2. BaseEnemy
   3. BaseObject
2. BaseObstacle
3. BaseInteractable

**Derived Classes / Component Compositions**

1. BasePlayer
   1. PlayerMain
   2. PlayerUnlockable
2. BaseEnemy
   1. EnemyWolf
   2. EnemyGoblin
   3. EnemyGuard (may drop key)
   4. EnemyGiantRat
   5. EnemyPrisoner
3. BaseObject
   1. ObjectRock (pick-up-able, throwable)
   2. ObjectChest (pick-up-able, throwable, spits gold coins with key)
   3. ObjectGoldCoin (cha-ching!)
   4. ObjectKey (pick-up-able, throwable)
4. BaseObstacle
   1. ObstacleWindow (destroyed with rock)
   2. ObstacleWall
   3. ObstacleGate (watches to see if certain buttons are pressed)
5. BaseInteractable
   1. InteractableButton

*Graphics*

**Style Attributes**

What kinds of colors will you be using? Do you have a limited palette to work with? A post-processed HSV map/image? Consistency is key for immersion.

What kind of graphic style are you going for? Cartoony? Pixel-y? Cute? How, specifically? Solid, thick outlines with flat hues? Non-black outlines with limited tints/shades? Emphasize smooth curvatures over sharp angles? Describe a set of general rules depicting your style here.

Well-designed feedback, both good (e.g. leveling up) and bad (e.g. being hit), are great for teaching the player how to play through trial and error, instead of scripting a lengthy tutorial. What kind of visual feedback are you going to use to let the player know they’re interacting with something? That they \*can\* interact with something?

**Graphics Needed**

1. Characters
   1. Party-goers
      1. Player (idle, walking, talking)
      2. Guests (idle, walking, eating, talking)
      3. Murderer (idle, walking, killing)
   2. Other
      1. Cats (idle, walking, running)
      2. Dogs (idle, walking)
2. House
   1. Floor
      1. Wood
      2. Tile
      3. Carpet
   2. Walls
   3. Doors
   4. Furniture
   5. Food
3. Ambient
   1. Light Fixtures
   2. Decorations
   3. Blood stains (on carpet or furniture)

*Sounds/Music*

**Style Attributes**

Again, consistency is key. Define that consistency here. What kind of instruments do you want to use in your music? Any particular tempo, key? Influences, genre? Mood?

Stylistically, what kind of sound effects are you looking for? Do you want to exaggerate actions with lengthy, cartoony sounds (e.g. mario’s jump), or use just enough to let the player know something happened (e.g. mega man’s landing)? Going for realism? You can use the music style as a bit of a reference too.

Remember, auditory feedback should stand out from the music and other sound effects so the player hears it well. Volume, panning, and frequency/pitch are all important aspects to consider in both music *and* sounds - so plan accordingly!

**Sounds Needed**

1. Effects
   1. Soft Footsteps (carpet)
   2. Sharper Footsteps (wood floor, tile)
   3. Victim’s Scream
   4. Door Opening
2. Feedback
   1. Speaking sound (male and female)
   2. Acknowledgement sound (male and female)

**Music Needed**

1. Mysterious title track
2. Bright but calm track for before murder
3. Darker and grim track for after murder

*Schedule*

*(what is a schedule, i don’t even. list is good enough, right? if not add some dates i guess)*

1. develop base classes
   1. base entity
      1. base player
      2. base enemy
      3. base block
   2. base app state
      1. game world
      2. menu world
2. develop player and basic block classes
   1. physics / collisions
3. find some smooth controls/physics
4. develop other derived classes
   1. blocks
      1. moving
      2. falling
      3. breaking
      4. cloud
   2. enemies
      1. soldier
      2. rat
      3. etc.
5. design levels
   1. introduce motion/jumping
   2. introduce throwing
   3. mind the pacing, let the player play between lessons
6. design sounds
7. design music