Project Andromedus  
(working title)

Design Document

#### Lackluster Software Engineering, Inc. 2022

## Index

1. [Index](#e1pmgau96gcf)
2. [Game Overview](#xthk3v7lq97k)
   1. [Summary](#raan6xudj5eb)
   2. [Gameplay](#fxytzxib1owi)
   3. [Mindset](#8zxra7igzw0j)
   4. [Highlights](#tfim2ti7q8ps)
3. [System Design](#p4vl5v43z96z)
   1. [Input](#zbokdsdefjtt)
   2. [Output](#w3nnp8hyo8d0)
   3. [Behaviors](#c1o8uh4doenn)
   4. [Requirements](#70sj49658smi)
      1. Unique Systems
      2. Other Systems
4. [World Design](#w0lj0vhvnapk)
   1. [Setting](#ezvi41t0pltj)
   2. [Subsettings](#bj0bu1603pmi)
5. [Level Designs](#d6feb8qxmgwk)
6. [Visual Design](#3zwibvw9917o)
   1. [Color Palette](#wwis5rfh62o1)
   2. [Stylistic Attributes / Elements](#nfbv3574fe99)
   3. [Effects](#sf7zt0yut0qz)
7. [Music/Sound Design](#m44lom9y3ath)
   1. [Sound Palette / Instrumentation](#fm089gw44vyz)
   2. [Stylistic Attributes / Elements](#sjnj25bv5m3n)
   3. [Effects](#3hz10q59qfbe)
8. [Required Assets](#6nuyp0zcsxjo)
9. [Schedule](#mkx3te5ni251)
10. [Other Notes](#r57nsvy2tsyz)

# Game Overview

## Summary

Brief summary of the game. A kind of elevator pitch. Keep it simple.

Interact and build relationships with your neighbors in this 2D, RPG-like game. Neighbors will also interact with the player via phone calls and text messages using the mobile app portion of the game client.

## Gameplay

What role will the player take on? What is their \*goal\*, what \*obstacles\* stand in their way, and what are the \*tactics\* they'll use to overcome them?

The player will be a single person living in a suburban neighborhood. The player will attempt to build relationships with their neighbors by speaking with them, assisting them, and hanging out with them. Potential obstacles might be a neighbor becoming upset if the player cannot assist them, doesn’t answer their call/text, or somehow offends them.

## Mindset

How will the player feel throughout the game? What is their initial \*expectation\*? Should they feel powerful enough to overcome all the \*obstacles\* right out of the gate? Or should they feel overwhelmed at the challenge, in search of the proper \*tactics\*?

What is the pace of the game? Is it a race against time, or does it require careful and deliberate planning/exploration?

The pace of the game, when playing via the desktop client, will be leisurely, but can be adjusted to a faster pace if the player is getting bored or waiting for something.

## Highlights

* **What makes this game stand out?** Neighbors will contact the player via the mobile app even when they are not actively playing the game.
* **Are there any new mechanics?** The mobile app, voice input for conversations
* **New uses of old mechanics?** TBD
* **Interesting aesthetics?** TBD
* **Twists?** TBD
* **Interesting algorithms?** The neighbor AI will (hopefully) be fairly robust. Also, the plan is to integrate natural language processing to interact with the neighbors.

# System Design

## Input

**Desktop**

* Standard keyboard/mouse 2D controls
* Click to move
* WASD to move
* Voice input to interact with neighbors (proximity-based)

**Mobile App**

* Standard mobile input via touchscreen
* Voice input for “phone calls”
* Software keyboard for “texts” and “emails”

## Output

What kind of reactions will the system generate?

What happens when a player collides with another entity?

Will platforms fall when walked on? Will enemies get knocked back on attack?

Will enemies need to respond to the actions of other enemies?

## Behaviors

* Player
  + Can walk, open doors, interact with objects, speak to neighbors, call neighbors with mobile app, text neighbors with mobile app, email neighbors with desktop client or mobile app.
* Neighbors
  + Can walk open doors, interact with objects, speak to player, call player on mobile app, text player on mobile app, email player (available from desktop client or mobile app)
  + Will react to player based on how well they like the player.
* Props
  + Can be interacted with
  + Can be taken
  + Can be placed
  + Can be given to others

## Requirements

### Unique Systems

Are there any unique systems/algorithms you’re going to have to implement? Describe them, and the process you’re planning to use to implement them.

Some examples might be Portal’s portal system, Spelunky’s level generation, or Spore’s character designer

### Other Systems

* Scene graph
* Application state management
* Nav meshes
* Bezier paths
* A\* path finding
* FSMs, decision trees
* Observer pattern (event listening)

# 

# World Design

## Setting

Time and place, if applicable.

## Subsettings

* Cell block
* Torture chamber
* Crypt
* Hell

# Level Designs

## Cell block

### Goals

Escape

### Obstacles

Cell door, guards

### Tactics

Murder guard, steal key

### Expectation

Might not succeed, but death comes either way

### Learned Mechanics

* Explicitly taught
  + Call for attention / distract.
  + Attack
* Implicitly learned
  + Enemies follow sounds
  + Attacks can be blocked
  + Dead enemies drop loot

… Add as many copies of the previous section as you need… Ctrl+Enter for a page break

# Visual Design

## Color Palette

What kind of color scheme are you planning to use?

Dull, desaturated colors?

Lively, vibrant colors?

How will they be applied to the game objects? Will they be silhouettes? Will they be outlined? Will they be shaded? How so?

Is there an existing work or two you want to draw from? Reference material is key.

## Stylistic Attributes / Elements

Any particular visual motifs (signature shapes/effects) you're planning to incorporate into the overall style?

Cute, round shapes? Aggressive, harsh triangles? Obscured silhouettes?

What about design languages? Should enemies or hazards be of a certain color or shape?

What about to call attention to specific parts of the game?

Nostalgic film noir, tense wide angle cameras, glitchy static?

Any visual influences (cultural, setting)?

Mayan, celtic, post-apocalyptic dystopia, alien, etc.?

## Effects

What kind of visual effects are you going to need for this game?

Motion blur? Bloom? Film noir?

# Music / Sound Design

## Sound Palette / Instrumentation

What are you going for with the instrumentation?

Dark, bassy tones? Happy, high-pitched tones?

Even electronically synthesized instruments can use a well-grounded basis. What real instruments do you want to base your palette on?

Where is the emphasis, which instruments (if any) will carry the focus?

Is there an existing song or two you want to draw from? Reference material is key.

## Stylistic Attributes / Elements

Any particular musical/auditory motifs (signature phrases/effects) you're planning to incorporate?

Does the game have a specific 'theme' you want to reuse throughout the soundtrack?

Will you use musical cues to draw attention to certain parts of the game?

## Effects

What post-processing techniques are you going to need?

Cross-fading? Reverberation? Filters?

# Required Assets

These should probably be roughly in order of use!

## Visual

1. Models
   1. Characters
      1. Player
      2. Enemy
   2. Props
      1. Vase
2. Animations
   1. Player
      1. Idle
      2. Walk
      3. Attack
      4. Hit
      5. Death
   2. Enemy
      1. Idle
      2. Walk
      3. Attack
      4. Hit
      5. Death
3. Maps
   1. Cell block
   2. Crypt

### 

### 

## Audio

1. Music
   1. Cell block theme
      1. Battle variation
2. Sounds
   1. Foot Steps
      1. Soft
      2. Hard
   2. Jumping
      1. Lift-off
      2. Landing
   3. Combat
      1. Hit
      2. Block

...

# Schedule

A rough list of steps towards release.

Aim for a portion of the milestone every X days/weeks.

* Pre-alpha
  + Player
  + Obstacles
  + Basic Enemies
  + Basic Level Designs
  + Temporary graphics
  + Game should be playable and the concept should be roughly defined
* Alpha
  + Advanced obstacles
  + Advanced enemies
  + Rough draft graphics/music/sounds
  + Rough level designs and level-specific assets
  + Game should be near complete and somewhat optimized
  + Assets should be roughed out but well-defined
* Beta
  + Complete set of obstacles/enemies
  + Complete levels
  + Finished assets
  + Debug, debug, debug
  + Optimize, optimize, optimize
* Gamma/delta
  + Release candidates
  + Finalize debugging
* Omega/final!
  + Done!

# Other Notes

blah blah blah additional notes, logs, etc.