**ZOOTOPIA GAME DESIGN DOCUMENT – SUBJECT TO CHANGE WHEN NECESSARY**

Feel free to add gameplay elements or details to this document when necessary. This document should serve only as a guideline towards the game’s completion.

SECTION 1 – Overview

* 1. – Pitch

A tense and strategic platformer utilizing two characters controlled individually, with puzzle elements and a stealth-based focus.

* 1. – Target

The current target audience is within the 15-28 age range, with a story and setting placed in a middle ground between ZooDystopia and the final Zootopia movie. Equivalent of a high E-10+ or Teen rated game. Fans of platformers (Mario, Fez), stealth games (Metal Gear, The Swindle) and action/puzzle games (Legend of Zelda, Castlevania, Metroid) should enjoy this game. Focused on making a compelling game first, with Zootopia being more of a set dressing and story base.

SECTION 2 – Gameplay

2.1 – Player Feel

Smooth and fast momentum based gameplay, with controls similar to platformers such as Duck Game, Fez, and Mario. Wall jumping, variable jump height, wall grabs, sliding, and basic melee attacks are all possible, with gameplay variations between the two characters to create asymmetric and varied gameplay.

2.2 – Game Elements

Gameplay is composed of a mixture of stealth and puzzles, taking full advantage of the asymmetric character abilities. Judy can jump higher, fit through smaller gaps, and wall jump, while Nick can hang on walls indefinitely, fly further when attacking, and see in dark areas.

Can switch between an individually controlled “hotswitch” system, and a simultaneously controlled “single” system. This allows for easier gameplay when both characters are heading to the same location. (Allows both characters to reach an area without having to constantly switch back and forth – think ‘follow’ behavior)

2.2.1 – Minigames

Minigames should be relegated to small actions used to expend time and build tension. A hacking minigame used in terminals, for example, should be designed to be completed as easily as possible. Main purpose should be tension building, i.e. initiating a hacking minigame as enemies funnel towards you, hacking a computer before detection systems come online, etc.

2.2.2 – Level Design

Levels should be built in a “tower” structure. Main direction for progression is up, starting from the ground floor or basement area, with the boss and/or level end at the highest floor. Each floor should be a level in its own right. Create small but compelling puzzles in each.

Separate each level into two paths: a “Nick” path and a “Judy” path. The paths can interconnect when necessary, and puzzles can connect objects between the paths. A terminal on the Judy path could open a door on the Nick path, etc.

Avoid creating overly precise stealth and obtuse puzzles. Precision and difficult puzzles =/= overly precise stealth and obtuse puzzles. No frame perfect jumps, no puzzles with illogical solutions. Game should feel good to play and not punish the player too much. Think Mario or Zelda, not Dark Souls.

Difficulty should build on experience and unique application of existing actions. Depth over breadth.

SECTION 3 - Enemies

3.1 – Enemy AI basics

Three enemy states - works on individual floors: enemies on floor 2 do not react to floor 1 detection.

**Detected**

**Searching**

**Undetected**

**Detected** - within line of sight, enemy chases player, alarms sounded, other guards go to detection point

**Searching** - enemies go back and forth around the area of last sighting, gains a slightly larger sight radius. Alarms reset.

**Undetected** - patrols around designated area, smaller line of sight

Alarms can be deactivated by hacking control terminal, allows for localized detection. (Only enemy that sees you is set to detected)

3.2 – Enemy Types

Boars - standard grunts, walk back and forth, chase when detected

Foxes - fast, large sight radius, low damage, easy to defeat

Rhinos - slow, heavy, high health, charge attack when detected

Tigers/panthers/lions - moderately fast, high-ish sight radius, high damage, easy kill

Cheetahs - moderately fast, short bursts of speed, high damage, easy kill, high-ish sight radius

Turrets - thin line of sight, when activated turret tracks and shoots player, sets off alarm.

Security cameras - swing side to side, when detected, alarm sounds. Follow wiring through walls to find terminal, terminal deactivates camera temporarily.

SECTION 4 - Summary

Game works floor by floor, each floor has set of enemies to avoid, traps, puzzle. Files are basic currency, more files = better ending. Files float like Mario coins, large stashes can be found in more risky areas (drawers, computers surrounded by enemies, etc). Goal to grab as much as possible. One puzzle per room, gain access to computer with large number of files. Puzzle can do with doors, security cameras, etc. Computer has an 'easy' hacking minigame. (Hitting button prompts to fill hacking bar)

Example puzzle: computer located in camera sight cone. One character deactivates camera using a terminal on their end. When deactivated, quickly switch to other character, hack computer before the camera comes back online, retrieve the files, and duck away before detected.

When killed, a percentage of files are lost, and both characters are sent back to the previous checkpoint.