**TECHNICAL DESIGN DOCUMENT:**

**Instructor: Brad Furminger**

**Game Title: Shadow Dash Ninja**

**Author: Kadeem Cherman**

**Course: VGC242 – Game Portfolio 3**



**Table Of Contents:**

Game Overview \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #1

Game Concepts \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #1

Physics Concepts \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #1 Players Goals & Conditions \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #2

Project Setup/ Folder Structure \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #2

Tasks Breakdown & Time Estimates \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #2 & 3

Prototyping Features/ Enemy FSM Pathfinding \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Page #3 & 4

**Game Overview:**

"Shadow Dash Ninjais a fast-paced 3D side-scrolling infinite runner. Players control a ninja character using keyboard and touch gesture inputs, dashing through procedurally generated track segments while dodging traps, jumping over hazards, collecting coins, and using power-ups.

Gameplay progressively increases in difficulty, and players are rewarded for distance, score, and items collected. The game supports level transitions, lives, a restart-able game loop, and UI feedback for scoring. **GAME CONCEPTS:**

Player Control:

* Auto-run forward
* Swipe left/right or arrow keys to change lanes
* Swipe up or spacebar to jump
* Swipe down or S key to slide Track Generation:
* Segmented track with dynamic spawning (Track Manager)
* Obstacles and collectibles randomly placed per segment Collectibles and Power-ups:
* Coins increase score
* Future: shields, magnets, energy boosts

UI:

* Live score display
* Coin count
* Remaining lives/health

**PHYSICS CONCEPTS:**

1. Kinematics and Rigid body
   * While player uses Character Controller (not Rigid body), we simulate grounded movement, jumping, and directional motion using velocity-based control.
2. Collision Triggers
   * Collectibles and obstacles use colliders (trigger or physical) to detect interactions
   * Player loses health when hitting an obstacle
   * Collectibles trigger coin collection via OnTriggerEnter

**Player Goals & Conditions:**

* **Objective:** Reach the point total needed to complete the level and move onto the next level.

* **Win Condition:** Player accumulates point score total 30000.

* **Lose Condition:** Player starts with (3) hearts &losses a heart per damage taken from level obstacles and level boundaries which damage the player and loose 1 heart per a hit/ damage taken.

**Project Setup/ Folder Structure:**

|  |
| --- |
| Assets – Containing project structure folders: |
| Scripts – Containing game logic / Mechanics |
| Prefabs – Any game objects turned into prefabs |
| Materials – Colors/ Textures used for game elements |
| Scenes – Level Scenes |
| UI – Elements of the UI |
| NavMesh – Level design |

**Task Breakdown & Time Estimates:**

|  |  |  |
| --- | --- | --- |
| **Tasks #: Task Descriptions:** | | **Estimated Time:** |
| **1** | Setup Unity Project Initial  & Folders Structure, along with GitHub repo. | 15 Mins |
| **2** | Create player with movement and swipe/touch support | 1 Hour |
| **3** | Design and script dynamic track spawning  (TrackManager) | 2 Hours |
| **4** | Create obstacle prefabs and placement logic | 1.5 Hours |
| **5** | Implement coin collection, score UI, and coin prefabs | 1 Hour |
| **6** | Add UI for score, lives, and game-over screen | 1.5 Hours |
| **7** | Add power-up placeholder (shield, magnet, etc.) | 1 Hour |
| **8** | Implement game over / win conditions and restart loop | 1 Hour |
| **9** | Playtesting & bug fixing | 1.5 Hour – 4 Hours estimates depending on the debugging process. |
| **10** | Polish environment visuals, materials, and  lighting | 1 Hour – 2 Hours |
| **11** | Final build & submission | 30 Mins |

Total estimated time of development: [17 Hours]

**Prototyping Features / Level Progression & Power-ups:**

Track Segmentation and Random Spawning: •Track Manager instantiates segments at runtime

* + Each Track Segment prefab includes:

oObstacles -> With 5 spawn points for random obstacle prefabs oCollectibles -> With 3 spawn points for coins oTrees -> Randomly toggled for background variation oEndpoint -> Used to place the next track segment Score, Coins, and Level Flow

* + Player collects coins, increasing score
  + Upon reaching 30,000 points, victory condition is displayed
  + Player loses health from trap collisions
  + Upon 0 lives, Game Manager triggers loss screen and reset button Next Iteration Steps:
  + Add swipe gesture controls via mobile input
  + Implement object pooling for better mobile performance
  + Add sound effects and background music
  + Power-up system for magnet, invincibility, or slow time
  + UI upgrades: pause menu, distance tracker, animated heart/lives