f

|  |  |
| --- | --- |
| Game Design Document  ***A Veggie Nightmare*** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Project Concept | | | |
| 1 **Player Control** |  | You control a   |  | | --- | | frightened running kid | | in this   |  |  | | --- | --- | | 2D action platformer | game | |
|  | where   |  | | --- | | touches on the screen | | make the player   |  | | --- | | - jump / double jump (touch on a UI button, right-down corner of the screen): one touch makes the player jump, double touch makes the player double-jump.  - attack (touch on a UI button, left-down corner of the screen): the player spawns a light-blue beam from her eyes towards the player’s forward direction. | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 2 **Basic**  **Gameplay** |  | During the game,   |  |  | | --- | --- | | evil vegetables and fruits | appear | | from   |  | | --- | | the right side of the screen | |
|  | and the goal of the game is to   |  | | --- | | reach the finish line without dropping dead from the veggies’ attacks | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3 **Sound**  **& Effects** |  | There will be sound effects   |  | | --- | |  | | and particle effects   |  | | --- | |  | |
|  | [optional] There will also be   |  | | --- | |  | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4 **Gameplay**  **Mechanics** |  | As the game progresses,   |  | | --- | | - health pickups (junk food such as hotdogs or burgers) will appear.  - evil veggies (rolling tomatoes, jumping carrots and static lettuces) will appear: some take out 1 point, others 2 points (the ones taking 2 points have a red point light on them, to let the player know they are tougher).  - The tougher enemies appear as from the middle of the stage.  - Every some fixed time, a full HP recharger will come at full speed from above, right to left, and the player must grab it with his/her finger. | | making it   |  | | --- | |  | |
|  | [optional] There will also be   |  | | --- | | obstacles such as high platforms / walls, being a must for the player to make a double jump (double touch on the screen) | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 5 **User**  **Interface** |  | The   |  | | --- | | **UP:**  - HP bar  - Score text  - Number of level  **DOWN:**  - Attack button (left)  - Jump button (right) | | will   |  | | --- | |  | | whenever   |  | | --- | |  | |
|  | At the start of the game, the title   |  |  | | --- | --- | | A VEGGIE NIGHTMARE | will appear | | | and the game will end when   |  | | --- | | the player reaches the goal (a portal that gets him/her out of the nightmare) | |

|  |  |  |  |
| --- | --- | --- | --- |
| 6 **Other Features** |  | |  | | --- | |  | |

# 

# Project Structure

**Assets:**

- Audio

- Materials

- Models

- Prefabs:

- Characters: Player and Enemies

- Pickups

- Scenes:

*\* Landing*

*\* Level 1*

*\* Level 2*

*\* Level 3*

- Scripts:

- Attributes:

**\*** *Health.cs*

**\*** *HealthBar.cs*

- Core

**\*** *GameManager.cs*

- Control

*\* EnemyController.cs*

*\* PlayerController.cs*

*\* PickupController.cs*

- UI

\* *LandingController.cs*

*\* HUDController.cs*

*\* GameOverController.cs* (controla el UI del canvas cuando pierde)

*\* GameEndController.cs* (controla el UI del canvas cuando gana)

- UI:

- Fonts

- Images

# Project Timeline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Milestone | Description | | Due | |
| **#1** | |  | | --- | | * GDD * Sketch: level design * Models * Project’s structure | | | |  | | --- | | *28/4 - 29/4* | | |
| **#2** | |  | | --- | | * Player controller: automatic forward movement, touch input, jump and attack, follow camera (Cinemachine), animations. * UML diagram for Player * Player’s Health controller * Events: onDeath(), * Game Manager | | | |  | | --- | | *2/5 - 6/5* | | |
|  | |  | |
| **#3** | |  | | --- | | * UML diagram for Enemy: EvilVeggie (inherits EvilMushroom, EvilTomato, EvilCarrot), each enemy has an enum (mild/tough) * Enemy controller: movement, inheritance for different kind of enemies. * HealthBoost class (hamburgers and hotdogs): use prefab and prefab variants (as they all behave the same way, the only change is its rendering model) * Use Raycast for detecting tap on full health boost. | | | |  | | --- | | *9/5 - 13/5* | | |
| **#4** | |  | | --- | | * Landing Scene * Game Over Scene * HighScore * Persistent data (PlayerPrefs): score * Usar Object Pooling para los laser beams que dispara el player. | | | |  | | --- | | *23/5 - 27/5* | | |
| **#5** | |  | | --- | | * Release: bug fixes, deployment to Google Play Store, Itch.io publication * Documentation: ReadMe | | | |  | | --- | | *30/5 - 3/6* | | |
| **Backlog** | |  | | --- | | * *Feature on backlog - not a part of the minimum viable product* * *Feature on backlog - not a part of the minimum viable product* * *Feature on backlog - not a part of the minimum viable product* | | | |  | | --- | |  | | |

# 

# Bugs



# Project Sketch