



TECNOLÓGICO  
NACIONAL DE MÉXICO®



# TECNOLÓGICO NACIONAL DE MÉXICO

## INSTITUTO TECNOLÓGICO DE CIUDAD MADERO

**Departamento: Sistemas**

**Carrera: Ingeniería en Sistemas  
Computacionales**

**Materia: Graficación**

**Tarea: Juego 2D: "Dodge the Creeps!"**

**Unidad: 3**

**Integrantes:**

- **Chiu Flores Fernando.  
22070307**
- **Hernández Martínez José Antonio.  
22070302**

**Maestro: Jorge Peralta Escobar**

**Lugar: Cd. Madero, Tamaulipas.**

**Fecha: 16 de Diciembre del 2024.**

# ÍNDICE

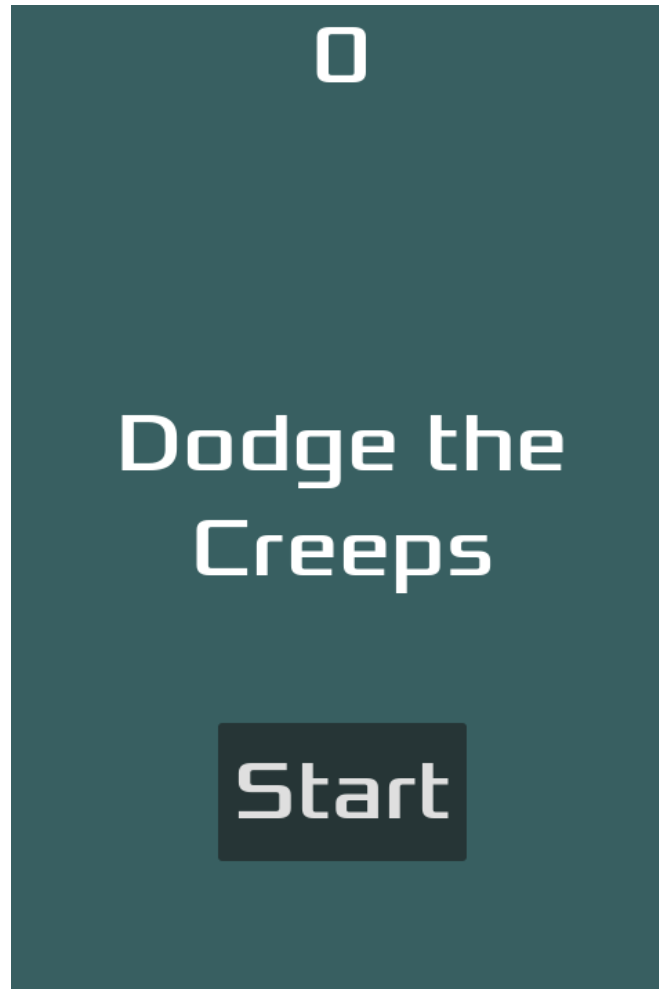
|   |           |
|---|-----------|
| <b>Resumen.....</b>                               | <b>2</b>  |
| <b>Objetivo del Juego .....</b>                   | <b>3</b>  |
| <b>Género y Estilo .....</b>                      | <b>4</b>  |
| <b>Mecánicas del Juego.....</b>                   | <b>4</b>  |
| <b>Personajes y Mundo del Juego .....</b>         | <b>7</b>  |
| <b>Historia.....</b>                              | <b>8</b>  |
| <b>Interfaz de Usuario (UI) .....</b>             | <b>8</b>  |
| <b>Controles .....</b>                            | <b>9</b>  |
| <b>Gráficos y Sonido.....</b>                     | <b>10</b> |
| <b>Tecnologías y Herramientas Utilizadas.....</b> | <b>10</b> |
| <b>Desarrollo y Progreso .....</b>                | <b>10</b> |
| <b>Conclusión .....</b>                           | <b>11</b> |

## Resumen

**Dodge the Creepts** es un juego 2D de acción y supervivencia desarrollado con Godot Engine. El jugador controla un personaje que debe moverse por el mapa para evitar que los enemigos lo toquen. Los enemigos aparecen aleatoriamente en diferentes puntos del mapa, y el jugador debe reaccionar rápidamente para esquivarlos. El objetivo principal es sobrevivir el mayor tiempo posible, lo que se refleja en la puntuación del juego. A medida que pasa el tiempo, la puntuación aumenta en uno por cada segundo que el jugador permanece con vida.



El juego cuenta con una pantalla inicial en la que se presenta un botón "**Start**" para iniciar la partida. Además, incluye efectos de sonido para diversas interacciones, como un sonido de derrota cuando el jugador pierde y un sonido general de fondo durante la partida.



### Objetivo del Juego

El objetivo principal de **Dodge the Creepts** es sobrevivir el mayor tiempo posible evitando que los enemigos toquen al jugador. Cada segundo que el jugador sobrevive, la puntuación incrementa en uno. El jugador debe usar las teclas de dirección (arriba, abajo, izquierda y derecha) para moverse y esquivar a los enemigos. El juego se reinicia cuando el jugador es tocado por un enemigo, y el tiempo de supervivencia se pierde.

El desafío principal radica en la habilidad del jugador para reaccionar con rapidez y precisión para evitar los enemigos mientras controla el movimiento del personaje en un espacio limitado.

## Género y Estilo

**Género:** Acción, Supervivencia.

**Estilo gráfico:** 2D

**Estilo de Juego:** Arcade. Este es un juego que pone a prueba los reflejos y la agilidad del jugador mientras evade a los enemigos.

**Dificultad:** A medida que el jugador progresa en el tiempo, el desafío aumenta debido a que los enemigos aparecen más rápido y de forma más impredecible.

## Mecánicas del Juego

**Main principal:** Este código gestiona el ciclo de un juego, incluyendo el inicio y el fin. En la función `game_over()`, se detienen los temporizadores, se muestra la pantalla de "Game Over", y se reproduce el sonido de muerte. En `new_game()`, se reinicia el juego, se elimina a los mobs existentes, se reinicia la puntuación, y se posiciona al jugador en su punto de inicio. Los mobs se generan con direcciones y velocidades aleatorias en `_on_MobTimer_timeout()`, mientras que la puntuación se incrementa y actualiza cada vez que el temporizador de puntuación se agota. Al finalizar el temporizador de inicio, los temporizadores de mobs y puntuación comienzan.

```
1  extends Node
2  @export var mob_scene: PackedScene
3  var score
4  func game_over():
5      >> $ScoreTimer.stop()
6      >> $MobTimer.stop()
7      >> $HUD.show_game_over()
8      >> $Music.stop()
9      >> $DeathSound.play()
10 func new_game():
11     >> get_tree().call_group(&"mobs", &"queue_free")
12     >> score = 0
13     >> $Player.start($StartPosition.position)
14     >> $StartTimer.start()
15     >> $HUD.update_score(score)
16     >> $HUD.show_message("Get Ready")
17     >> $Music.play()
18 func _on_MobTimer_timeout():
19     >> var mob = mob_scene.instantiate()
20     >> var mob_spawn_location = get_node(^"MobPath/MobSpawnLocation")
21     >> mob_spawn_location.progress = randi()
22     >> var direction = mob_spawn_location.rotation + PI / 2
23     >> mob.position = mob_spawn_location.position
24     >> direction += randf_range(-PI / 4, PI / 4)
25     >> mob.rotation = direction
26     >> var velocity = Vector2(randf_range(150.0, 250.0), 0.0)
27     >> mob.linear_velocity = velocity.rotated(direction)
28     >> add_child(mob)
29 func _on_ScoreTimer_timeout():
30     >> score += 1
31     >> $HUD.update_score(score)
32 func _on_StartTimer_timeout():
33     >> $MobTimer.start()
34     >> $ScoreTimer.start()
```

**Movimiento del Jugador:** El jugador controla al personaje con las teclas de dirección (arriba, abajo, izquierda y derecha). El personaje se mueve de manera fluida y rápida, permitiendo al jugador evadir los enemigos con agilidad.

Este código es para un jugador en Godot que se mueve usando las teclas direccionales (arriba, abajo, izquierda, derecha). Controla la velocidad de movimiento, muestra animaciones de sprite según la dirección del movimiento, y asegura que el jugador no salga de los límites de la pantalla. Al ser tocado por otro objeto, el jugador se oculta y emite una señal hit, además de desactivar su colisión. El jugador se inicia en una posición específica con la función start, y las animaciones se ajustan según la dirección del movimiento.

```
1 extends Area2D
2 signal hit
3 @export var speed = 400 # How fast the player will move (pixels/sec).
4 var screen_size # Size of the game window.
5 func _ready():
6     screen_size = get_viewport_rect().size
7     hide()
8 func _process(delta):
9     var velocity = Vector2.ZERO # The player's movement vector.
10    if Input.is_action_pressed(&"move_right"):
11        velocity.x += 1
12    if Input.is_action_pressed(&"move_left"):
13        velocity.x -= 1
14    if Input.is_action_pressed(&"move_down"):
15        velocity.y += 1
16    if Input.is_action_pressed(&"move_up"):
17        velocity.y -= 1
18    if velocity.length() > 0:
19        velocity = velocity.normalized() * speed
20        $AnimatedSprite2D.play()
21    else:
22        $AnimatedSprite2D.stop()
23    position += velocity * delta
24    position = position.clamp(Vector2.ZERO, screen_size)
25    if velocity.x != 0:
26        $AnimatedSprite2D.animation = &"right"
27        $AnimatedSprite2D.flip_v = false
28        $Trail.rotation = 0
29        $AnimatedSprite2D.flip_h = velocity.x < 0
30    elif velocity.y != 0:
31        $AnimatedSprite2D.animation = &"up"
32        rotation = PI if velocity.y > 0 else 0
33 func start(pos):
34     position = pos
35     rotation = 0
36     show()
37     $CollisionShape2D.disabled = false
38 func _on_body_entered(_body):
39     hide() # Player disappears after being hit.
40     hit.emit()
41     # Must be deferred as we can't change physics properties on a physics callback.
42     $CollisionShape2D.set_deferred(&"disabled", true)
```

**Pantalla de Inicio:** Al inicio del juego, aparece un panel 2D con un botón "Start". El jugador debe hacer clic en este botón para comenzar la partida. Este botón actúa como un punto de inicio para la experiencia de juego.



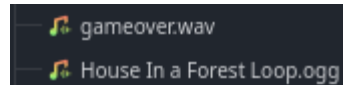
**Aparición de Enemigos:** Los enemigos aparecen de manera aleatoria en diferentes puntos del mapa, lo que agrega un desafío constante. El jugador debe estar atento para detectar sus ubicaciones y moverse rápidamente para evitar el contacto con ellos.

**Puntuación:** La puntuación se incrementa cada segundo que el jugador sobrevive sin ser tocado por los enemigos. La puntuación se muestra en la parte superior de la pantalla y aumenta en una unidad cada vez que pasa un segundo.



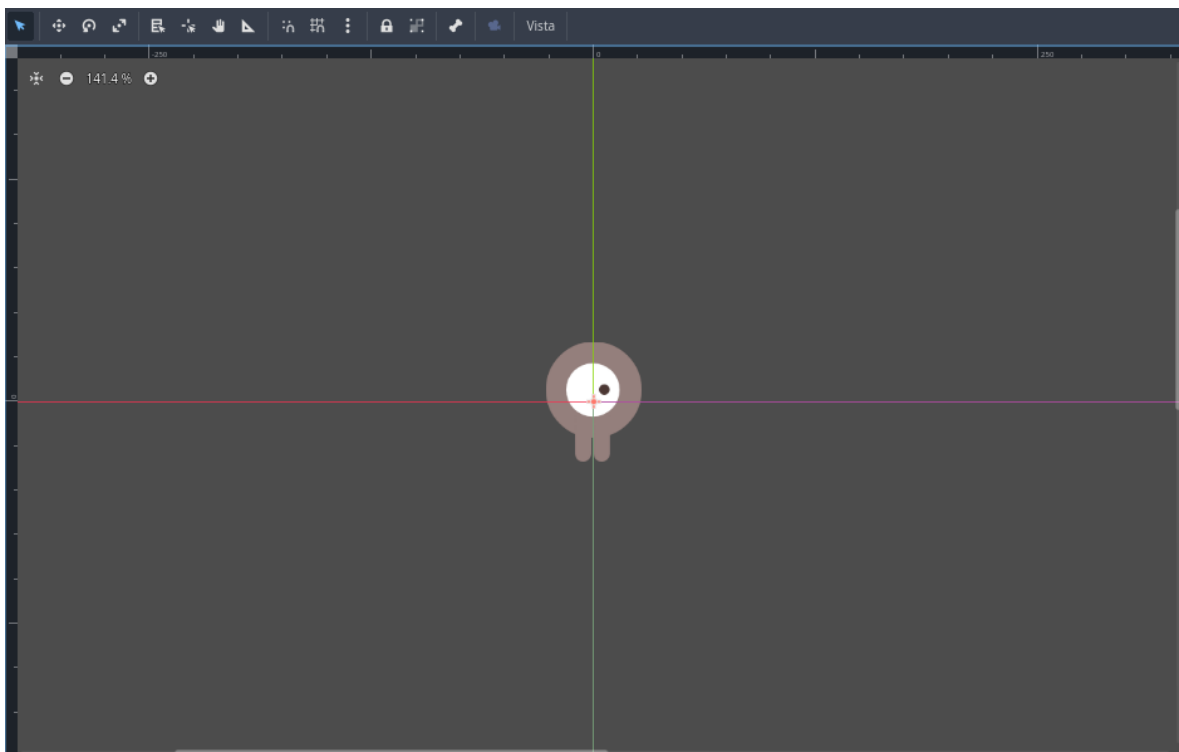
**Finalización del Juego:** Si el jugador es tocado por un enemigo, se activa un sonido de derrota y el juego regresa a la pantalla de inicio. Desde allí, el jugador puede presionar nuevamente el botón "Start" para intentar una nueva partida.

**Sonidos:** Durante el juego, suena música de fondo para mantener la atmósfera activa y dinámica. Al perder el juego (cuando el jugador es tocado por un enemigo), suena un sonido de derrota que indica el final de la partida. Además, cuando se presiona el botón "Start", se reproduce un sonido breve que señala el inicio de la partida.



## Personajes y Mundo del Juego

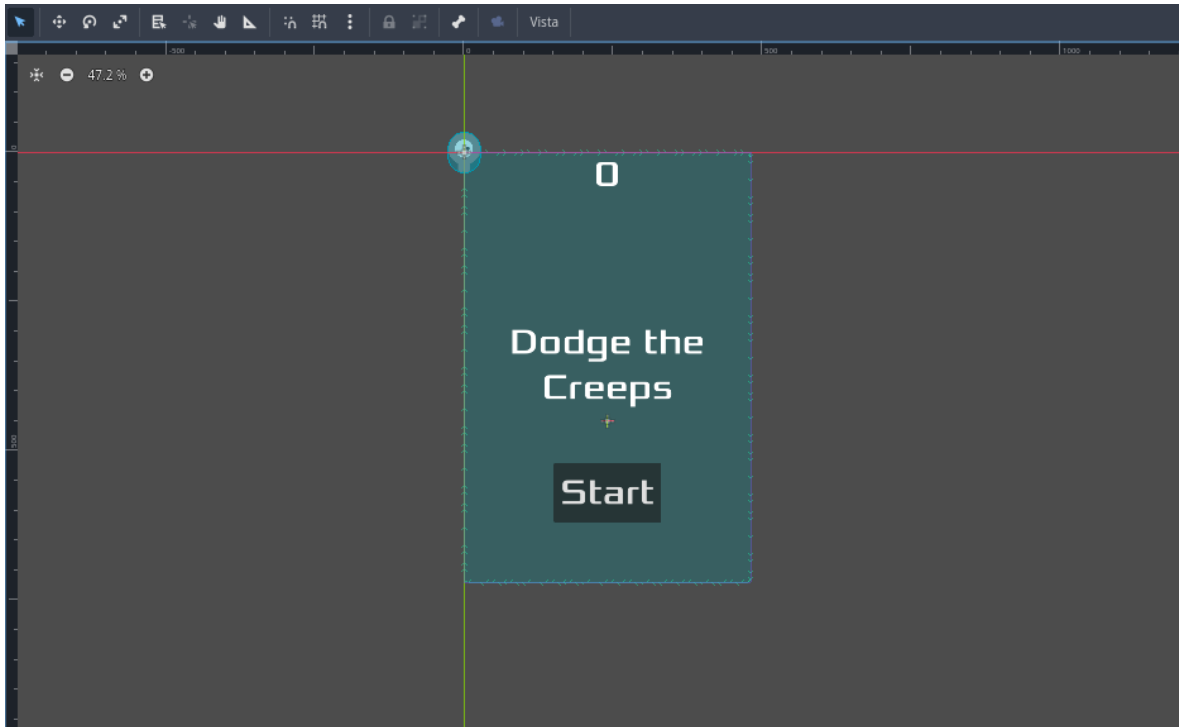
**Personaje:** El jugador controla un personaje cuya principal habilidad es moverse rápidamente para evitar los enemigos. El personaje no tiene otras habilidades especiales, por lo que la clave del éxito radica en el control preciso de su movimiento.



**Enemigos:** Los enemigos aparecen de forma aleatoria en el mapa. Estos enemigos no tienen patrones de movimiento complejos, pero su aparición impredecible aumenta la dificultad del juego. El jugador debe estar constantemente atento y preparado para esquivar los enemigos, lo que requiere rapidez y concentración.



**Entorno y Mapas:** El juego se desarrolla en un entorno 2D simple y colorido. El mapa es limitado, lo que hace que el jugador tenga que gestionar bien el espacio y los movimientos para evitar quedar atrapado entre los enemigos o los bordes del mapa.



## Historia

El concepto principal de **Dodge the Creepts** es la supervivencia pura. El jugador debe evitar que los enemigos lo toquen y sobrevivir el mayor tiempo posible. No obstante, si se deseara añadir una historia, esta podría centrarse en el personaje principal siendo atacado por extrañas criaturas (los "Creepts") y buscando una manera de escapar o sobrevivir en un mundo hostil.

## Interfaz de Usuario (UI)

**Pantalla de Inicio:** Al inicio, el jugador se encuentra con una pantalla sencilla con un botón "Start". Esta pantalla 2D le permite al jugador dar inicio a la partida.

**Pantalla de Juego:** Durante el juego, la pantalla muestra la puntuación actual en la parte superior, la cual se actualiza constantemente mientras el jugador sobrevive.

**Pantalla de Fin de Juego:** Cuando el jugador pierde, se reproduce un sonido de derrota, y el juego vuelve a la pantalla de inicio donde se puede ver nuevamente el botón "Start" para reiniciar el juego.

Este código maneja la interfaz de usuario (UI) en Godot, mostrando mensajes y controlando el inicio del juego. Tiene una señal `start_game` que se emite al iniciar el juego. La función `show_message(text)` muestra un mensaje en la pantalla y lo oculta después de un tiempo usando un temporizador. `show_game_over()` muestra el mensaje "Game Over", luego cambia el texto a "Dodge the Creeps" antes de mostrar un botón de inicio. La función `update_score(score)` actualiza la puntuación mostrada. Al presionar el botón de inicio, se oculta y se emite la señal `start_game`. Finalmente, el temporizador oculta el mensaje cuando se agota.

```
1 extends CanvasLayer
2 signal start_game
3 func show_message(text):
4     >> $MessageLabel.text = text
5     >> $MessageLabel.show()
6     >> $MessageTimer.start()
7 func show_game_over():
8     >> show_message("Game Over")
9     >> await $MessageTimer.timeout
10    >> $MessageLabel.text = "Dodge the\nCreeps"
11    >> $MessageLabel.show()
12    >> await get_tree().create_timer(1).timeout
13    >> $StartButton.show()
14 func update_score(score):
15    >> $ScoreLabel.text = str(score)
16 func _on_StartButton_pressed():
17    >> $StartButton.hide()
18    >> start_game.emit()
19 func _on_MessageTimer_timeout():
20    >> $MessageLabel.hide()
```

## Controles

**Movimiento del Personaje:** El jugador utiliza las teclas de flecha (arriba, abajo, izquierda, derecha) para mover al personaje por el mapa. El movimiento es suave y rápido, permitiendo una fácil evasión de los enemigos.

**Botón de Inicio:** En la pantalla de inicio, el jugador debe hacer clic en el botón "Start" para comenzar el juego. Esto activa la partida y comienza el conteo del tiempo de supervivencia.



## Gráficos y Sonido

**Gráficos:** El estilo visual del juego es simple y minimalista, lo que permite que el jugador se enfoque en el objetivo principal: evitar a los enemigos. Los gráficos 2D tienen colores llamativos para los personajes y los enemigos, lo que facilita su visibilidad.

**Sonido:** El juego incluye música de fondo que se reproduce durante toda la partida. Además, hay efectos de sonido que hacen la experiencia más inmersiva:

**Sonido de Inicio:** Un sonido breve y alegre cuando el jugador hace clic en "Start" para comenzar el juego.

**Sonido de Derrota:** Un sonido característico de derrota que suena cuando el jugador es tocado por un enemigo.

**Música de Fondo:** Música dinámica que crea una atmósfera tensa y emocionante mientras el jugador juega.

## Tecnologías y Herramientas Utilizadas

**Motor de Juego:** Godot Engine, un motor de código abierto ideal para el desarrollo de juegos 2D.

**Lenguaje de Programación:** GDScript, el lenguaje de programación de Godot, utilizado para crear la lógica del juego.

**Gráficos:** Herramientas 2D dentro de Godot para crear los personajes y enemigos, o bien programas de diseño gráfico externos si se utilizaron.

**Sonido:** Archivos de sonido en formatos como .ogg o .wav, usados en el juego para los efectos de sonido y la música de fondo.

## Desarrollo y Progreso

El juego se encuentra en la fase inicial de desarrollo, con las mecánicas principales implementadas (movimiento del personaje, aparición de enemigos, puntuación y sonido de derrota). El juego ya es jugable y está listo para ser probado.

A medida que el desarrollo continúe, se añadirán características adicionales como diferentes tipos de enemigos, niveles con mayor dificultad, mejoras en los gráficos y la posibilidad de que el jugador gane power-ups u otras mecánicas.

## Conclusión

**Dodge the Creepts** es un juego que pone a prueba las habilidades de los jugadores para reaccionar rápidamente y sobrevivir en un entorno lleno de enemigos. El desarrollo en Godot ha sido una excelente oportunidad para aprender sobre la programación de juegos y las interacciones entre diferentes elementos como la puntuación, los enemigos y los efectos de sonido. Aunque es un proyecto relativamente simple, ha sido una gran experiencia para comprender los fundamentos del desarrollo de videojuegos y la importancia del diseño de jugabilidad.