DOKUMENTASI "FROGGIN' RUN!"

== NKBB Team ==

Anggota Kelompok:

1. Naufal Gastiadirrijal Fawwaz Alamsyah   (1201220446)
2. Rizky Hidayatullah   (1201220027)
3. Farhan Nugraha Sasongko Putra   (1201220449)

Dokumentasi Demo Game:

Script *main.gd*:

```gdscript
extends Node

#preload obstacles
var stump_scene = preload("res://scenes/stump.tscn")
var rock_scene = preload("res://scenes/rock.tscn")
var barrel_scene = preload("res://scenes/barrel.tscn")
var bird_scene = preload("res://scenes/bird.tscn")
var other_scene = preload("res://scenes/other.tscn")
var obstacle_types := [stump_scene, rock_scene, barrel_scene]
var obstacles : Array
var bird_heights := [200, 390]
var other_heights := [300, 490]

#game variables
const DINO_START_POS := Vector2i(150, 485)
const CAM_START_POS := Vector2i(576, 324)
var difficulty
const MAX_DIFFICULTY : int = 2
var score : int
const SCORE_MODIFIER : int = 10
var high_score : int
var speed : float
const START_SPEED : float = 10.0
const MAX_SPEED : int = 25
const SPEED_MODIFIER : int = 5000
var screen_size : Vector2i
var ground_height : int
var game_running : bool
var last_obs

# Called when the node enters the scene tree for the first time.
func _ready():
	screen_size = get_window().size
	ground_height = $Ground.get_node("Sprite2D").texture.get_height()
	#if game_running :
		#if Input.is_action_pressed("ui_accept"):
			#new_game()
		#
	$GameOver.get_node("Button").pressed.connect(new_game)
	new_game()
#...41#
```

```gdscript
#...41#

func new_game():
	#reset variables
	score = 0
	show_score()
	game_running = false
	get_tree().paused = false
	difficulty = 0

	#delete all obstacles
	for obs in obstacles:
		obs.queue_free()
	obstacles.clear()

	#reset the nodes
	$Dino.position = DINO_START_POS
	$Dino.velocity = Vector2i(0, 0)
	$Camera2D.position = CAM_START_POS
	$Ground.position = Vector2i(0, 0)

	#reset hud and game over screen
	$HUD.get_node("StartLabel").show()
	$GameOver.hide()

# Called every frame. 'delta' is the elapsed time since the previous frame.
func _process(delta):
	if game_running:
		#speed up and adjust difficulty
		speed = START_SPEED + score / SPEED_MODIFIER
		if speed > MAX_SPEED:
			speed = MAX_SPEED
		adjust_difficulty()

		#generate obstacles
		generate_obs()

		#move dino and camera
		$Dino.position.x += speed
		$Camera2D.position.x += speed
	#...81#
```

```
1   #...81#
2   #update score
3   score += speed
4   show_score()
5
6   #update ground position
7   if $Camera2D.position.x - $Ground.position.x > screen_size.x * 1.5:
8       $Ground.position.x += screen_size.x
9
10  #remove obstacles that have gone off screen
11  for obs in obstacles:
12      if obs.position.x < ($Camera2D.position.x - screen_size.x):
13          remove_obs(obs)
14  else:
15      if Input.is_action_pressed("jump"):
16          game_running = true
17          $HUD.get_node("StartLabel").hide()
18
19  func generate_obs():
20      #generate ground obstacles
21      if obstacles.is_empty() or last_obs.position.x < score + randi_range(300, 500):
22          var obs_type = obstacle_types[randi() % obstacle_types.size()]
23          var obs
24          var max_obs = difficulty + 1
25          for i in range(randi() % max_obs + 1):
26              obs = obs_type.instantiate()
27              var obs_height = obs.get_node("Sprite2D").texture.get_height()
28              var obs_scale = obs.get_node("Sprite2D").scale
29              var obs_x : int = screen_size.x + score + 100 + (i * 100)
30              var obs_y : int = screen_size.y - ground_height - (obs_height * obs_scale.y / 2) + 5
31              last_obs = obs
32              add_obs(obs, obs_x, obs_y)
33          #additionally random chance to spawn a bird
34          if difficulty == MAX_DIFFICULTY:
35              if (randi() % 2) == 0:
36                  #generate bird obstacles
37                  obs = bird_scene.instantiate()
38                  var obs_x : int = screen_size.x + score + 100
39                  var obs_y : int = bird_heights[randi() % bird_heights.size()]
40                  add_obs(obs, obs_x, obs_y)
41                  #...121#
```

```
1       #...121#
2       elif (randi() % 2)==0:
3           #generate other obstacles
4           obs = other_scene.instantiate()
5           var obs_x : int = screen_size.x + score + 100
6           var obs_y : int = other_heights[randi() % other_heights.size()]
7           add_obs(obs, obs_x, obs_y)
8
9   func add_obs(obs, x, y):
10      obs.position = Vector2i(x, y)
11      obs.body_entered.connect(hit_obs)
12      add_child(obs)
13      obstacles.append(obs)
14
15  func remove_obs(obs):
16      obs.queue_free()
17      obstacles.erase(obs)
18
19  func hit_obs(body):
20      if body.name == "Dino":
21          game_over()
22
23  func show_score():
24      $HUD.get_node("ScoreLabel").text = "SCORE: " + str(score / SCORE_MODIFIER)
25
26  func check_high_score():
27      if score > high_score:
28          high_score = score
29          $HUD.get_node("HighScoreLabel").text = "HIGH SCORE: " + str(high_score / SCORE_MODIFIER)
30
31  func adjust_difficulty():
32      difficulty = score / SPEED_MODIFIER
33      if difficulty > MAX_DIFFICULTY:
34          difficulty = MAX_DIFFICULTY
35
36  func game_over():
37      check_high_score()
38      get_tree().paused = true
39      game_running = false
40      $GameOver.show()
```

Script *dino.gd*:

```
1   extends CharacterBody2D
2
3   const GRAVITY : int = 4200
4   const JUMP_SPEED : int = -1700
5   const FALL : int = 3000
6   const DASH : int = 200
7   const BACK : int = -500
8
9   # Called every frame. 'delta' is the elapsed time since the previous frame.
10  func _physics_process(delta):
11      velocity.y += GRAVITY * delta
12      if is_on_floor():
13          if not get_parent().game_running:
14              $AnimatedSprite2D.play("idle")
15          else:
16              $RunCol.disabled = false
17              if Input.is_action_pressed("jump"):
18                  velocity.y = JUMP_SPEED
19              elif Input.is_action_just_pressed("ui_right"):
20                  velocity.x = DASH
21              elif Input.is_action_just_released("ui_right"):
22                  velocity.x = 0
23              elif Input.is_action_just_pressed("ui_left"):
24                  velocity.x = BACK
25                  #...25#
```

```
1               #...25#
2               elif Input.is_action_just_released("ui_left"):
3                   velocity.x = 0
4                   #$JumpSound.play()
5               elif Input.is_action_pressed("ui_down"):
6                   $AnimatedSprite2D.play("duck")
7                   $RunCol.disabled = true
8               else:
9                   $AnimatedSprite2D.play("run")
10      elif !is_on_floor():
11          if Input.is_action_pressed("ui_down"):
12              velocity.y = FALL
13          elif Input.is_action_just_pressed("ui_right"):
14              velocity.x = DASH
15          elif Input.is_action_just_released("ui_right"):
16              velocity.x = 0
17          elif Input.is_action_just_pressed("ui_left"):
18              velocity.x = BACK
19          elif Input.is_action_just_released("ui_left"):
20              velocity.x = 0
21          else:
22              $AnimatedSprite2D.play("jump")
23
24      else:
25          $AnimatedSprite2D.play("jump")
26
27      move_and_slide()
```