Easter camp adventure

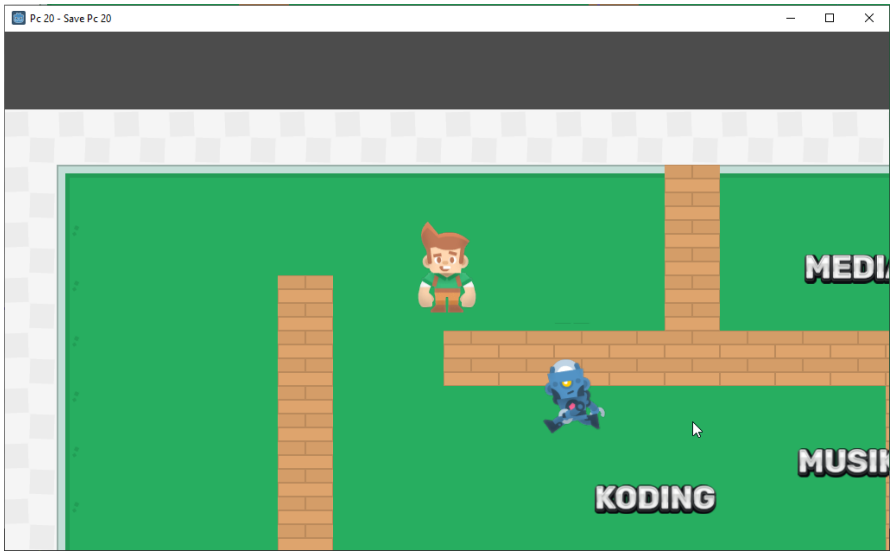
Godot 3.2 Tutorial, Påskecamp Online 2020 Koding

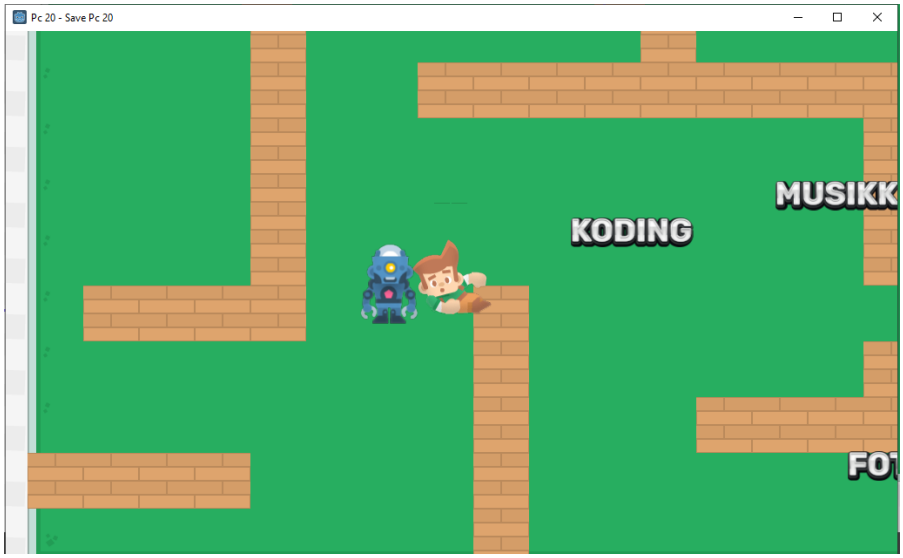
Introduction

This is a tutorial prepared for the Easter camp online. Goal of this tutorial is to make you familiar with the Godot Game Engine. Together, we'll be creating an adventure game, where players goal will be to collect keys to unlock different activities for the Easter camp and avoid the corona virus.

Game to be created

Some screen shoots:



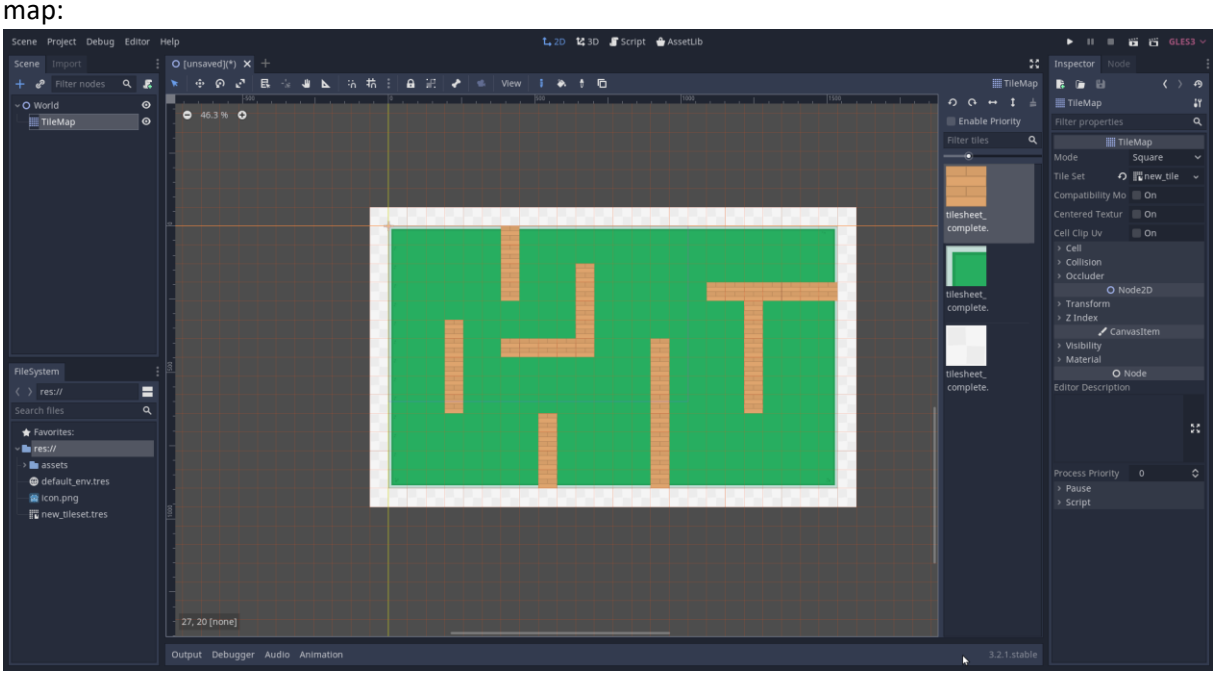


Chapter 1

Creating the world and map

Steps to do:

1. Download Godot 3 Game Engine,
2. Download all the assets from this page: https://github.com/akselon/tutorials/blob/master/godot3-eastercamp_adventure/assets.zip?raw=true
3. Create Node 2D and change the name to “World”, add TileMap, choose tiles and create a map:



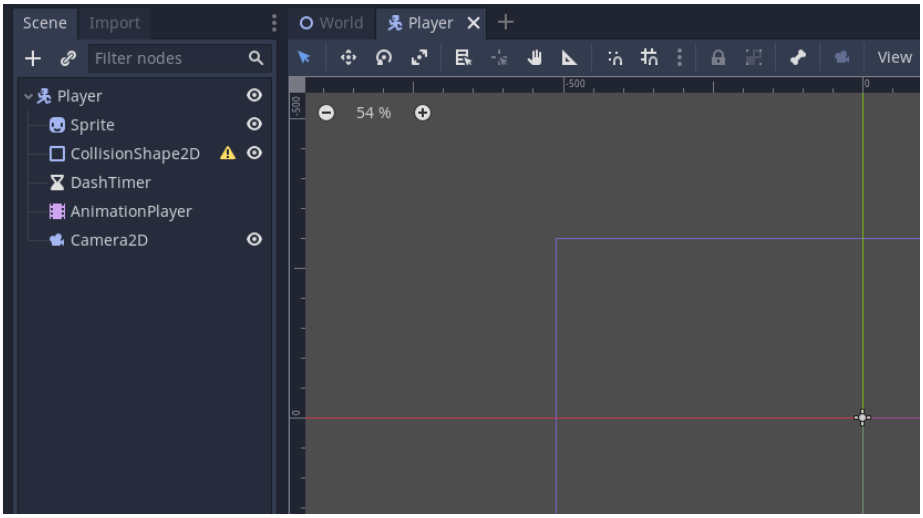
Download the project after Chapter 1 from here: Link

Chapter 2

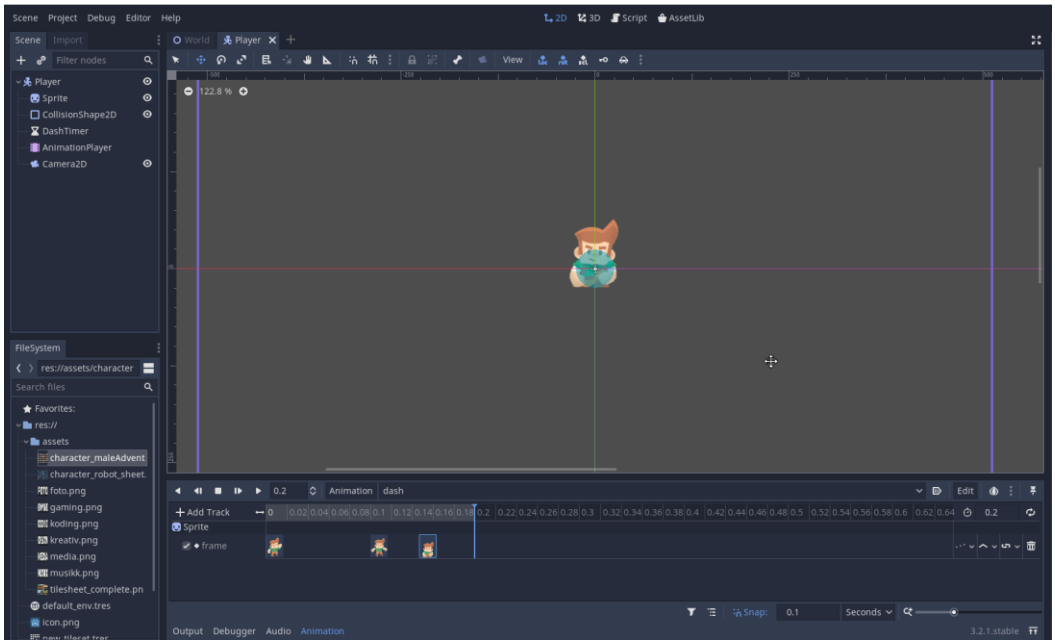
Player scene

Steps to do:

1. Create new scene with those nodes:



2. Add:
    a. The character_maleAdventure Sprite,
    b. Collision shape,
    c. Set Camera as current,
    d. Add animations (don't use Bezier Curves!): idle, move, dash.



3. Add Input Map,
4. Add the Players script:

```
extends KinematicBody2D
class_name Player

signal end_game

export (int) var speed = 200
export (int) var dash_speed = 800
export (int) var activities_to_collected = 1

onready var anim_player = $AnimationPlayer
onready var sprite = $Sprite
onready var dash_timer = $DashTimer

var facing_right = false
var is_dashing = false
var velocity = Vector2()
var activities_collected = 0

func get_input():
    velocity = Vector2()
    if Input.is_action_pressed("move_right"):
            velocity.x += 1
    if Input.is_action_pressed("move_left"):
            velocity.x -= 1
    if Input.is_action_pressed("move_down"):
            velocity.y += 1
    if Input.is_action_pressed("move_up"):
```

```
                velocity.y -= 1

        if is_dashing:
                velocity = velocity.normalized() * dash_speed
        else:
                velocity = velocity.normalized() * speed

        if facing_right and velocity.x > 0:
                flip()
        if !facing_right and velocity.x < 0:
                flip()

        if (velocity.x != 0 or velocity.y != 0) and is_dashing:
                play_anim("dash")
        elif (velocity.x != 0 or velocity.y != 0) and !is_dashing:
                play_anim("move")
        else:
                play_anim("idle")

        if Input.is_action_just_pressed("dash"):
                start_dashing()

func _physics_process(delta):
        get_input()
        velocity = move_and_slide(velocity)

func flip():
        facing_right = !facing_right
        sprite.flip_h = !sprite.flip_h

func play_anim(anim_name):
        if anim_player.is_playing() and anim_player.current_animation == anim_name:
                return
        anim_player.play(anim_name)

func start_dashing():
        if !is_dashing:
                is_dashing = true
                dash_timer.start()

func _on_DashTimer_timeout():
        is_dashing = false

func take_damage():
        emit_signal("end_game", false)

func collect_activity():
        activities_collected = activities_collected + 1
        if activities_collected >= activities_to_collect:
                emit_signal("end_game", true)
```

5. Connect the Timer signal and change Wait Time to 0.2,

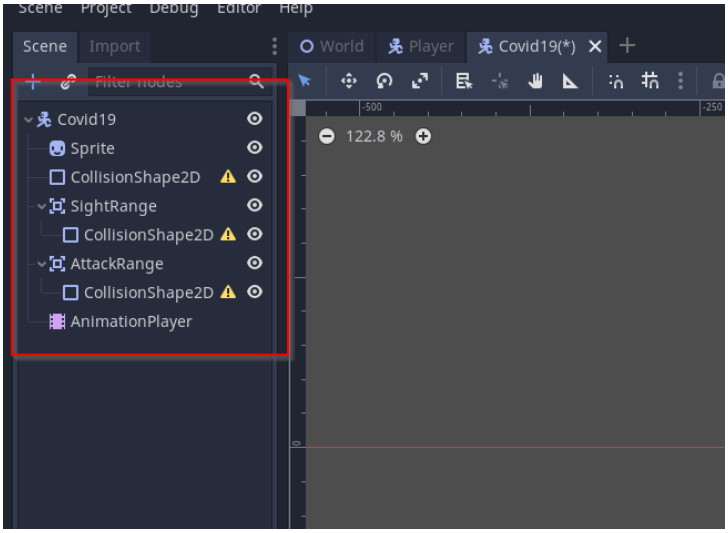
In needed, you can also download the project after Chapter 2 from here: Link (it does not contain all textures, so if you are missing any, use the link above)

Chapter 3

COVID-19 scene

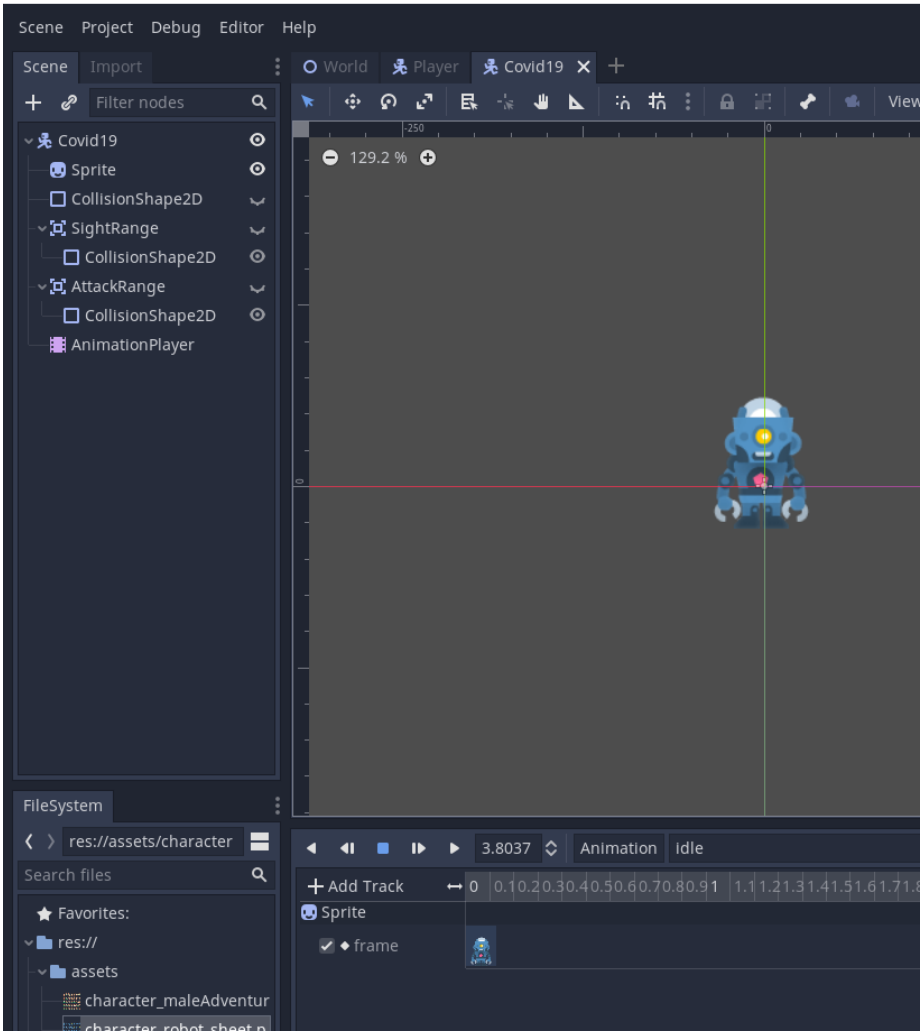
Steps to do:

1. Create a new scene with these Nodes:



2. Add the sprite,
3. Set the collision shapes,

4. Create the animations: attack, idle, move,



5. Add the script:

```
extends KinematicBody2D

export (int) var speed = 200

onready var anim_player = $AnimationPlayer
onready var sprite = $Sprite

var chasing_player = null
var facing_right = false
var velocity = Vector2()

func _physics_process(delta):
    velocity = Vector2.ZERO

    if chasing_player:
            velocity = position.direction_to(chasing_player.position) * speed
            play_anim("move")
    else:
            play_anim("idle")

    if facing_right and velocity.x > 0:
```

```
                flip()
        if !facing_right and velocity.x < 0:
                flip()

        velocity = move_and_slide(velocity)

    func flip():
        facing_right = !facing_right
        sprite.flip_h = !sprite.flip_h

    func _on_SightRange_body_entered(body):
        if body is Player:
                chasing_player = body

    func _on_SightRange_body_exited(body):
        if body == chasing_player:
                chasing_player = null

    func play_anim(anim_name):
        if anim_player.is_playing() and anim_player.current_animation == anim_name:
                return
        anim_player.play(anim_name)

    func _on_AttackRange_body_entered(body):
        if body.has_method("take_damage"):
                body.take_damage()
```

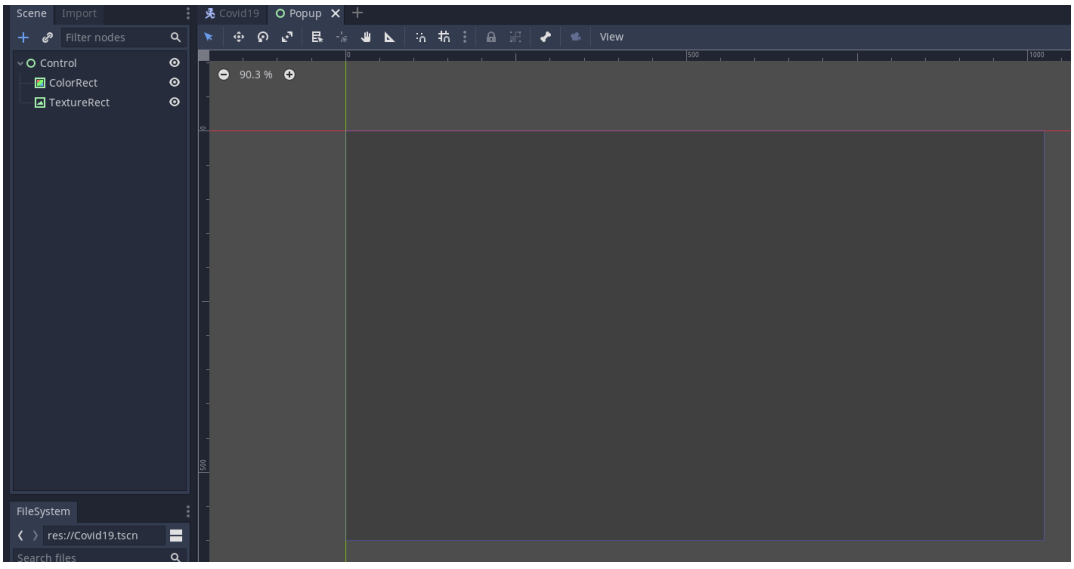
In needed, you can also download the project after Chapter 3 from here: Link (it does not contain all textures, so if you are missing any, use the link above)

Chapter 4

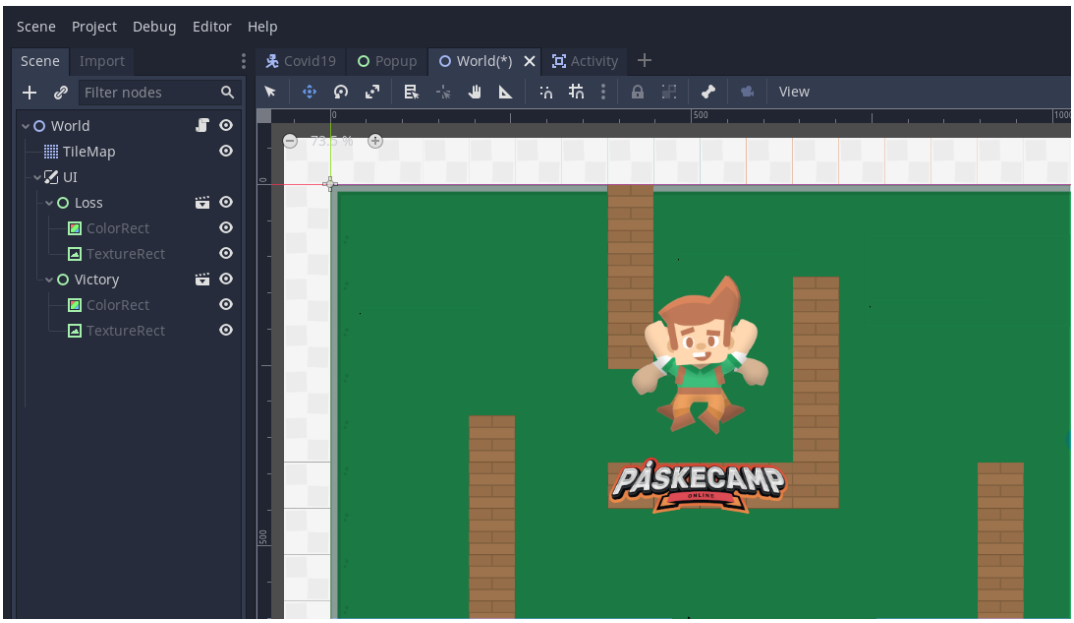
UI (User Interface) elements and final adjustments

Steps to do:

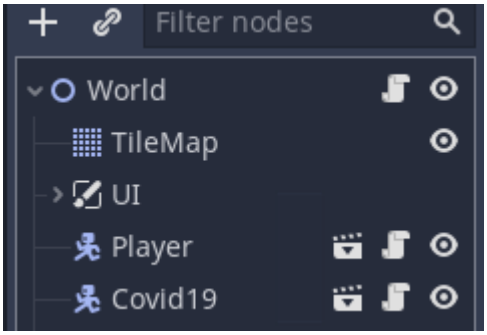
1. Add a new scene for the UI elements:



2. Add a new CanvasLayer to the World Scene and attach two times the Popup scene.
3. Add different Textures to these Popup scenes. Change also name of those to Loss and Victory:



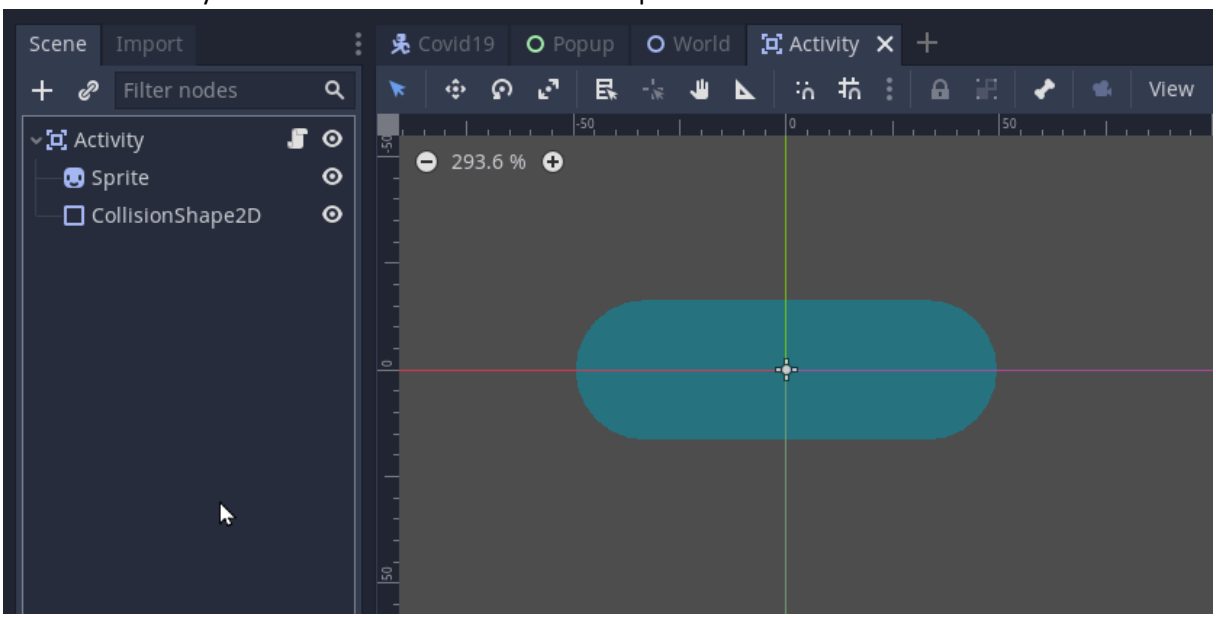
4. Add Player and Covid19 Scenes to the World scene:



5. Add this Script to the World Scene:

```
extends Node2D

func _on_Player_end_game(victory):
    if victory:
        $UI/Victory.visible = true
    else:
        $UI/Loss.visible = true

    get_tree().paused = true
```

6. Create an Activity Scene without Texture chosen on Sprite:

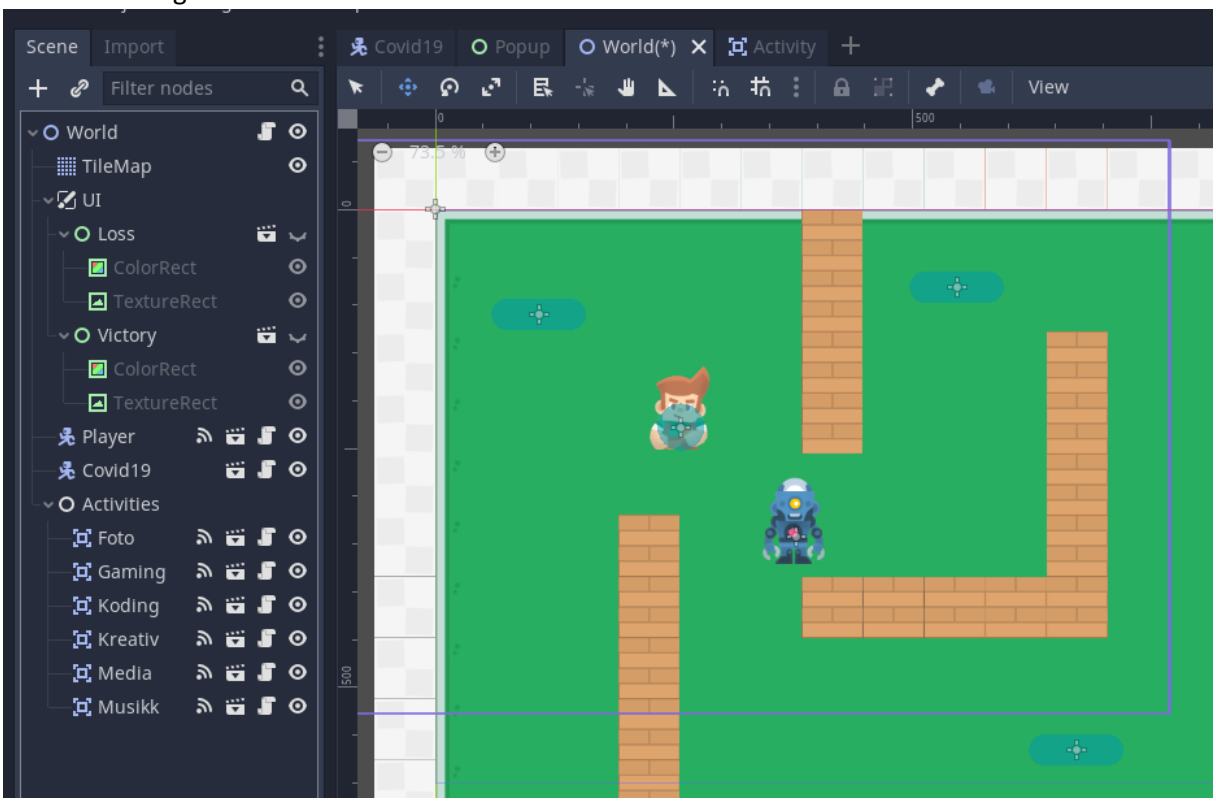


7. Add this script to the Activity Scene:

```
extends Area2D

export (Texture) var activity_texture = null

onready var sprite = $Sprite

func _ready():
    sprite.texture = activity_texture

func _on_Activity_body_entered(body):
    if body is Player:
            body.collect_activity()
            queue_free()
```

8. Place the activites on the World Scene.

9. Connect the signals:



In needed, you can also download the project after Chapter 4 from here: Link

Resources and assets

These assets were used in the game:

- https://kenney.nl/assets/topdown-shooter
- https://kenney.nl/assets/toon-characters-1
- Official logo and logos of activities from PÅSKECAMP ONLINE