

Dodge the Creeps!

El joc es diu ***Dodge the Creeps!***. El teu personatge ha de moure's i evitar els enemics tant de temps com sigui possible.

En aquesta fitxa veurem com:

- Crear un joc 2D complet amb l'editor **Godot**.
- Estructurar un projecte de joc senzill i pujar-lo a **GitHub**.
- Moure el personatge del jugador i canviar-ne l'sprite quan calgui.
- Spawnejar enemics de forma aleatòria.
- Comptar puntuacions.

Disposeu dels assets necessaris per començar al **Moodle**.

1. Per començar, crearem el nostre repositori a **GitHub**, clonarem el mateix al nostre ordinador i crearem un projecte de **Godot** a la nostra carpeta local.

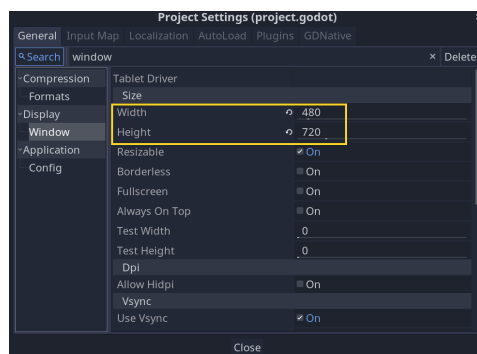
2. Descarreguem **assets.zip**. L'arxiu conté les imatges i els sons que utilitzarem per fer el joc.

Extreu l'arxiu i mou els directoris **art/** i **fonts/** al directori del projecte!

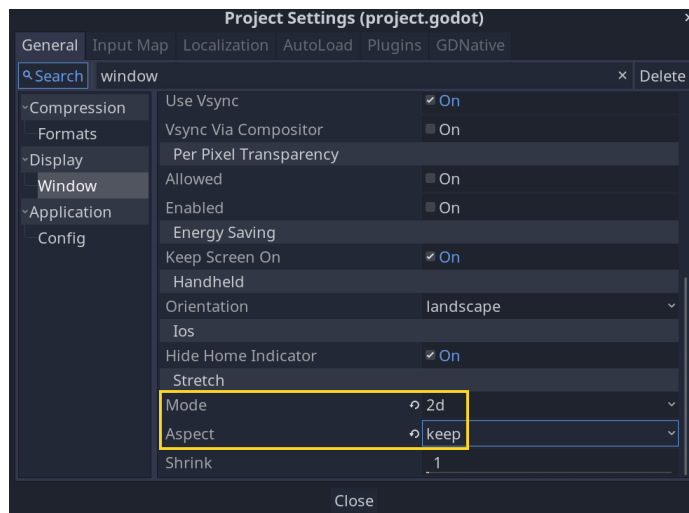
[COMMIT] Pujada inicial.

3. Aquest joc està dissenyat per al mode *portrait*, de manera que hem d'ajustar la mida de la finestra del joc.

Feu clic a **Projecte/Configuració del projecte** per a obrir la finestra de configuració del projecte i al menú, obriu la pestanya **Visualitza/Finestra**. Establiu **Width** a **480** i **Height** a **720**.



4. A més, desplaçe-vos cap avall fins a la part inferior de la secció i, sota les opcions **Estira**, establiu el **Mode** a **2d** i **Aspect** a **keep**. Això assegura que el joc s'escala coherentment en pantalles de diferents mides.

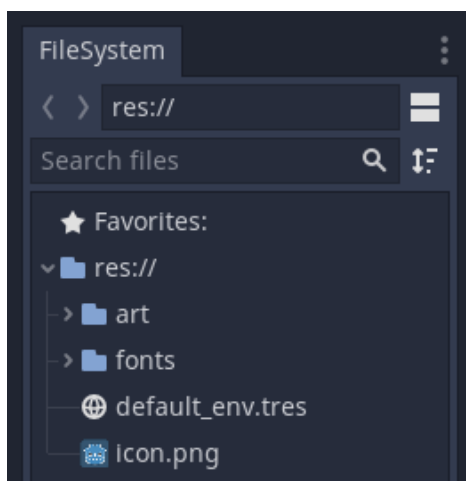


[COMMIT] Configuració inicial.

En aquest projecte, realitzarem 3 escenes independents: **Player**, **Mob** i **HUD**, que combinarem en l'escena principal del joc.

En un projecte més gran, pot ser útil crear carpetes per a contenir les diverses escenes i els seus scripts, però per a aquest joc relativament petit, podeu desar les vostres escenes i scripts a la carpeta arrel del projecte, identificada per **res://**.

Podeu veure les carpetes del vostre projecte al menú **FileSystem** a la cantonada inferior esquerra:

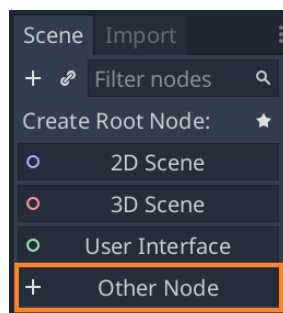


5. Amb la configuració del projecte, podem començar a treballar en el personatge controlat pel jugador.

La primera escena definirà l'objecte **Player**. Un dels avantatges de crear una escena de jugador separada és que podem provar-la per separat, fins i tot abans que hàgim creat altres parts del joc.

6. Per començar, hem de triar un node arrel per a l'objecte reproductor.

Com a regla general, el node arrel d'una escena ha de reflectir la funcionalitat desitjada de l'objecte (què és). Feu clic al botó **Altres nodes** i afegiu un node **Area2D** a l'escena.

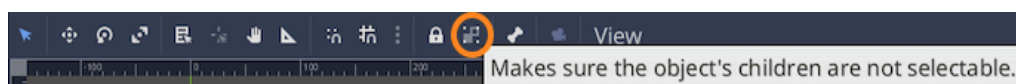


7. **Godot** mostrarà una icona d'avís al costat del node a l'arbre d'escena. Pots ignorar-ho de moment. Ho abordarem més tard.

8. Amb **Area2D** podem detectar objectes que se superposen o es topen amb el jugador. Canvieu el nom del node a **Player** fent-hi doble clic. Ara que hem establert el node arrel de l'escena, podem afegir nodes addicionals per donar-li més funcionalitat

9. Abans d'afegir qualsevol fill al node del **Player**, volem assegurar-nos que no es mouen accidentalment o es redimensionen fent clic sobre seu.

Seccioneu el node i feu clic a la icona a la dreta del bloqueig; el seu consell d'eina diu **Assegureu-vos que els fills de l'objecte no siguin seleccionables**.



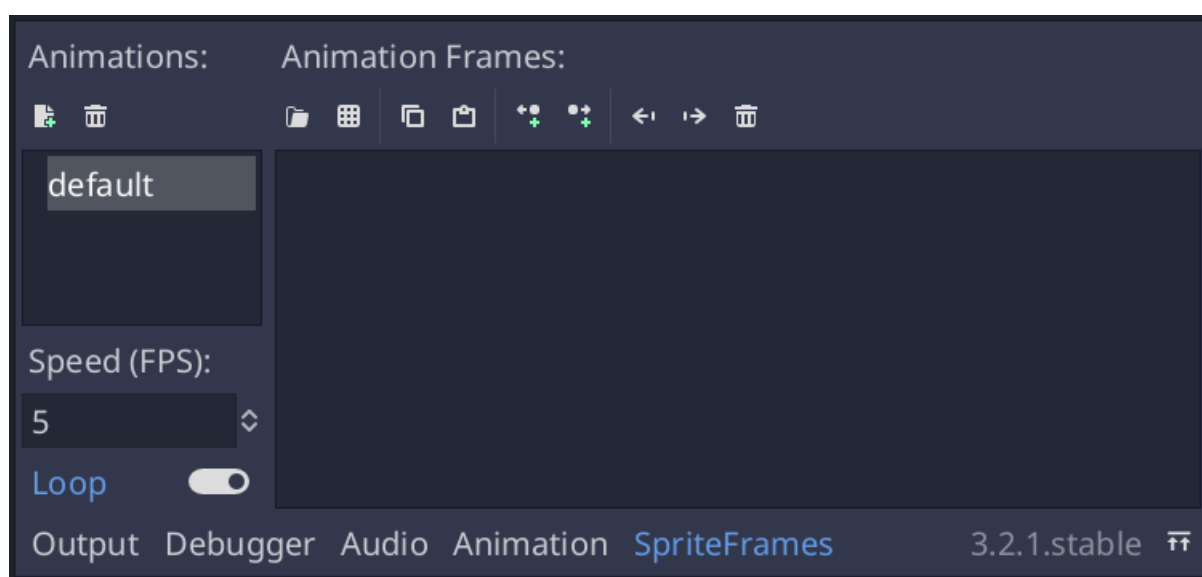
Guarda l'escena.

[COMMIT] Creació Node Jugador.

10. Feu clic al node **Player** i afegiu un node **AnimatedSprite** com a fill. **AnimatedSprite** gestionarà l'aparença i les animacions per al nostre jugador.

11. Tingueu en compte que hi ha un símbol d'avís al costat del node. Un **AnimatedSprite** requereix un recurs **SpriteFrames**, que és una llista de les animacions que pot mostrar.

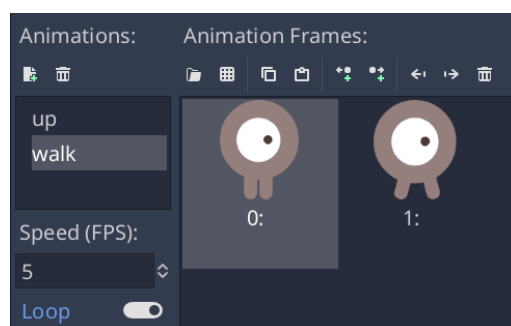
Per crear-ne una, trobeu la propietat **Frames** al manú i feu clic a **[empty]/New SpriteFrames**. Feu clic de nou per a obrir el plafó **SpriteFrames**:



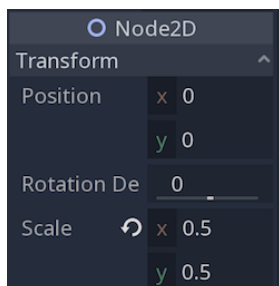
12. A l'esquerra hi ha una llista d'animacions. Feu clic al **per defecte** i canvieu el nom per **walk**. A continuació, feu clic al botó **Nova Animació** per a crear una segona animació anomenada **up**.

13. Cerqueu les imatges del reproductor a la pestanya **FitxerSistema** (es troben a la carpeta d'art que heu descomprimit abans).

14. Arrossegueu les dues imatges per a cada animació, anomenades **playerGrey_up[1/2]** i **playerGrey_walk[1/2]**, al costat dels **marcs d'animació (Animation Frames)** del menú per a l'animació corresponent:



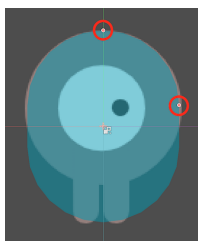
15. Les imatges del jugador són una mica massa grans per a la finestra de joc, així que hem de reduir-les. Feu clic al node **AnimatedSprite** i establiu la propietat **Scale** a **(0,5, 0,5)**. El podeu trobar al menú sota **Node2D**.



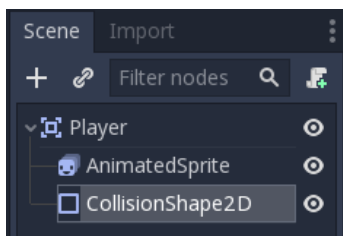
16. Finalment, afegeix un **CollisionShape2D** com a fill de **Player**. Això determinarà la **hitbox** del jugador, o els límits de la seva àrea de col·lisió.

17. Per a aquest personatge, un node **CapsuleShape2D** ens proporciona el millor ajust, de manera que al costat de **Shape** al menú, feu clic a **[empty]/New CapsuleShape2D**.

18. Utilitzant les dues boletes de mida, redimensiona la forma per cobrir l'sprite:



19. Quan hàgiu acabat, l'escena del jugador s'hauria de veure així:



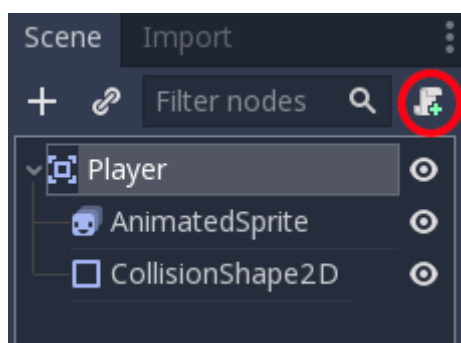
Guarda l'escena de nou.

[COMMIT] Animació Sprite Jugador.

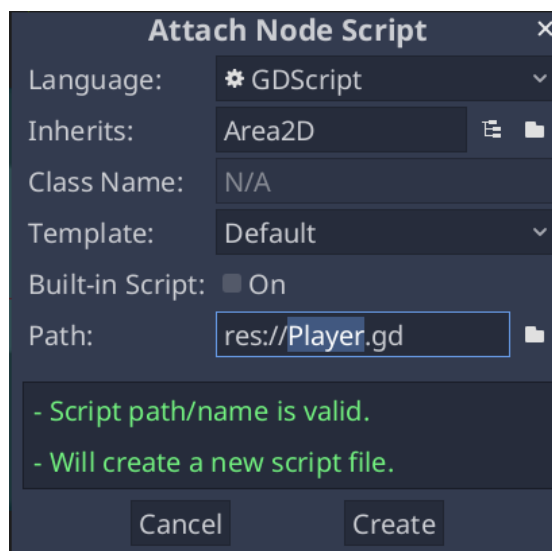
20. Ara afegirem el moviment del jugador, l'animació i la configurarem per detectar col·lisions.

Per fer-ho, hem d'afegir algunes funcionalitats que no podem obtenir d'un node integrat, de manera que afegirem un script.

21. Feu clic al node del jugador i feu clic al botó **Adjunta un script**:



22. A la finestra de configuració de l'script, podeu deixar la configuració predeterminada. Només has de fer clic a **Crea**:



23. Comenceu declarant les variables de membre que aquest objecte necessitarà:

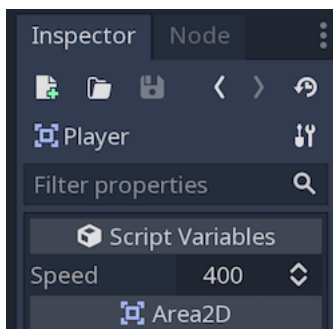
```
extends Area2D
```

```
export var speed = 400 # A quina velocitat es mourà el jugador (píxels/seg).
```

```
var screen_size # Mida de la finestra de joc.
```

24. L'ús de la paraula clau **export** a la primera variable **speed** ens permet establir el seu valor al menú. Això pot ser útil per als valors que voleu poder ajustar com les propietats integrades d'un node.

25. Feu clic al node del jugador i veureu que la propietat ara apareix a la secció **Variables de script** del menú. Recordeu, si canvieu el valor aquí, sobreescriurà el valor escrit a l'script.



Guarda l'escena de nou.

[COMMIT] Creació Script Jugador.

26. La funció **_ready()** es crida quan un node entra a l'arbre d'escena, que és un bon moment per a trobar la mida de la finestra de joc:

```
func _ready():  
    screen_size = get_viewport_rect().size
```

27. Ara podem utilitzar la funció **.process()** per a definir què farà el jugador. **.process()** es crida cada fotograma, de manera que l'utilitzarem per actualitzar elements del nostre joc, que esperem que canviï sovint. Per al jugador, hem de fer el següent:

- Comprovar si hi ha entrada.
- Moure'l en la direcció indicada.
- Reproduir l'animació adequada.

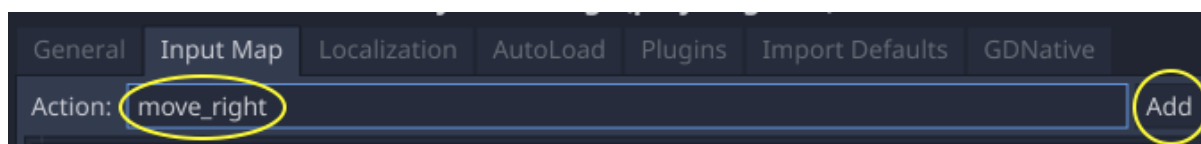
28. En primer lloc, hem de comprovar si hi ha entrada (el jugador està prement una tecla?). Per a aquest joc, tenim 4 entrades de direcció per comprovar.

Les accions d'entrada es defineixen a la configuració del projecte a **Mapa d'entrada**.

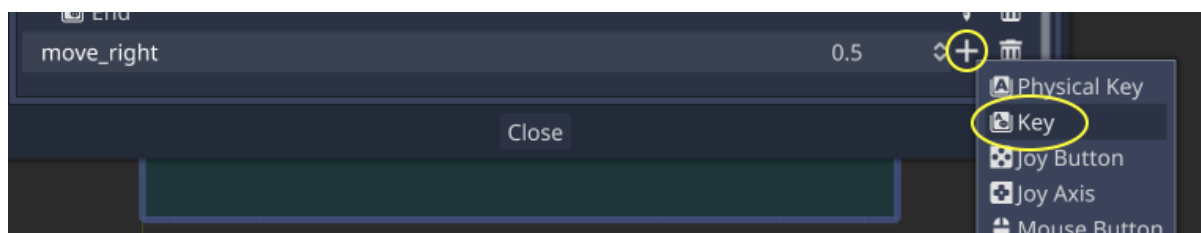
Aquí, podeu definir esdeveniments personalitzats i assignar-los diferents claus, esdeveniments del ratolí o altres entrades. Per a aquest joc, mapejarem les tecles de fletxa a les quatre direccions.

29. Feu clic a **Projecte/Configuració del projecte** per a obrir la finestra de configuració del projecte i feu clic a la pestanya **Mapa d'entrada** a la part superior.

Escriviu **move_right** a la barra superior i feu clic al botó **Afegeix** per afegir l'acció **move_right**.



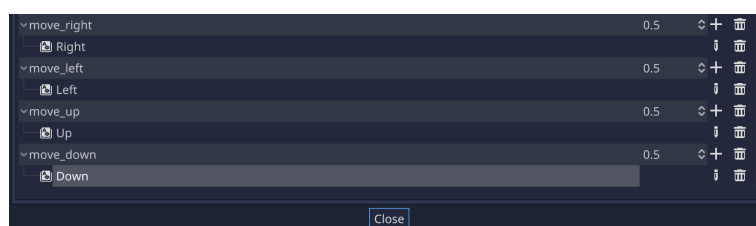
30. Hem d'assignar una clau a aquesta acció. Fes clic a la icona **+** de la dreta i, després, a l'opció **Clau** del menú desplegable. Un diàleg us demana que escriviu la clau desitjada. Premeu la fletxa dreta del teclat i feu clic a **D'acord**.



31. Repetiu aquests passos per afegir tres mapatges més:

- **move_left** assignat a la tecla de fletxa esquerra.
- **move_up** assignat a la tecla de fletxa amunt.
- **move_down** mapejat a la tecla de fletxa cap avall.

La pestanya del mapa d'entrada hauria de tenir aquest aspecte:



[COMMIT] Assignació de tecles.

32. Podeu detectar si es prem una tecla utilitzant *Input.is.action.pressed()*, la qual retorna cert si es prem o fals si no.

```
func _process(delta):  
    var velocity = Vector2.ZERO # Vector de moviment del jugador.  
    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 velocity.length() > 0:  
        velocity = velocity.normalized() * speed  
        $AnimatedSprite.play()  
    else:  
        $AnimatedSprite.stop()
```

33. Comencem establint la velocitat a *(0, 0)* per defecte, el jugador no s'ha de moure. A continuació, comprovem cada entrada i afegim/treiem de la *velocity* per obtenir una direcció total.

Per exemple, si es manté a la *right* i *down* al mateix temps, el vector de *velocity* resultant serà *(1, 1)*. En aquest cas, com que estem afegint un moviment horitzontal i vertical, el jugador es mouria més ràpid en diagonal que si simplement es mogués horitzontalment.

Podem evitar-ho si normalitzem la velocitat, el que significa que establim la seva *length* a *1*, i després multiplicar per la velocitat desitjada. Això significa que no hi ha més moviment diagonal ràpid.

També comprovem si el jugador s'està movent de manera que puguem cridar *play()* o *stop()* a *AnimatedSprite*.

34. Ara que tenim una direcció de moviment, podem actualitzar la posició del jugador. També podem utilitzar **clamp()** per evitar que surti de la pantalla. Fer *clamping* d'un valor significa restringir-lo a un interval determinat.

Afegeix el següent a la part inferior de la funció **_process** (assegureu-vos que no està sagnat sota la resta):

```
position += velocity * delta  
position.x = clamp(position.x, 0, screen_size.x)  
position.y = clamp(position.y, 0, screen_size.y)
```

35. Feu clic a **Reprodueix escena** i confirmeu que podeu moure el personatge per la pantalla en totes les direccions.

[COMMIT] Moviment jugador.