

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.

Continuació de la fitxa anterior!

77. La part final que necessita el nostre joc és una interfície d'usuari per mostrar coses com la puntuació, un missatge de *Game Over* i un botó de reinici.

Creeu una escena nova i afegiu un node ***CanvasLayer*** anomenat ***HUD***.

HUD significa ***heads-up display***, una pantalla informativa que apareix com una superposició a la part superior de la vista del joc.

El node ***CanvasLayer*** ens permet dibuixar els nostres elements d'interfície d'usuari en una capa per sobre de la resta del joc, de manera que la informació que mostra no està coberta per cap element de joc com el jugador o els enemics.

78. L'***HUD*** ha de mostrar la informació següent:

- Puntuació, canviada per ***ScoreTimer***.
- Un missatge, com ara ***Game Over*** o ***Get Ready!***
- Un botó d'***Start*** per a començar el joc.

El node bàsic per als elements d'interfície d'usuari és ***Control***. Per crear la nostra interfície d'usuari, utilitzarem dos tipus de nodes de control: ***Label*** i ***Button***.

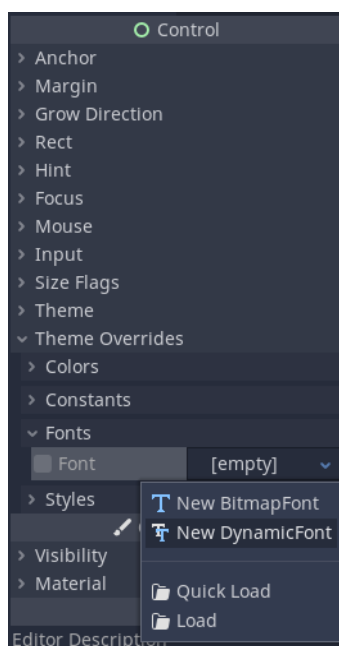
Crea els següents com a fills del node ***HUD***:

- ***Label*** anomenada ***ScoreLabel***.
- ***Label*** anomenada ***Missatge***.
- ***Button*** anomenat ***StartButton***.
- ***Timer*** anomenat ***MessageTimer***.

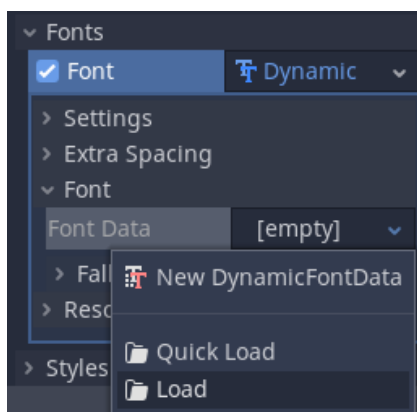
79. Feu clic a **ScoreLabel** i escriviu un número al camp **Text** del menú. El tipus de lletra per defecte per als nodes **Control** és petit i no escala bé.

Hi ha un fitxer de tipus de font inclòs en els assets del joc anomenat **Xolonium-Regular.ttf**. Per a utilitzar aquest tipus de lletra, feu el següent:

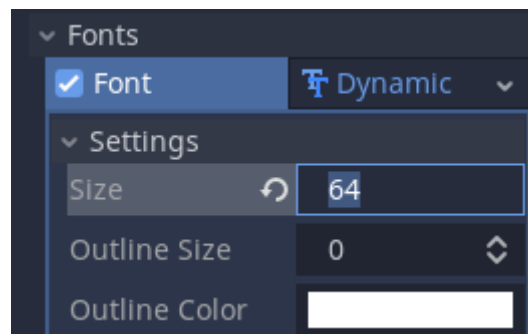
79.1. A **Theme overrides/Fonts** feu clic en el quadre buit i seleccioneu **New DynamicFont**.



79.2. Feu clic al **DynamicFont** afegit, i a **Font/FontData**, trieu **Load** i seleccioneu el fitxer **Xolonium-Regular.ttf**.

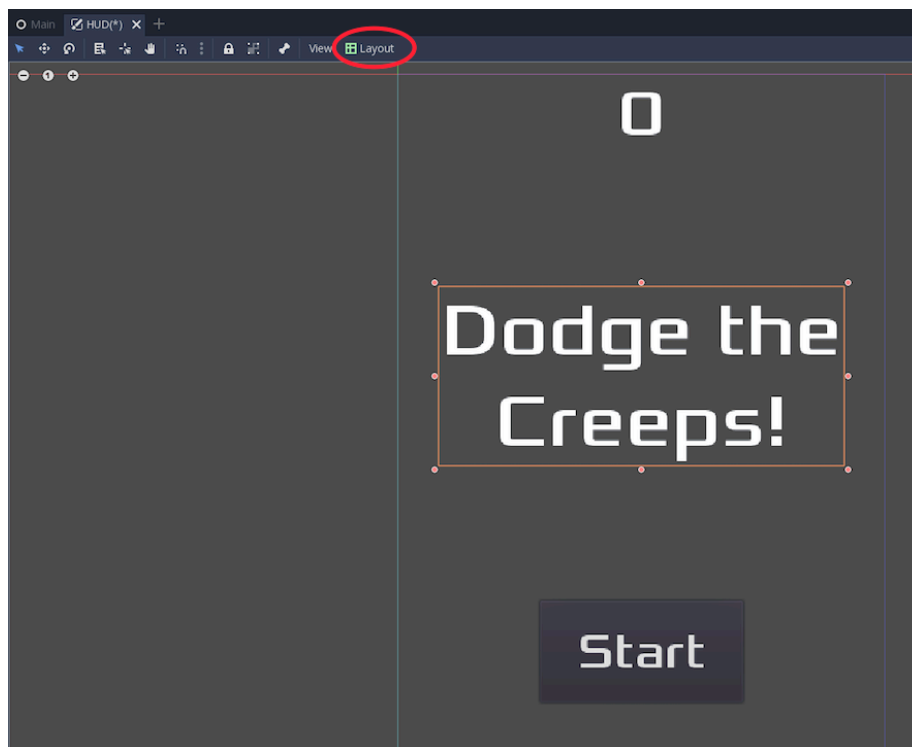


79.3. Estableix la propietat **Size** a **Settings**, **64**.



80. Un cop configurada l'**ScoreLabel**, podeu fer clic a la fletxa cap avall al costat de la propietat **Font** i triar **Copy**, després **Paste** al mateix lloc en els altres dos nodes de control.

81. Organitza els nodes com es mostra a continuació. Feu clic al botó **Layout** per a definir la disposició d'un node de control:



82. Podeu arrossegat els nodes per a col·locar-los manualment. Per a una col·locació més precisa, utilitzeu els paràmetres següents:

ScoreLabel

- *Layout* : "Top Wide"
- *Text* : 0
- *Align* : "Center"

Message

- *Layout* : "HCenter Wide"
- *Text* : Dodge the Creeps!
- *Align* : "Center"
- *Autowrap* : "On"

StartButton

- *Text* : Start
- *Layout* : "Center Bottom"
- *Margin* :
 - *Top*: -200
 - *Bottom*: -100

83. Al **MessageTimer**, establiu el **Wait Time** a 2 i establiu la propietat **One Shot** a **On**.

84. Ara afegiu aquest script a **HUD**:

extends CanvasLayer

signal start_game

El senyal **start_game** indica al node **Main** que s'ha premut el botó.

86. Aquesta funció es crida quan volem mostrar un missatge temporalment, com ara **Estigues a punt**.

```
func show_message(text):
    $Message.text = text
    $Message.show()
    $MessageTimer.start()
```

87. Aquesta funció es crida quan el jugador perd. Es mostrarà **Game Over** durant 2 segons, després es tornarà a la pantalla del títol i, després d'una breu pausa, es mostrarà el botó **Start**.

```
func show_game_over():
    show_message("Game Over")
    # Espereu fins que el MessageTimer hagi acabat de comptar.
    yield($MessageTimer, "timeout")

    $Message.text = "Dodge the\nCreeps!"
    $Message.show()
    # Feu un temporitzador one-shot i espereu que acabi.
    yield(get_tree().create_timer(1), "timeout")
    $StartButton.show()
```

87. Aquesta funció és cridada per **Main** sempre que la puntuació canviï.

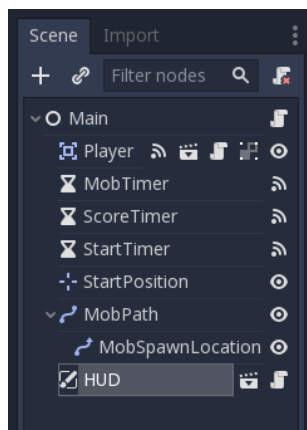
```
func update_score(score):
    $ScoreLabel.text = str(score)
```

87. Connecteu el senyal **timeout()** del **MessageTimer** i el senyal **pressed()** de **StartButton** i afegiu el codi següent a les noves funcions:

```
func _on_StartButton_pressed():
    $StartButton.hide()
    emit_signal("start_game")

func _on_MessageTimer_timeout():
    $Message.hide()
```

88. Ara que hem acabat de crear l'escena **HUD**, torneu a **Main**. Instància l'escena **HUD** a **Main** com vas fer l'escena **Player**. L'arbre de l'escena hauria de tenir aquest aspecte, així que assegura't que no falta res:



89. Ara necessitem connectar la funcionalitat **HUD** al nostre script **Main**. Això requereix algunes addicions a l'escena **Main**:

A la pestanya **Node**, connecteu el senyal **start_game** de l'**HUD** a la funció **new_game()** del node **Main** escrivint **new_game** al **Receiver Method** a la finestra **Connect a Signal**. Verifiqueu que la icona de connexió verda ara apareix al costat de **func new_game()** a l'script.

90. A **new_game()**, actualitzeu la visualització de la puntuació i mostreu el missatge **Get Ready**:

```
$HUD.update_score(score)
$HUD.show_message("Get Ready")
```

91. A **game_over()** necessitem cridar la funció **HUD** corresponent:

```
$HUD.show_game_over()
```

92. Finalment, afegiu això a **_on_ScoreTimer_timeout()** per a mantenir la visualització sincronitzada amb la puntuació canviant:

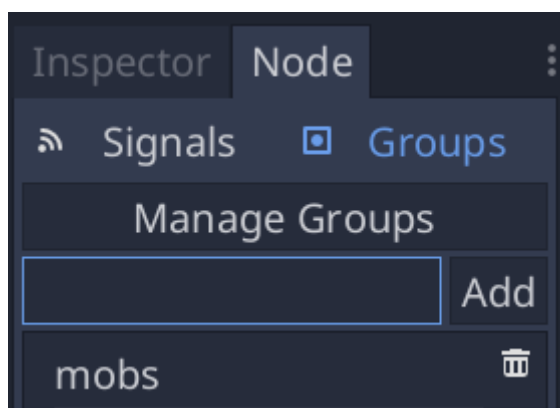
```
$HUD.update_score(score)
```

[COMMIT] Joc jugable V1

93. Si jugues fins al **Game Over** i llavors comença una partida nova de seguida, els enemics de la partida anterior encara poden estar a la pantalla.

Seria millor que tots desapareguessin a l'inici d'un nou joc. Només necessitem una manera de dir a totes les multituds que s'eliminin. Podem fer-ho amb la característica **group**.

A l'escena **Mob**, seleccioneu el node arrel i feu clic a la pestanya **Node** al costat de l'Inspector (el mateix lloc on trobareu els senyals del node). Al costat de **Signals**, feu clic a **Groups** i podreu escriure un nom de grup nou i fer clic a **Add**.



94. Ara totes les mobs estaran en el grup **mobs**. A continuació, podem afegir la següent línia a la funció **new_game()** a **Main**:

```
get_tree().call_group("mobs", "queue_free")
```

La funció **call_group()** crida la funció anomenada en cada node d'un grup, i en aquest cas, li estem dient a cada **Mob** que s'elimini a si mateixa.

[COMMIT] Joc jugable V2

El joc està pràcticament sencer en aquest punt. A la següent i última fitxa, el polirem una mica afegint un fons, música en bucle i algunes dreceres de teclat.