# Guia Space Shooter – Parte 2

## Criando a Nave do Jogador (Jogador)

### Objetivo desta parte

• Criar a cena Jogador (jogador.tscn).  
• Adicionar os nós necessários (Sprite2D, CollisionShape2D).  
• Colocar o script completo do Jogador.  
• Entender como o jogador se move, atira e respeita os limites da tela.

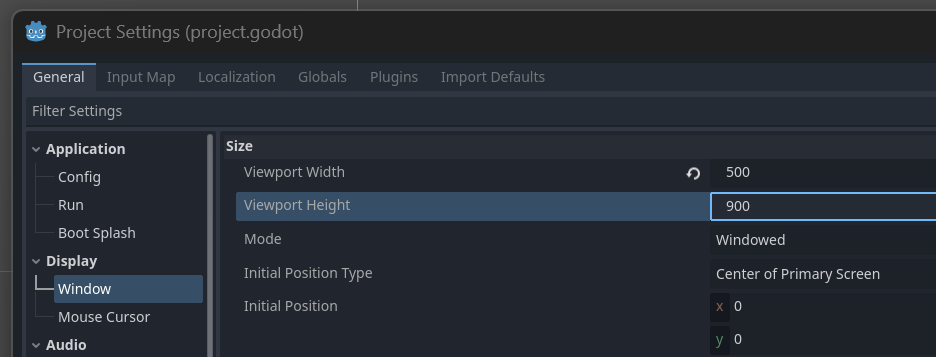
### 1. Criar a Cena Jogador

1. Clique em Scene > New Scene.  
2. Escolha o tipo Area2D (pois o jogador precisa detectar colisões).  
3. Renomeie para Jogador.  
4. Salve a cena como jogador.tscn.

👉 Dentro da cena Jogador, adicione:  
- Sprite2D → será a imagem da nave.  
- CollisionShape2D → é a área invisível que detecta batidas com inimigos e asteroides.

### 2. Configure o tamanho da Janela *(Viewport/Window)*

Configure o tamanho da tela para 500 pixels de largura e 900 pixels de altura.



### Grupos

👉 Coloque o Jogador no grupo Player (Node > Groups > Add > escreva Player > OK).

### 2. Adicionar o Script do Jogador

### 1. Selecione o nó Jogador. 2. Clique em Attach Script. 3. Salve como jogador.gd na pasta Scripts. 4. Cole exatamente este Código

### Script do Jogador

extends Area2D  
@export var velocidade := 300  
@onready var cena\_tiro = preload("res://Cenas/tiro.tscn")  
  
const SCREEN\_WIDTH = 474  
const SCREEN\_HEIGHT = 873  
  
func \_process(delta):  
 var mov := Vector2.ZERO  
 if Input.is\_action\_pressed("ui\_right"): mov.x += 1  
 if Input.is\_action\_pressed("ui\_left"): mov.x -= 1  
 if Input.is\_action\_pressed("ui\_up"): mov.y -= 1  
 if Input.is\_action\_pressed("ui\_down"): mov.y += 1  
 position += mov.normalized() \* velocidade \* delta  
   
 if Input.is\_action\_just\_pressed("ui\_accept"):  
 atirar()  
   
 position.x = clamp(position.x, 26, SCREEN\_WIDTH)  
 position.y = clamp(position.y, 27, SCREEN\_HEIGHT)  
  
func atirar():  
 var t = cena\_tiro.instantiate()  
 t.position = position  
 get\_parent().add\_child(t)

### 3. O que esse script faz

• velocidade → controla a rapidez com que a nave se move.  
• cena\_tiro → carrega a cena do tiro que o jogador dispara.  
• SCREEN\_WIDTH e SCREEN\_HEIGHT → são os limites da tela (largura e altura).  
• \_process(delta) → executa a cada frame do jogo:  
 - Verifica quais teclas estão pressionadas (setas ou WASD).  
 - Move a nave para o lado correspondente.  
 - Se a tecla de tiro (ui\_accept, normalmente a barra de espaço) for pressionada, chama atirar().  
 - Usa clamp() para impedir que a nave saia da tela.  
• atirar() → cria um novo tiro na posição da nave e adiciona ele na cena.

👉 Agora a nave já pode se mover e poderá disparar projéteis, mas somente após a criação da cena tiro.tscn.