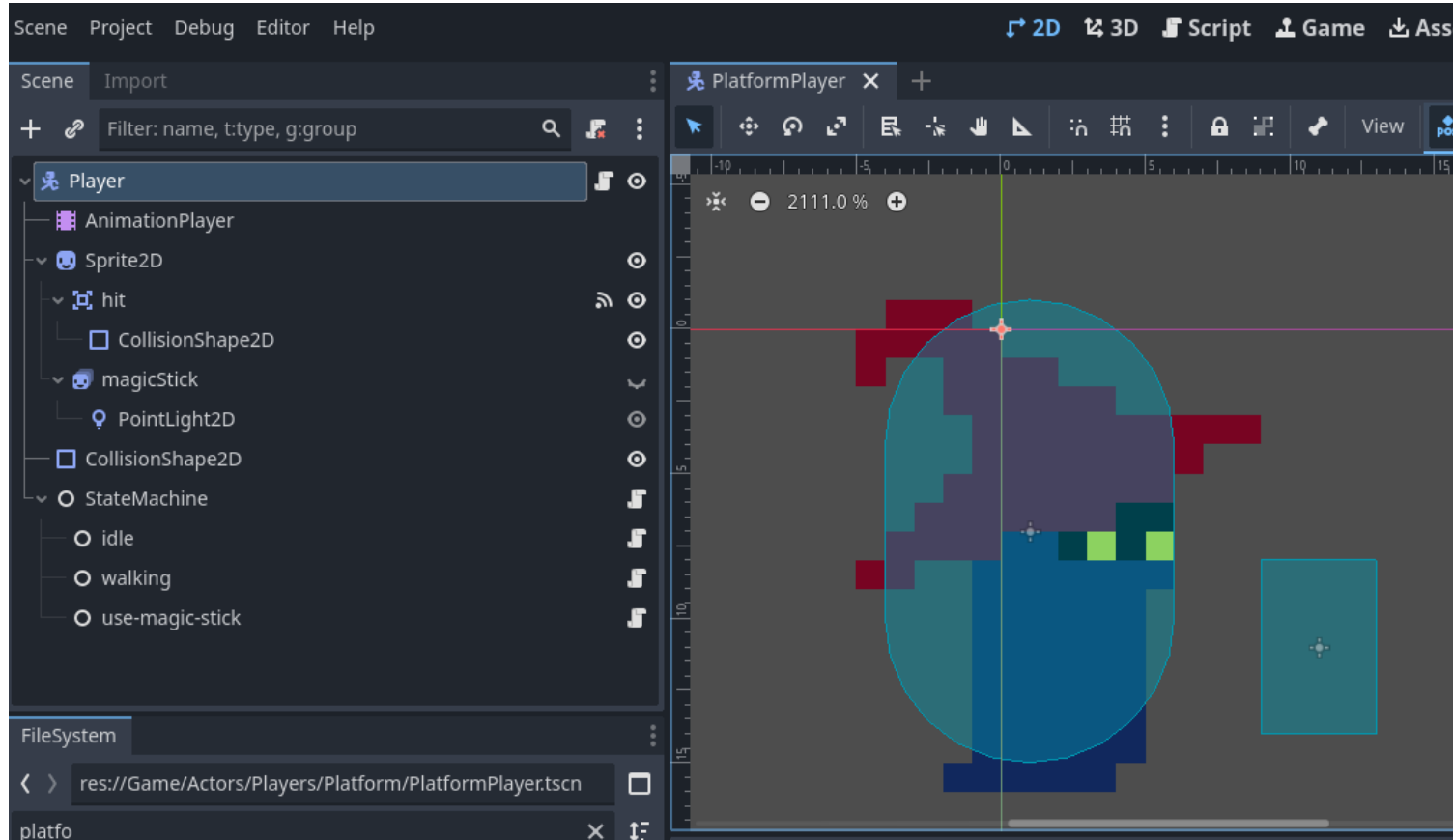


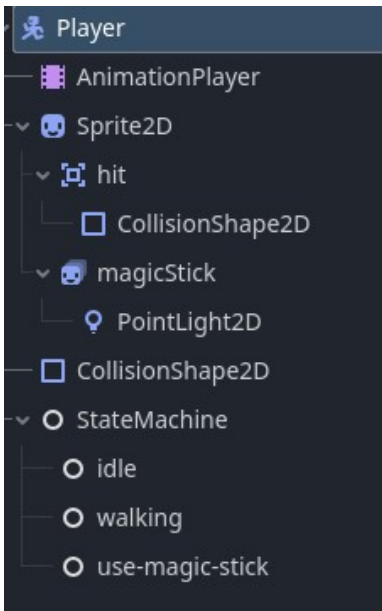
Interface



On voit dans menu de gauche
un arbre d'objets

ON peut remarquer avec l'icone
À quoi ils correspondent
Mais on va regarder ça en détail

Dans les pages suivantes



Player

Le type de noeud ici est `CharacterBody2D`: c'est un type d'objet de Godot qui gère déjà plein de chose pour les personnages en général

Notamment la collision, on y revient juste après

AnimationPlayer

Comme son nom l'indique, cet objet va permettre de gérer les différentes animations de la scène, que ce soit de jouer un défilement d'image ou un déplacement d'un objet

Sprite2D

C'est l'objet qui gère les images dans Godot, on a plein de propriétés qui vont nous permettre ensuite de gérer les animations, texture..

hit/collisionShape2D

Dans Godot on peut avoir des objets zones, et y attacher une forme pour définir son périmètre d'action, et ainsi au moment de détecter qu'un autre objet entre dans ce périmètre d'exécuter du code

CollisionShape2d

Je disais sur Player qu'on avait la gestion des collisions intégrées à l'objet racine, ici "`CharacterBody2d`", nommé "Player"

Cet objet représente la forme qui représentera la zone de collision pour notre héros

StateMachine

On aura un chapitre dédiée dessus, ce qu'on appelle en français une "machine à état", permet de gérer pas mal de choses en fonction de cet "état/status" du personnage à un instant T

On peut lire ici: `idle`: il fait rien, `walking`: il marche, `use-magic-stick`: il utilise sa baguette magique et en fonction du status à cet instant on va définir ce qu'il a le droit de faire ou non à ces moments là

Mais j'y reviendrais en détail plus tard

AnimationPlayer

En cliquant sur cet objet, on peut voir en bas cette fenêtre, on va voir comment elle fonctionne en détail

