

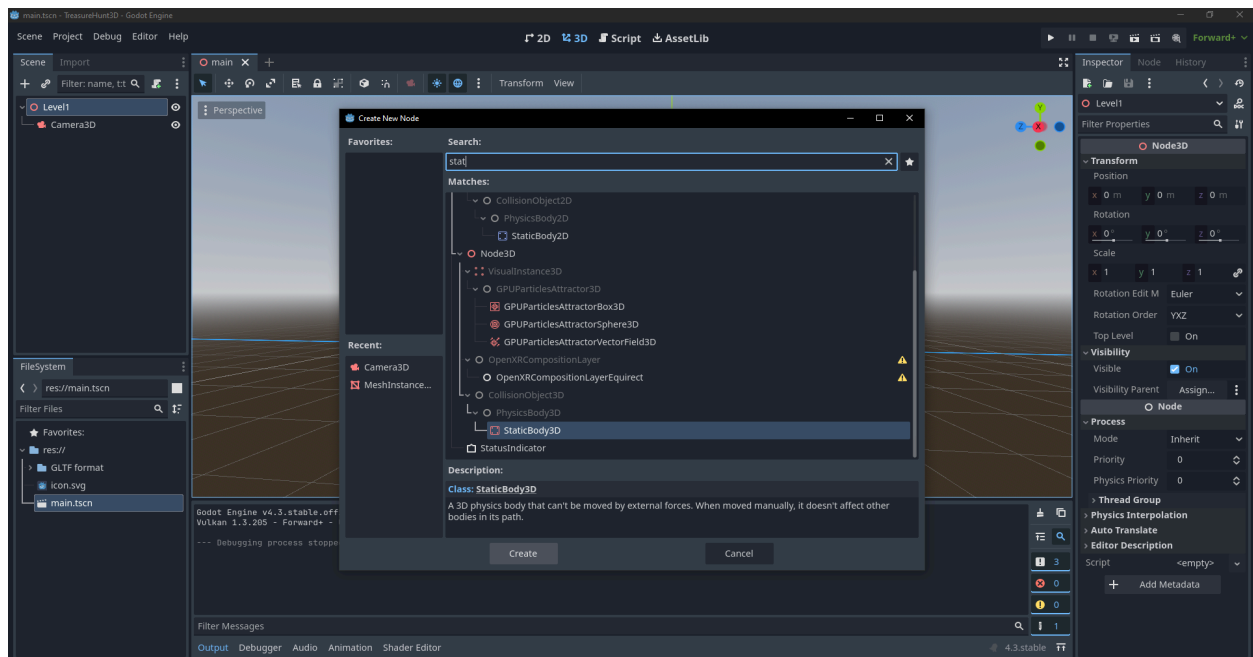
Програмирање на видео игри - домашна работа во Godot

Вовед

Според барањата на домашната задача, направив едноставна видео игра која се вика TreasureHunt, има играч кој шета низ паркот и собира парички и ковчези и добива поени. Ја работев со помош на Godot 4.0, додека текстурите ги преземав од Kenney. Играта може да се проба на itch.io. Видео од играта може да се погледне [тука](#). Кодот е достапен [тука](#).

Карактеристики на играта

- **Движење на играчот**
 - Со помош на стрелките на тастатурата играчот оди напред, назад, лево и десно
 - Со помош на спресе играчот скока нагора
 - Со поминување низ паричките, играчот ги собира.
 - Со помош на копчињата WASD играчот ја движи камерата во различни агли.



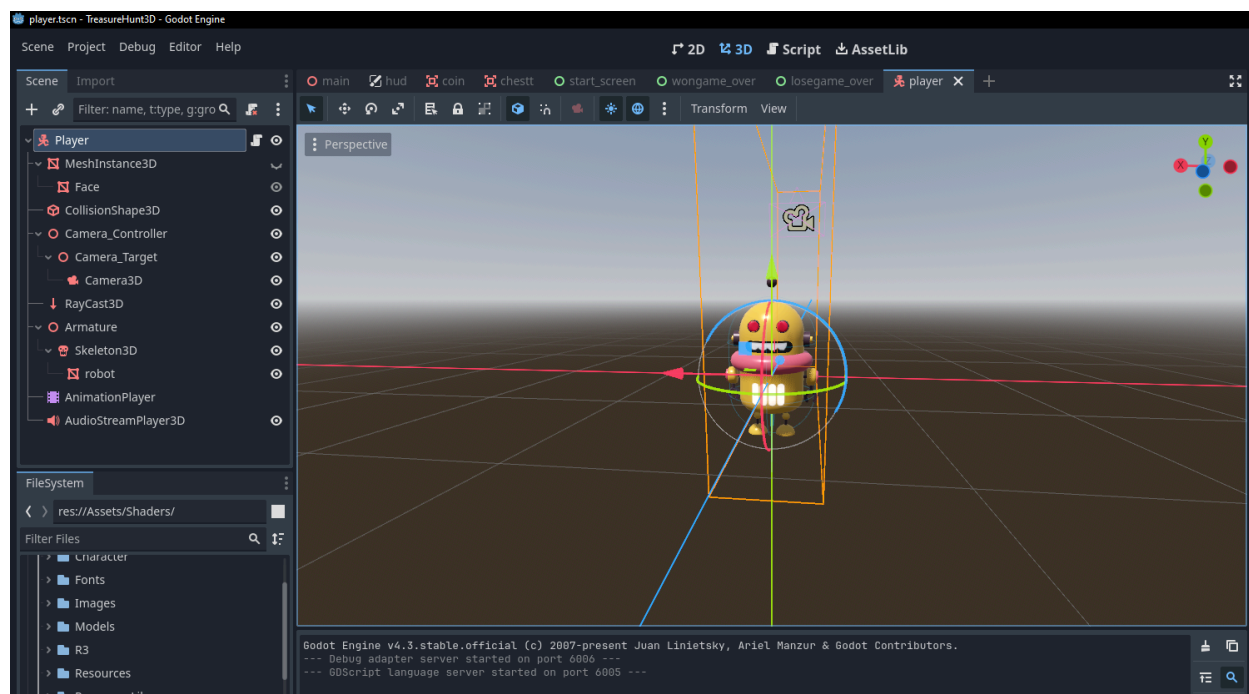
- **Играчот победува ако**
 - собере 30 парички
 - или собере 10 ковчези
- **Играчот губи ако**
 - истече времето пред да собере доволен број богатства
 - падне во водата

Својства

Графички

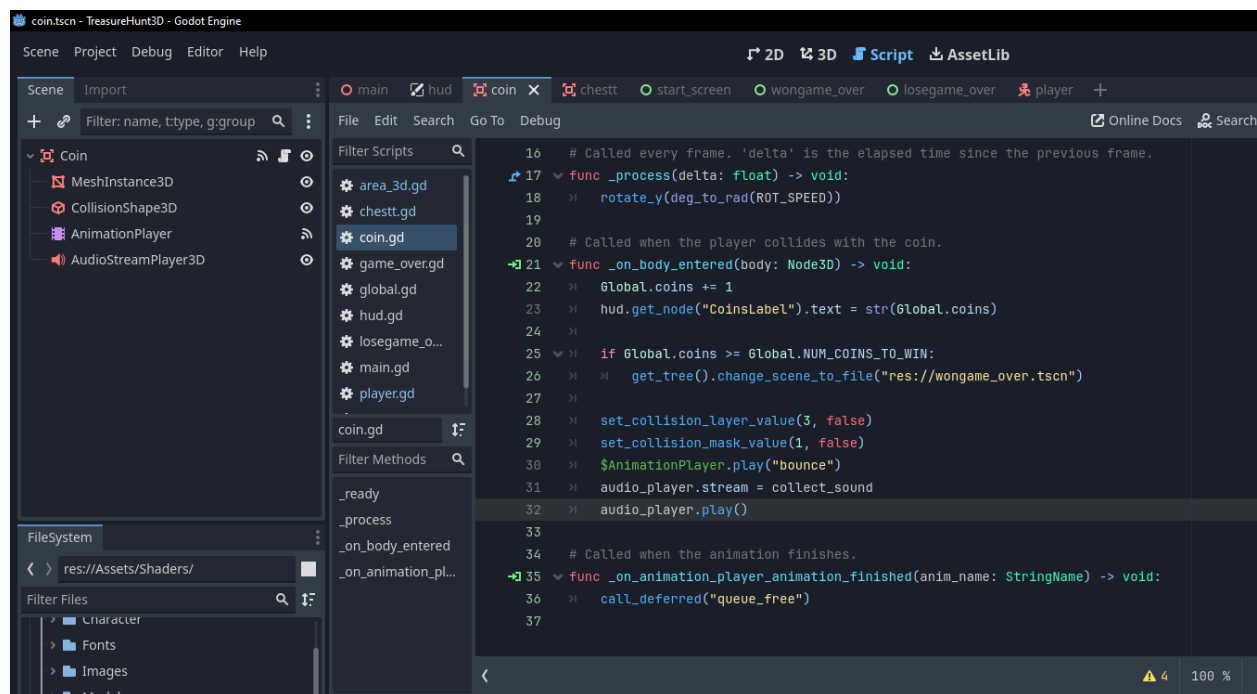
- Играчот

Го преземав од [следниот линк](#) и му додадов CharacterBody3D за да може д се движи и да има интеракција со околината. Му додадов и Camera_Controller кој ја контролира камерата што е прикачена за играчот. Во неговата скрипта има и функција која детектира паѓање на играчот од мапата, со што играчот автоматски ја губи играта.



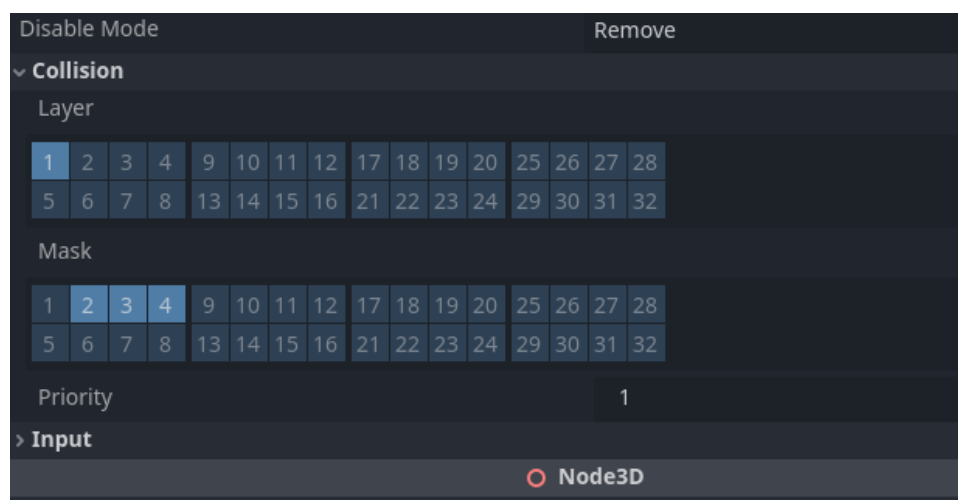
- Богатства

Направив 2 типа на богатства, парички coins, чијшто assets го преземав од [следниов линк](#) и ковчежи chests од [тука](#). Додадов Area3D кои се од помош за да откријат кога играчот комуницира со некој предмет. Испратениот сигнал се емитува при контакт, а собраното богатство се отстранува од сцената, овозможувајќи механика за собирање предмети. Исто така додадов и HUD сцена, која го прикажува бројот на собрани предмети. Ова го направив со помош на глобални променливи кои се ресетираат ако играчот изгуби..

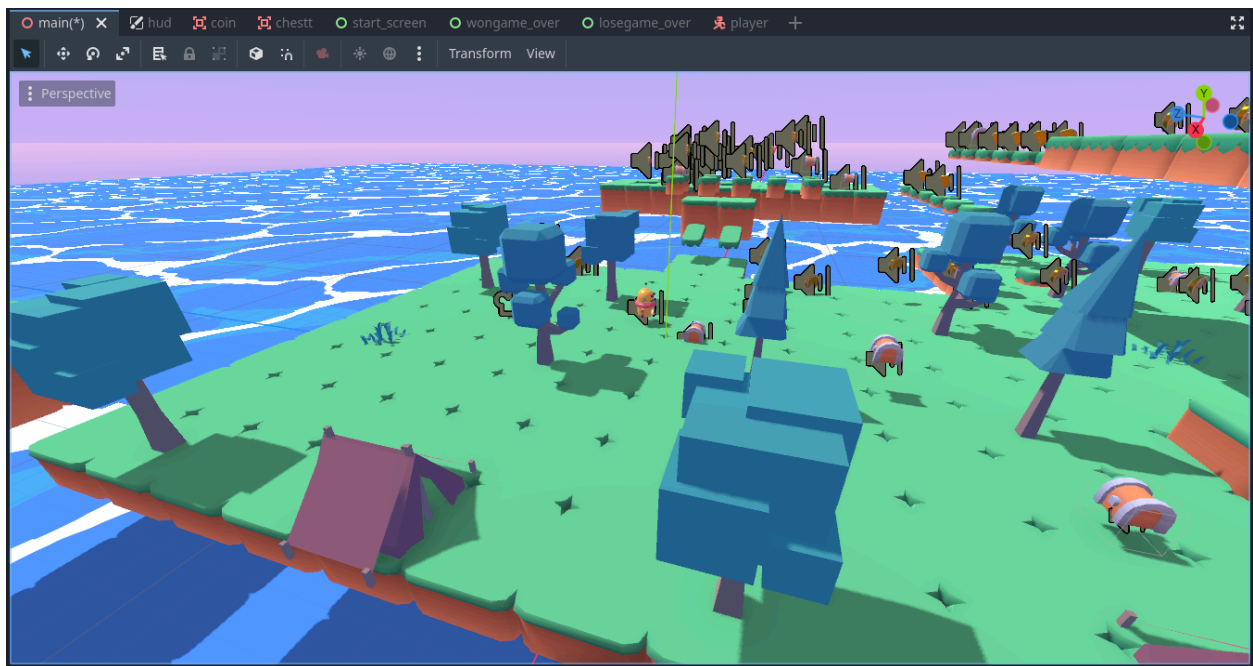


- **Околина**

Изградил едноставна мапа за движење на играчот со помош на блокови кои ги преземав од истото место како и ковчезите. Дополнително додадов уште една гريد мапа за дрвата за да изгледа на парк. Овие предмети ги преземав од [тука](#). Дополнително, имплементирав систем за детекција на пад, активирајќи звучен ефект и преминувајќи на екран дека играчот изгубил. Додадов и посебно засенчување за водата да изгледа брановито. Ги преземав од [овде](#) и ги додадов на нова MeshInstance3D.



За колизија на играчот

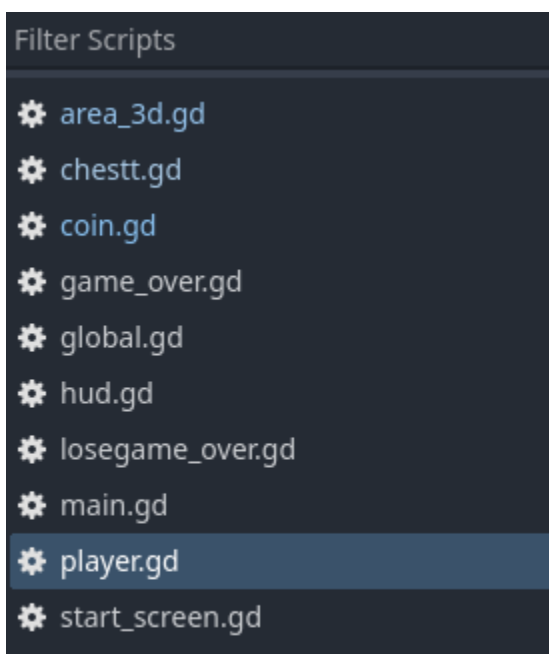


Звук

- Музика во позадина, преземена од [овде](#)
- Звучни ефекти
 - При собирање на богатства: [collect_sound_short.wav](#)
 - При пад во водата: [f.wav](#)

Ги поврзав со помош на `AudioStreamPlayer3D`, кој го додадов во соодветните сцени.

Скрипти

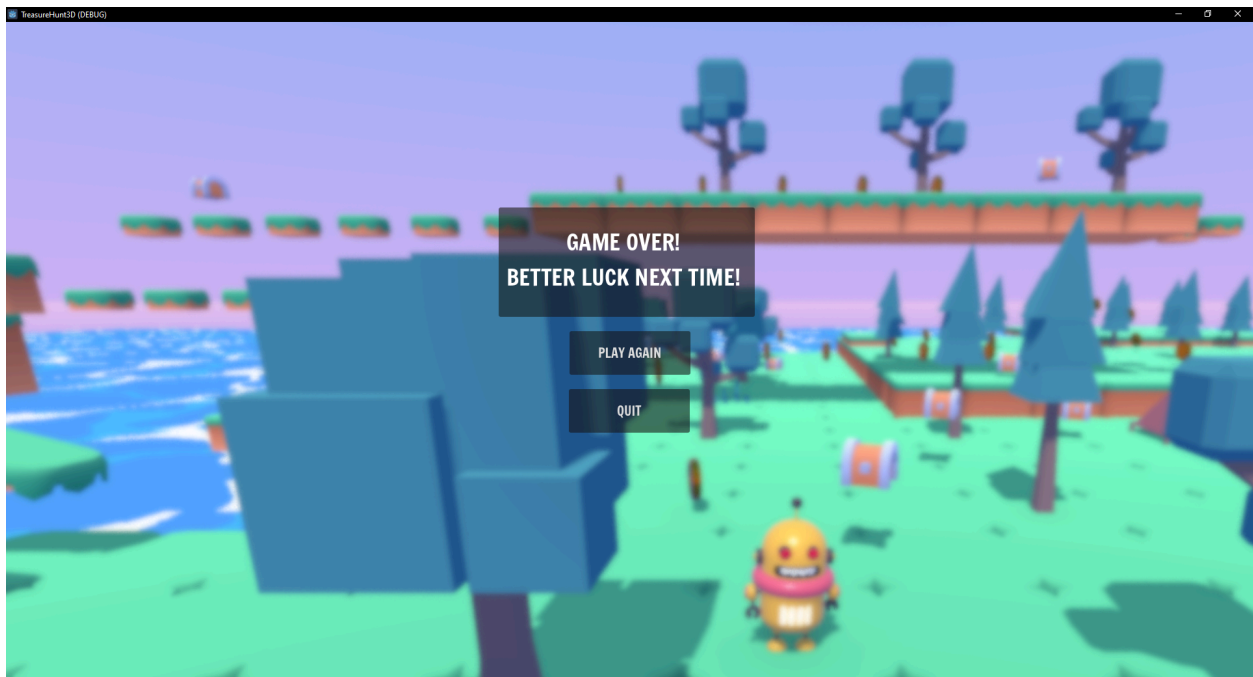
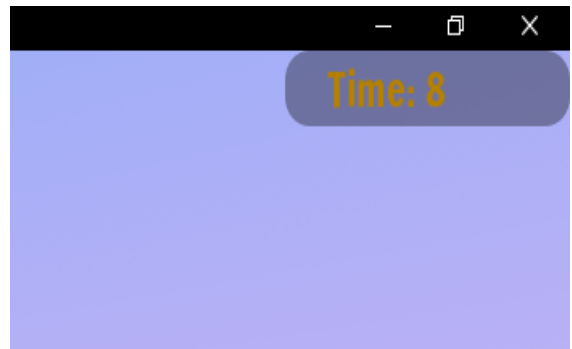


- `player.gd` - овозможува движење, скокање и интеракција на играчот.
- `coin.gd` и `chestt.gd` - скрипти за собирање предмети
 - `main.gd` - главна сцена на играта која чува глобални променливи и ажурира UI.
 - `hud.gd` – за број на собрани богатства и тајмер
 - `start_screen.gd` – почетен екран со копче за започнување на играта и излез

- `game_over.gd` – сцена што се прикажува кога играта ќе заврши со опција за рестарт.

UI Елементи

- HUD е 2D сцена која прикажува колку монети се собрани.
- Почетен екран: Се појавува кога играчот ќе ја стартува играта.
- Краен екран: Се појавува кога играчот победува или губи.



Идни подобрувања

Играта дозволува да се додадат повеќе нивоа кои стануваат потешки, бројот на парички се намалува. Така играчот добива поголем предизвик за пократко време да победи. Исто може да се имплементираат различни парички кои носат различен број на поени.

Користени материјали

1. [Godot Engine.](#)
2. [Nature Kit by Kenney](#)
3. [3D Coin Model by BornCG](#)
4. [Godot 3D Platformer Tutorial by Miziziziz](#)
5. [3D Coin Pickup Animation by BornCG](#)
6. [Background Music - Fun Retro Game](#)
7. [Collect Sound Effect by Kenneth Cooney](#)
8. [Jump Sound Effect by Plasterbrain](#)