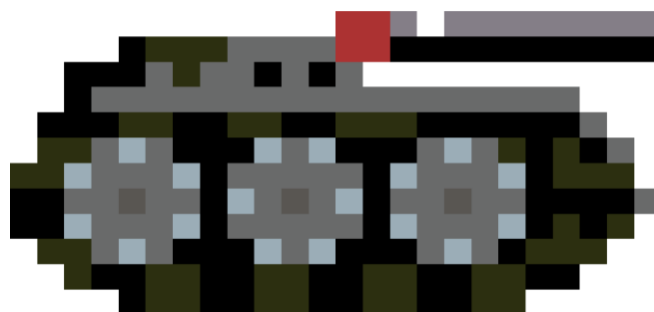


Jungle Jumble

Descriere Tehnica



versiunea: 1.0

-2019-

User Interface

UI-ul este format din 4 ecrane: meniul principal, meniul de alegere al profilului, ecran de leaderboard-uri, meniul de pauza.

Meniul de alegere al Profilului

Cand aplicatia este pornita jucatorului ii va fi prezentat un meniu cu 3 obiecte de tip buton, fiecare va avea numele profilului si nivelul la care se afla respectivul jucator, daca un profil nu este initializat atunci numele este „None” si nivelul este „0”.

Daca jucatorul apasa pe un profil neinitializat meniul actual va fii dezactivat si un textbox va fii creat unde jucatorul isi poate scrie numele dorit, va apasa „enter” si noul profil va fi salvat iar meniul principal va fi lansat.

Daca jucatorul apasa un profil existent meniul principal va fi lansat cu configuratia din profilul respectiv (numele jucatorului si nivelul la care a ajuns).

Meniul Principal

Meniul principal contine 4 butoane: New Game, Continue, Leaderboard, Quit.

Butonul „ New Game” va incarca primul nivel al jocului, jocul va fi salvat numai daca nivelul la care jucatorul a ajuns este mai mare decat nivelul din profilul sau; butonul „Continue” va incarca ultimul nivel neterminat de jucator; butonul „Leaderboard” va incarca ecranul de Leaderboard al jocului; butonul de Quit va inchide aplicatia

Ecran de Leaderboard

Ecranul contine un obiect de tip text unde sunt afisate toate profilele initializate cu scorul lor, in ordinea scorului lor, si un buton ce intoarce jucatorul la meniul principal.

Obiectul text este creat prin citirea celor 3 profile din fisierul „profiles.txt”, izolarea celor initializate si sortarea lor dupa nivel.

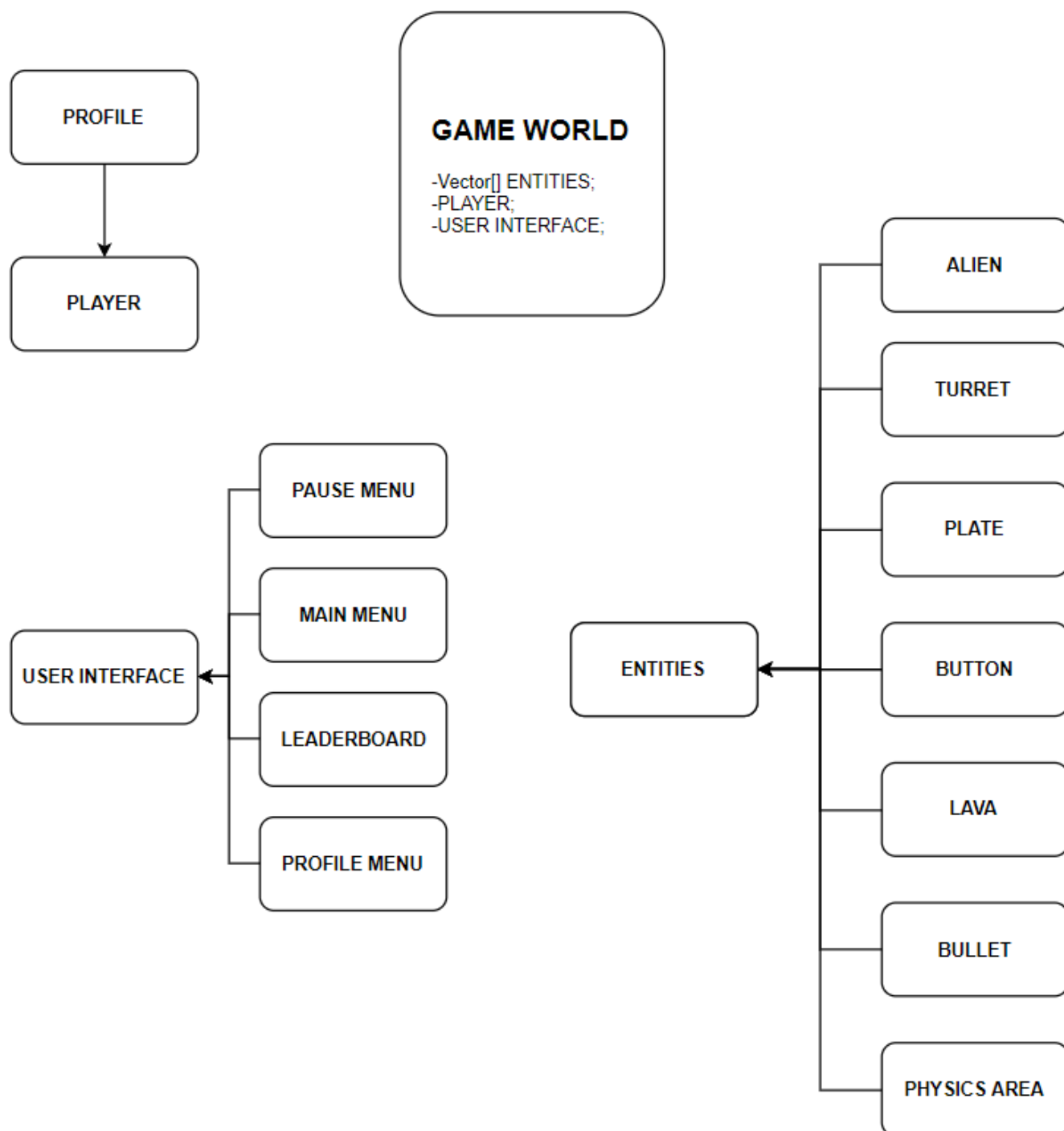
Meniu de Pauza

Daca in timpul jocului jucatorul apasa tasta „escape” viteza jocului va fi setata la 0 si meniul de pauza va fi setat ca vizibil. Acesta contine 3 butoane: Resume, Main Menu si Quit.

Butonul „Resume” seteaza viteza jocului inapoi la viteza initiala si seteaza meniul invizibil; butonul „Main Meniu” incarca meniul principal al jocului, progresul in nivel este pierdut; butonul „Quit” inchide aplicatia.

SCHEMA CLASELOR

ierarhia claselor



Descriere

Jocul are ca si clasa principala clasa GAME WORLD ce contine un PLAYER, un vector de ENTITIES si o instanta a USER INTERFACE.

PLAYER este clasa derivata din clasa PROFILE, el contine numele pofilului, nivelul actual si un GameObject de tip player pe care jucatorul il controleaza in lumea jocului.

Clasa ENTITIES este clasa de baza pentru toate entitatile non-player din joc, fiecare entitatie are, pe langa caracteristicile specifice fiecarei entitati, un ID unic si un set de coordonate din lumea jocului.

Clasa USER INTERFACE contine toate meniurile din joc si permite jucatorului sa ajunga la diferitele servicii ale jocului cum ar fii ecranul de Leaderboards.

DESCRIEREA SCRIPT-URILOR

Prezenarea functionalitatii script-urilor.

PlayerScript

PlayerScript.cs este scriptul responsabil cu controlul player-ului si interactiunea acestuia cu scripturi externe. Fisierul contine o singura clasa cu acelasi nume.

Player-ul este controlat prin intermediul tastaturii, folosind tastele WASD sau sagetile de pe tastatura si tasta Space.

Pentru a putea merge pe pereti, am dezactivat gravitatiea Rigidbody-ului si in locul acesteia am adaugat player-ului componenta „Constant Force 2D” care are rolul de a simula gravitatiea artificiala, in orice directie. Astfel, atunci cand este necesara rotirea player-ului se schimba si directia in care actioneaza forta constanta.

Pentru a sti cand este necesara rotirea player-ului, se amplaseaza pe acesta 4 senzori: 2 in colturile stanga si dreapta jos, care verifica daca player-ul este inca in contact cu terenul; si inca 2 la mijlocul intalimii, in dreapta si in stanga, care verifica daca am intrat in coliziune cu perete. In functie de datele citite de acesti senzori (care sunt doar niste GameObject-uri fara nimic adaugat, pozitionate in locurile mentionate mai sus, din care se cauta coliziuni folosind functia Raycast) player-ul este rotit corespunzator.

Clasa PlayerScript furnieaza scripturilor externe valoarea HP-ului printr-un getter si permite modificare acestuia (scaderea vietii).

AlienScript

AlienScript.cs este scriptul responsabil cu miscarea extraterestrilor in felul urmator: se misca cu o viteza constanta un anumit timp intr-o directie (TimpDirectie) si dupa ce timpul se termina sau extraterestrul loveste un zid, isi schimba directia.

Scriptul detecteaza coliziunea cu playerul si cand aceasta se intampla apeleaza functia getDamage din PlayerScript.

ButtonScript

ButtonScript.cs este scriptul care se ocupa cu interactiunea player-ului cu butoanele din joc. Astfel, cand playerul este in dreptul buton el poate apasa tasta F pentru a apasa butonul. Butonul poate avea diferite actiuni: schimbarea sprite-ului unei usi, mutarea in stanga sau in dreapta a unei usi, distrugerea unei usi. Rolul butonului caruia ii este atasat scriptul este salvat in variabila TypeOfInteraction.

Frecare

Frecare.cs este scriptul cu ajutorul caruia viteza de miscare a player-ului este influentata de un factor. Astfel asupra tuturor miscarilor player-ului se adauga un factor de frecare care duce la incetinirea sau miscarea mai rapida a playerului.

Acest script intra in actiune atunci cand player-ul intra intr-o zona speciala, marcata ca trigger.

LavaScript

LavaScript.cs este responsabil cu aplicarea de damage player-ului atunci cand player-ul intra in lava. Damage-ul este aplicat in intervale de 2 secunde.

TurretScript

TurretScript.cs se ocupa cu instantierea de proiectile la o anumita rata. Proiectilele pot fi trase in toate cele 4 directii, codificate ca 1, -1, 2, -2. Dupa instantierea proiectilului se apeleaza o functie pentru directionarea acestuia.

ProjectileScript

ProjectileScript.cs este atasat proiectilelor si are rolul de a misca proiectilul intr-o anumita directie la cu o viteza constanta. Cand acesta loveste playerul cauzeaza damage, iar cand loveste un zid se autodistruge.

GameManager

GameManager.cs se ocupa cu citirea, incarcarea, creerea de noi profile. Datele despre playeri sunt tinute intr-un fisier local.

NextLevel

NextLevel.cs are ca scop actualizarea informatiilor salvate despre player in fisier si incarcarea noului nivel atunci cand cel actual a fost terminat.