

C0re

Proiect realizat de Imbrea Daniel

- Gen de joc:

Asteroids-like arcade / Rogue-lite

- Povestea:

Au trecut mii de ani de la prima planetă colonizată și primul contact cu ființe extraterestre, iar între timp zeci de galaxii au devenit casă pentru oameni. În timp, aceste colonii s-au consolidat și astfel a apărut Imperiul Galactic Terestru.

IGT era un stat corporat-autocratic, condus atât de o elită selectă, cât și de cele mai puternice companii. O revoluție abruptă a dus la fragmentarea IGTului, în diferite facțiuni, de la companii militare precum ADW-Tech, dictaturi regionale ca Dictatul Absterian și militanții religioși ai cultului Kertham, iar vechiul imperiu, sub o nouă conducere s-a redenumit în Hegemonia pentru Unitate Umană.

În acest climat, un proaspăt absolvent al Institutului de Aeronautică Spațială care dorește să își facă un nume, se află în poziția de a-și oferi abilitățile uneia din facțiuni. După un timp acesta devine din ce în ce mai renumit pentru talentele sale în lupta și navigație și astfel, acesta ajunge să facă parte din elită facțiunii. Aici află că undeva la marginile cunoscute ale spațiului, se află o veche baza de cercetare, unde se presupune că se află o armă atât de puternică, că ar putea opri toate conflictele prin simplă menționare a acesteia.

Astfel, tânărul pilot alături de un convoi militar este trimis în căutarea acestei arme. După un traseu de câteva zeci de ani lumina însă, o ambuscadă inamică duce la decimarea acestui convoi, din care, prin manevre de evadare, doar pilotul nostru reușește să scape. Singur, și cu o nava de dimensiuni mici, acesta se hotărăște să își continue drumul către misterioasă destinație.

În traseul sau, acesta va întâlni pirați, nave inamice și situații precum centuri de asteroizi și găuri negre, obstacole pe care acestea le va învinge, iar la diverse service-uri galactice, să își îmbunătățească nava. În final reușește să ajungă la baza de cercetare, în cel mai rău moment: flagship-ul inamic, alături de un convoi deja au preluat armă. În aceste condiții, singură soluție a pilotului este de a înfrânge inamicul și de a recupera obiectul mult căutat.

- Prezentarea jocului:

Campanie pentru un singur jucator in care scopul este parcurgerea unui set de nivele generate semi-aleator in care imbunatatirile si inamicii evolueaza odata cu parcurgerea jocului. Un nivel este parcurs cand toti inamicii sunt infranti. Daca nava jucatorului ramane fara durabilitate, jocul este pierdut, scorul jucatorului este inregistrat iar jocul trebuie luat de la inceput.

- Regulile jocului:

La inceput se alege o factiune, care afecteaza nava de inceput si armamentul acesteia.

Gameplay loop-ul consta intr-o insiruire de valuri de inamici care trebuie invinsi. In fiecare val, eroul se lupta cu o selectie de inamici, care cresc in dificultate odata cu inaintarea in joc. În afară de inamici, există posibilitatea ca un grup de asteroizi să treacă prin nivel, făcând astfel navigarea nivelului mai dificilă.

Infrangerea inamicilor ofera jucatorului valuta "Credits". Cu aceasta valuta, jucatorul are abilitatea de a cumpara imbunatatiri pentru arme, module si imbunatatiri generale pentru nava. Accesul la magazin are loc odata la 5 valuri și de fiecare data selectia de imbunatatiri e diferita. Odata la 10 valuri, v-or aparea elite, inamici mai dificili care prezintă o provocare crescuta. Jocul se termina cand nava jucatorului este distrusa, caz în care se consideră că jucătorul a pierdut, sau cand valul 30 este invins, caz în care jucătorul a câștigat.

Daca jucatorul pierde inainte de a atinge obiectivul final, acesta isi pierde tot progresul pentru acea sesiune de joc, si primeste un scor care contribuie la deblocarea de noi factiuni de joc.

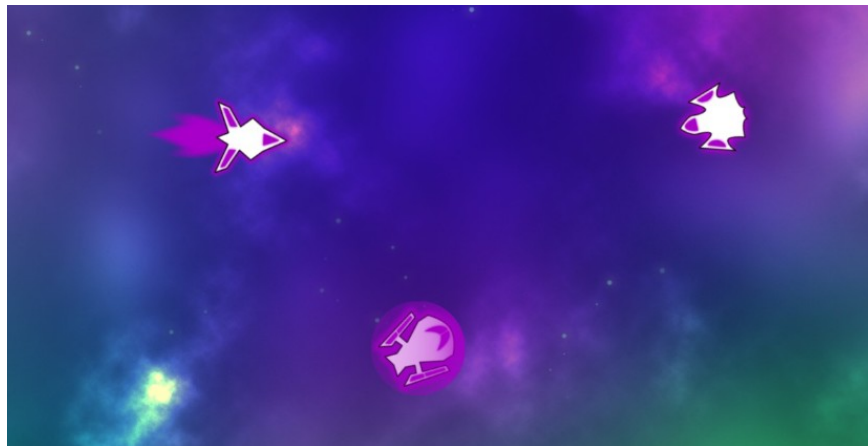
- Prezentarea personajelor:

Personajele constau in diferitele factiuni si sunt reprezentate in joc prin flotele acestora:

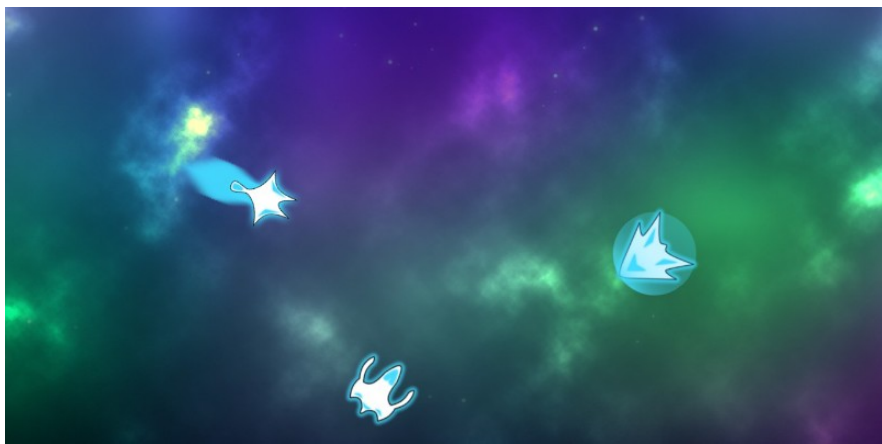
- Hegemonia pentru Unitate Umana (culoare tematica: Portocaliu): este ceea ce a ramas din fostul imperiu, iar conducerea este alcatuita din fosti oligarhi si elita care nu a cedat in urma revolutiei, fiind un stat condus cu o mana de fier si cu o dorinta clara de reintregire. In ciuda dificultatilor aduse de dezertori si pirati, flota hegemoniei este inca puternica si cuprinde nave de talie mare, cu un armament pe masura.



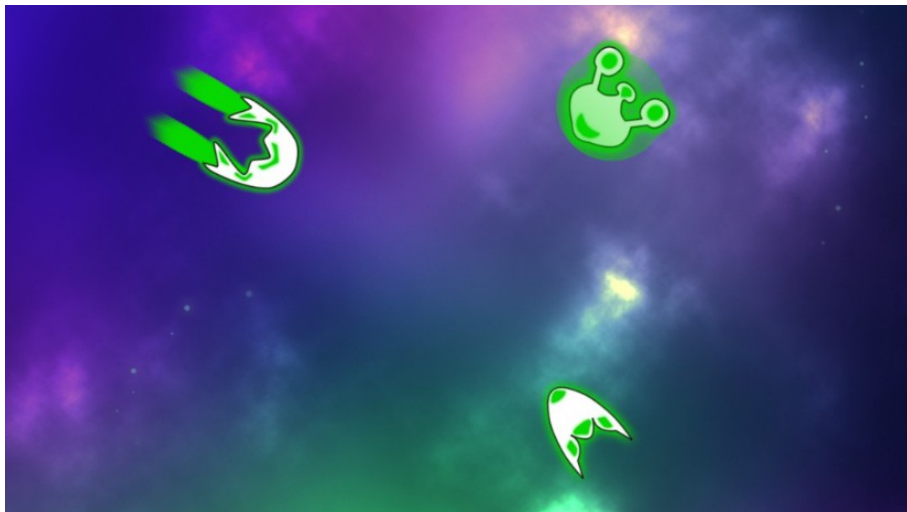
- Dictatul Absterian (culoare tematica: Violet): este clica marelui general Abster Conrad, care a declarat independenta domeniilor sale dupa caderea fostului imperiu. Acesta isi doreste intregirea imperiului sub conducerea lui si de aceea cauta misterioasa baza de cercetare. El foloseste o flota de nave de mercenari, abilitatile navelor diferind de la una la alta.



- ADW-Tech (culoare tematica: Albastru): este cea mai cunoscuta si puternica companie de tehnologie militara. Aceasta pune mare importanta pe inovatie, astfel ca navele lor sunt cele mai avansate din fostul imperiu, cu capabilitati precum invizibilitate, scuturi foarte rezistente si o afinitate pentru lasere. Isi doresc arma secreta pentru a domina complet competitia.



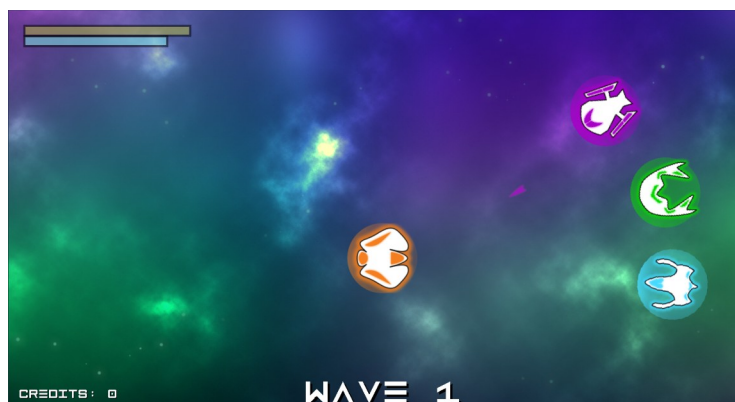
- Cultul lui Kertham (culoare tematica: Verde): este un grup misterios, despre care se zvoneste ca ar dori invocarea unei bestii cosmice, numita Yor'han, pentru implinirea unei profetii. Cultul nu permite accesul persoanelor avute, astfel ca toti membrii folosesc nave de acum cateva generatii, fara scuturi performante, dar cu un sasiu foarte rezistent si mitraliere de calibru mare.

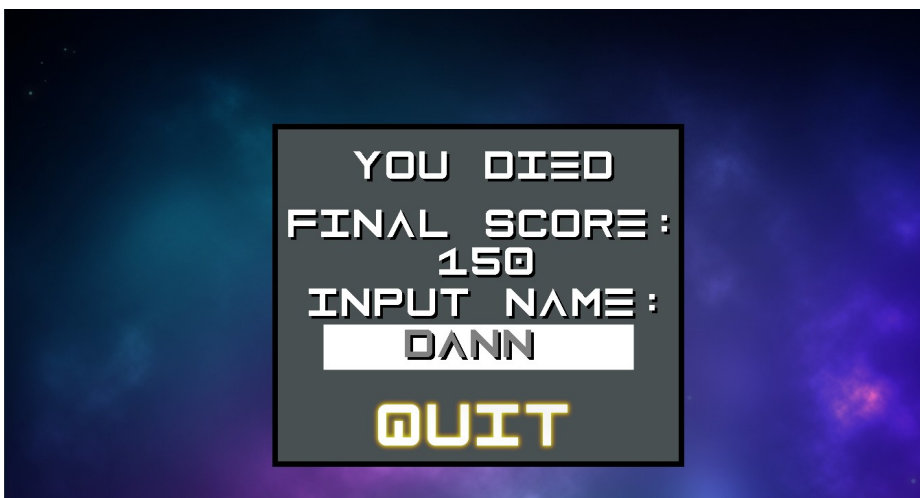
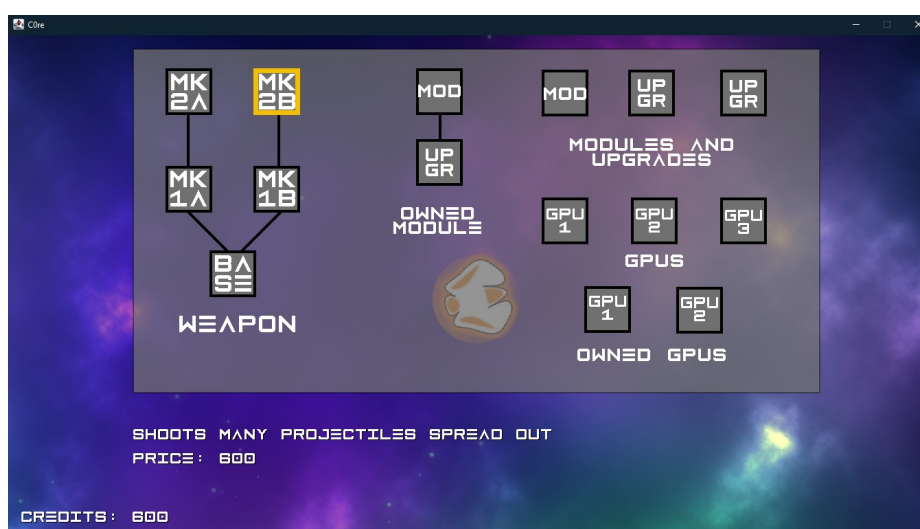


- Tabla de joc:

Tabla de joc este campul de lupta dintre jucator si valurile de inamici. Aceasta contine elemente active precum jucatorul, inamicii și asteroizii cât și proiectilele și scuturile acestora dar și elemente pasive, precum interfața, propulsoarele navelor și fundalul. Aceste elemente fluctuează de la un val la altul, inamicii fiind generați în numeroase configurații. Inamicii și asteroizii funcționează ca obstacole, formând astfel un nivel care se modifică constant.

Exemple de situații din joc:





- Mecanicile jocului:

Jocul incepe prin alegerea unei factiuni, actiune care va afecta nava cu care incepe jucatorul. Fiecare factiune ofera anumite avantaje si dezavantaje fata de celelalte factiuni, reprezentate prin diferite tipuri de arme și module.

Jucătorul are la dispoziție 2 modalități de a controla nava: folosind mouse-ul, sau folosind tastatura, acestea facilitând la o experiență confortabilă de joc, în funcție de preferințe.

Odata aleasă o facțiune, jucătorul este pus pe câmpul de luptă, unde trebuie să înfrunte valuri de inamici generați semi-aleator. Inamicii au acces la toate funcționalitățile la care are acces jucătorul și anume arme, scuturi și module, fiecare cu variațiile lor. Fiecare val constă din 3 inamici, care pot fi generați în numeroase configurații, făcând parte din diferite facțiuni, având arme și îmbunătățiri pentru acestea și chiar module și îmbunătățiri pentru ele. Gradul de dificultate al inamicilor crește pe parcursul jocului, prin faptul că aceștia primesc arme de o calitate mai buna, au mai multă viață și au module mai puternice.

Jucătorul are la dispoziție o arma, care diferă în funcție de facțiunea aleasă. Această armă este principala unealtă ofensivă, jucătorul având însă la dispoziție opțiunea de a se izbi de navele inamice pentru a cauza daune. Armele sunt de 3 tipuri: plasma, balistice și explozive, fiecare tip de armă cauzând daune diferite. De exemplu, plasma este mai eficientă la distrus scuturi, pe când armele balistice sunt eficiente la distrus navele propriu-zise.

Totodată, de la o facțiune la alta diferă și modulul cu care se începe jocul. Aceste module sunt abilități speciale care permit, spre exemplu, îmbunătățirea armei pentru o perioadă scurtă de timp, regenerarea rapidă a scutului, sau chiar teleportarea într-o anumită direcție. Aceste module pot fi schimbate pe parcursul jocului, spre deosebire de arme.

În cele din urmă, scutul este cea mai folosită unealtă defensivă, protejând nava de proiectilele inamice. Acesta se regenerează după o perioadă în care nu este lovit și poate fi îmbunătățit pentru o regenerare mai rapidă.

Înfrangerea unui inamic crește contribuie la creditele jucatorului, iar aceste credite pot fi folosite în magazine pentru a debloca îmbunătățiri de diferite feluri. Îmbunătățirile variază de la simple creșteri ale vitezei sau vieții, până la obținerea de abilități unice, precum teleportarea, sau evoluarea armei și schimbarea comportamentului acesteia. Jucătorul are acces la magazin la fiecare 5 runde și de fiecare dată acesta are în stoc lucruri diferite.

Odată la 10 runde, în loc de 3 inamici, va apărea unul singur, dar care este de o dificultate crescută, având mult mai multă viață și folosind 3 arme în loc de una singură.

Jocul se încheie când jucătorul moare, sau când este învins valul 30, urmând apoi înscrierea unui nume și apariția a acestuia alături de scorul obținut, în leaderboard, dacă scorul se află printre cele mai bune. Totodata, la finalizarea unei sesiuni de joc, scorul final este convertit în credite globale, care pot fi folosite pentru a debloca noi facțiuni de joc.

- Prezentarea meniului:



- Meniul principal conține 4 butoane: Play, Options, Quit și Leaderboard.

Butonul de Play are 2 funcționalități:

În cazul în care nu există nici un save state salvat, acesta duce jucătorul la meniul de selectare a unei facțiuni. Aici jucătorul poate alege una din clasele deja deblocate, sau poate consuma credite pentru a debloca o facțiune nouă. Totodata există opțiunea de a te întoarce înapoi la meniul principal.

Dacă există deja un save state salvat, atunci se trece direct în jocul propriu-zis, continuând sesiunea anterioară.

- Meniul de selectare a facțiunii



Butonul Options duce la meniul de modificare a opțiunilor, și anume: modificarea volumului, și schimbarea modului de control.

- Meniul de opțiuni:



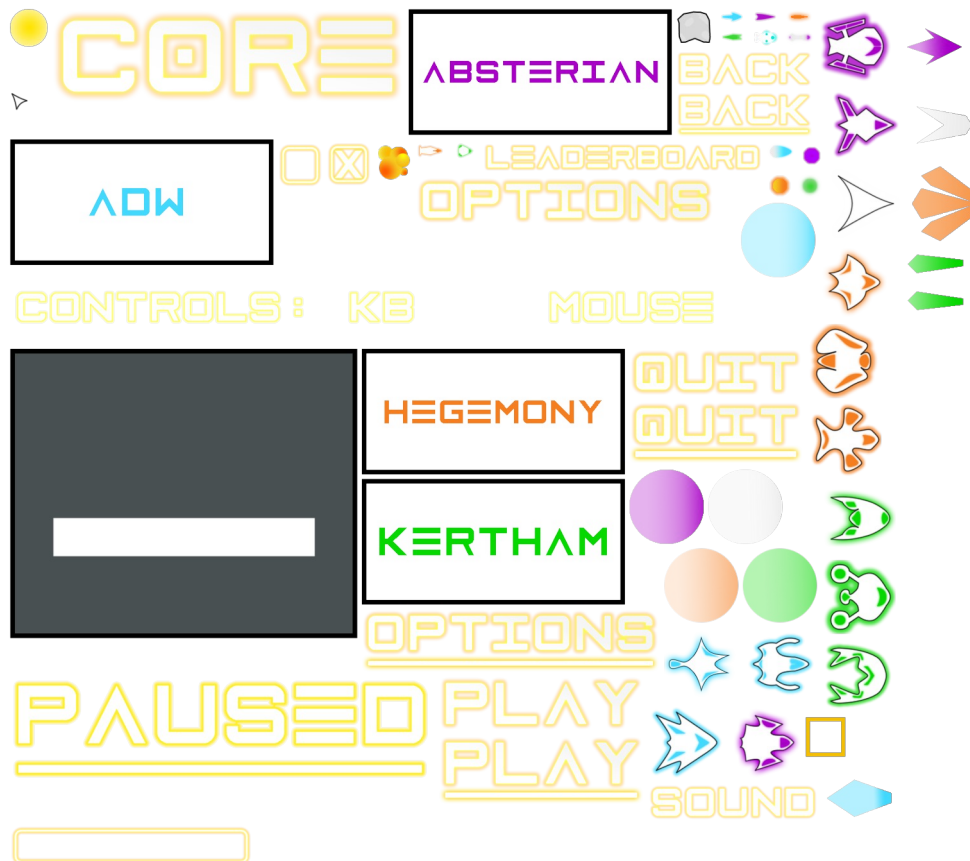
Butonul Leaderboard duce la prezentarea unui “Hall of fame”, mai exact cele mai bune 10 scoruri care au fost salvate.

- Meniul Leaderboard:



În cele din urmă, Butonul Quit închide jocul.

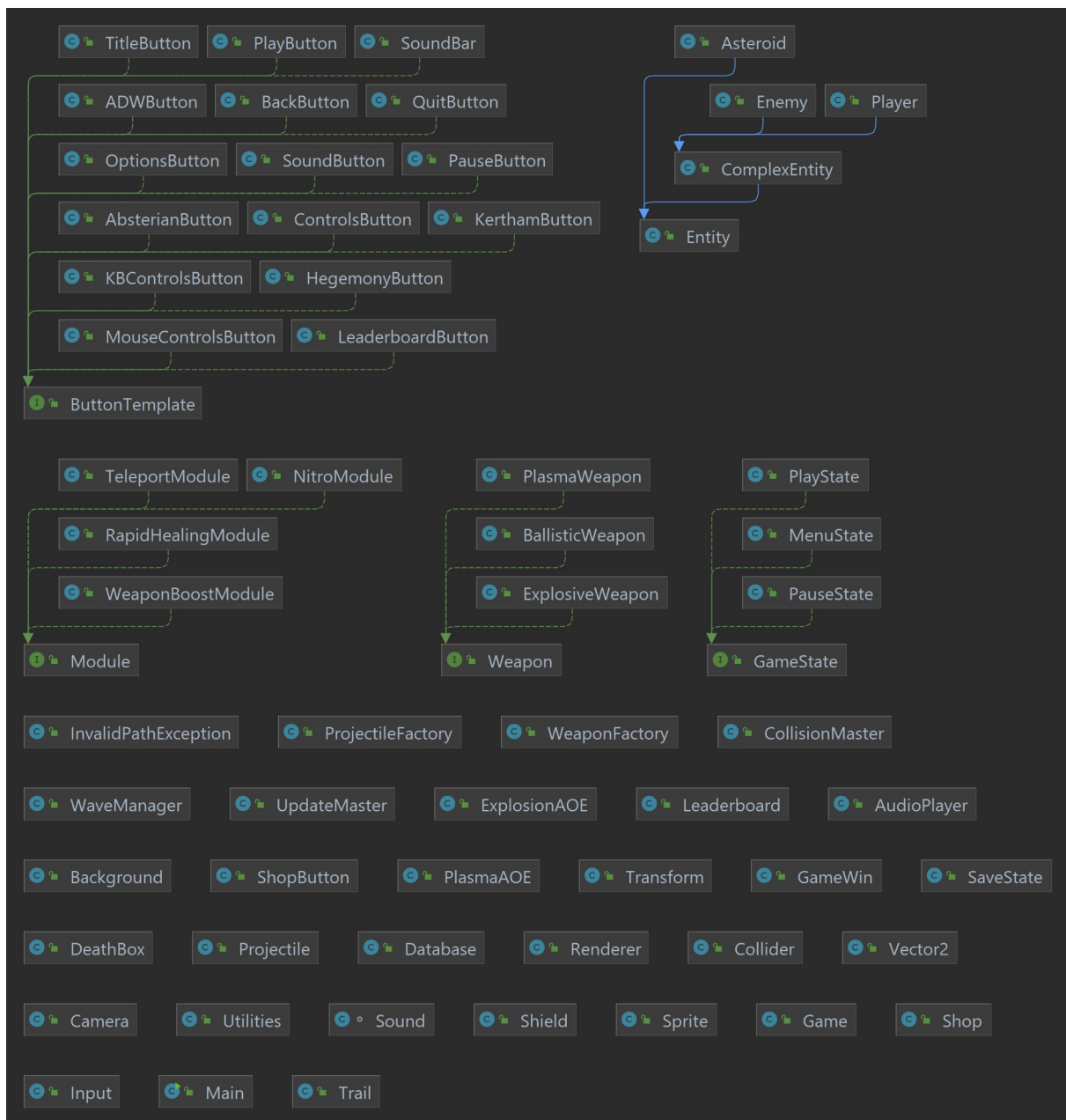
- Sprite-urile:



- Bibliografie:

Toate asset-urile sunt create de mine, cu excepția fontului, care face parte din Public Domain.
Fontul a fost preluat de la: <https://www.fontspace.com/arizone-unicase-font-f32250>

- Elemente de implementare:
- Diagrama generala a claselor:



Aceasta diagrama cuprinde toate clasele utilizate în proiect, și relațiile de extindere și implementare.

- Diagrama entităților:

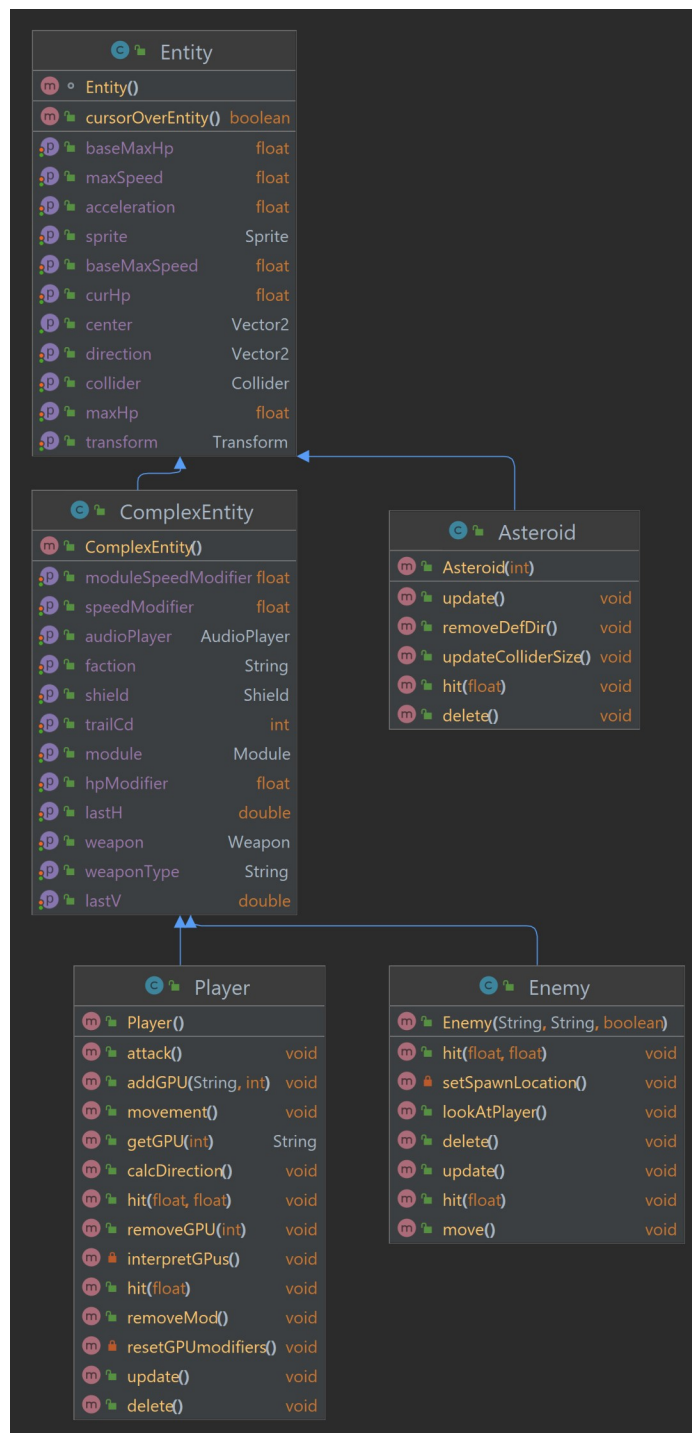
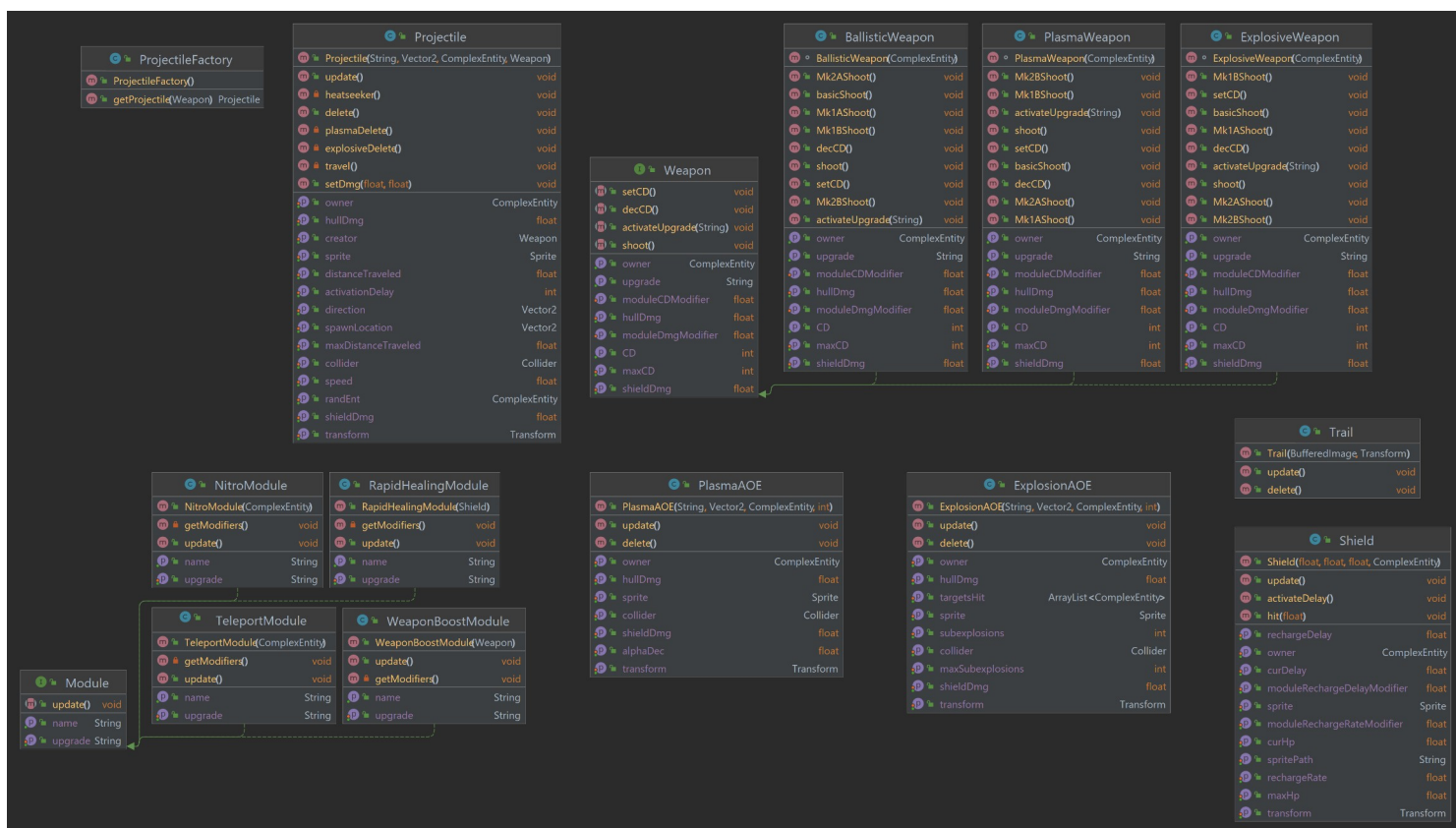


Diagrama entităților cuprinde jucătorul, asteroizii și inamicii. Asteroizii sunt entități simple, astfel ca nu au arma, scut sau module, pe când jucătorul și inamicii au mai multe câmpuri, și funcții mai complexe.

- Diagrama addon-urilor entităților:



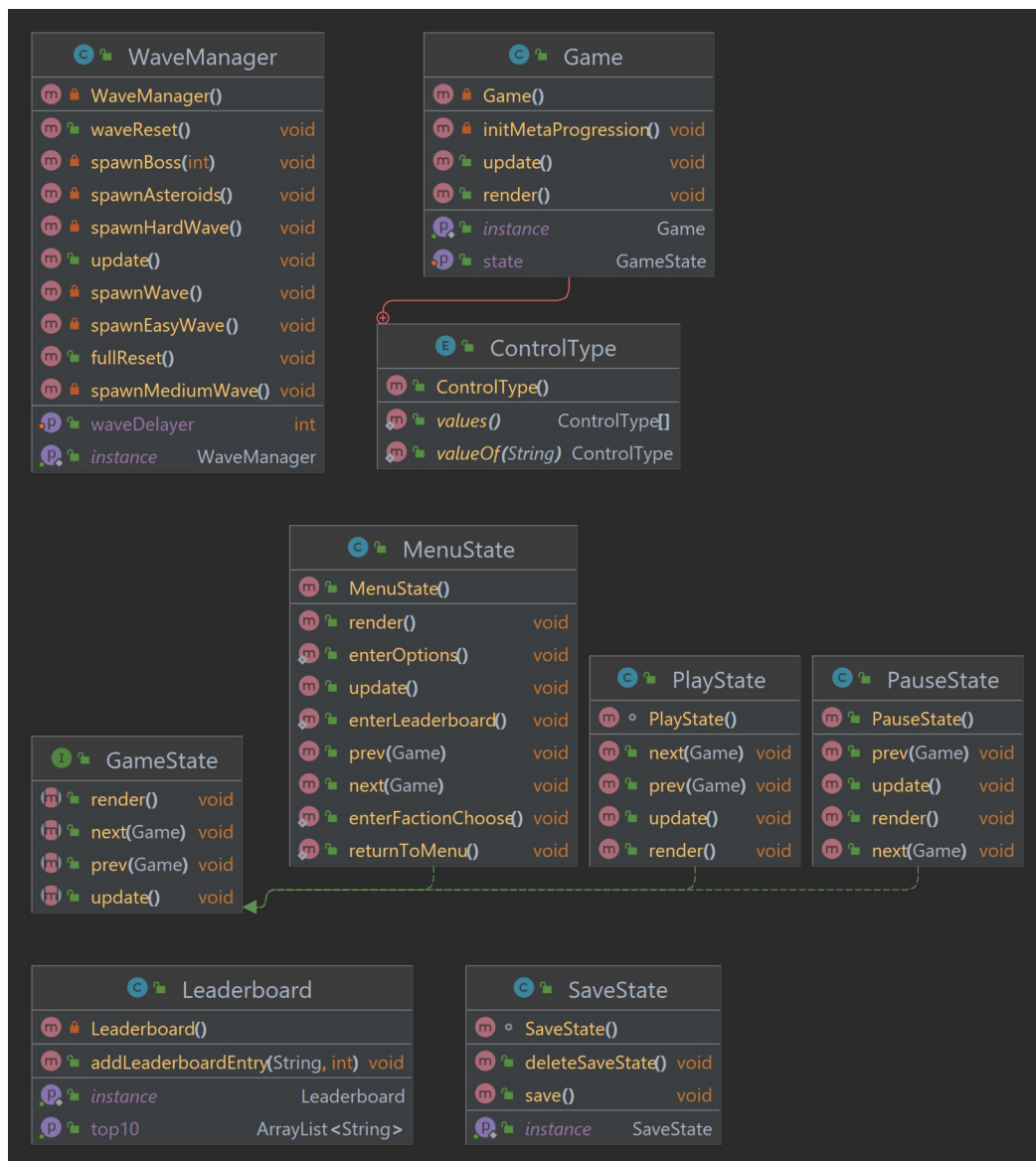
Această diagramă cuprinde toate obiectele care sunt folosite de entitățile complexe, sau care sunt generate de entități complexe.

Armele sunt generate de weapon factory și implementează clasa weapon, din care rezultă variațiile în comportament ale armelor balistice, explozive și plasma.

Armele la rândul lor, folosesc un projectile factory pentru a genera proiectile de diferite feluri (și anume proiectile specifice armelor de tip explozive și plasma).

Modulele implementează o clasă de bază, care este folosită pentru a putea fi schimbat cu ușurință modulul utilizat de jucător.

- Diagrama jocului:



Jocul este controlat printr-un state machine, în funcție de situația în care se află. Stările implementate sunt meniu, joc propriu-zis și pauză, iar cu ajutorul butoanelor aceste stări se schimbă de la una la alta, în mod ciclic. Fiecare stare are o funcție de randare și una de actualizare, astfel că în funcție de stare, resursele sunt folosite doar pentru lucrurile folosite. Alte elemente generale ale jocului sunt: Wave managerul, care se ocupă cu generarea valurilor de inamici, SaveState-ul, contrar numelui nu face parte din state machine, ci se refera la ultima sesiune joc care a fost salvată și este folosit pentru a salva progresul. În cele din urmă Leaderboard-ul se ocupă cu salvarea și afișarea celor mai bune scoruri.

- Diagrama elementelor grafice:

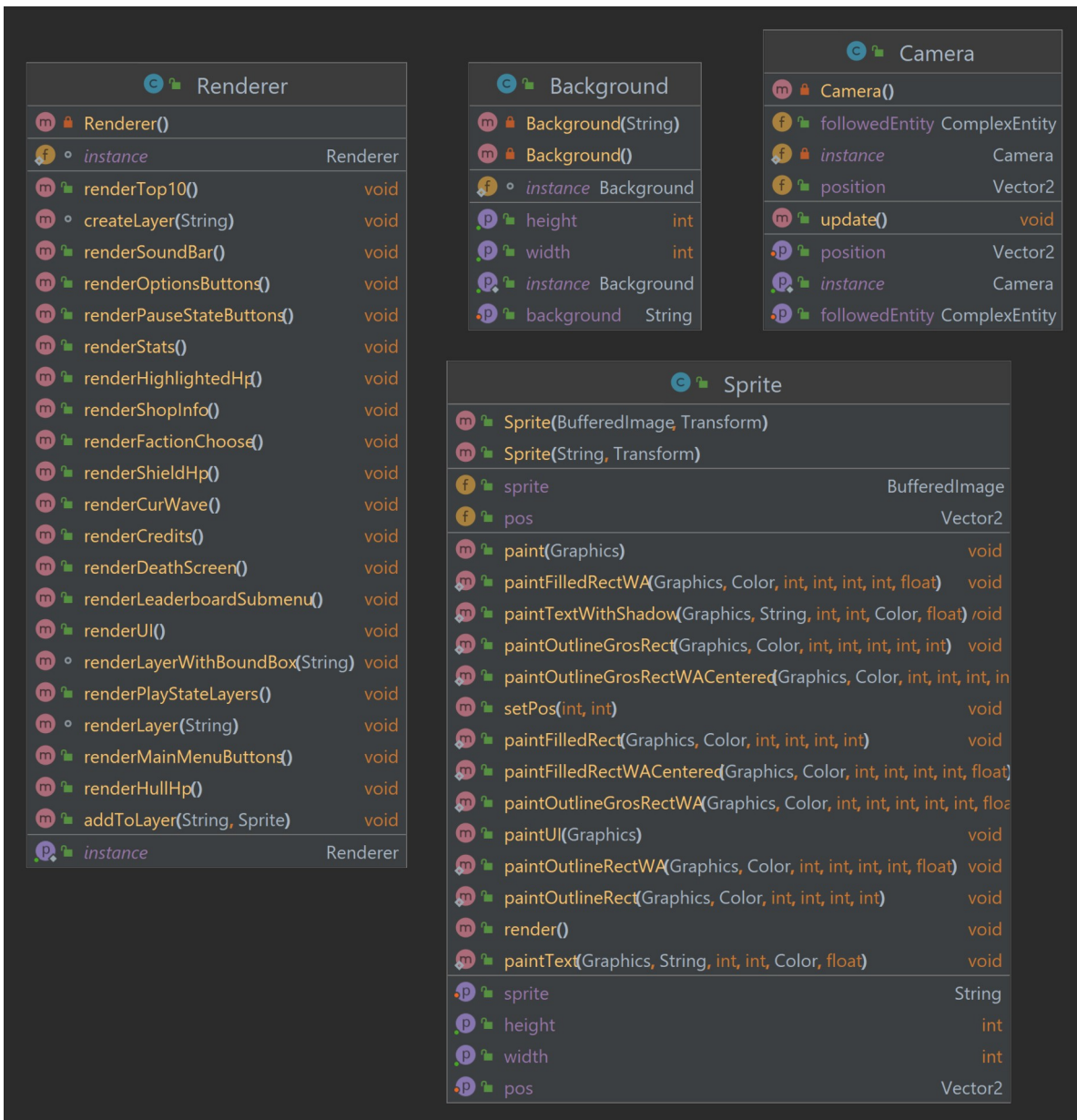


Diagrama elementelor grafice cuprinde elementul de baza, Sprite-ul, în care se găsește atât funcția de randare a unei imagini încărcate în memorie, cât și metode statice pentru desenare de elemente grafice primitive, precum dreptunghiuri, sau text.

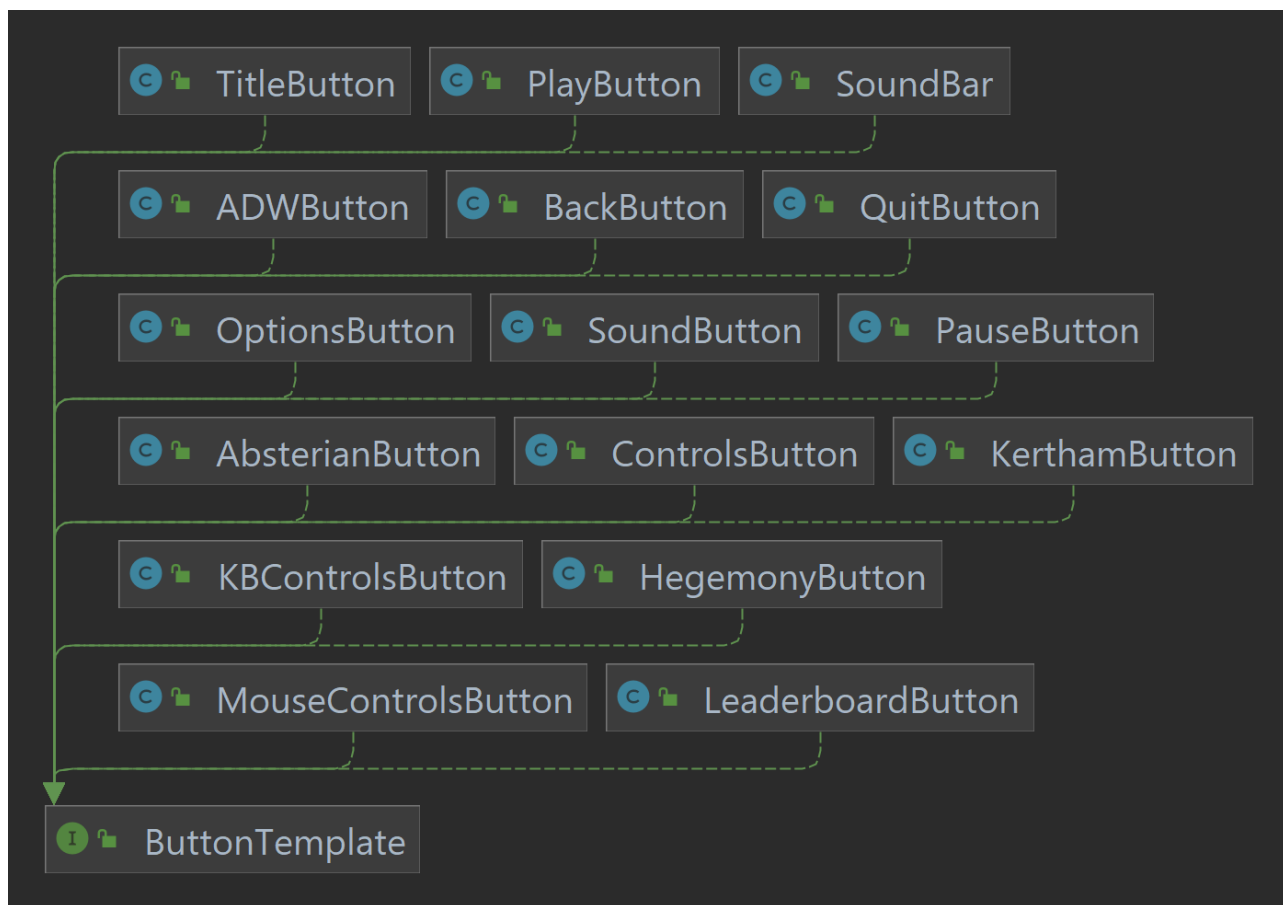
Background-ul se folosește de sprite, dar este diferențiat pentru că modul de randare al acestuia este tratat diferit de un element grafic obișnuit.

Renderer-ul se ocupa cu randarea tuturor elementelor grafice din joc, cu ajutorul layerelor de randare. Fiecare sprite poate fi atribuit unui astfel de layer, care este apoi randat odată cu toate

elementele din acel grup. Astfel, la randare se poate garanta ordinea corectă a afișărilor, și desenarea corectă a interfeței peste aceste sprite-uri.

În cele din urmă, camera este folosită pentru a putea muta jucătorul pe tabla de joc și scopul ei este de a offseta pozițiile tuturor sprite-urilor, pentru a crea efectul de deplasare.

- Diagrama butoanelor:



Butoanele implementează template-ul de butoane, care conține o funcție de hover, și o funcție de click, urmând ca funcționalitatea să fie făcută în funcție de nevoie.

- Diagrama interfeței:

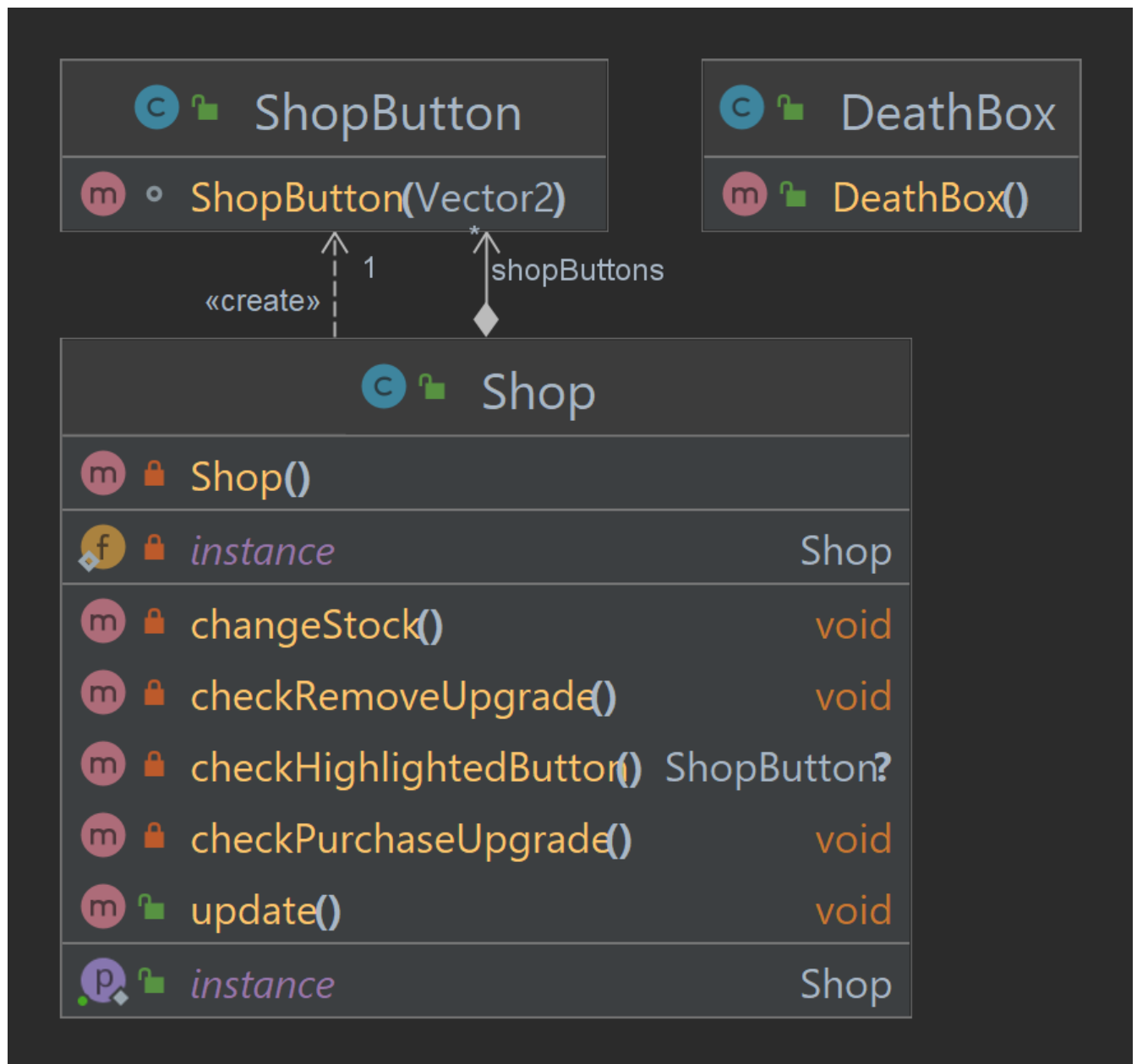


Diagrama aceasta cuprinde magazinul folosit pentru obținerea de îmbunătățiri pentru navă, alături de butoanele folosite în magazin, și meniul care apare atunci când jucătorul pierde.

- Diagrama utilităților:



Această diagramă cuprinde clasele folosite cu scop general sau care nu se încadrează într-una din pachetele menționate anterior.

Clasa Transform reține poziția, dimensiunea și rotația a unui obiect.

Clasa Vector2 este folosită pentru a reprezenta puncte în spațiu și cuprinde metode utile pentru a calcula relația între 2 puncte.

Clasa Collider conține metoda de verificare a suprapunerii a 2 obiecte.

Clasele Update Master și respectiv Collision Master se ocupă cu actualizarea și respectiv verificarea coliziunilor obiectelor, în funcție de contextul jocului.

Input se ocupă cu preluarea și prelucrarea controalelor.

Database este implementarea Java a bazei de date SQLite și este folosită pentru a memora leaderboardul, setările, progresul între sesiuni de joc și ultimul save state.

Audio player și Sound sunt folosite pentru a porni sunetul în joc.

- Tratarea erorilor:

În cadrul proiectului excepțiile au fost verificate și tratate folosind excepții predefinite, cât și o clasă de excepții creată special pentru verificarea corectitudinii path-ului la schimbarea sprite-urilor.

```
package Exceptions;

public class InvalidPathException extends Exception{

    public InvalidPathException(String s) { super(s); }

}
```

```
public void setSprite(String filepath) throws InvalidPathException {

    if(filepath == null)
        throw new InvalidPathException("Sprite path is null");

    sprite = null;
    System.gc();

    try {
        sprite = ImageIO.read(new File(filepath));
    } catch (IOException e) { System.out.println("Cannot load image"); }

}
```



```

public void setBackground(String filepath){
    try {
        sprite.setSprite(filepath);
    } catch (InvalidPathException e){
        e.printStackTrace();
    }
}

```

- Structura bazei de date:

Table: Leaderboard

	ID	SCORE	NAME
	Filter	Filter	Filter
1	1	150	dann

Table: MetaProgression

	ID	GlobalCredits	Volume	HegemonyUnlocked	ADWUnlocked	AbsterianUnlocked	KerthamUnlocked
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0	125	1.0	1	0	0	0

Table: SaveState

	ID	PlayerFaction	Credits	CurrentWave	GPU1	WeaponUpgrade	GPU2	Module	ModuleUpgrade	CurHP
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	0	Hegemony	0	-1				Weapon boost		100.0

- Șabloane de proiectare:

În cadrul proiectului au fost folosite 3 șabloane de proiectare și anume: Singleton, State Machine și Factory.

Singleton a fost folosit în numeroase clase pentru a asigura existența unei singure instanțe din acea clasă. Principalele clase care utilizează singleton sunt Game, Renderer, butoanele, cât și alte elemente de back-end.

Exemple de Singleton din cod:

```
static Game instance = null;

public static Game getInstance(){
    if(instance == null){
        synchronized (Game.class){
            if(instance == null)
                instance = new Game();
        }
    }

    return instance;
}
```

```
static Renderer instance = null;

public static Renderer getInstance(){
    if(instance == null){
        synchronized (Renderer.class){
            if(instance == null)
                instance = new Renderer();
        }
    }

    return instance;
}
```

State Machine a fost folosit pentru separarea sarcinilor jocului, în funcție de contextul curent. Starile jocului sunt: Meniu, Joc propriu-zis și Pauza. Aceste stări dictează ce elemente vor fi randate și actualizate în fiecare frame.

Exemple ale utilizării state machine-ului:

```
Game.gameState.prev(Game.getInstance());
```

```
//set game state  
gameState = new MenuState();
```

```
gameState.update();
```

```
gameState.render();
```

Factory a fost folosit în 2 situații: pentru generarea armelor pentru jucător și pentru inamic, cât și la generarea proiectilelor în momentul folosirii acestor arme.

```
weapon = weaponFactory.getWeapon( owner: this);
```

```
Projectile proj = projectileFactory.getProjectile( creator: this);
```