

UNIVERSITATEA POLITEHNICA DIN BUCUREȘTI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

Grafică, Multimedia și Realitate Virtuală

PROIECT DE DIPLOMA

ÎNTRUMĂTOR ȘTIINȚIFIC
Prof. Dr. Ing. Moldoveanu Alin

ABSOLVENT
Soare Robert Daniel

Bucuresti, 2023

UNIVERSITATEA POLITEHNICA DIN BUCUREȘTI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE

Grafică, Multimedia și Realitate Virtuală

Aprobat Decan ` Prof. Dr. Ing. Mihnea Alexandru Moisescu

PROIECT DE DIPLOMA

Design extensibil pentru Tower Defense

ÎNTRUMĂTOR ȘTIINȚIFIC
Prof. Dr. Ing. Moldoveanu Alin

ABSOLVENT
Soare Robert Daniel

Bucuresti, 2023

Cuprins

Introducere	4
1 Genul de joc Tower Defense.....	5
2 Design-ul unui joc de Tower Defense.....	7
2.1 Modele hibride de jocuri.....	7
2.2 O nouă metodă de extindere a mecanicii de joc.....	8
2.3 Compunerea jetoanelor de acțiune.....	8
2.4 Tipurile de turnuri.....	9
2.5 Valul de inamici.....	10
2.6 Economia de joc.....	10
3 Implementarea sistemelor.....	11
3.1 Primele minute de joc.....	11
3.2 Sistemul de jetoane de acțiune.....	12
3.3 Sistemul de turnuri de apărare.....	14
3.4 Sistemul de economie de joc.....	15
3.5 Sistemul de inamicii.....	16
3.6 Interfața de utilizator.....	16
Bibliografie	17

Introducere

Genul “tower defense” (sau “apărare prin turnuri”) este un gen de joc video în care jucătorul are rolul de a construi și de a upgrada turnuri defensive pentru a împiedica invazia inamicilor și a proteja o anumită zonă sau resurse. Jucătorul trebuie să plaseze strategic turnuri cu diferite abilități și funcții, cum ar fi turnuri de tragere, turnuri de aruncare a proiectilelor sau turnuri magice, pentru a opri inamicii să ajungă la punctele cheie ale hărții sau să distrugă baza jucătorului. Pe măsură ce jocul avansează, inamicii devin tot mai puternici, iar jucătorul trebuie să-și îmbunătățească strategiile de apărare și să facă alegeri strategice pentru a reuși să reziste valurilor de inamici.

De-a lungul anilor, genul de joc Tower Defense a evoluat și a rămas unul dintre cele mai populare genuri de jocuri în rândul jucătorilor din întreaga lume. Cu toate acestea, odată cu creșterea continuă a pieței jocurilor video, există nevoia de a inova și de a oferi jucătorilor o experiență de joc nouă și interesantă. Iar această experiență poate varia de la o valoare educativă [1] la una pur distractivă.

Acest gen de joc este tot mai întâlnit pe platformele de jocuri mobile, precum Google Play și App Store. Punctele forte care îl fac să fie atât de popular sunt:

- Sesiuni de joc scurte care pot fi jucate oricând și oriunde.
- Nu necesită atenție continuă, jucătorul poate să se concentreze pe alte activități în timp ce jocul rulează în fundal.
- O rundă de joc poate fi câștigată prin mai multe moduri, astfel oferă o experiență de joc variată.
- Mecanică joc de simplu de învățat.
- Joc de strategie care îi oferă jucătorului satisfacția de a reuși să reziste valurilor de inamici prin prisma deciziilor strategice pe care le ia.

Un punct slab al acestui gen de joc este găsirea unui echilibru în relația dintre turnurile de apărare. În marea majoritate a jocurilor de aceste gen care au avut succes, turnurile de apărare sunt independente și nu au nevoie de ajutorul unui alt turn pentru a funcționa. Acest lucru simplifică mecanica jocului, însă poate duce la o experiență de joc monotună. Implementarea unui sistem interdependent aduce un grad de complexitate ridicat atât pentru jucător cât și pentru dezvoltator. Cea mai mare problemă fiind design hărții de joc care trebuie să fie concepută astfel încât să pună în evidență interdependența turnurilor de apărare.

În această lucrare, ne propunem să cercetăm un model semi-interdependent de colaborare pentru turnurile de apărare. Modelul propus va fi implementat într-un joc de tip Tower Defense, care va fi dezvoltat în programul de dezvoltare pentru jocuri video Unity. Acest model va folosi un sistem de comunicare bazat pe mesaje pentru a comunica între turnuri. Iar aceste mesaje vor fi purtate de către inamici cu denumirea de **jetoane de acțiune**.

Așadar, această lucrare va descrie un set de specificații pentru un joc de tip Tower Defense care va implementa un sistemul de colaborare între turnurile de apărare care va urma să fie descris.

1. Genul de joc Tower Defense

Tower Defense este un gen de joc de strategie în care jucătorii trebuie să-și construiască și să-și îmbunătățească turnuri defensive pentru a împiedica inamicii să ajungă la obiectiv.

Jocurile de Tower Defense se desfășoară de obicei pe o hartă cu un traseu prestabilit pe care inamicii încearcă să avanseze, în timp ce jucătorii plasează și își upgradează turnurile defensive pentru a-i opri. Fiecare turn are caracteristici unice, cum ar fi raza de acțiune și puterea de foc, și trebuie plasat strategic pentru a maximiza efectivitatea sa. Jucătorii trebuie să ia decizii strategice importante în timpul jocului, cum ar fi ce tipuri de turnuri să construiască, când să le construiască și cum să le upgradeze, pentru a se asigura că pot gestiona cu succes amenințările inamice.

Genul de joc Tower Defense își are rădăcinile într-un mod popular de joc numit "Maze Games" care a început să fie jucat în anii 1980. În aceste jocuri, jucătorii trebuiau să-și construiască un traseu labirintic pentru a împiedica inamicii să ajungă la destinație. Aceste jocuri au fost inspirația pentru dezvoltarea jocurilor de Tower Defense, care au început să apară în anii 1990, în special în Japonia. Prin intermediul modurilor create pentru jocul *Warcraft 3: The Frozen Throne*, popularitatea acestui gen a fost sporită și acestea au deservit ca sursă de inspirație pentru viitoarele titluri care urmau să apară [2].

Punctele puternice ale acestui gen de joc sunt:

- Gameplay-ul relativ simplu și ușor de înțeles, ceea ce le face accesibile pentru toți jucătorii, indiferent de nivelul lor de experiență.
- Sesiuni scurte de joc împărțite în niveluri relativ scurte, ceea ce le face perfecte pentru a fi jucate în timpul pauzelor de la muncă, în transportul public sau în orice moment liber.
- Implică planificarea, gestionarea resurselor și luarea deciziilor strategice pentru a proteja o anumită zonă de atacurile inamicilor.

În următoarea listă avem exemple de jocuri populare și emblematice ale acestui gen de joc care au aparut de-a lung timpului:

- *Plants vs. Zombies* - În acest joc, jucătorii trebuie să își planteze flori și alte plante pentru a împiedica zombii să ajungă la casa lor. Plantele au rol de turn de apărare, iar fiecare are abilități unice. Jucătorii trebuie să le plaseze strategic pentru a opri zombii înainte ca aceștia să ajungă la locuințele lor. Jocul oferă o varietate de niveluri și moduri de joc. Acesta a fost dezvoltat de către PopCap Games și a fost lansat în anul 2009.
- *Kingdom Rush* - Acest joc are loc într-o lume de basm, unde jucătorii trebuie să își construiască turnuri defensive și să își organizeze trupele pentru a învinge hoardele de creaturi mitice. Jocul oferă o gamă largă de turnuri și trupe, fiecare cu abilități și caracteristici unice. De asemenea, acesta este unul dintre cele mai emblematice jocuri ale genului de joc Tower Defense. Jocul a fost lansat în anul 2011 și a fost dezvoltat de către Ironhide Game Studio. Au mai fost lansate și alte versiuni ale jocului, cum ar fi *Kingdom Rush: Frontiers*, *Kingdom Rush: Origins* și *Kingdom Rush: Vengeance* care s-au bucurat de un succes imens.
- *Bloons TD 6* - Jocul este centrat pe o tematică simplă în care trebuie să apară baza de baloane, jucătorii trebuie să își plaseze turnurile defensive (care au forma unor maimute) și să își upgradeze abilitățile pentru a împiedica baloanele să ajungă la final. Jocul oferă o mulțime de turnuri și abilități, precum

și niveluri diferite cu diverse provocări. Jocul a avut parte de mai multe versiuni de-a lungul anilor. Versiunea 6 este ultima lansată și care este îmbunătățită în continuare. Acest a fost dezvoltat de către Ninja Kiwi și a fost lansat în anul 2018.

- Orcs Must Die! - O combinație interesantă între jocuri de strategie și jocuri de acțiune. Jucatorul are posibilitatea să participe în mod activ la oprirea valului de inamici prin intermediul unui personaj. Jocul oferă o gamă largă de arme și abilități pentru personaje, iar jucătorii trebuie să le folosească strategic pentru a supraviețui nivelelor și a învinge inamicii. Harta de joc se aseamănă unui tunel, iar partea creativă constă în faptul cu turnurile de apărare (care apar sub forma unor capcane) pot fi poziționate pe podea, pereți și tavan. Jocul a fost lansat în anul 2011 și a fost dezvoltat de către Robot Entertainment.
- Dungeon Defenders - Acest joc combină elemente de joc de rol și Tower Defense, jucătorii trebuie să-și construiască turnurile defensive și să-și antreneze eroii pentru a învinge inamicii. Jocul oferă o gamă largă de personaje cu abilități unice, iar jucătorii trebuie să le upgradeze și să le echipeze înainte de a se aventura în nivelele următoare. Acesta prezintă o combinație interesantă între strategie și joc de rol. Jocul a fost lansat în anul 2011 și a fost dezvoltat de către Trendy Entertainment.

2. Design-ul unui joc de Tower Defense

Design unui joc de Tower Defense este relativ simplu. În general, jocurile de Tower Defense au următoarele elemente:

- O bază care trebuie apărată de atacurile inamicilor.
- O hartă cu un traseu pe care inamicii încearcă să avanseze.
- Inamici care atacă baza jucătorului.
- Turnuri defensive care trebuie plasate strategic pentru a opri inamicii înainte ca aceștia să ajungă la baza jucătorului.
- Resurse care trebuie colectate pentru a construi turnurile defensive.

Design-ul jocurilor de Tower Defense a evoluat semnificativ în ultimii ani. Iată câteva exemple de evoluție a design-ului pentru jocurile de Tower Defense:

- Varietate în tipurile de structuri defensive. Creativitatea dezvoltatorilor a fost foarte inovatoare pentru acest aspect. Multe jocuri asemănând turnurile de apărare cu alte structuri, cum ar fi: capcane, arme, aparate sau chiar personaje. Acest lucru a oferit o flexibilitate în dezvoltarea jocurilor hibride care se îmbină cu alte genuri de jocuri.
- Inamici pot fi rezistenți la anumite tipuri de atacuri sau pot avea abilități de evitare a atacurilor din partea structurilor defensive.
- Map design si-a păstrat structura de bază, schimbările au fost mai pronunțate în ceea ce privește tematica și complexitatea traseului pentru inamici.
- Economia de joc a fost extinsă, introducerea mai multor tipuri de resurse și a unor mecanisme de colectare mai complexe au avut un impact pozitiv în ceea ce privește partea strategică de gestionare a resurselor. Mulți dezvoltatori folosindu-se de acesta pentru a integra elementele din jocurile de tip “resource management”.

2.1. Modele hibride de jocuri

Jocurile de Tower Defense au evoluat și au început să se îmbine cu alte genuri de jocuri, ceea ce a dus la apariția unor noi genuri de jocuri. Iată câteva exemple de modele hibride:

- Cu elemente de joc de rol și acțiune. Acestea aduc în prim-plan partea de poveste și acțiune continuă pentru a crea o experiență de joc mai interesantă. Ele se concentrează pe crearea unui univers care să-l captiveze pe jucător. Jocurile de acest tip oferă o varietate de eroi cu abilități și caracteristici unice care se îmbină cu tematica elementară a genului Tower Defense.
- Cu elemente de joc de puzzle. Acest gen se pliază foarte ușor pe genul Tower Defense, iar introducerea de mici schimbări în mecanica de joc poate duce la crearea unui joc de acest tip. Jocurile de acest tip se concentrează pe crearea unui puzzle care să fie rezolvat de către jucător. Acest puzzle poate consta în găsirea unei anumite combinații de structuri defensive care să oprească valul inamic în anumite condiții. Unele jocuri, introduc acest concept sub formă de “challenge mode” care poate fi jucat de către jucători după ce au terminat jocul. Acesta consta în adăugarea de noi constrângeri pentru jucător, cum ar fi: limitarea numărului de turnuri defensive, limitarea numărului de resurse, creșterea numărului de inamici, etc.

2.2. O nouă metodă de extindere a mecanicii de joc

Având în vedere toate variantele prin care a evoluat genul Tower Defense, încă există arii care nu au fost explorate. Unul dintre acestea este crearea unui sistem colaborativ între turnurile de apărare.

În mare majoritate a jocurilor create, turnurile de apărare funcționează independent, fiecare având propriile sale abilități și caracteristici. Unele jocuri au introdus mici schimbări, prin care turnurile de apărare pot fi îmbunătățite, iar unele din îmbunătățiri fiind în a oferi turnurilor vecine mici bonusuri.

Pentru a dezvolta această arie, explorăm următoarea idee de mecanică de joc: Fiecare turn de apărare poate crea și consuma un **jeton de acțiune** de pe inamicul din raza de apărare a turnului. Un jeton de acțiune reprezintă o acțiune care poate fi efectuată de unele turnuri de apărare și acesta este purtat de către inamicii. Un turn de apărare poate crea un jeton de acțiune care poate fi consumat de un alt turn de apărare prin intermediul inamicilor.

În mod practic, noi construim un mod de comunicare între turnuri, acestea având un comportament diferit asupra unui inamic în funcție de tipurile și numărul de jetoane de acțiune pe care le au. Un turn poate crea un jeton de acțiune prin interceptarea unui inamic în raza sa, iar apoi acel jeton poate fi consumat de un alt turn pentru a efectua o acțiune specifică. De exemplu, un turn poate crea un jeton de acțiune care să activeze un atac special. Cu cât un turn are mai multe jetoane de acțiune, cu atât importanța sa este mai mare.

Harta joc are un rol crucial în design, ea este cea care de cele mai multe ori oferă unicitate unui sesiuni de joc. În multe jocuri clasice, dimensiunea mărită a hărții de joc nu oferă prea multe oportunități deoarece de multe ori jucătorul aplică aceeași combinație turnuri, fapt care crează redundanță. Trecând către acest sistem colaborativ, putem sesiza un nou beneficiu, și anume: hărțile de dimensiune mai mare ne oferă mai mult spațiu pentru a crea mai multe turnuri defensive, fapt care ne permite să creăm mai multe interacțiuni între acestea.

2.3. Compunerea jetoanelor de acțiune

Flexibilitatea jetoanelor de acțiune vine din faptul că acestea pot fi compuse din mai multe tipuri de jetoane. Presupunem că avem următoarele tipuri de jetoane de bază:

- Jeton de bonus atac: inamicii primesc daune bonus de la proiectilele turnurilor.
- Jeton de încetinire: inamicii au viteză de mișcare redusă.
- Jeton de explozie: proiectilele turnurilor au un efect de explozie asupra inamicului și vecinilor săi.

Rangul unui jeton reprezintă numărul de jetoane de același tip pe care un inamic le deține, fiecare jeton are prestabilit un rang maxim. Un jeton de rang mai mare are un efect mai pronunțat decât unul de rang mai mic. Jetoanele pot fi combinate pentru a obține noi tipuri. De exemplu, putem avea următoarele tipuri de jetoane compuse:

- Jeton de înghețare: inamicii vor fi înghețați (viteză de mișcare 0) atâta timp cât jetonul este activ. Format dintr-un jeton de încetinire de rang 2.
- Jeton de daune de-a lungul timpului: inamicii care dețin acest jeton vor primi daune atâta timp cât jetonul este activ. Format dintr-un jeton de bonus atac de rang 3.
- Jeton de explozie la eliminare: inamicii care dețin acest jeton vor crea o explozie când sunt eliminați. Format dintr-un jeton de explozie de rang 2.
- Jeton de explozie la încetinire: inamicii care dețin acest jeton vor crea o explozie când sunt încetiniți. Format dintr-un jeton de explozie de rang 2 și un jeton de încetinire de rang 2.
- Jeton de explozie pulsantă: inamicii care dețin acest jeton vor crea o explozie în jurul lor la fiecare 3 secunde. Format dintr-un jeton de explozie de rang 2 și un jeton de bonus atac de rang 3.

Putem observa marele avantaj al acestui sistem de jetoane de acțiune, și anume: flexibilitatea. Acest sistem ne permite să creăm o varietate de jetoane de acțiune, iar acestea pot fi combinate pentru a crea noi tipuri de jetoane.

Așadar, un joc care urmează acest design poate fi extins foarte ușor fără schimbări majore în mecanica de joc. Această flexibilitate poate fi observată în jocurile de tip cărți de joc precum: *Hearthstone* sau *Magic: The Gathering*. Aceste jocuri au o mecanică de bază simplă, însă prin intermediul extensiilor, acestea pot fi extinse cu noi tipuri de cărți care aduc varietate.

2.4. Tipurile de turnuri

Pentru sistemul de jetoane de acțiune trebuie să avem o gamă largă de turnuri care să fie capabile să creeze și să consume jetoanele de acțiune. Vom împărți turnurile în două categorii: *active* și *pasive*.

Proiectilele pot fi create doar de către turnurile active și acestea reprezintă principalul mod a elimina inamicii. Dar turnurile active nu pot crea jetoane de acțiune, ele pot doar consuma jetoanele de acțiune pentru a-și îmbunătăți atacul. Turnurile pasive nu pot crea proiectile, însă acestea pot crea jetoane de acțiune care pot fi consumate de către turnurile active sau de alte turnuri pasive.

Această relație de interdependență între turnuri este un aspect important în design-ul jocului, el fiind forma de colaborare care este creată de acest sistem de jetoane.

În acest sens, putem avea următoarele tipuri de turnuri pasive:

- Turn pasiv de atac bonus: crează un jeton de bonus atac pentru fiecare inamic din raza sa de acțiune.
- Turn pasiv de încetinire: crează un jeton de încetinire pentru fiecare inamic din raza sa de acțiune.
- Turn pasiv de înghețare: crează un jeton de înghețare pentru fiecare inamic din raza sa de acțiune care are un jeton de încetinire de rang 2.
- Turn pasiv de explozie pulsantă: crează un jeton de explozie pulsantă pentru fiecare inamic din raza sa de acțiune care are un jeton de explozie de rang 2.

Turnurile active se aseamănă cu turnurile clasice de apărare, acestea având rolul de a elimina inamicii. Dacă turnurile pasive variază prin tipul de jeton pe care îl produc, turnurile active se vor diferenția prin modelul de proiectil create și rata de atac. Iată câteva exemple de turnuri active:

- Turn activ cu proiectil simplu: acesta crează un proiectil simplu care aplică pagube primului inamic cu care intră în contact. Acesta atacă la o rată medie și aplică pagube medii.
- Turn activ de tip mortar: acesta crează un proiectil care explodează la impact și aplică pagube tuturor inamicilor din raza de acțiune a exploziei. Proiectilul explodează când ajunge la destinație, acesta ignorând inamicii din cale.
- Turn activ cu atac rapid: acesta crează un proiectil simplu care aplică pagube primului inamic cu care intră în contact. Acesta are o rată de atac ridicată, dar care provoacă pagube mici.
- Turn activ cu proiectil inteligent: acesta crează un proiectil simplu care urmărește inamicul cel mai apropiat din raza sa de acțiune.

Având în vedere exemple de mai sus, putem observa diferite cum fiecare turn îl completează pe celălalt:

- Pentru ca turnul activ să fie eficient, ar avea nevoie de niște turnuri pasive care să încetinească inamicii.
- Turnul de tip mortar ar fi mai bun dacă inamicii ar fi mult mai încetiniți astfel încât aceștia să fie mai grupați astfel încât explozia să fie mai eficientă.
- Turnul cu proiectil inteligent nu ar avea probleme cu țintirea inamicilor așa că ar beneficia mai mult dacă inamicii ar avea un jeton de explozie pulsantă care să le reducă viața cât mai repede.

2.5. Valul de inamici

Scopul unui inamic este să ajungă la obiectivul pe care jucătorul încearcă să-l protejeze. Pentru a ajunge la destinație, acesta trebuie să supraviețuiască atacurilor turnurilor de apărare. Inamicii au o viață și o viteză de mișcare. Viața reprezintă numărul de pagube pe care un inamic le poate suporta înainte de a fi eliminat. Viteza de mișcare reprezintă cât de repede se deplasează inamicul pe traseu.

Ca un inamic să ajungă la obiectiv, viața lui trebuie să fie mai mare decât pagubele pe care le pot produce turnurile de apărare de-a lungul traseului. Sau, poate fi mai mică, însă viteza de mișcare trebuie să fie mai mare decât viteza proiectilelor. Sunt multe moduri în care putem varia caracteristicile sale.

Proiectarea corectă a inamicilor este un aspect important în design-ul jocului. Inamicii trebuie să fie echilibrați astfel încât să ofere o provocare jucătorului și să fie în armonie cu sistemul de jetoane. Pentru inamicii nu avem un anumit tip ci un set de recomandări de design de care trebuie ținut cont:

- Un inamic nu poate fi eliminat doar prin intermediul turnurilor active – ne dorim să existe o colaborare între turnuri, așadar turnurile pasive trebuie să aibă și ele o contribuție.
- Caracteristicile inamicului (precum: viață, viteză de mișcare) trebuie să fie în concordanță cu evoluția jocului. Inamicii devin mai puternici pe măsură ce jocul avansează.
- Generarea valului de inamici trebuie să fie consistent. În loc să generăm aleatoriu pozițiile de start al inamicilor, putem folosi un algoritm de generare care să producă un traseu de la punctul de start la punctul final. Acest lucru ne permite să controlăm mai bine dificultatea jocului.

Unele jocuri, introduc noi mecanici de joc pentru inamicii, precum:

- Armură: atacurile de un anumit tip au un efect redus asupra inamicilor care au armură (exemplu: -50% pagube primite de la proiectil). Aceasta, poate să introducă la rândul său noțiune de *tipuri de atac* (exemplu: atac de foc, atac magic) unde fiecare tip de armură acționează diferit.
- Regenerare: inamicii își pot regenera viața în timpul jocului.
- Abilități speciale: inamicii care oferă un bonus altor inamici din jurul lor (exemplu: +50% viață pentru inamicii din jurul său) sau care produc o acțiune care le oferă avantaj (exemplu: crearea de noi inamici de-a lungul traseului).

Sistemul de jetoane nu prezintă niciun impediment în implementarea acestor mecanici de joc. Chiar putem crea jetoane de acțiune care să contracareze inamicii care prezintă aceste mecanici de joc. De exemplu, putem avea un jeton de acțiune care să reducă armura inamicilor sau un jeton de acțiune care să reducă regenerarea inamicilor.

2.6. Economia de joc

Într-un joc de tip *Tower Defense*, turnurile de apărare sunt construite și îmbunătățite prin intermediul resurselor. Aceste resurse pot fi colectate de către jucător prin intermediul unor structuri speciale sau pot fi obținute prin eliminarea inamicilor. Acest sistem reprezintă economia de joc.

Acest sistem joacă un rol important în partea strategică a jocului. Jucătorul trebuie să ia decizii strategice în ceea ce privește gestionarea resurselor. De exemplu, jucătorul poate alege să construiască mai multe turnuri de apărare de la începutul jocului, sau poate alege să construiască mai puține turnuri de apărare și să-și îmbunătățească structurile existente.

De regulă, pentru a stabili costul resurselor pentru turnurile de apărare, se pot folosi următoarele întrebări:

- Câți inamici trebuie să eliminăm pentru a obține resursele necesare pentru a construi un turn de apărare?

- Câți inamicii trebuie să eliminăm pentru a obține resursele necesare pentru a îmbunătăți un turn de apărare?
- Câte resurse se pot obține în total la eliminarea unui val de inamici?
- Ce turnuri putem construi cu resursele obținute după ce eliminăm X valuri de inamici?
- Care este performanța unui turn în raport cu costul său?

Un alt mod în care poate fi folosit acest sistem este cel în generarea de valuri inamice. În loc ca tipurile de inamici să fie prestabilite, acestea pot fi alese în funcție de valoarea valului de inamici. Fiecare inamic având o valoare, inamicii pot fi aleși aleatoriu până ajungem la valoarea totală de resurse pe care dorim s-o avem pentru valul respectiv.

3. Implementarea sistemelor

O parte importantă al oricărei idei de joc este implementarea acesteia. În capitolele următoare vom descrie implementarea sistemelor principale din care va fi compus jocul care se folosește de sistemul de jetoane de acțiune.

3.1. Primele minute de joc

Înainte de a intra în detalii despre implementarea sistemului de jetoane de acțiune, vom descrie primele minute de joc. Acest lucru ne va ajuta să înțelegem mai bine cum funcționează jocul și cum trebuie jucătorul să interacționeze cu acesta.

Prima dată când pornim jocul vom vedea meniul principal. În acest meniu avem următoarele opțiuni:

- *New Game*: începe un joc nou.
- *Load Game*: încarcă un joc salvat.
- *Options*: setările jocului.
- *Exit*: închide jocul.

Când apăsăm pe butonul *New Game* se va încărca scena de joc. În această scenă avem următoarele elemente:

- Harta de joc: reprezintă zona de joc, aici se vor desfășura toate acțiunile jocului.
- Interfața pentru magazin: reprezintă meniul de unde putem cumpăra turnuri de apărare și putem vedea informații despre acestea.
- Interfața pentru statusul jucătorului: elemente vizuale care arată informații despre resursele acumulate, viața obiectivului care trebuie protejat, numărul valului de inamicii, timpul rămas până la următorul val de inamicii, etc.
- Interfața pentru statusul inamicilor: elemente vizuale care arată informații despre inamicii care se află pe hartă, cum ar fi: viața, tipul de jetoane de acțiune deținute, armura, abilități, etc.
- Turnurile de apărare: elemente vizuale care reprezintă turnurile de apărare care au fost construite pe hartă și elementele conexe acestora, cum ar fi: raza de acțiune, proiectilele, etc.
- Inamicii: elemente vizuale care reprezintă inamicii care se află pe hartă.

Înainte să apăsăm pe butonul de start al sesiunii, trebuie să amplasăm primele turnuri de apărare. Avem mai multe opțiuni de plasare, de exemplu:

1. Începem cu un turn activ care are un proiectil simplu și o rată de atac medie. Iar ca să ne asigurăm că acesta va fi eficient, vom folosi turnurile pasive de încetinire și îngheț. În acest fel, vom încetini

inamicii și vom îngheța inamicii care au un jeton de încetinire de rang 2. Acest lucru ne va permite să ne asigurăm că proiectilele turnului activ vor lovi inamicii înghețați.

2. Începem cu un turn activ cu rată mare atac și proiectil rapid. Ca să-l folosim la potențialul său maxim, vom folosi turnurile pasive de atac bonus și încetinire. Ne vom asigura că inamicii vor fi încetiniți înainte de a intra în raza de acțiune a turnului de atac bonus, astfel încât inamicul să ajungă să aibă un jeton de atac bonus de rang înalt. Astfel, proiectilele turnului activ vor avea un efect mai mare asupra inamicilor.

După ce am plasat primele turnuri de apărare, putem apăsa pe butonul de start al sesiunii. În acest moment, primul val de inamici va fi generat și va începe să se deplaseze pe traseu. În acest moment, putem observa dacă amplasarea turnurilor a fost corectă sau nu. Dacă inamicii sunt eliminați înainte de a ajunge la obiectiv, este un semn că am făcut decizia corectă. Dacă inamicii ajung la obiectiv, atunci trebuie să ne gândim la o altă strategie.

Odată terminat valul, e timpul să ne folosim de resursele acumulate pentru a achiziționa noi turnuri de apărare sau pentru a îmbunătăți turnurile existente. Cu fiecare val care trece inamicii devin mai puternici și numărul lor crește, iar noi trebuie să ne adaptăm strategia de joc pentru a face față provocărilor care apar.

În funcție de ce turn am ales la început, încercăm să maximăm potențialul acestuia, dar în același timp să ne gândim cum putem combina turnurile pasive pentru a obține jetoane mai puternice. Dacă am încerca să investim în turnurile pasive de atac bonus și exploziv, ne-ar putea ajuta cu valurile care au mulți inamici cu viață mică. Dacă am încerca să investim în turnurile pasive de încetinire și îngheț, ne-ar putea ajuta cu valurile care au inamici cu viață mare.

3.2. Sistemul de jetoane de acțiune

Acest sistem se ocupă de gestionarea jetoanelor de acțiune. Un jeton de acțiune reprezintă o acțiune care poate fi efectuată de unele turnuri de apărare și acesta este purtat de către inamicii. Un turn de apărare poate crea un jeton de acțiune care poate fi consumat de un alt turn de apărare prin intermediul inamicilor.

Propunem următoare structură pentru definirea unui jeton de acțiune:

- Tip efect: reprezintă tipul de efect pe care îl produce jetonul de acțiune.
- Valoare efect: reprezintă intensitatea efectului.
- Rang maxim: reprezintă rangul maxim pe care îl poate avea jetonul de acțiune.
- Rang curent: reprezintă rangul curent pe care îl are jetonul de acțiune.
- Durată: reprezintă durata de timp pentru care jetonul de acțiune este activ.
- Durata curentă: reprezintă durata de timp rămasă pentru care jetonul de acțiune este activ.
- Condiții de creare: reprezintă condițiile care trebuie îndeplinite pentru a crea un jeton de acțiune.

Tipul de efect reprezintă acțiune care se va produce atunci când jetonul de acțiune este consumat. Aceste acțiuni reprezintă funcționalități ale mecanicii de joc. Iată câteva exemple de tipuri de efecte:

- Scăderea parțială sau completă a vitezei de mișcare al inamicilor.
- Scăderea parțială sau completă a vieții inamicilor.
- Creșterea pagubelor primite de inamici din partea proiectilelor turnurilor.

Valoarea efectului reprezintă intensitatea efectului. Aceasta poate fi un număr întreg sau un procentaj. De exemplu, un jeton de acțiune care scade viteza de mișcare a inamicilor cu 50% are o valoare efect de 50%. Acest valori fie pot fi constante sau calculate prin intermediul unor funcții sau formule matematice. Exemplu:

$$v = \frac{r}{10}$$

, unde $r \in [0, 10]$ reprezintă rangul jetonului de acțiune și $v \in [0, 1]$ reprezintă valoarea efectului.

Rangul maxim reprezintă numărul maxim de jetoane de acțiune de același tip pe care un inamic le poate deține. Un jeton de acțiune de rang mai mare are un efect mai pronunțat decât unul de rang mai mic. Când un inamic este expus de mai mult ori atacului unui turn de apărare, rangul jetonului de acțiune asociat cu turnul de apărare crește.

Rangul curent reprezintă rangul curent pe care îl are jetonul de acțiune. Acesta poate fi mai mic sau egal cu rangul maxim. Rangul curent crește atunci când un inamic este expus de mai mult ori atacului unui turn de apărare asociat cu jetonul respectiv.

Durata reprezintă durata de timp pentru care jetonul de acțiune este activ. Aceasta poate fi un număr real. De exemplu, un jeton de acțiune care scade viteza de mișcare a inamicilor cu 50% pentru 5 secunde are o durată de 5 secunde. Această valoare poate fi constantă sau calculată prin intermediul unor funcții sau formule matematice.

Durata curentă reprezintă durata de timp rămasă pentru care jetonul de acțiune este activ. Aceasta este un număr real care este actualizat la un interval de timp dat. Când durata curentă ajunge la 0, jetonul de acțiune este eliminat.

Condiții de creare reprezintă condițiile care trebuie îndeplinite pentru a crea un jeton de acțiune. Aceste condiții iau considerare următoarele aspecte: tipul de efect și rangul său curent. De exemplu, un jeton de înghețare poate fi creat doar dacă inamicul are un jeton de încetinire de rang 2.

Cum acest sistem trebuie să aibă o **implementare concretă într-un limbaj de programare** pentru a fi integrat într-un joc, vom folosi limbajul de programare Rust[3] pentru a descrie o posibilă implementare. Exemplele pot fi prezentate și sub formă de pseudocod, dar realizarea lor cu un limbaj de programare face ca totul să fie mai tangibil și chiar să facă parte din implementarea jocului.

Un exemplu concret de structură pentru un jeton de acțiune este următoarea:

```
struct Token {
    effect_type: EffectType,
    effect_value: float,
    max_rank: int,
    current_rank: int,
    duration: float,
    current_duration: float,
    creation_conditions: List<Condition>
}
```

Pentru tipul de efect putem avea un simplu *enum* care să conțină toate tipurile de efecte pe care le putem avea:

```
enum EffectType {
    Slow,
    Freeze,
    BonusAttack,
    Explosion
}
```

, iar pentru condiția de creare avem următoarea structură:

```
struct Condition {
    effect_type: EffectType,
    min_rank: int
}
```

3.3. Sistemul de turnuri de apărare

Acest sistem se ocupă de gestionarea turnurilor de apărare. Un turn de apărare reprezintă o structură care exercită o acțiune asupra inamicilor care duce în mod direct sau indirect la eliminarea acestora. În acest sistem, vom avea două tipuri de turnuri de apărare: turnuri active și turnuri pasive.

Turnurile active au următoarele caracteristici:

- Produc proiectile care pot elimina inamicii.
- Au o rată de atac proprie.
- Au sistem de țintire.
- Au rază de atac.

Proiectilele fac parte din mecanismul turnurilor active, dar cum pot varia de la un turn la altul, vom avea o structură separată pentru acestea cu următoarele caracteristici:

- Viteză de mișcare: reprezintă viteza de deplasare a proiectilului.
- Pagube: reprezintă pagubele pe care le produce proiectilul.
- Durată de viață: reprezintă durata de timp pentru care proiectilul este activ.
- Durată de viață curentă: reprezintă durata de timp rămasă pentru care proiectilul este activ.

Turnurile pasive au următoarele caracteristici:

- Produc jetoane de acțiune care pot fi consumate de alte turnuri.
- Au rază de influență.
- Jetoanele se aplică la un interval de timp dat tuturor inamicilor din rază de influență.

Codul structurii pentru un turn de apărare activ și proiectil este următorul:

```
struct ActiveTower {
    projectile: Projectile,
    attack_rate: float,
    attack_rate_timer: float,
    range: float,
}

struct Projectile {
    speed: float,
    damage: float,
    life_time: float,
    current_life_time: float,
}
```

Pentru turnurile pasive avem următoarea structură:

```
struct PassiveTower {
    token: Token,
    range: float,
    token_rate: float,
    token_rate_timer: float,
}
```

Acest sistem este simplu de implementat, cea mai complicată parte fiind partea de țintire a turnurilor active. În mod ideal, am dori să trimitem către inamic un număr minim de proiectile care să-l elimine. Astfel, turnul devine mai eficient prin faptul că nu pierde timp pentru a elimina un inamic care oricum va fi eliminat de către alt turn sau de proiectilele create precedent.

O exemplu de algoritm de țintire ar fi următorul:

1. Alegem cel mai apropiat inamic din raza de acțiune.

2. Calculăm câte proiectile sunt necesare pentru a-l elimina folosind punctele de viață și pagubele proiectilului.
3. Calculăm pozițiile inamicilor la momentul când aceștia vor fi loviți de proiectile.
4. Verificăm dacă la pozițiile calculate vor lovi și proiectilele de la alte turnuri.
5. Dacă există asemenea caz, recalculăm numărul de proiectile ținând cont de punctele de viață rămase după ce inamicul este lovit de proiectilele de la alte turnuri.

O problemă care acest algoritm nu o ia în considerare este efectul provocat de jetoanele de acțiune asupra inamicilor. Și anume că inamicul nu va avea o viteză de mișcare constantă pe traseu, iar acesta poate primi pagube de la inamicii învecinații care au asupra lor jetoane de explozie care îl pot elimina înainte ca proiectilele să ajungă la el. Acest scenariu este prea complex pentru un simplu joc dar reprezintă un caz interesent de cercetare.

Pentru turnurile pasive, nu e nevoie de niciun algoritm de țintire, acestea având un efect asupra tuturor inamicilor din rază de influență. Există un timer care la un interval de timp dat, toate turnurile pasive aplică jetoanele de acțiune.

3.4. Sistemul de economie de joc

Acest sistem se ocupă de gestionarea resurselor. Resursele sunt folosite pentru a construi turnuri de apărare și pentru a le îmbunătăți. Resursele pot fi obținute prin intermediul unor structuri speciale sau pot fi obținute prin eliminarea inamicilor.

În jocuri, acestea au denumiri generice precum: *gold* (aur), *coins* (monede), *gems* (nestemate), etc. Numărul lor variază în funcție de joc, iar unele jocuri pun mai mare accent pe gestionarea resurselor decât altele.

În principal, acest sistem nu are un impact prea mare asupra mecanicii de joc la un joc Tower Defense clasic, dar prin combinarea mai multor genuri de jocuri, se poate crea ceva unic. De exemplu, în jocuri populare care se combină cu *Role Play Game* (RPG), această parte devine fundamentală pentru joc. Un astfel de joc este *Arknights*[4] care combină genul Tower Defense cu RPG, unde jucătorul trebuie să colecteze o varietate de resurse pentru a avansa în joc. Și cum este un proces încet, se pot folosi bani reali pentru a cumpăra resurse – aceasta fiind și principala sursă de monetizare a jocului.

Un tip de resursă poate fi definit astfel:

```
enum ResourceType {  
    Gold,  
    Wood,  
    Stone,  
}
```

Atunci când eliminăm un inamic, primim o recompensă sub formă de resurse. Această recompensă compusă din tipul de resursă și suma. De exemplu, un inamic poate să ofere 10 aur, 5 lemn și 2 piatră când este eliminat. O structură pentru recompensă poate fi definită astfel:

```
struct Reward {  
    resource_type: ResourceType,  
    amount: float,  
}
```

3.5. Sistemul de inamicii

Inamicii reprezintă entitățile care trebuie să ajungă la obiectivul pe care jucătorul încearcă să-l protejeze. Pentru a ajunge la destinație, acesta trebuie să supraviețuiască atacurilor turnurilor de apărare. Inamicii au o viață și o viteză de mișcare. Viața reprezintă numărul de pagube pe care un inamic le poate suporta înainte de a fi eliminat. Viteza de mișcare reprezintă cât de repede se deplasează inamicul pe traseu.

Un inamic are următoarele caracteristici:

- Viață: reprezintă numărul de pagube pe care le poate suporta inamicul înainte de a fi eliminat.
- Viteză de mișcare: reprezintă cât de repede se deplasează inamicul pe traseu.
- Jetoane de acțiune: reprezintă jetoanele de acțiune pe care le deține inamicul.
- Recompensă: reprezintă recompensa pe care o oferă inamicul când este eliminat.

În baza acestor caracteristici putem avea următoarele tipuri de inamicii:

- Inamic rapid: acesta are o viață mică și o viteză de mișcare mare.
- Inamic normal: acesta are o viață medie și o viteză de mișcare medie.
- Inamic rezistent: acesta are o viață mare și o viteză de mișcare mică.

La primă vedere pare să nu avem prea multe opțiuni de variație pentru inamicii. Unele jocuri de Tower Defense încearcă să crească acest număr prin introducerea unor noi mecanici de joc pentru inamicii, precum:

- Rezistență la anumite tipuri de atac. Turnurile de apărare pot avea diferite tipuri de atac, iar inamicii pot avea rezistență la un anumit tip de atac. De exemplu, un inamic poate avea rezistență la atacurile de foc, iar turnurile de apărare pot avea atac de foc.
- Regenerare viață.
- Imunitate la anumite efecte date de către turnuri.
- Abilități speciale. Exemplu: poate deveni invulnerabil pentru o perioadă de timp; poate crește viteze de mișcare a inamicilor din jurul său;

Structura unui inamic poate fi definită astfel:

```
struct Enemy {  
    health: float,  
    speed: float,  
    tokens: List<Token>,  
    reward: Reward,  
}
```

3.6. Interfața de utilizator

Bibliografie

- [1] A. Hernández-Sabaté, M. Joanpere, N. Gorgorió, and L. Albarracín, “Mathematics learning opportunities when playing a tower defense game,” *Int. J. Serious Games*, vol. 2, no. 4, pp. 57–71, 2015.
- [2] P. Avery, J. Togelius, E. Alistar, and R. P. Van Leeuwen, “Computational intelligence and tower defence games,” in *2011 IEEE Congr. Evol. Computation (Cec)*, 2011, pp. 1084–1091.
- [3] “Rust.” [Online]. Available: <https://www.rust-lang.org/>
- [4] “Arknights wiki.” [Online]. Available: https://arknights.fandom.com/wiki/Arknights_Wiki