

# One Crew

## First Author

Ghinea Theodor Traian

## Second Author

Ocnaru Mihai Octavian

## Third Author

Accente Robert

### Abstract

Proiectul constă într-un prototip de joc care simulează experiența piratilor în bătălie, încercând să redea lumea lor cât mai fidel, dar într-o manieră plăcută.

## 1 Introducere

Problema abordată este proiectarea și implementarea unui sistem de inteligență artificială pentru personaje non-jucător (NPC), în special pirati, care să se comporte într-un mod realist, percepând mediul, luând decizii și acționând în funcție de context. Scopul este de a simula un comportament convingător și dinamic al acestora, folosind concepte precum mașini de stări finite și memorii contextuale.

De asemenea, un alt aspect important a fost realizarea player-ului și a tuturor animațiilor. De aceea sunt incluse reacții fizice prin animații blend-uite, care simulează impactul unui asupra corpului în momentul loviturii.

Această problemă a fost aleasă deoarece combină mai multe concepte de inteligență artificială studiate în cadrul cursului, cum ar fi comportamentele bazate pe stări, percepția mediului și simularea unor interacțiuni fizice simple. În plus, oferă o oportunitate practică de a implementa un sistem în care agenții virtuali reacționează în mod credibil la acțiunile jucătorului și la schimbările din mediu. Este, de asemenea, un mod plăcut și vizual de a aplica teoria în practică.

## 2 Lucrări conexe

Inteligența artificială (AI) aplicată în jocuri video a evoluat semnificativ în ultimele decenii, de la comportamente scriptate rigide la sisteme complexe, reactive și adaptabile. Inițial, majoritatea AI-urilor foloseau sisteme simple bazate pe reguli sau scripturi liniare. Odată cu creșterea puterii de procesare și dezvoltarea motoarelor grafice moderne (precum Unreal Engine și Unity), a devenit

posibilă implementarea unor arhitecturi modulare mai sofisticate, cum ar fi mașinile de stări finite (FSM), arborii comportamentali (behavior trees) și sistemele de percepție (line of sight, raycast, stimuli auditivi etc.).

Tendințele recente în AI pentru jocuri pun accent pe realism, emergență și modularitate. Sistemele moderne includ memorii contextuale, planificatori de acțiuni (HTN – Hierarchical Task Networks), comportamente învățate (inclusiv machine learning în unele cazuri) și reacții fizice bazate pe sistemul de animații și fizica din motor.

Un exemplu notabil de joc care folosește un astfel de sistem este Sea of Thieves, care se axează pe explorare nautică, luptă navală și interacțiuni între echipaje de pirati. Deși jocul este orientat în principal spre multiplayer, NPC-urile inamice (cum ar fi scheleții sau piratii fantomă) prezintă un comportament reactiv și adaptiv, jocul presupune și alte mecanici precum inundarea bărcii, sistemul de damage al bărcii și tragerea cu tunul. Totodată, stilul artistic și tematica marină au constituit o sursă de inspirație directă pentru proiectul de față.

## 3 Method

În dezvoltarea acestui joc de acțiune, sistemul de inteligență artificială al inamicilor reprezintă coloana vertebrală a experienței de gameplay. Arhitectura AI nu este doar o colecție de funcții și variabile, ci un ecosistem complex care modelează comportamente credibile, adaptative și care oferă jucătorului provocări variate și memorabile.

### 3.1 Arhitectura Modelului AI

Arhitectura inteligenței artificiale pentru inamici este concepută ca un organism viu, capabil să perceapă, să decidă și să acționeze într-un mod care imită comportamentul natural al unui adversar inteligent. La baza acestui sistem stă o triadă conceptuală: percepție, decizie și acțiune.

### 3.1.1 Filozofia Sistemului de Decizie

În centrul arhitecturii AI se află arborele de comportament (*Behavior Tree*), o structură care nu doar organizează deciziile, ci modelează psihologia inamicului. Spre deosebire de o simplă mașină de stări finite, arborele de comportament permite o ierarhizare naturală a priorităților — similar modului în care o ființă vie își prioritizează nevoile și răspunsurile.

Imaginați-vă un paznic medieval care patrulează un castel. În mod normal, acesta urmează o rutină prestabilită, verificând punctele cheie ale fortificației. Însă în momentul în care observă un intrus, prioritățile sale se schimbă instantaneu — abandonează patrularea rutinată și se concentrează pe noua amenințare. Exact acest tip de comportament adaptativ este modelat prin sistemul nostru de AI.

### 3.1.2 Memoria Contextuală - Sistemul Blackboard

Blackboard-ul funcționează ca o memorie de lucru pentru AI, similar modului în care creierul uman păstrează informații temporare relevante pentru situația curentă. Acest sistem nu stochează doar date brute, ci contextul și semnificația acestora:

- **Starea cognitivă** - Nu doar poziția inamicului, ci și interpretarea sa asupra situației (este alertat? urmărește pe cineva? se simte amenințat?)
- **Memoria amenințărilor** - Păstrează informații despre ultima poziție cunoscută a jucătorului, chiar și atunci când acesta dispare din câmpul vizual
- **Intențiile comportamentale** - Planurile pe termen scurt ale inamicului, permițând continuitatea acțiunilor chiar și atunci când circumstanțele se schimbă

Această arhitectură permite inamicilor să manifeste “persistență cognitivă” — ei nu uită instantaneu prezența jucătorului doar pentru că acesta s-a ascuns temporar, creând astfel o experiență mai realistă și provocatoare.

### 3.1.3 Straturile de Percepție

Sistemul de detecție al inamicilor este stratificat, imitând complexitatea percepției biologice:

**Primul strat - Vigilența pasivă:** Inamicul menține o stare constantă de alertă minimă,

scanând mediul pentru anomalii. Acest strat consumă resurse minime de calcul, similar modului în care atenția periferică umană funcționează fără efort conștient.

**Al doilea strat - Focalizarea activă:** Când o anomalie este detectată (sunet, mișcare, prezență), sistemul AI alocă mai multe resurse pentru investigație. Inamicul își modifică comportamentul, devenind mai atent și mai reactiv.

**Al treilea strat - Angajamentul total:** Odată ce amenințarea este confirmată, toate resursele cognitive ale inamicului sunt redirectionate către gestionarea conflictului. Patrularea este abandonată, viteza de mișcare crește, și toate deciziile sunt luate în contextul confruntării.

### 3.1.4 Dinamica Tranziției între Stări

Trecerea între diferitele stări comportamentale nu este abruptă, ci graduală și contextuală. Când un inamic pierde din vedere jucătorul, nu revine instantaneu la patrulare, ci intră într-o stare de “căutare activă”, verificând ultimele poziții cunoscute și zonele adiacente. Această persistență comportamentală creează momente de tensiune autentică, unde jucătorul știe că este căutat activ.

Sistemul implementează conceptul de “memorie comportamentală” — inamicii își amintesc interacțiunile recente și își ajustează comportamentul în consecință. Un inamic care a fost atacat recent va fi mai vigilent, va patrula mai alert și va reacționa mai rapid la stimuli suspecti.

### 3.1.5 Optimizarea Inteligentă a Resurselor

Pentru a menține performanța optimă într-un mediu 3D complex, sistemul AI utilizează o abordare de “inteligentă la cerere”. Calculele costisitoare, precum verificările de vizibilitate sau planificarea rutelor complexe, sunt efectuate doar când sunt absolut necesare.

Această optimizare este inspirată din principiile cognitive umane — nu procesăm constant toate informațiile disponibile, ci ne concentrăm pe ceea ce este relevant pentru situația curentă. Inamicii aflați la distanță mare de jucător funcționează într-un mod simplificat, economisind resurse pentru momentele când interacțiunea devine iminentă.

### 3.1.6 Extensibilitate și Modularitate

Arhitectura este construită cu gândul la evoluție. Fiecare componentă — fie că este un nod în arborele de comportament, o variabilă în Blackboard sau un eveniment — poate fi modificată, în-

locuită sau extinsă fără a perturba sistemul global. Această modularitate permite:

- Adăugarea de noi tipuri de inamici cu comportamente unice
- Ajustarea dificultății prin modificarea parametrilor de percepție și reacție
- Implementarea de tactici de grup pentru inamici care cooperează
- Integrarea de noi sisteme senzoriale (auz, detectarea urmelor, etc.)

Prin această abordare holistică, sistemul AI transcende simpla programare și devine un element narativ activ, contribuind la crearea unei lumi vii și reactive care răspunde organic la acțiunile jucătorului.

### 3.2 Animații

Animațiile au un rol esențial în construirea unei experiențe vizuale fluide și expresive în jocuri. În cadrul acestui proiect, animațiile sunt utilizate atât pentru elemente decorative ale mediului, cât și pentru entitățile active din joc.

Pe marginile insulei se află tentacule uriașe animate, care contribuie la atmosfera misterioasă a jocului. Acestea folosesc o animație preluată dintr-un pachet de asseturi și sunt integrate pasiv în decor, mișcându-se constant pentru a simula prezența unei creaturi ascunse sub insulă.

Pentru jucător și inamici, animațiile de bază (*idle*, *run* și *attack*) sunt de asemenea preluate dintr-un pachet de asseturi. Aceste animații asigură tranziții line între stările de mișcare și contribuie la claritatea vizuală a acțiunilor efectuate de entități. Sistemul de animații este gestionat printr-o mașină de stare (*state machine*) care comută automat între diferitele animații în funcție de contextul jocului (ex: dacă jucătorul apasă tastele de mișcare, se activează animația de *run*, altfel rămâne în *idle*).

Atacurile beneficiază de animații dedicate, distincte pentru jucător și pentru fiecare tip de inamic. Acestea sunt sincronizate cu logica jocului, astfel încât efectele (ex: daunele) să fie aplicate într-un moment corespunzător al animației — de obicei în cadrul unui *animation event*.

Pentru inamicii de tip schelet a fost creată o animație personalizată, denumită `customHitReact`, care este redată atunci când

inamicul primește daune. Această animație a fost realizată utilizând *skeletal animation* și presupune combinarea a două animații existente:

- de la osul Hip în jos se utilizează animația de tip *idle*, pentru a păstra stabilitatea picioarelor;
- de la Hip în sus se utilizează animația de tip *death*, pentru a simula o reacție violentă a trunchiului și a brațelor la impact.

Această abordare, cunoscută sub numele de *animation blending pe ierarhia de oase*, permite reutilizarea eficientă a animațiilor și obținerea unor reacții realiste fără a necesita animații complete noi. Astfel, jocul oferă un feedback vizual clar și convingător în timpul confruntărilor.

### 3.3 Imersiunea în joc

Imersiunea în joc este obținută printr-o combinație între stilul vizual, elementele ambientale și designul sonor, care lucrează împreună pentru a crea o lume convingătoare, deși jocul nu are o poveste explicită. În ciuda complexității reduse a lumii, aceasta oferă suficiente detalii pentru a menține jucătorul conectat și implicat.

Stilul vizual al jocului este unul **low-poly**, caracterizat prin utilizarea unui număr redus de poligoane și forme geometrice simple. Acest stil, popularizat de jocuri precum *Totally Accurate Battle Simulator*, *Islanders* sau *Valheim* (care îmbină low-poly cu iluminare avansată), este adesea folosit pentru a obține un look artistic clar și expresiv, fără a depinde de realism. În jocul de față, această estetică contribuie la o atmosferă jucăușă, relaxată, dar în același timp misterioasă și captivantă, mai ales datorită elementelor neașteptate, cum ar fi tentaculele uriașe care înconjoară insula.

O trăsătură importantă a stilului low-poly este că reduce complexitatea vizuală, ceea ce face ca obiectele, inamicii și zonele importante să fie ușor de identificat pentru jucător. În același timp, prin lipsa detaliilor inutile, se stimulează imaginația jucătorului, ceea ce duce la o formă subtilă de imersiune: jucătorul umple golurile vizuale cu propriile așteptări și interpretări.

Pe lângă aspectul vizual, imersiunea este susținută de o coloană sonoră ambientală cu tematică de piraterie. Aceasta oferă o fundal sonor coerent care completează experiența de explorare a insulei și întărește identitatea lumii de joc. Chiar dacă nu există dialoguri sau poveste, muzica și

efectele sonore contribuie esențial la crearea unei atmosfere convingătoare.

Prin folosirea intenționată a unui stil minimalist și coerent, jocul creează un spațiu credibil în care jucătorul poate intra ușor și rămâne angajat, fără a fi nevoie de o narațiune explicită sau de efecte grafice complexe.

### 3.4 Mecanici de joc

Jucătorul poate interacționa cu inamicii într-un spațiu 3D. Mecanica principală implică lupta corp-la-corp. Sistemul de luptă include:

- **Atacuri bazate pe hitbox-uri** – Detecția atacurilor se face folosind coliziuni definite pentru fiecare tip de atac. Dacă jucătorul sau inamicul intră în raza de coliziune activă, se aplică daune.
- **Detecție și urmărire** – Inamicii detectează prezența jucătorului și trec în starea de atac. Dacă jucătorul se îndepărtează de un anumit prag de distanță, inamicul revine la patrulare.
- **Focalizare și direcție de atac** – Inamicul setează activ un focus către jucător, astfel încât să direcționeze animațiile și loviturile în mod realist.
- **Controlul vitezei de mișcare** – Inamicul se deplasează mai lent în timpul patrulării și accelerează în timpul atacului, crescând tensiunea interacțiunilor.

Aceste mecanici sunt esențiale pentru a oferi o experiență de joc coerentă și credibilă, sprijinind imersiunea și provocarea.

#### 3.4.1 Barcă plutitoare interactivă

Lângă insulă este poziționată o barcă pe care jucătorul o poate explora. Pentru a simula comportamentul realist al plutirii pe apă, a fost utilizată o tehnică bazată pe *float points*, inspirată din tutorialul (Gorka Games, 2023).

Modelul de plutire implică definirea mai multor puncte atașate la barcă, fiecare dintre acestea verificând în fiecare frame dacă se află sub nivelul apei. Dacă un punct este sub apă, asupra acestuia se aplică o forță ascensională proporțională cu adâncimea sa. Astfel, barca răspunde dinamic la forma și mișcarea apei, putând fi înclinată sau balansată în funcție de poziția acestor puncte.

Această abordare oferă o simulare fizică simplificată, dar convingătoare, fără a fi necesar un sistem complet de fizică a fluidelor. Barca poate fi rotită și înclinată natural, păstrând realismul vizual chiar și în absența post-procesărilor grafice sau a unui sistem complex de valuri.

#### 3.4.2 Puncte de salvare

Insula este împărțită în zone delimitate natural de teren sau de prezența inamicilor. În anumite locații cheie, de obicei după un grup de inamici, sunt amplasate puncte de salvare (*save points*). Acestea pot fi activate de către jucător și servesc drept puncte de respawn în cazul în care personajul moare. Acest sistem încurajează explorarea și asumarea de riscuri, fără a penaliza excesiv progresul jucătorului.

#### 3.4.3 Sistem de sănătate și damage

Jucătorul are la dispoziție o bară de viață vizibilă, care se reduce în urma atacurilor primite de la inamici. Sistemul de damage este echilibrat în funcție de dificultatea inamicilor, inamicii obișnuiți (schelete) producând daune mai mici, iar inamicul de tip boss (piratul-rechin) producând daune semnificative. Când viața ajunge la zero, jucătorul este eliminat și reîncepe din ultimul punct de salvare activat. Acest mecanism creează o buclă clară între explorare, luptă și supraviețuire.

## 4 Direcții viitoare

Proiectul prezintă un potențial semnificativ de extindere și îmbunătățire. Printre funcționalitățile care pot fi adăugate se numără implementarea unui sistem de navigare cu barca, posibilitatea de a trage cu tunul și mecanici care permit distrugerea bărcilor inamice. Aceste elemente ar spori gradul de interactivitate al aplicației și ar contribui la o experiență de simulare mai completă.

Un alt aspect vizat pentru dezvoltări viitoare este aprofundarea și rafinarea poveștii jocului. Deși narațiunea nu este un element de simulare în sine, aceasta joacă un rol esențial în crearea unui context coerent și captivant pentru utilizator, sporind astfel implicarea și imersiunea în universul jocului.

Proiectul ar putea fi utilizat ca punct de plecare pentru aplicații educaționale sau pentru dezvoltarea altor jocuri care combină explorarea, lupta navală și elementele narative. De asemenea, el poate fi integrat într-un cadru mai larg de sim-

ulare interactivă, oferind oportunități de experimentare și învățare într-un mediu virtual dinamic.

## **5 Concluzie**

Proiectul a oferit oportunitatea de aprofundare a cunoștințelor legate de Unreal Engine, în special în ceea ce privește simularea fizicii și interacțiunile din mediul 3D. A fost apreciată libertatea creativă oferită de dezvoltarea unui prototip de joc și satisfacția obținută prin implementarea mecanicilor de bază. În schimb, dificultățile întâmpinate în gestionarea fizicii obiectelor și a interacțiunilor complexe au reprezentat o provocare majoră, dar și o sursă importantă de învățare.

## **Limitari**

Proiectul prezintă o serie de limitări specifice stadiului actual de dezvoltare. În primul rând, implementarea este realizată exclusiv în Unreal Engine 5, ceea ce presupune o cerință hardware ridicată, în special în ceea ce privește resursele GPU. Scalabilitatea este deocamdată limitată, întrucât simularea fizicii și redarea graficii realiste pot afecta performanța pe sisteme mai slabe.

În timpul proiectului nu au fost realizate cercetări legate de performanță, iar odată cu creșterea complexității, această problemă va deveni mult mai proeminentă.

## **References**

Gorka Games. 2023. [How to make a boat float in unreal engine 5 - buoyancy tutorial](#).