

# Antiquarian

## 1. *Introducere*

Jocul nostru se încadrează în subgenul survival horror game. Dezvoltarea acestuia se va face cu ajutorul game engine-ului Godot folosind o abordare hibridă: înglobăm elemente 3D dar ne folosim de o grafică 2D. Gameplay-ul va fi axat în jurul experienței unui singur jucător, care își va controla caracterul într-o lume bidimensională.

### 1.1.Scopul sistemului

Sistemul are ca scop relaxarea jucătorului, permitându-i acestuia să evadeze din cotidian într-o lume nouă, asumându-și un caracter și acționând într-un mod cu totul diferit. Pe măsură ce petreci mai mult timp în lumea misterioasă iar povestea începe să se desfășoare vei observa o dorință de a descoperi fiecare colțisor al mapei și de a acumula cât mai multe informații despre noua lume. Perspectiva asupra jocului ți se va schimba și vei realiza că nu îți mai dorești atât de mult să duci misiunea la bun sfârșit ci vei dori să faci din călătoria ta în lumea plină de monștri feroși și pericole ce te pândesc la tot pasul una cât mai palpitantă, profitând într-un mod cât mai creativ de resursele întâlnite și obiectele adunate.

### 1.2.Documente referințe

- Documentul de specificare a cerințelor

## 2. *Obiective de proiectare*

Obiectivul principal al proiectării este realizarea unei arhitecturi modulare care să faciliteze implementarea funcționalităților descrise în documentul de specificație.

## 3. *Arhitectura propusă*

Arhitectura pe care ne-am propus să o folosim este MVC (Model – View – Controller).

### 3.1.Prezentarea generală a arhitecturii sistemului

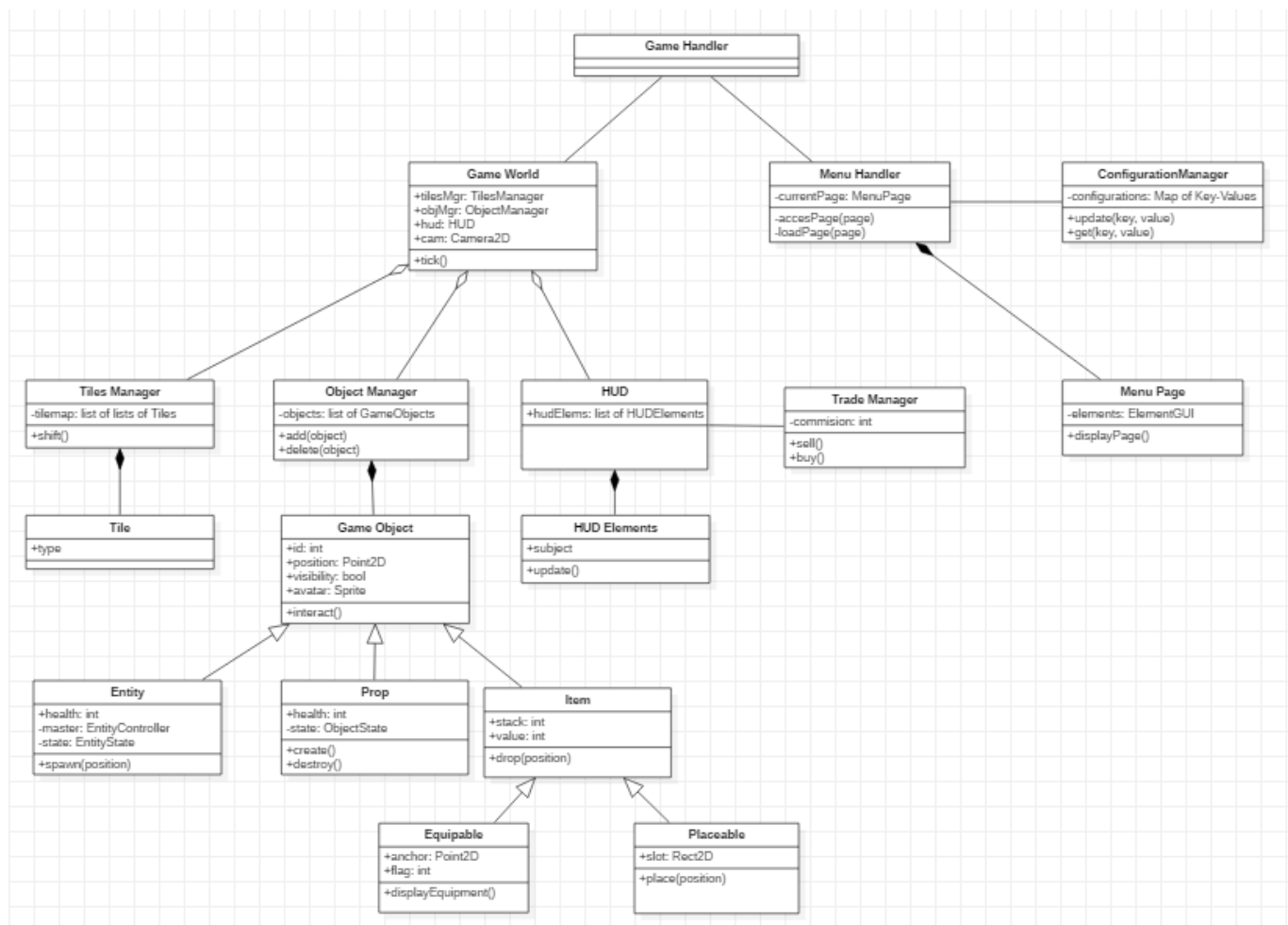
Arhitectura va fi construită peste engine-ului Godot și va urma modelul MVC (Model - View - Controller), fiind un model foarte popular în lumea jocurilor video.

Acesta va folosi lumea jocului (care include un număr de obiecte "ascunse" pentru a învârti roțile jocului) ca Model în care se vor petrece toate acțiunile cu adevărat. Aceste acțiuni includ orice interacțiuni între jucător și joc.

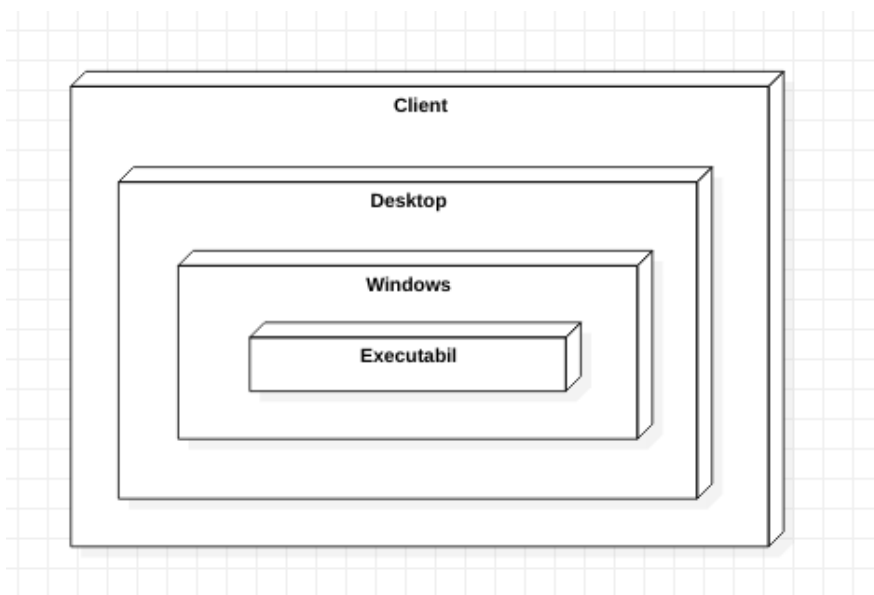
View-ul va fi constituit din camera care prezinta lumea jocului dintr-un anumit unghi, oferind o viziune de ansamblu asupra modelului, totodata fara a dezvalui detalii de functionare si logica din spatele evenimentelor. In categoria aceasta intra si HUD-ul in sine, care reprezinta o interfata de vizualizare a datelor din spate pentru ochii utilizatorului. Diverse interactiuni se vor desfasura prin intermediul ferestrelor de HUD, acestea fiind un hibrid intre actiune pe model (Controller) si vizualizari ale acestuia (View-uri).

Controllerul este insusi caracterul jucatorului, dupa care se orienteaza Viewul si care (din motive de optimizare) activeaza in jurul sau diferite evenimente in asteptare din Modelul jocului. Caracterul jucatorului este aproape exclusiv singurul mod de interactiune a jucatorului cu mediul, exceptie facand meniurile si ferestrele de HUD care cer interactiuni directe prin intermediul mouseului.

### 3.2. Decompozitia in subsisteme si responsabilitatile fiecarui subsistem



### 3.3. Distributia subsistemelor pe platforme hardware/software (diagrama de distributie)



### 3.4. Managementul datelor persistente (fișiere, sistemul de baze de date folosit, schema conceptuala a bazei de date)

- **Fisierul de save** va fi salvat pe disc si actualizat periodic in timpul jocului. Daca nu exista se va considera ca este primul joc al jucatorului.
- **Asseturile** sunt incarcate dinamic in timpul jocului.
- **Fisierul de configuratii** va exista numai daca utilizatorul a facut cel putin o modificare peste configuratiile default

### 3.5. Controlul accesului utilizatorilor la sistem

Data fiind natura pur recreativa a programului si lipsa accesului acestuia la un set de date ce pot fi folosite in scopuri malițioase, consideram ca nu este necesara implementarea unui mecanism de securizare a jocului sau a accesului la datele generate de utilizator in acesta.

### 3.6. Fluxul global al controlului

#### **Functionalitatea de modificare a dimensiunii fontului:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de modificare a nivelului de luminozitate:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de modificare a volumului:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de setare a tastelor:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de creare a unui joc nou:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de continuare a unui joc:**

- module active:
  - MenuPage: modulul care afiseaza setarile.
- pasare control:
  - MenuHandler este cel care detine controlul.

#### **Functionalitatea de negot cu un NPC:**

- module active:
  - TradeHandler: modulul care efectueaza negotul
- pasare control:
  - TradeHandler preia controlul, se deschide o fereastră. Cand tasta de iesire este apasata fereastra se inchide si se intra din nou in lumea jocului .

#### **Functionalitatea de atac al monstrului:**

- module active:
  - Entity: NPC, caracterul
- pasare control:
  - Game world este cel care detine controlul.

#### **Functionalitatea de navigare:**

- module active:
  - Entity: caracterul
- pasare control:
  - Game world este cel care detine controlul.

#### **Functionalitatea de culegere a unor elemente dintr-o resursa:**

- module active:
  - Item: elementul dat de resursa
  - Prop: resursa
- pasare control:
  - Game world este cel care detine controlul.

#### **Functionalitatea de adunare a unui element:**

- module active:
  - Item: elementul
  - ObjectManager: sterge instanta obiectului
- pasare control:
  - Game world este cel care detine controlul.

#### **Functionalitatea de plasare a capcanelor:**

- module active:
  - Placeable: capcana
  - ObjectManager: memoreaza instanta obiectului
- pasare control:
  - Game world este cel care detine controlul.

#### **Functionalitatea de echipare cu o arma:**

- module active:
  - Equipable: arma ce urmeaza a fi echipata
- pasare control:
  - Game world este cel care detine controlul.

### 3.7. **Condițiile limita** (cazurile de utilizare limită)

- Plasarea de obiecte (de către jucător) ce conduce la încărcarea unui număr prea mare de entități în spațiul în lumii activ la un moment dat (spațiul din jurul jucătorului).
- Consumarea resurselor neregenerabile (de către jucător) poate conduce într-un punct în care jucătorului nu mai poate învinge.