**Drive To Survive**

*Spridon Ioan*

*1212A*

*2023*

**Cuprins**

**I – Povestea jocului**

**II – Gameplay**

**III – Continut**

**IV – Niveluri**

**V – Interfata grafica**

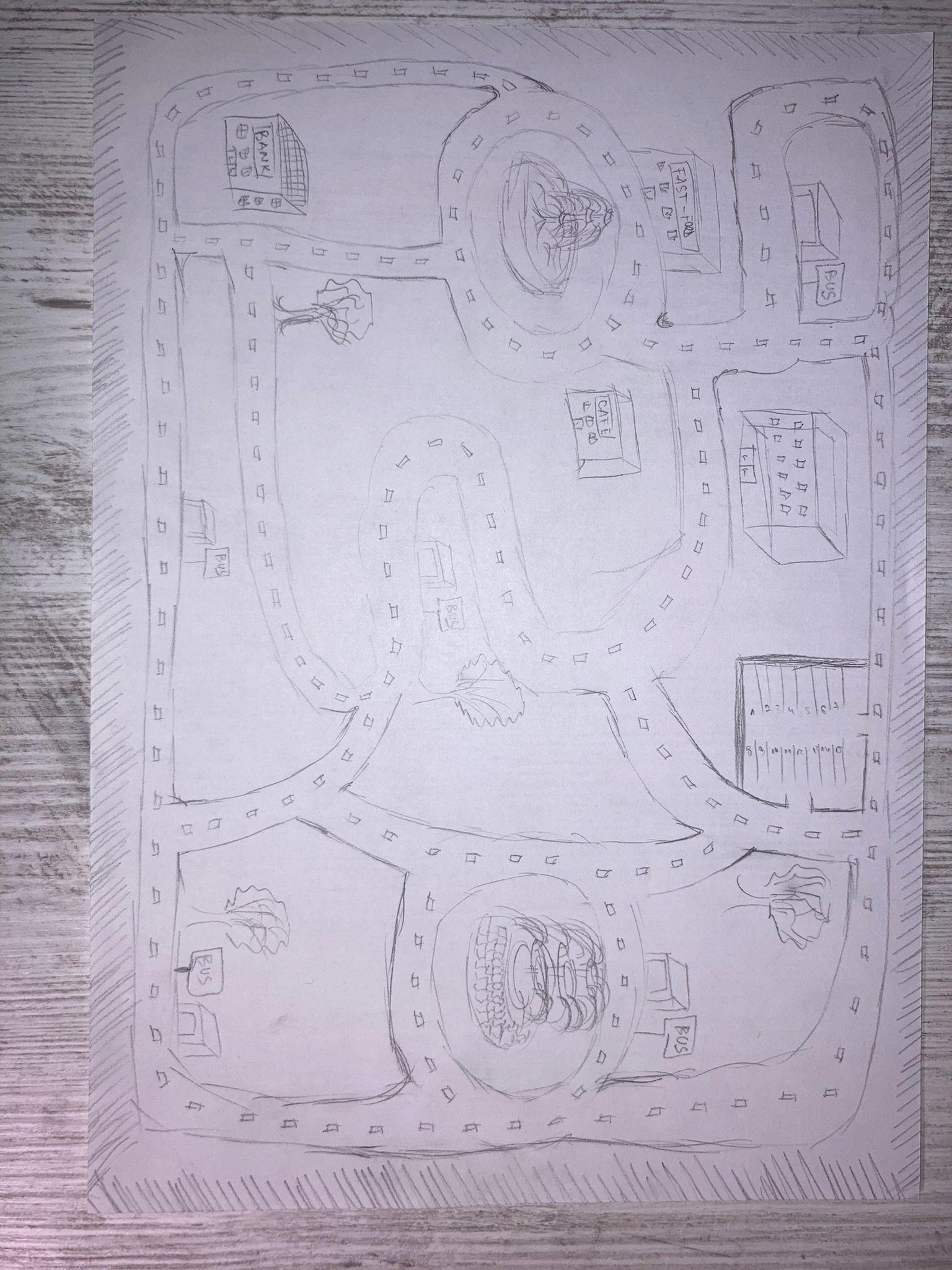
**VI – Diagrame**

**VII - Clase**

1. **Povestea jocului**

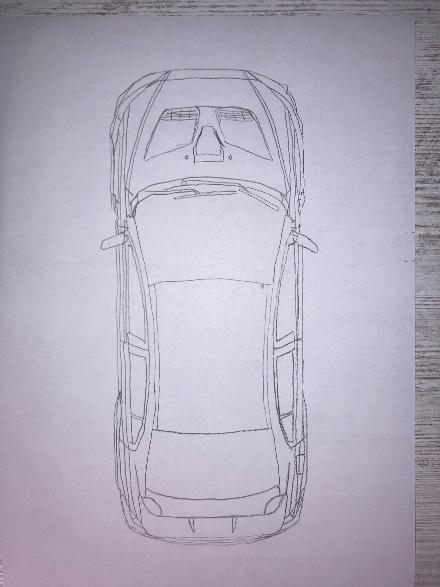
La începutul anilor 2000, printr-un cartier din București, un băiat pe nume Ryan a decis că trebuie să facă ceva cu viața lui și că nu mai poate trăia așa așteptând să se întâmple o minune care să-i schimbe soarta. Astfel într-o seară plimbându-se pe bulevardul [Theodor Pallady](https://stirileprotv.ro/stiri/actualitate/tineri-soferi-vitezomani-filati-cu-lunile-de-politisti-ce-faceau-noaptea-pe-strazile-din-capitala.html), a văzut câțiva băieți, împătimiți ai curselor ilegale, arătându-și mașinile la adevărata lor putere, sute de cai putere fiind implicați într-un loc în care cea mai rapidă mașină era o [Dacia](https://www.eurooldtimers.com/data/galerie/photos/31161.jpg). Și-a luat toți banii strânși și a plecat la părinții lui pentru a le cumpăra mașina deoarece i-a venit o idee care i-a plăcut atât de mult, încât a realizat că și-a găsit scopul. Ai lui având un [Mitsubishi Evo IV RS](https://farm1.staticflickr.com/827/42055654581_6c15219a66_k.jpg), el s-a gândit că prin mici reparații și dotări, va putea ajunge la performanța acelora din acea seară, iar cu timpul, să învingă tot mai mulți băieți ca să le ia banii, iar mai apoi, într-o zi, având o mașină destul de puternică, va ajunge rege al șoselelor din capitală.

1. **Gameplay**



Playerul va fi pus în perspectiva de sus, precum in GTA 1 si [Bomberman](https://en.wikipedia.org/wiki/Bomberman), iar folosindu - se de literele “W” ”A” ”S” ”D”, el va controla mașina și va ocoli obstacolele astfel incat sa ajunga cu succes la linia de finish. Prin apăsarea tastei “ESC”, jocul se va pune pe pauza.

Fiecare misiune va avea un sprite sheet diferit, adică va fi dus într-o alta lume, cu alte condiții de vreme și trafic, unde se va desfasura cursa.

1. **Continut** 

Playerul va avea de la început o mașina care va arăta asemănător cu Mitsubishi Evo, iar mai apoi, parcurgând nivelele, va putea da upgrade la mașină, făcând-o mai puternică sau va putea cumpăra alta pe baza bugetului pe care îl are. 

1. **Niveluri**

Jocul va avea 3 misiuni în care jucătorul va trebui să se întreacă și să câștige.

Rookie Challenge :

În primul nivel, totul contează doar din punct de vedere al vitezei. Dacă playerul are o mașină în care top speed-ul este mai mare decât al adversarului. Runda este ca și câștigată.

Master of the Streets:

În al doilea nivel, vor apărea obstacole: gropi, mașini parcate, semne de circulație care să încetinească jucătorii.

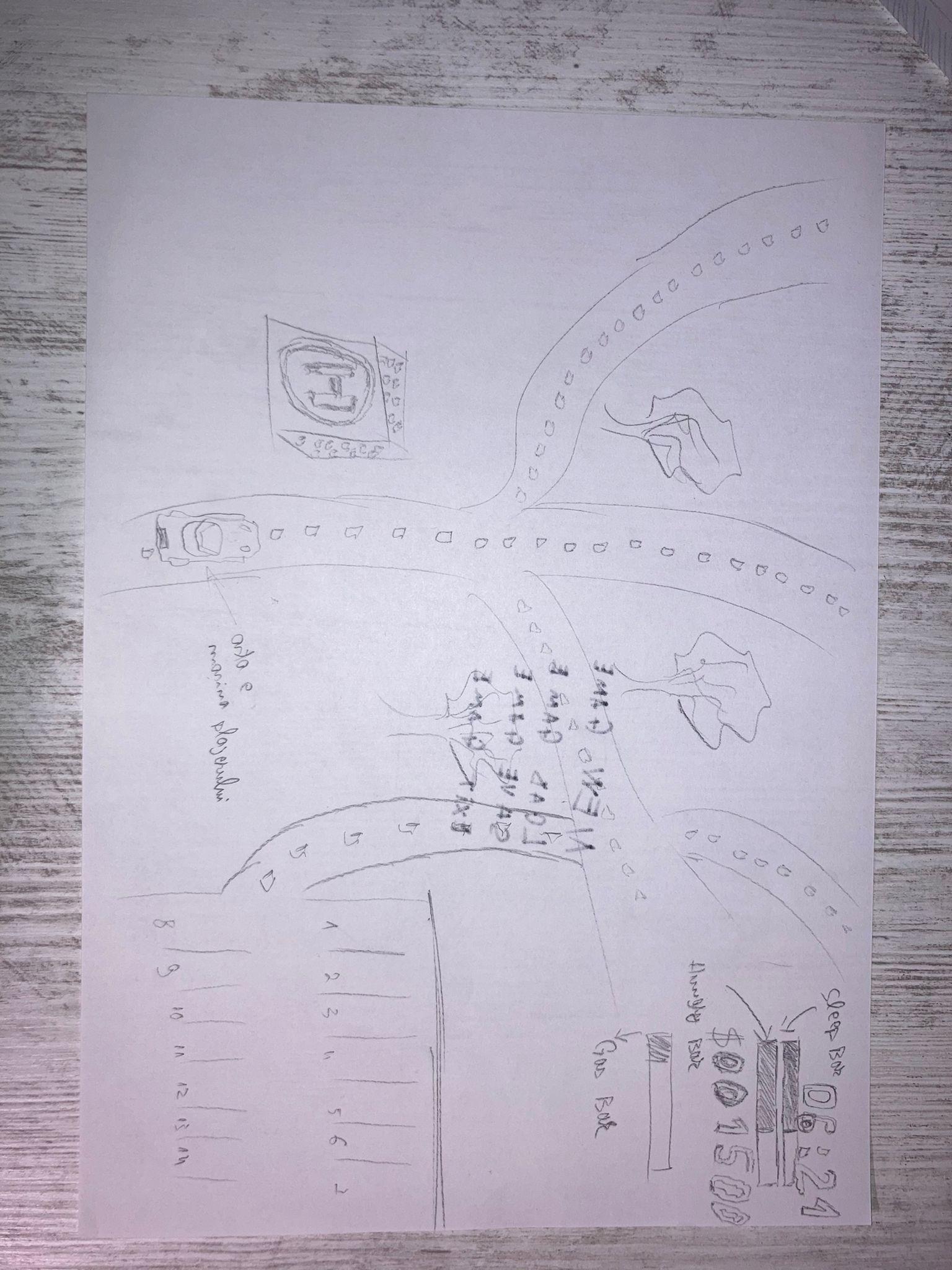


Emperor of the Streets:

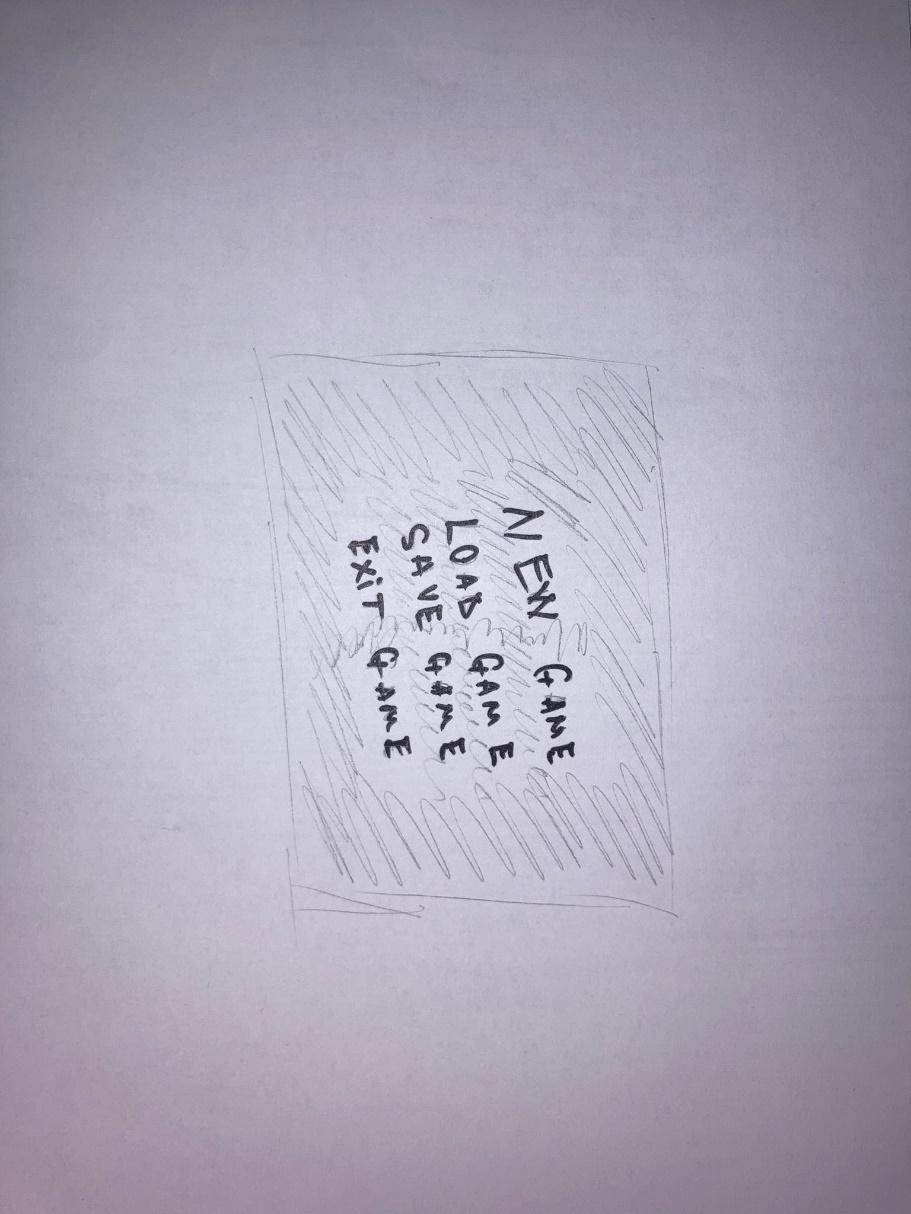
La Boss-ul final, Top Speed-ul mașinii lui va fi una mare astfel playerul va trebui sa dețină o mașină care să aibe minim top speed-ul adversarului, altfel, nu va putea câștiga. Pe drumul pe care se desfășoară cursa vor fi obstacole care vor obstructiona buna desfășurare a lucrurilor.

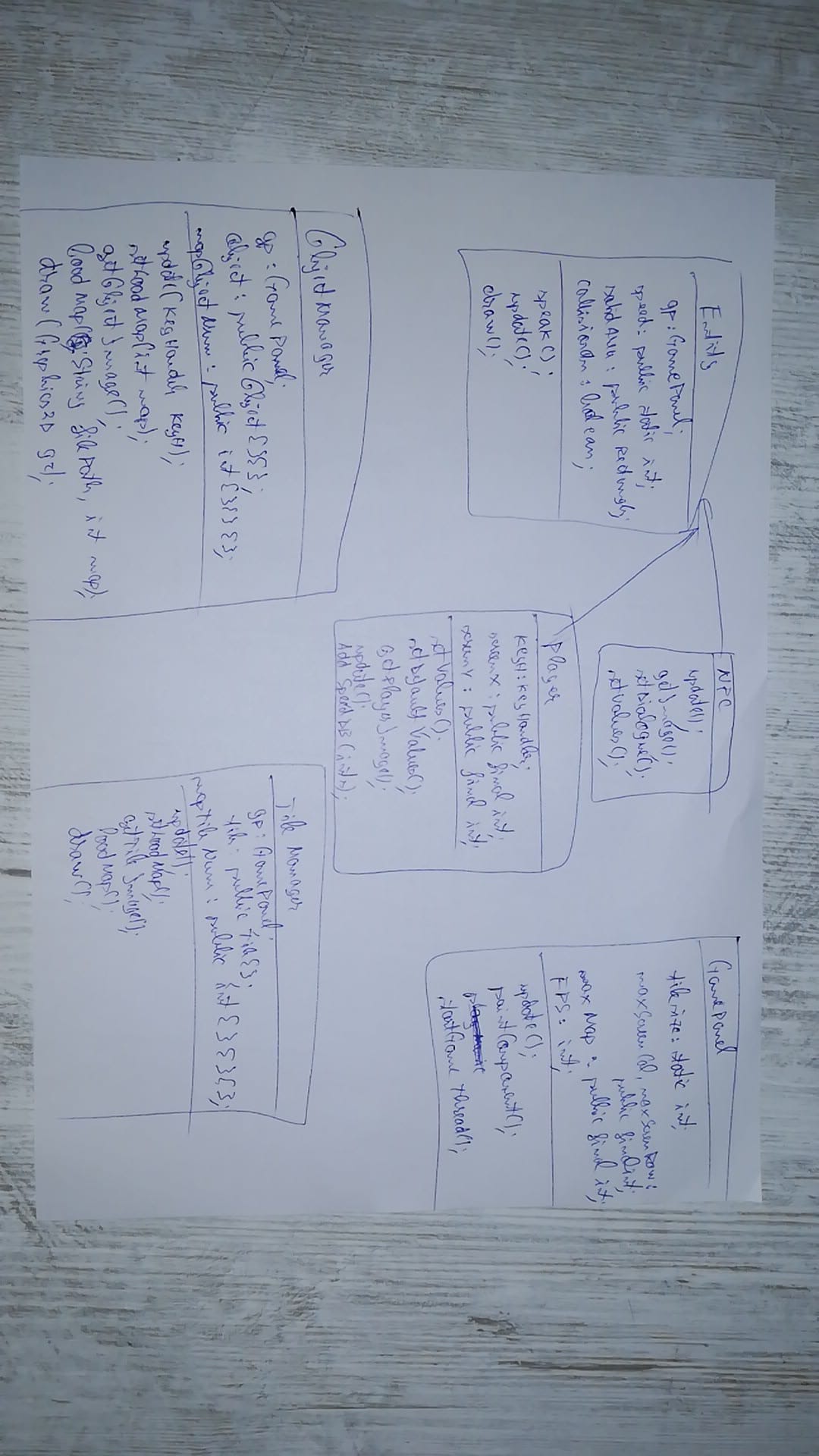
1. **Interfata grafica**

Utilizatorul va avea la dispoziție mici elemente grafice în care își va putea da seama de starea jocului.



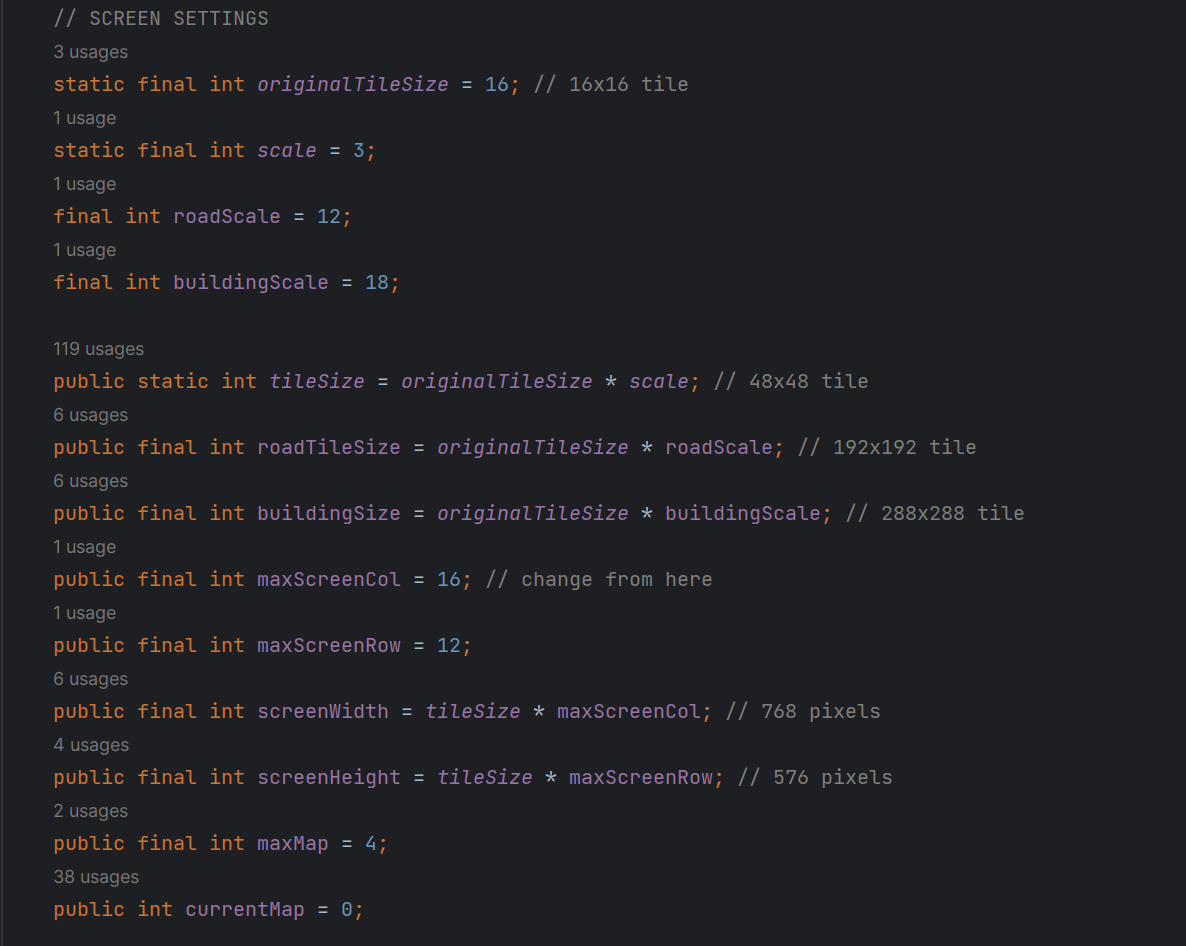
Când va apăsa tasta “ESC” vor apărea pe ecran următoarele elemente.



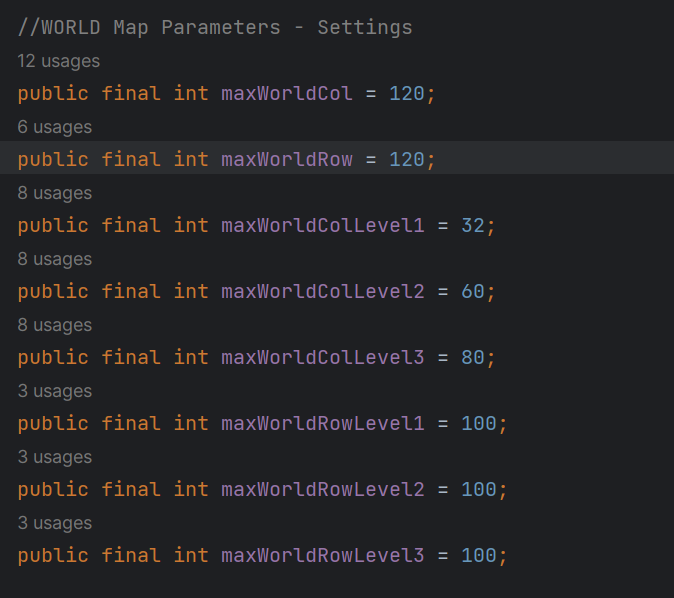
1. **Diagrame**
2. **Clase**

In proiectul de fata sunt 17 clase.

Printre cele mai importante dintre ele ar fi GamePanel impreuna cu TileManager, ObjectManager si cu clasa Entity.

GamePanel: ****

Contine atribute pentru setarile ecranului, precum marimea la tile-uri, dimensiunile matricelor care se incarca din fisiere text.

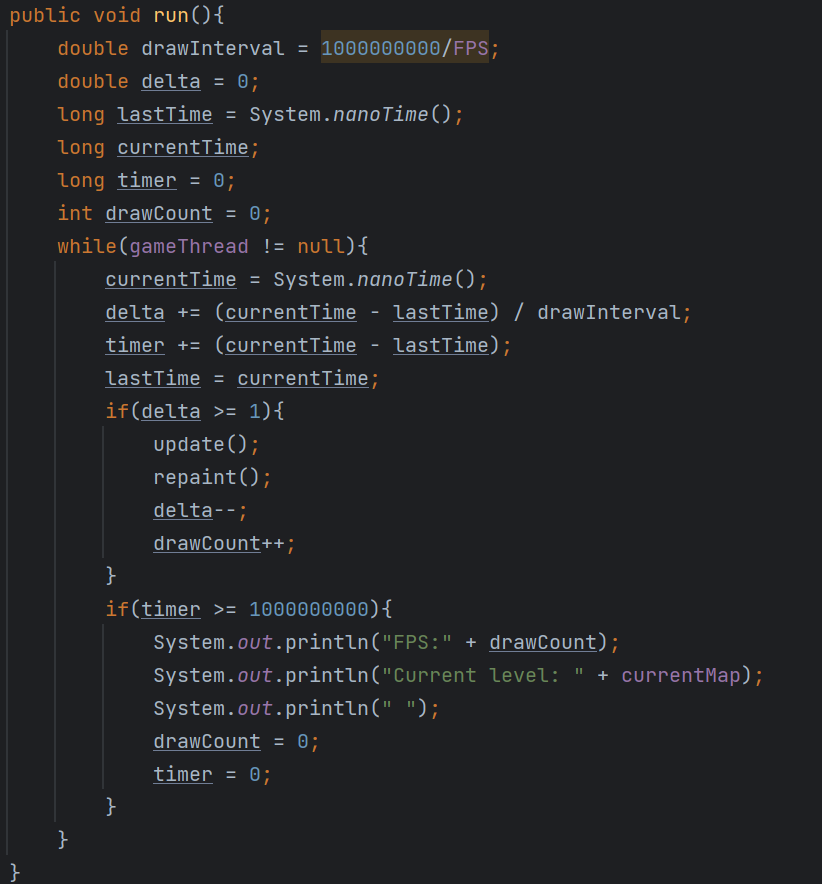


Printre metodele mai importante regasim paintComponent:

****

Aici sunt desenate pe ecran toate elementele grafice, obiecte, tile-uri, player, elemente de UI precum mesaje.

O alta metoda importante este cea de run:



Aici se executa cadrele per secunda al jocului.

In clasa TileManager, printre metode se regaseste si setLoadMap:

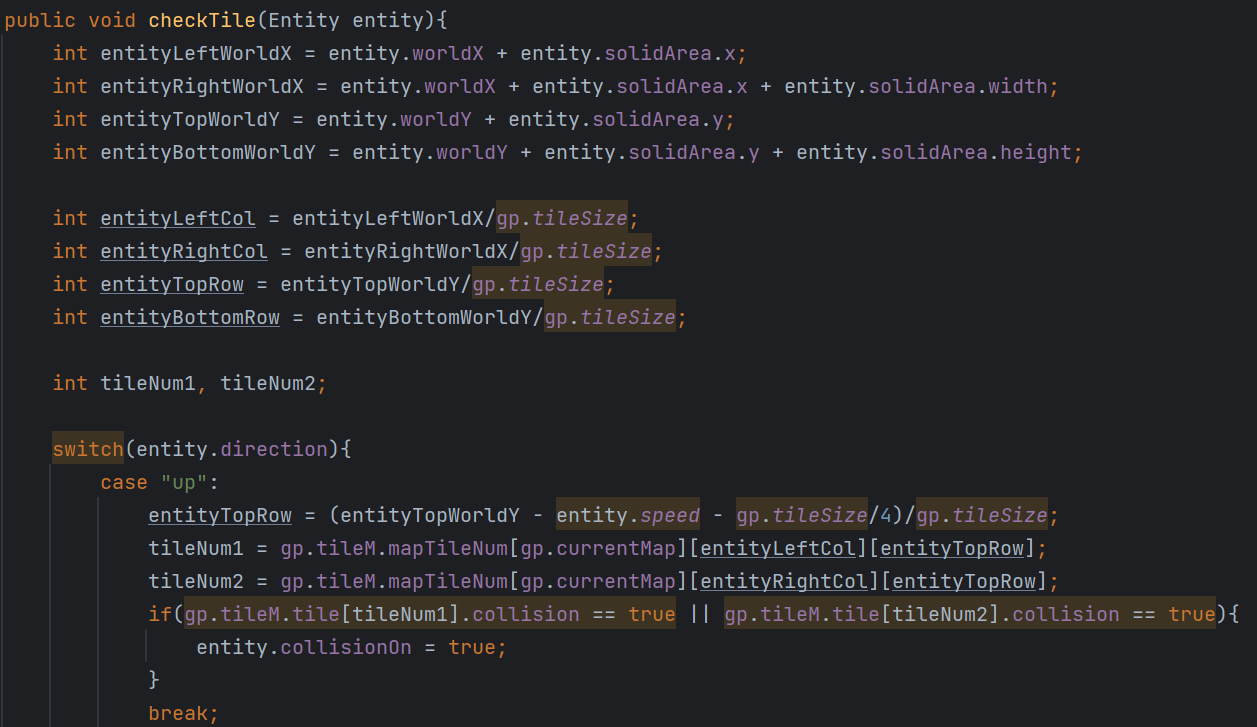


Scopul ei principal este de a schimba fisierul din care se va incarca matricea in memorie pe baza hartii curente in care se desfasoara jocul.

****

In exemplul de fata, desenam pe ecran in functie de datele pe care le primim din matrice.

Acel prim if face ca, daca parametrii in care se playeru este adevarat, sa deseneze doar un dreptunghi care sa incapa pe ecran si sa nu deseneze toata harta odata pentru a optimiza randarea jocului.

****

Aceasta metoda se regaseste in clasa CollisionChecker iar scopul ei este de a identifica coliziunile care se petrec intre Entity si tile.

Mai intai aflam diferiti parametri precum entityLeftWorldX pentru a sti cu exactitate pozitia pe ecran. Mai apoi vom afla pozitia in matricea cu tile-uri, iar in functie de directie vom verifica daca tile-ul respectiv are coliziunea setata pe “true”, ca mai apoi sa trimitem informatia adevarata iar dupa ce va fi procesata, entitatea sa ramana pe loc.

**Resursele biografice utilizate bibliografie**

Pentru jocul de față, m-am inspirat din jocuri precum [Gta San Andreas](https://www.rockstargames.com/games/sanandreas), [Gta 1](https://www.rockstargames.com/games/gta), [CSR Racing](https://play.google.com/store/apps/details?id=com.naturalmotion.csrracing&gl=US).