BATTLESHIPS

In cadrul disciplinei „Tehnici de programare” am ales proiectul „Battleships” deoarece mi se pare ca este un joc care ne aduce amintiri placute din copilare si are un nivel mediu de dificultate.

**Mediul de dezvoltare:**

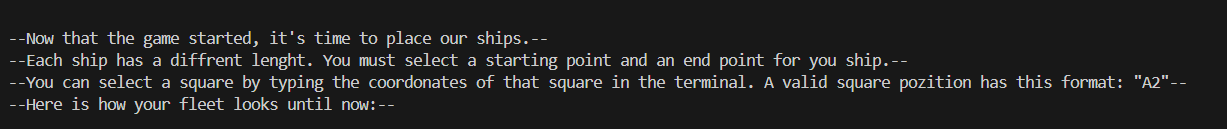
Am ales sa realizez acest joc in sistemul de operare Ubuntu 24.04 folosind editorul Visual Studio Code. Am folosit o biblioteca externa pentru partea grafica precum RayLib, codul sursa pentru jocul in terminal avand aproximativ 1400 de linii, iar codul pentru jocul cu interfata grafica avand apoximativ 900 de linii fara librariile commune cu jocul din terminal.

**Prezenatarea jocului din terminal:**

A screenshot of a computer screen

AI-generated content may be incorrect.

Chiar la lansarea jocului suntem intampinati de un mic meniu cu care putem selecta impotriva la cine sa jucam, sau daca dorim sa iesim din joc. Pentru a selecta o optiune trebuie doar ca sa tastam cifra dintre parantezele patrate.



Imediat suntem intampinati de un mic text care ne descrie jocul si modul in care putem selecta pozitia dorita, precum tastand litera dorita urmata de o cifra.

A computer screen with white text

AI-generated content may be incorrect.

Aici putem observa table noastra pe care vom juca si mai multe nave pe care le putem selecta tastand numarul corespunzator fiecarei nave.

Dupa fiecare introducere corecta, vom putea vizualiza nava pe table noastra. In cazul introduceri eronate a pozitiei, ne va aparea un mesaj de eroare corespunzator.

A screenshot of a computer

AI-generated content may be incorrect.

Dupa plasarea tutror navelor, ne va aparea optiunea de a reseta cateva nave dorite, iar in cazul in care nu dorim acest lucru putem contiuna cu jocul.

A screen shot of a computer code

AI-generated content may be incorrect.

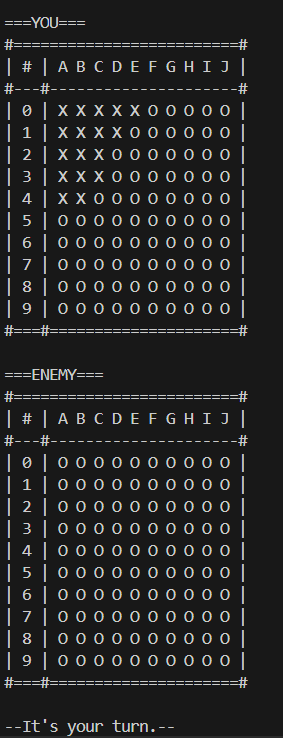
Aici ne este descrisa micuta harta a simbolurilor care le putem intalnii si semnificatia fiecaruia astfel:

-O- reprezinta un patrat care inca nu a fost lovit

- + - reprezinta o nava avariata

- # - reprezinta o nava distrusa

- \* - reprezinta un rateu.

Aici putem observa cele 2 table pe care le avem la dispozitie, table noastra unde vom vedea loviturile inamicului si table noastra unde putem observa loviturile facute de catre jucator.

Pentru a selecta patratul pe care dorim sa il lovim trebuie sa scriem in terminal pozitia dupa codificarea anterioara.

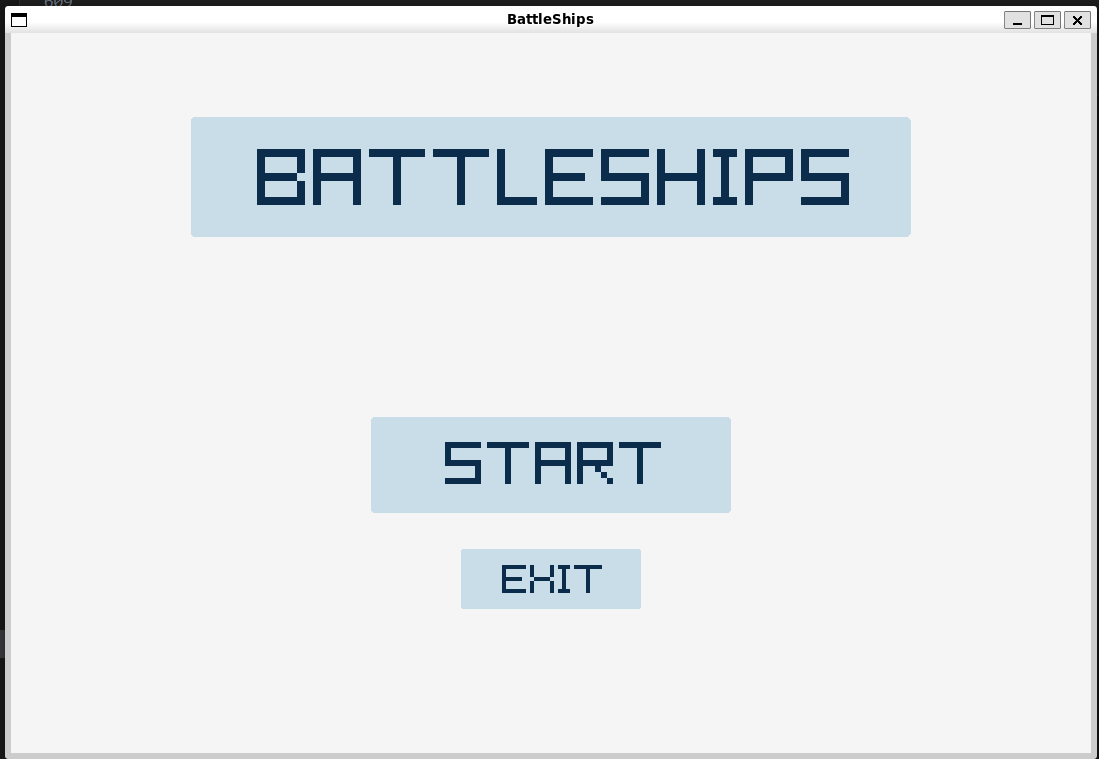
Castigatorul este reprezentat de catre cel care distruge toate navele adversarului.

Calculatorul selecteaza pozitia pe care o va lovi aleator, si daca acesta loveste una din navele jucatorului atunci acesta va retine pozitia care a lovit-o prima data si va verifica cele 4 directii alaturate acestui patrat pentru a determina directia in care a fost amplasata nava. Odata ce acesta descopra directia corecta, va continua sa loveasca in acea directie pana cand va distruge nava.

De asemenea, navele calculatorului sunt selectate aleator si verificate pentru a nu se suprapune, asemanator navelor jucatorului.

**Prezenatarea jocului cu interfata grafica:**

Odata ce deschidem jocul, suntem intampinati de un mic meniu care ne poate duce la partea de amplasare a navelor noastre prin apasarea butonului de START, sau un buton care ne poate ajuta sa iesim din joc.



Odata ce apasam pe butonul de start suntem intampinati de meniul de amplasare a navelor. Acestea pot fi trase pe tabla de joc din partea stanga. De asemenea putem sa si rotim navele pe care dorim sa le amplasam prin apasarea tastei [R] dupa cum ne sugereaza si texul din dreapta jos.

A screenshot of a computer

AI-generated content may be incorrect.

Dupa amplasarea tuturor navelor pe tabla de joc putem observa aparitia unui buton care ne va conduce la urmatorul ecran.

A screenshot of a video game

AI-generated content may be incorrect.

Dupa ce apasam pe butonul NEXT vom fi intampinati de ecranul:

A screenshot of a game

AI-generated content may be incorrect.

Putem observa in partea stanga a ecranului o tabla unde putem selecta patratul unde vrem sa lovim prin apasarea patratului dorit si in partea dreapta navele noastre unde vom putea observa loviturile calculatorului. Ca si in jocul din terminal, patratele sunt selectate aleator pana cand vom lovi o nava, dupa care vom cauta directia navei si o vom distruge.

A grid with squares and a number

AI-generated content may be incorrect.In cazul in care este lovita o nava, patratul in care se afla se va colora galben, iar cand distrugem o nava, toate patratele se vor colora in rosu.

A screenshot of a video game

AI-generated content may be incorrect.

La finalizarea jocului ne va aparea castigatorul impreauna cu un buton pe care putem apasa pentru a iesi din joc.

De asemenea, in cazul in care nu mai dorim sa continuam jocul putem sa apasam pe tasta[ESC] pentru a iesi complet din joc.

**Generarea Random a navelor calculatorului**

Navele calculatorului sunt generate in mod aleator prin generarea unei pozitii de inceput a navei si o directie, dupa care este calculata pozitia de final a navei in functie de lungimea si directia ei. In cazul in care pozitia finala a navei depaseste liniile sau se intersecteaza cu alte nave, se creste directia si se verifica din nou. In cazul in care niciuna dintre directii nu este buna, atunci se alege o noua pozitie de start si se reia procesul de la inceput.

Functia de generare a unei nave random se gaseste in biblioteca „computer.h”, unde ne este si descrisa.

**Jocul impotriva calculatorului**

Cand jucam impotriva calculatorului, de fiecare data cand este tura lui isi genereaza o pozitie random care nu a mai fost deja selectata si verifica daca a lovit sau nu o nava. In cazul in care pozitia generata loveste o nava, acest lucru este marcat pt algoritm prin variabila **hit** care ia valoarea 1 si este retinuta aceasta pozitie initiala.

Odata ce variabila **hit** ne indica ca am lovit o nava, se va genera o directie random, se va calcula urmatoarea coordonata in functie de directie si verificam ca aceasta sa fie corecta(sa nu iasa din conturul tablei de joc). In cazul in care directia iese din contur, atunci crestem directia si calculam o alta pozitie care sa o verificam. Odata calculata aceasta pozitie viitoare, se verifica daca nava se afla in directia calculata. In cazul in care aceasta se afla in directia dorita, atunci o variabila denumita **correct\_dir** va lua valoarea 1 si ne va indica ca directia care am ales-o este corecta, in caz contrar se va reveni la pozitia loviturii initiale, vom creste directia si vom verifica daca nava se afla in noua pozitie calculata.

Odata ce acea nava este distrusa, variabilele care ne-au ajutat sa distrugem nava isi vor lua valorile initiale si vom creste numarul de nave distruse al jucatorului.