

KULÁKSWEERЯ

Az világ első
kommunista aknakeresője

SZEREGÜNK



KŐVÁCS
CSABA

Csapatunk célja egy olyan játék megalkotása volt, amely visszaidézi a retró klasszikusokat, és ötvözi a kommunista ideológiát, valamint ennek okán a Szovjetunió hangulatát. A játékmenet kiválasztása, (külső tanács alapján) a mindenki által ismert aknakeresőre esett.



GÁL
BALÁZS

UI design
front-end
back-end
AI fejlesztés
projekt management

JÁTÉKMELET



Az aknakereső egy logikai táblajáték, melyben a játékosoknak a pályán elhelyezett aknákat kell megtalálniuk anélkül, hogy azokba belelőnének. A játék során a számok segítenek a játékosnak, melyek azt jelzik, hány olyan akna található a szomszédos mezőkön, amelyek a kiválasztott mezővel közvetlenül érintkeznek. A sikeres játékhoz logikai gondolkodásra és kockázatkezelési stratégiára van szükség.

FEJLESZTÉS

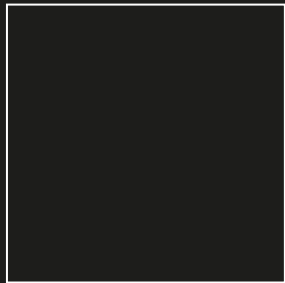
A projekt tanulmányi célból jött létre, Asztali alkalmazás fejlesztés tanórán kaptuk feladatnak egy WPF és C# nyelven írodott program elkészítését



A mocskos kapitalista szoftverek, programozási nyelvek használata miatt előre tudtuk hogy a végeredmény korántsem lesz olyan tökéletes mint például Marx Tőke c. műve, vagy mint a proletáruralomi törekvések.

FΦЯЯÁΣΦK

Színek



Fekete

Háttér

#1d1d1b



Vörös

Elsődleges szín

#aa1921



Arany

Kiemelőszín

#fccb39

Képek



Sarló,
Kalapács,
Csillag

Aknajelölő,
ikon

BETÜTÍPUS

ΣΦVIET

Készítő:

www.iconian.com

A stílus és hangulat megteremtése érdekében a leginkább szovjet stílusú betűtípust kerestük. Ez a betűkészlet egyből megmelengette kicsi komcsi szivecskénket

A	B	C	D	E	F
G	H	I	J	K	L
M	N	Φ	P	Q	Я
Σ	T	U	V	W	X
		Y	Z		

MŰKÖDÉS

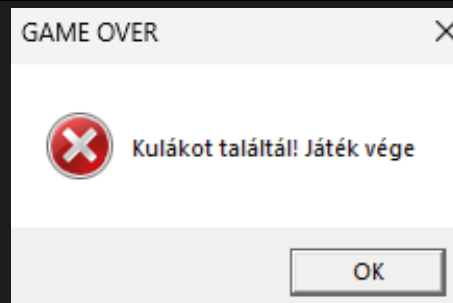


A játékos elvtárs a játékot egérekattintással irányítja

Bal klikk:
Megnyitja a mezőt, ha bomba, akkor felrobban, ha nem akkor kiírja hány szám található mellette

Jobb klikk:
Megjelöli a mezőt, egy sarló-kalapács-csillag jellel.

(Ezt utánna le is lehet venni, vagy megnyitni a blokkot)



KÓD

```
1 reference
private bool[,] GenerateMineGrid()
{
    Random random = new Random();
    bool[,] grid = new bool[Rows, Columns];

    for (int i = 0; i < Mines; i++)
    {
        int row, col;

        do
        {
            row = random.Next(0, Rows);
            col = random.Next(0, Columns);
        } while (grid[row, col]);

        grid[row, col] = true;
    }

    return grid;
}
```

```
1 reference
private void CreateGridButtons()
{
    gridButtons = new Button[Rows, Columns];
    var bc = new BrushConverter();

    for (int i = 0; i < Rows; i++)
    {
        for (int j = 0; j < Columns; j++)
        {
            Button button = new Button
            {
                Width = 40,
                Height = 40,
                Margin = new Thickness(2),
                Tag = new Tuple<int, int>(i, j),
                Background = (Brush)bc.ConvertFrom("#db1017"),
                BorderBrush = null,
                FontFamily = new FontFamily("Soviet"),
                FontSize = 30
            };

            button.Click += Button_Click;
            button.MouseRightButtonDown += Button_Flag;
            gridButtons[i, j] = button;
            Grid.SetRow(button, i);
            Grid.SetColumn(button, j);
            MinefieldGrid.Children.Add(button);
        }
    }
}
```

Először a megadott
mennyeiségű aknák
helyét generálja le a
program, többdimenziós
tömbökbe, majd a többi
helyet tölti ki. A
számot kattintáskor
számolja ki a játék.

JO SZÓRÁKΦZÁST
KÍNÁΠΠK,
ELVTÁRÁK!