Szakképesítés neve: Szoftverfejlesztő

OKJ száma: 54 481 02 0010 54 04

**Záró dolgozat**

Aknakereső

Krajnyák Attila Sztrehoszki Attila

Konzulens 14. S

**Budapest**

**2015.02.10.**

***NYILATKOZAT***

*Alulírott, Sztrehoszki Attila kijelentem, hogy ez a szakdolgozat saját tudásom, önálló munkám terméke. Kijelentem, hogy a szakdolgozat beköttetett és elektronikus formában leadott példányai mind formátumban, mind tartalomban egyezőek, eltérést nem tartalmaznak.*

Dátum: Hallgató Aláírása:

2015.03.31

Tartalom

[1. Bevezető - 1 -](#_Toc415253459)

[2. Felhasználói dokumentáció - 2 -](#_Toc415253460)

[2.1. Program általános leírása - 3 -](#_Toc415253461)

[2.2. Rendszerkövetelmények - 4 -](#_Toc415253462)

[2.2.1. Szoftverkövetelmények - 4 -](#_Toc415253463)

[2.2.2. Hardverkövetelmények - 4 -](#_Toc415253464)

[2.2.2.1. Minimális - 4 -](#_Toc415253465)

[2.2.2.2. Ajánlott - 4 -](#_Toc415253466)

[3. Fejlesztői dokumentáció - 5 -](#_Toc415253467)

[3.1. Alkalmazott fejlesztői eszközök - 5 -](#_Toc415253468)

[3.2. Adatmodell leírása - 5 -](#_Toc415253469)

[3.3. Feladat specifikáció, algoritmusok - 5 -](#_Toc415253470)

[3.3.1. Fontosabb algoritmusok - 5 -](#_Toc415253471)

[3.4. Tesztelési dokumentáció - 6 -](#_Toc415253472)

[3.4.1. Ismert hibák - 6 -](#_Toc415253473)

[3.5. Továbbfejlesztési lehetőségek - 6 -](#_Toc415253474)

[3.5.1. Segítség - 6 -](#_Toc415253475)

[3.5.2. Nehézségi szintek, MI - 6 -](#_Toc415253476)

[3.5.3. Tutorial - 7 -](#_Toc415253477)

[3.5.4. További játékmód - 8 -](#_Toc415253478)

[3.5.5. Új játék - 8 -](#_Toc415253479)

[3.5.6. Kilépés - 8 -](#_Toc415253480)

[4. Összegzés - 9 -](#_Toc415253481)

[4.1. Forrásjegyzék - 9 -](#_Toc415253482)

[4.1.1. Fejlesztői eszközök ikonjai (2015.03.26.) - 9 -](#_Toc415253483)

[4.1.2. Játékon belül található háttérképek ikonjai (2015.03.27.) - 9 -](#_Toc415253484)

[4.1.3. Dokumentáció: - 10 -](#_Toc415253485)

# Bevezető

A játék azok számára lehet érdekes akik szeretnek logikai játékot játszani, vagy akik szeretnek a barátjuk ellen játszani... Vagy mind kettő. Sokat lehet nevetni például amikor valaki véletlenül a semmi közepén eltalál egy aknát és erre a másik fél azt mondja neki hogy hogyan csal. Vagy amikor eltalál egyszerre hármat majd azt mondja „*Ez csak matek*”. Persze tudomásom szerint nem lehet benne csalni. Ha lehetne tudnék róla.

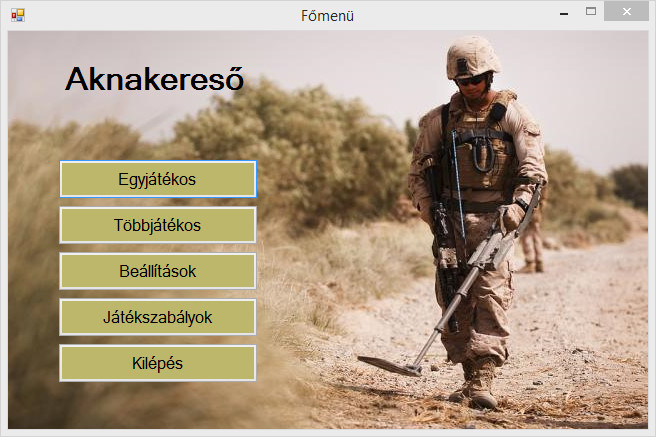
Szerintem sokakban felmerült az kérdés, miért ezt a témát választottam és talán jogosan kérdezik, hiszen van már aknakereső talán nem is egy. Nos egyrészt azért, mert sokat és régóta játszok és foglalkozok játékokkal ebből adódóan szeretek is velük foglalkozni, másrészt azért, mert szerettem volna én is csinálni egy játékot, hogy megtudjam, valójában milyen munka állhat egy játék elkészítése mögött. Továbbá szeretném, ha egy olyan cégnél tudnék elhelyezkedni ahol játékokat, fejlesztenek és szerintem ez nem egy rossz kiindulás ahhoz, hogy bekerülhessek egy ilyen helyre. Úgy éreztem hogy ez az a cél amit kitűztem magamnak ezt teljesíteni is tudom. Tudom hogy nagy különbség van egy grafikus 3D-s játék között és az én játékom között de ettől függetlenül megtanított arra hogy logikailag hogyan épül fel egy játék és hogy akármilyen nagy vagy kicsi is legyen az a játék azt el kell tárolni valamilyen adatszerkezetben és miután minden megvan csak azután következik a megjelenítés. Megtanított a játék elkészítése arra is, hogy az osztályok és a metódusok hogyan használják és hívják meg egymást.

# Felhasználói dokumentáció

Játékszabályok a következők:

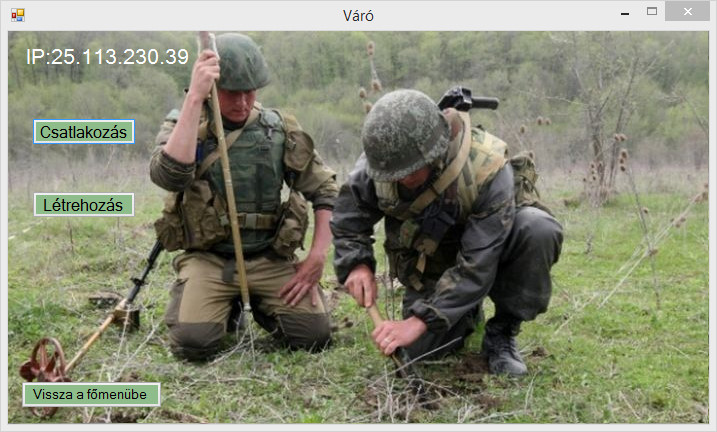
* Egyjátékos módban a játékos kezd többjátékosban pedig aki a szobát csinálta
* akinek a kattintása aknára került az újra kereshet és kattinthat a térképen egészen addig, amíg el nem rontja a sorozatot
* ha egy szám mezőre nyomott a játékos (azaz elrontotta a tippelést), akkor az ellenfél köre jön
* csak az aknák felderítése ér pontot
* a játéknak akkor van vége, ha már minden akna fel lett derítve
* az a játékos nyer, akinek több pontja van
* Lehet egy „X”-et rakni a kívánt gombra amiben biztosak vagyunk hogy ott nincs akna. Ha mégis meggondoltuk volna magunkat lehetőség van a levételére is.

A program indításakor a főmenüben találjuk magunkat (1. ábra). Itt van négy opció, amiből választhatunk. Az első az egyjátékos mód. Itt gépi ellenfél ellen kell majd játszani. A második a többjátékos mód. A harmadik lehetőség a beállítások. A negyediknél találod meg a játékszabályok részletesebb leírását. Az ötödik opció pedig a kilépés amivel a játékból lehet kilépni.



*1. ábra – A főmenü*

Visszatérve a többjátékos módra (2-es ábra), itt három újabb választási lehetőség van. Az első, amikor egy másik játékos csatlakozik hozzánk. A másik lehetőség, amikor mi csatlakozunk másokhoz. Illetve a harmadik, amivel a főmenübe lehet visszalépni. A bal felső sarokban mindig a saját IP címünk látható.



*2. ábra – Többjátékos módhoz való váró*

A beállításokban lehet beállítani a térkép méretét, ami mind a két játékmódra, vonatkozik.

## Program általános leírása

Mielőtt bárki azt hinné, hogy a játék az eredeti aknakereső hasonmása elmondanám az alapvető különbséget az eredeti, klasszikus aknakereső között, és az én aknakeresőm között. Az eredetiben elmondható, hogy kerülni kellett az aknákat azáltal, hogy a számok, amik körülötte vannak, elárulják a pozícióját és megjelölni egy zászlóval. Ha rányomtunk egy aknára elvesztettük a játékot és akkor nyertük meg, ha minden olyan mezőt felfedtünk, ami nem akna (azaz szám mező). Az én aknakeresőmben viszont pont hogy az aknákat kell elsősorban meg találni és a számokat lehetőleg kerülni mivel azok nem érnek pontot az aknák pedig igen. Ezt a szabályt azért hoztam létre, mert a program a többjátékos módra összpontosít, hogy izgalmasabbá és élvezhetőbbé tegye játékot. Ebből az okból be kellett vezetnem az egyjátékos módba egy gépi ellenfelet. A szabály ezek után már egyszerű. Az nyer, aki több ponttal rendelkezik a játék végén. A játék akkor ér véget, ha az összes akna felderítésre került.

## Rendszerkövetelmények

## Szoftverkövetelmények

## Hardverkövetelmények

## Minimális

Operációs rendszer: Windows 7 32 vagy 64 bites

Processzor: 1GHz

Memória: 1GB (64 bites rendszer esetében 2GB)

Videokártya: 512MB DirectX 9 támogatással

Merevlemez: 100 MB szabad hely

## Ajánlott

Operációs rendszer: Windows 7 64 bites

Processzor: Intel Core i7 950 3GHz, vagy AMD FX 8350, vagy ezeknél jobbak

Memória: 6GB

Videokártya: 1GB DirectX 10 támogatással

Merevlemez: 100 MB hely

# Fejlesztői dokumentáció

## Alkalmazott fejlesztői eszközök

|  |  |  |
| --- | --- | --- |
| c#.jpg | C# | Programnyelv, amiben a program íródott. |
| mvc#.jpg | Micosoft Visual C# Express 2010 | Fejlesztői környezet. |
| word.jpg | Micosoft Office World 2007 | Ebben a szövegszerkesztő programban lett megírva a dokumentáció. |
| Paint.png | Paint | Rajzok készítése és módosítása. |
| onedrive.png | One Drive | Fájlok tárolására alkalmas felhőszolgáltatás. |
| AI.png | Adobe Illustartor CS5 | Képek létrehozásához használható. |
|  | Bandicam | Jó minőségű képmentéseket illetve videókat lehet vele készíteni. |

## Adatmodell leírása

## Feladat specifikáció, algoritmusok

## Fontosabb algoritmusok

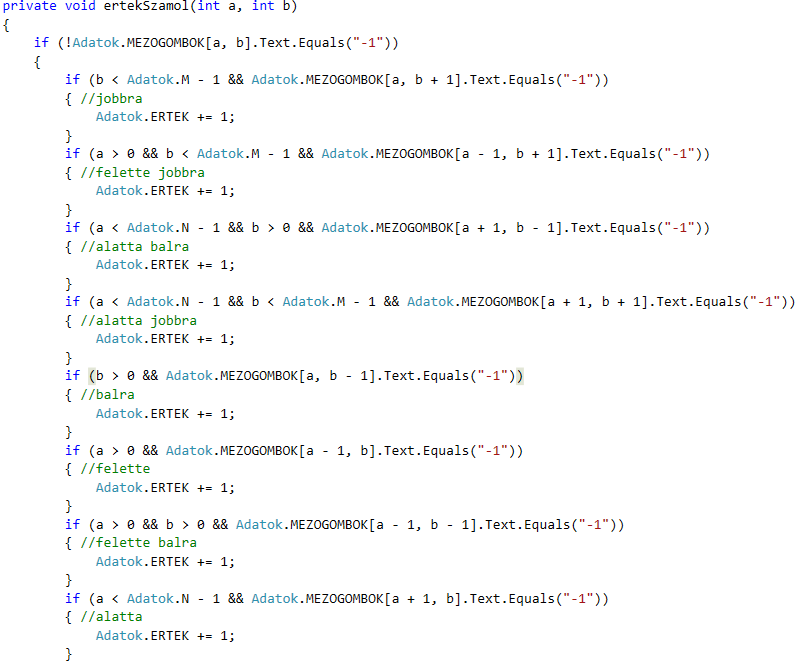
private void MezoGeneralas()

Ez a legfontosabb és legfőbb metódus a játékban. Ez felelős a mező legenerálásáért. Itt töltődik fel a 2 mátrix gombokkal. Pozícionálja a gombokat. Kiszámolja mennyi akna legyen a térképen, legenerálja a gombokra véletlenszerűen az aknákat és le is tárolja egy két dimenziós tömbben a helyüket, ami arra szükséges, hogy az MI tudjon majd hova kattintani, illetve hogy többjátékos módban majd el tudjuk küldeni hogy biztosan ugyanaz a térkép legyen minkét játékosnál. Ezeken felül még hozzáad az ablak szélességéhet 200 pixelt illetve a magassághoz 100-at, hogy kiférjenek a gombok és a *labelek*, (többjátékos módban pedig az üzenetek). A metódus meghívja a *fedoGombok\_MouseUp*-ot, a *gombLetrehoz* metódust, az *ertekSzamol* metódust.

private void ertekSzamol(int a, int b)

A metódus a mezőgombok értékeinek számolásáért felelős, ezáltal kiíratódik a gombokra az értékük. A két paraméter a mátrixban lévő gombok meghatározása miatt szükséges (3. ábra).

Ebben a metódusban hívódik meg a *pontszamol* metódus.



*. Ábra - Érték számolás*

private void pontSzamol(FedoGomb aktualisGomb)

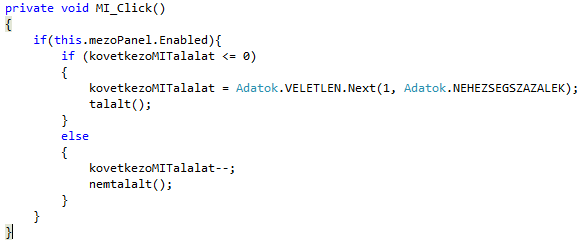
Ez a metódus a játékosok pontszámait számolja össze, változtatja az aktuális játékost illetve a *labelek* színeit. Meghívja a *jatekVege* metódust is.

private void JatekAblak\_Load(object sender, EventArgs e)

Itt a *mezoPanel* szélességét és magasságát növeli a beállításokban megadott értékekkel hogy ki tudjanak férni a gombok. Itt hívódik meg a *MezoGeneralas* metódus illetve az *AknaszamL*-re kiíratja az összes akna számát ami a játékban fellelhető.

private void MI\_Click()

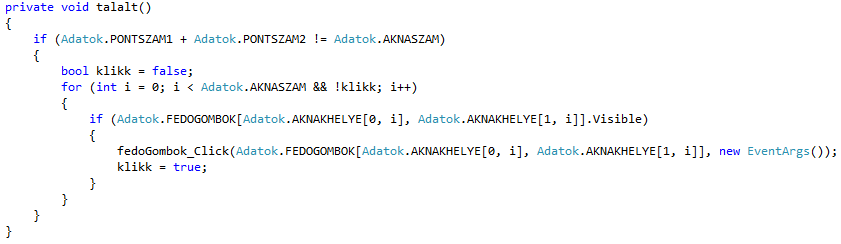
Olyan metódus ami azt csinálja ha a panelen még lehet kattintani és a *kovetkezoMITalalat* változó kisebb vagy egyenlő mint nulla akkor meghívja *talalt* metódust ha pedig nagyobb akkor a *nemTalalt* metódust. Ezek megadják hányadik rossz tippelése után találjon aknát a gép.



*.Ábra – Nem vagyok robot*

private void talalt()

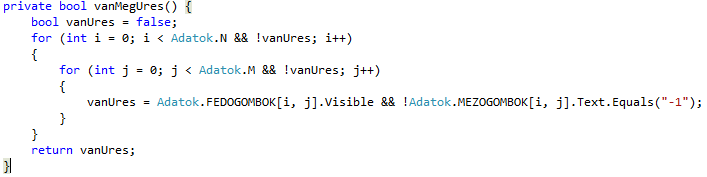
Található benne egy elágazás ami eldönti, a játékosok pontszámai összegének alapján hogy az kisebb-e mint amennyi akna van. Ha ez igaz akkor belép, amin belül található egy ciklus és egy lokális *klikk* nevű változó. A ciklusba csak akkor lép be ha a *klikk* hamis és végiglépked az aknákon a ciklus *i* változója amíg kisebb mint az aknaszám. Ebben a ciklusban van egy elágazás aminek a feltétele az, hogy az aktuális indexen lévő akna felett lévő fedőgomb látható legyen. Ha teljesül a feltétel, belép és kattintást hajt végre ezen a fedőgombon, majd a *klikk-*et igazra állítja, hogy ne tudjon újra lefutni a ciklus és ezáltal a gép ismét kattinthasson.



*.Ábra - Találat*

private bool vanMegUres()

Van benne egy lokális *vanUres* nevű logikai változó illetve egy egymásba ágyazott léptetős ciklus amik végigmennek a beállított térkép méreteinek szélességén és magasságán, a ciklusoknak pedig az a feltételük hogy a *vanUres* hamis legyen. Ha belépett a ciklusmagba akkor levizsgálja hogy az aktuális fedőgomb még látható-e és hogy ugyanazon a ponton lévő mezőgomb nem akna-e. Ha mindez igaz akkor a *vanUres* igazra állítódik. A ciklus visszadob majd egy igaz vagy egy hamis értéket.



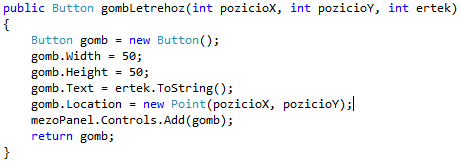
*.Ábra – Van-e még elérhető gomb?*

private void nemtalalt()

Az első elágazás ami benne található annak az a feltétele hogy a *vanMegUres* metódus visszatérési értéke igaz legyen. Ha ez megvan továbbmegy ahol lesz ismét egy *klikk* logikai

public Button gombLetrehoz(int pozicioX, int pozicioY, int ertek)

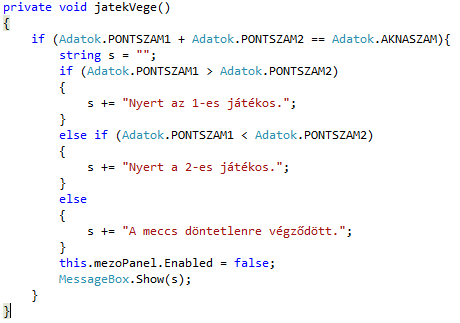
Gombokat hoz létre, megadja nekik a pozíciójukat hol helyezkedjenek el a *mezoPanel*-en, megkapják az értékeiket, kapnak szélességet illetve magasságot. A metódus pedig egy gombbal fog visszatérni.



*.Ábra – Gomb létrehozás*

private void jatekVege()

Ebben a metódusban található egy fő elágazás ami eldönti hogy a két játékos pontszámának összege megegyezik-e az aknák számával. Utána következik egy kisebb elágazás, ami eldönti ki nyert az alapján kinek mennyi pontszáma van a játék végén. Ennek eredményéül kiírja egy felugró ablakban, hogy ki nyert vagy hogy döntetlen lett-e, majd letiltja a mezoPanelt, hogy ne lehessen tovább kattintani a panelen.



*.Ábra – GAME OVER*

## Tesztelési dokumentáció

## Ismert hibák

A játék bizonyos esetekben ha kicsi a pálya akkor a játék vége felé lefagy.

Többjátékos módban ha a játék közben lépünk ki hibaüzenetet fog kidobni.

## Továbbfejlesztési lehetőségek

Amiket még érdemes lenne megvalósítani, de nem volt már rá időm az egyszeri segítség a vesztésre álló játékosnak, akinek 5 pont különbsége van a nyerésre álló játékostól. Ezek egyikének áll a rendelkezésére az adott játékosnak és csak egyszer használhatja. A segítséget lehetne alkalmazni mind az egyjátékos módban, mind a többjátékos módban.

## Segítség

A segítségek részletes leírása:

* Négyzet alakú felderítés:

Az éppen kattintott mező és a körülötte levő nyolc mező felderítése (ez összesen kilenc felderített mező)

* Vízszintes vonal alakú felderítés:

Az éppen kattintott mező, a tőle balra levő négy mező és a tőle jobbra levő négy mező (ez szintén kilenc darab felderített mező)

* Függőleges vonal alakú felderítés

Az éppen kattintott mező, a felette lévő négy illetve az alatta lévő négy mező felderítése (ez ugyancsak kilenc mező összesen)

## Nehézségi szintek, MI

A segítségeken kívül még szükség lehet a játék egyjátékos mód részéhez: további nehézségi szintek létrehozására illetve a mesterséges Intelligencia (MI) megalkotására

Tippek a nehézségi szintekhez illetve a Mesterséges Intelligenciához:

Nehézségi szintek részletesebb leírása:

* Könnyű, közepes, nehéz: Ezekben a módokban más-más lenne a gép találati aránya.

Mesterséges Intelligencia részletesebb lírása:

* A Mesterséges Intelligencia alatt gondoltam azt, hogy a gép ne kattintson, olyan mezőre ahol nem nagyon van felderítve valami, azaz lehetőleg olyan mezőkre nyomjon, ami körül már van felderített mező. A Mesterséges Intelligencia kattintásai függhetnek a nehézségi szinttől is. Például nehéz szinten a gép találhat aknát olyan helyen is ahol még nincs, felderítve semmi könnyű szinten pedig csak a felderített mezők körül találgasson gépi ellenfél. Közepes szint a pedig kettő keveréke.

## Tutorial

Egy Tutorial (oktató játékmód) opció is lehetne a főmenüben, ami lépésről-lépésre bemutatja a játékot a játékosnak.

Tippek az Oktató Játékmódhoz:

Lehetne *szimulálni* bizonyos helyzeteket a játékos számára. Statikusan feltöltve mondjuk egy hatszor hatos térképpel és mindig csak a kiemelt mezőre tudjon kattintani. Lépésről lépésre valahogy így nézhetne ki.

1. Elsőre még semmilyen mező nem lenne felderítve. Feldobunk a játékos részére egy üzenetet, hogy hova kattintson, Ok-t nyom majd lehetne valamilyen színnel kiemelt mező (például zöld vagy sárga) ami mögött egy szám-mező lenne, a többi mező pedig nem lehetne elérhető a játékosnak.
2. Ismét következhetne egy üzenet, amiben leírjuk, hogy mivel ő nem aknára kattintott az ellenfél köre jön megvárva, míg az ellenfél valahova nem kattint. Miután bezárta az üzenetet a gép nyom egy aknára majd elmondjuk a játékosnak, hogy az ellenfele egy aknát talált ezért ismét az ellenfél kattinthat, illetve megmutatjuk, hogy a gép pontszáma az első találat után növekedett. Elmagyarázzuk neki azt is, hogy minden akna egy pontot ér. Ezek után a gép további négy aknát fog találni majd szám mezőre nyom.
3. Miután lement a gép köre elmondjuk neki, hogy a gép öt ponttal vezet, ezért elérhetővé válik számára egy egyszeri segítség, amit felhasználhat. Kiemeljük ismét valamilyen színnel a segítséget majd azt is hova kattintson. Felderít hat-hét aknát és megmondjuk neki, hogy akkor van vége a játéknak, ha minden akna meg lett találva, ezek után elmondjuk neki, hogy a maradék párat már egyedül kell megtalálnia.
4. Miután megtalálta az összes aknát elmondjuk, hogy itt az oktató rész vége, illetve hogy bővebb információt a játék működéséről a főmenüben lévő játékszabályok menü alatt talál majd. Ok-t nyom és kilépteti a program a főmenübe.

## További játékmód

További választási lehetőség a játék további két módját kiválasztani az egyjátékos és a többjátékos részhez is. Klasszikus és Egyedi játékmód.

Egyedi:

* Ez az, ami már meg van alkotva. Itt elérhetőek a segítségek a vesztésre álló játékos számára.

Klasszikus:

* Ide azt terveztem, hogy itt nincsen lehetőség segítségeket használni.

## Új játék

Új játékkezdés lehetőségének megvalósítása többjátékos módban. Azért lenne érdemes megcsinálni mert akkor nem kellene kilépni a váróba új játék kezdésének céljából.

## Kilépés

Kilépés a játékból vagy kilépés a főmenübe. Megerősítés kérdése, hogy biztosan kilép-e a játékból.

# Összegzés

## Forrásjegyzék

## Fejlesztői eszközök ikonjai (2015.03.26.)

c#:

<https://s-media-cache-ak0.pinimg.com/736x/cf/e4/b1/cfe4b1de7406ff78e5f710f27cd7430f.jpg>

Micosoft Visual C# Express 2010:

<http://screenshots.en.sftcdn.net/en/scrn/70000/70368/microsoft-visual-c-2008-16.jpg>

Micosoft Office World 2007:

<http://www.schoonzie.com/sites/default/files/Word2007Icon.jpg>

Paint: <http://img1.wikia.nocookie.net/__cb20120808070321/angrybirdsfanon/images/3/3e/MS_Paint_Icon.png>

One Drive:

<https://lh5.ggpht.com/YlM1Od8VyNSo64B6KIFAQUhW8ArLWCNZISYdNup45ArcLcKJySMiz24RFo-dhYui7Lc=w300>

Adobe Illustartor CS5:

<http://upload.wikimedia.org/wikipedia/commons/7/79/Adobe_Illustrator_CS5_icon.png>

Bandicam:

<https://lh3.googleusercontent.com/-0yyXZTvaoes/AAAAAAAAAAI/AAAAAAAAACs/wMm6yGxufYM/photo.jpg>

## Játékon belül található háttérképek ikonjai (2015.03.27.)

Főmenü:

<https://www.defencetalk.com/wp-content/uploads/2011/01/afghanistan-IED-metal-detector.jpg>

Beállítások:

<https://c1.staticflickr.com/5/4146/5041010323_3eeaf221c5_b.jpg>

Váró:

<http://i57.fastpic.ru/big/2013/1029/91/441034f426d3a923cf7ba83b5b182291.jpg>

Egyjátékos:

<http://71513.stablerack.com/images/falklands-minefield1.jpg>

## Dokumentáció:

c# dokumentáció:

<https://msdn.microsoft.com/en-us/library/gg145045(v=vs.110).aspx>