**Záródolgozat**

Tanuló neve: Tóth Dániel Mátyás

Osztálya: 5/13.s

Elkészítési éve: 2022/23

**Záródolgozat adatlap**

**Záródolgozat készítőjének neve:**

Tóth Dániel Mátyás

**Elérhetősége(i):**

[dani.toth2003@gmail.com](mailto:dani.toth2003@gmail.com)

[furryvagyokihhi@gmail.com](mailto:furryvagyokihhi@gmail.com)

+36-30-603-1890

**A záródolgozat témája:**

Egy C#-ban fejlesztett program, aminek a feladata hogy tároljon a Minecraft nevű játékból tárgyakat és az elkészítési módjukat, és annak az adatait kezelje.

**Záródolgozat címe:**

MCraftingTree

**Konzulens tanár:**

Szegedi Barnabás

Kelt: Budapest, 2023.

.................................................................. .................................................................

a záródolgozat készítőjének aláírása a konzulens tanár aláírása

EREDETISÉG NYILATKOZAT

Alulírott tanuló kijelentem, hogy a záródolgozat saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült záródolgozatomban található eredményeket az intézmény egy példányban archiválhatja.

Budapest, 20

tanuló aláírása

**Záródolgozati konzultáció igazoló lap**

Alulírott Szegedi Barnabás konzulens tanár aláírásommal igazolom.

Tóth Dániel Mátyás nevű tanuló konzultációkon való részvételével.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dátum** | **Téma** | **Tanuló aláírása** | **Tanár aláírása** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tartalom

[1. Bevezetés 6](#_Toc130481339)

[2. Feladatspecifikáció 8](#_Toc130481340)

[3. Fejlesztői dokumentáció 9](#_Toc130481341)

[3.1 9](#_Toc130481342)

[4. Felhasználói dokumentáció 10](#_Toc130481343)

[5. Irodalomjegyzék 10](#_Toc130481344)

[6. Mellékletek 10](#_Toc130481345)

# Bevezetés

A programom készítésétéhez az inspiráció akkor jött, mikor óra közben játszottam a Minecraft nevű játékkal, és egy barátomnak teszteltem egy modifikált verzióját. Sokkal több tárgy, ellenség és barkácsolási recept volt benne, mint az alap játékban, ezért rengeteg mindent nem tudtam, hogy hogy kell megcsinálni, és hogy milyen tárgyakba kerülne. Annyit tudtam, hogy melyik tárgyat kell megszereznem ahhoz, hogy „győzzek”, amihez egy nagyon komplex recept tartozott. Ekkor volt egy gondolatom, hogy: „Hmm, de jó lenne hogyha lenne egy program vagy egy modifikáció ahhoz, hogy megmutassa nekem a szükséges alapanyagokat ehhez a tárgyhoz…”, és úgy döntöttem, hogy megcsinálom.

Először is, ki kellett találnom, hogy hogyan és mivel hozom létre, és mi kell ahhoz, hogy úgy működjön, ahogy azt én akarom. Ideálisan Java-ban lett volna jó megírni, hiszen a játék is Java 8, majd Java 17-ben íródott, és könnyen tudtam volna importálni az osztályfájlokat magából a .jar fájlból, de helyette a C# mellett döntöttem, mert ahhoz valamilyen szinten értek is, Java-val ellentétben. Megterveztem az adatbázis alapjait draw.io-ban, hogy könnyen tudjak változtatni az oszlopokon, és hogy ne kelljen szenvednem azzal, hogy valaminek a tárolására nem jó. A grafikai felülethez WPF-et használtam, mert gyorsabb, modernebb és több funkcionalitást nyújt a WinForms-hoz képest. Emellé még Material Design-t is használtam, ami a Bootstrap-hez hasonlóképpen egy felhasználói felület keretrendszer, hogy jobban nézzen ki a program, mint ahogy én azt meg tudnám írni nélküle. Kitaláltam, hogy akarom hogy a program kinézzen, milyen funkciói legyenek, és nekikezdtem megírni a programot, az adatbázissal kezdve. Entity Framework 6-ot használva megcsináltam a terveim alapján, és nagyon alap funkciókkal teszteltem, hogy be tud-e olvasni és lehet-e rá írni. Ezután megírtam, hogy lehessen hozzáadni tárgyakat az adatbázishoz, hogy dobja-e a megadott tárgyat egy ellenség, ha igen melyik és mekkora eséllyel, a tárgyakkal asszociált típus, tárgyakból készített receptek, legyen az kemencében kisütés, alkímiában készült főzetek, vagy csak egyszerűen barkácsolások. Ezután kidolgoztam, hogy megjelenjenek a tárgyak egy listában, a velük asszociált képekkel együtt, megcsináltam, hogy be lehessen tölteni recepteket azzal, hogy behúzol egy képet a barkácsasztal menüjébe a program közepén, és hogy kiszámolja hogy milyen tárgyak kellenek a megadott receptnek az elkészítéséhez, és mennyi belőlük. Végül hozzáadtam alap adatokat az adatbázishoz, tárgyak, recepteket és típusok amik megjelennek az alap játékban is, modifikációk nélkül.

Végül csak a tesztelés és ellenőrzés maradt, amik sikeresek voltak.

# Feladatspecifikáció

A szoftver lényege a Minecraft játékból tárgyak és receptek felvétele és kezelése, hogy tudjuk hogy a játékban a mennyibe kerülnek megadott tárgyak. Ezt egy egyszerűbb felülettel igyekeztem megoldani.

Funkciói:

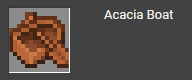
* Tárgyak kezelése és felvétele, típusokkal és dobási eséllyel
* Receptek felvétele
* Alap tárgyak megkeresése egy megadott tárgy receptjéhez

# Fejlesztői dokumentáció

## Általános ismertetés

Először ismerkedjünk meg az alap fogalmakkal:

* Modifikáció: Játékosok által készített változtatások a játékban, például extra receptek és/vagy tárgyak, legtöbbszőr CurseForge-ról vagy Modrinth-ról szerezhetőek be, de saját magunk is tudunk csinálni, ha ismerjük a Java-t.
* Tárgy: egy Minecraft-ból származó elem, amivel tudjuk befolyásolni a világot azzal, hogy letesszük a világba, felveszünk velünk valamit a világból, vagy csinálunk belőlük másik tárgyakat amik képesek az ezelőtt felsoroltakra.



1. ábra

* Recept: Egy megadott felhasználói felületen milyen tárgyakat kell felhasználni ahhoz, hogy más tárgyakat lehessen létrehozni belőlük. Ezeknek van 3 fajtája:
  + Barkácsoló asztal: egy háromszor hármas felületen lehet azonnal más tárgyakat létrehozni recepttől függően, vagy egy előre megadott alakzatban, vagy alakzat nélkül. Van egy megadott mennyisége hogy mennyi tárgyat kapsz egy megadott receptből.

A képen diagram látható

Automatikusan generált leírás

2. ábra

* + Kemence: Egy az egyhez átalakítás egy tárgyból a másikba, nem azonnali, de viszont ezt nem reprezentálom a programban. Használható ha valamit ki kell sütni/beolvasztani.

A képen szöveg, clipart, névjegykártya látható

Automatikusan generált leírás

3. ábra

* + A képen szöveg, csatlakozó látható

    Automatikusan generált leírásAlkímia asztal: Egy alkímiai hozzávalót belerakunk egytől háromig terjedő vízzel/főzettel teli üvegbe, és belefőzzük. Leginkább főzetek készítéséhez használatos.

4. ábra

Ezt az alkalmazást túlnyomó többségben a Minecraft játékosai fogják  
használni, azon belül azok akik egyedi és/vagy másik játékosok által terjesztett modifikációkat használnak a játékhoz, és leginkább ahhoz fogják használni, hogy megnézzék milyen nyersanyagokba, azaz annál jobban le nem bonthatóbb tárgyakba, kerül egy megadott tárgy a hozzá asszociált receptet használva. Ezért észben kell tartani hogy mi a legfontosabb miközben változtatásokat hozunk létre a programhoz: a tárgyak és a receptek vannak az előnyben, minden más extra; Ha valami blokkolja a tárgyak és/vagy receptek rendes működését, azt vagy törölni vagy változtatni kell kiadás előtt.

## Fejlesztői környezet

Az általam használt fejlesztői környezet a Visual Studio. Egy IDE *(Integrált Fejlesztői Környezet)* a C#-hoz. Ideális a könnyed Git hozzáférés és a NuGet bővítmények beszerzése miatt, de viszont csak Windows-on használható. Ha egy Linux disztribúción szeretné fejleszteni, akkor a MonoDevelop-ot tudom ajánlani, de tesztelni a programot Wine-ban kell, mert a WPF nem támogatott Linux-on.

# Felhasználói dokumentáció

# Irodalomjegyzék

# Mellékletek

## Ábrajegyzék

[1. ábra: Egy tárgy a programból 9](#_Toc130483132)