



Soter-Line Oktatási, Továbbképző és Szolgáltató Kft.

1073 Budapest, Erzsébet krt. 7. Tel: 887-2323; Fax: 887-2324, www.soterline.hu

Nyilvántartásba vételi szám: E-000201/2014

ZÁRÓDOLGOZAT

Mobilalkalmazás Cross Platform alapokon

Szoftverfejlesztő

OKJ száma: 54 213 05

Témavezető:

Nagy András

Készítette:

Mátyus Annamária

2020

Tartalomjegyzék

Köszönetnyilvánítás	3
1.Bevezetés.....	4
1.1. Problémafelvetés	4
1.2. Az applikáció célja	5
1.3. Felhasznált technológiák	6
1.3.1. .NET Framework	6
1.3.2. C# programozási nyelv	6
1.3.3. Microsoft Visual Studio	6
1.3.4. Xamarin.....	6
1.3.5. Android.....	7
1.3.6. iOS.....	7
1.3.7. Android vs iOS	7
1.3.8. Fejlesztői környezet.....	8
2. Felhasználói dokumentáció	10
2.1. Az applikációról	10
2.2. Rendszerkövetelmények.....	10
2.3. Használata	10
2.3.1. Telepítés és indítás	10
2.3.2. Első lépések, kezdőoldal	10
2.3.3. Új recept hozzáadása	12
2.3.4. Receptek megtekintése	13
2.3.5. Menü.....	15
2.3.6. Keresés	16
2.3.7. Információk	18
3. Fejlesztői dokumentáció.....	20
3.1. Az alkalmazás projekt	20
3.2. A dolgozat során használt fő cross-platform elemek.....	20
3.3. A projekt szerkezete	21
3.4. Az alkalmazás életútja és a projekt osztályai	22
3.4.1. App	22
3.4.2. StarterPage.....	23
3.4.3. MenuPage	25
3.4.4. RecipePage	26
3.4.5. NewRecipe	28
3.4.6. Recipe.....	30
3.4.7. SearchPage	31

3.4.8.	InfoPage.....	33
3.4.9.	Az MRecipe modell.....	34
4.	Tesztelés	36
4.1.	Black Box tesztelés	36
4.1.1.	A kezdőképernyő tesztelése	36
4.1.2.	Az új recept hozzáadásának tesztelése	37
4.1.3.	A recept adatlapjának tesztelése	40
4.1.4.	A menü tesztelése	40
4.1.5.	A keresés tesztelése	40
4.1.6.	Az információs oldal tesztelése	42
4.2.	Unit tesztelés	43
4.2.1.	Unit Tesztelés menete.....	43
5.	Összegzés	45
6.	Irodalomjegyzék.....	46
7.	Ábrák és táblázatok jegyzéke	47
8.	Mellékletek.....	48
9.	CD lemez tartalma.....	48
10.	Nyilatkozat	49

Köszönetnyilvánítás

Köszönettel tartozom Nagy Andrásnak a konzulensi és az oktató munkájáért.

A Soter-Line Oktatási Központ munkatársainak, akik munkájukkal támogatták tanulmányaimat az intézményben.

Végül, de nem utolsósorban köszönöm családomnak, hogy mindig bíztattak és motiváltak célom elérésében.

1.Bevezetés

Záródolgozatomban először ismertetem az applikáció fejlesztése folyamán használt technológiákat – többek között a programozási nyelvet, illetve a fejlesztői környezetet. Ezt követően bemutatom a programom felépítését és működését a felhasználói- és a fejlesztői dokumentáción keresztül. Bemutatom a tesztelés folyamatát és annak eredményeit. Végezetül pedig összegzem a fejlesztés folyamán szerzett tapasztalataimat és leírom az applikáció továbbfejlesztési lehetőségeit.

1.1. Problémafelvetés

Az évek során rengeteget gondolkoztam azon, hogyan lehetne a legegészségesebben táplálkozni úgy, hogy más fontos dolgokra is maradjon időm. Az egészséges táplálkozás az egyik alapköve annak, hogy hosszan és egészségesen éljünk. Ennek viszont ára van, mégpedig az időnk. A mindennapjaim fontos részét képezi az egészséges táplálkozás és ezzel együtt a főzés. Pedig a főzés időigényes, kezdve azzal, hogy ki kell találni mit is főzzünk magunknak vagy a családnak. Sokat beszélgettem erről az ismerőseimmel, hogy mégis mi veszi el a legtöbb időt, min lehetne spórolni, hogy maradjon időnk másra is és ne csak a főzésről és a munkáról szóljanak a hétköznapijaink. A megkérdezettek többségének az egészséges receptek megtalálása okozza a legnagyobb problémát. Sokan követnek különböző étrendeket – én magam is próbálok tejmentesen főzni és minimalizálni a húsfogyasztást– és ezeknek a kritériumoknak megfelelő receptek megtalálása nem könnyű dolog. De mi történik, miután megtaláltuk az igényeinknek megfelelő receptet?

Én személy szerint ezeket a recepteket sokszor a telefonom böngészőjében akár hónapokig megnyitva hagyom, viszont így pár hét elteltével harmincnál is több megnyitott oldal gyűlik össze, ami lassítja a telefon működését, illetve megnehezíti a megnyitott receptek kezelését is. Ezt szoktam orvosolni azzal, hogy üzenetben küldöm el magamnak, de egy idő után már ott sem tudom visszakeresni és számon tartani a receptjeimet, így jó eséllyel többé nem fogom megnyitni ezeket.

Ennek a problémának a megoldására találtam ki és fejlesztettem a *RecipeApp* névre keresztelt telefonos applikációt. Itt össze lehet gyűjteni kedvenc recepteket és könnyedén vissza is lehet keresni őket akár név, címke vagy hozzávaló szerint.

1.2. Az applikáció célja

Az applikáció fejlesztése során arra törekedtem, hogy egy olyan telefonos alkalmazást hozzak létre, ami szélesebb körben is képes kiszolgálni a felhasználóit.

Az okostelefonok térnyerésének és gyors fejlődésének köszönhetően ma már szinte nincs is olyan tevékenység, amit ne tudnánk a mobiltelefonunkról elvégezni – legyen szó banki átutalásról, jegyvásárlásról, szállásfoglalásról. Jelenleg körülbelül 5 milliárd mobil felhasználó van a világon, így nem lehet figyelmen kívül hagyni ezt a piacot sem. [1] Ha vállalkozásunk nem rendelkezik mobilalkalmazással, hatalmas piactól esik el.

Ekkora fejlődés és felhasználói szám mellett jogos az elvárás, hogy az egyszerűbb teendőinket is zavartalanul tudjuk akár utazás közben intézni mobilunkról, legyen szó vásárlásról, receptkeresésről, videónézésről, chatelésről vagy weboldalak megnyitásáról. Ma már a fiatalabb és idősebb korosztály is széles körben kihasználja az okostelefonok által biztosított technológiai szabadságot. Egy dolog viszont közös minden felhasználóban: szeretik az egyszerűen és átláthatóan kezelhető felületeket.

A következő szempontokat tartottam a fejlesztés során szem előtt:

- letisztult megjelenés, egyszerű használat
- receptek megjelenítése a főoldalon abc rendben
- összetett keresési funkció
- receptek szerkesztésének és törlésének lehetősége
- információ a jelenleg működő funkciókról, valamint a várható fejlesztésekről
- kapcsolatfelvételi lehetőség, amennyiben problémát észlelnének vagy javaslatuk lenne a fejlesztést illetően

Nem volt céлом egy közösségi platform létrehozása, ahol a felhasználók megoszthatják egymással a receptjeiket. Ilyen telefonos alkalmazások már léteznek, de a sok lehetőség, egymás felhasználói profiljának követése és egyéb funkciók miatt elterelhetik az ember figyelmét a lényegről – a megfelelő recept megtalálásáról – és hosszú percekre, akár órákra köthetik le a felhasználó figyelmét. Ebből kiindulva szerettem volna egy lényegre törő, valóban segítséget nyújtó applikációt létrehozni.

1.3. Felhasznált technológiák

1.3.1. .NET Framework

A .NET egy fejlesztői platform, amelyet a Microsoft hozott létre számos különféle alkalmazás létrehozására. (Eszközből, programozási nyelvekből és könyvtárakból áll, különféle alkalmazások felépítéséhez.) A .NET keretrendszer segítségével több nyelvet, szerkesztőt és könyvtárat használhatunk webes, mobil, asztali, játék, IoT alkalmazások létrehozására. A kiterjedt osztálykönyvtárakkal, az API-kal, a többnyelvű támogatással a .NET az egyik leghatékonyabban használható platform a fejlesztők számára. Az alap platform olyan komponenseket biztosít, amelyek minden különféle alkalmazástípusra vonatkoznak – C# programozói nyelv, alap könyvtárak, szerkesztők és eszközök (tools) Windows, Linux, macOS és Docker részére.[2]

1.3.2. C# programozási nyelv

A C # egy egyszerű, modern, objektumorientált és típusbiztos programozási nyelv, ami lehetővé teszi a programozók számára, hogy a .NET keretrendszerben különböző alkalmazásokat hozzanak létre. A .NET fő programozási nyelve. 1999-ben kezdték meg a fejlesztését, amit Anders Hejlsberg vezetett. Platform független nyelv – létezik Linux és Mac fordító is -, de a leghatékonyabban a Microsoft implementációjával működik. [3]

1.3.3. Microsoft Visual Studio

Ez egy integrált fejlesztői környezet (integrated development environment - IDE). Többek között biztosít szerkesztés könnyítő- és hibakereső funkciót, kódkiegészítő funkciót, fordítókat, grafikus tervezőt a fejlesztői folyamatok megkönnyítésére.

A Visual Studio elérhető Windows és Mac rendszerekhez egyaránt. A Visual Studio 2019 a platformok közötti és a mobil alkalmazások fejlesztésére lett optimalizálva. [4]

A fizetős változatok mellett létezik egy ingyenesen elérhető csomag is, cserébe telepítés után 30 nappal egy regisztráció szükséges, az ezután kapott kóddal viszont a továbbiakban zavartalanul tudjuk használni. [5]

1.3.4. Xamarin

A Xamarin a Microsoft tulajdonában lévő, nyílt forráskódú, ingyenes, cross-platform alapú szoftverplatform Android és iOS alkalmazások készítéséhez .NET és C# segítségével. A

Xamarin kiegészíti a .NET fejlesztői platformot eszközökkel és könyvtárakkal, kifejezetten Android és iOS alkalmazások készítéséhez. A Xamarin.Forms egy nyílt forráskódú mobil felhasználói felület, amellyel iOS, Android és Windows alkalmazásokat lehet készíteni a .NET segítségével egyetlen megosztott kódbázisból. [6]

1.3.5. Android

Az Android az egyik legelterjedtebb mobil operációs rendszer elsősorban okostelefonokra és táblagépekre tervezve, ami a Linux kernel és más nyílt forráskódú szoftverek módosított verzióján alapul. Fejlesztését az Android Inc. kezdte el, 2005-ben a Google felvásárolta, majd az Open Handset Alliance nevű szövetség vette át. Java nyelven lehet fejleszteni, vezérlése Java programkönyvtárakon keresztül lehetséges. 2008-ban jelent meg az első androidos telefon. A Google Playből lehet letölteni új alkalmazásokat a készülékünkre, ahol vannak ingyenes és fizetős alkalmazások egyaránt.

Az Android platform létrehozásának célja egy egységes nyílt forrású operációs rendszer létrehozása mobiltelefonokra. Jelenleg a legfrissebb verziója az Android 10, ami 2019 őszén jelent meg. [7]

1.3.6. iOS

Az Apple mobil operációs rendszere. Kizárólag az Apple saját termékeire fejlesztik (pl: iPhone, iPod touch, iPad). Az i (én) a cég szokásos személyes névmása, az OS pedig az „operating system”, vagyis az operációs rendszer rövidítése. 2009. nyarán már több mint 50.000 alkalmazás volt elérhető az iPhone-ra. Az alkalmazásokat az Apple készülékekre az Apple Storeból lehet letölteni.

Nehézség, hogy a fejlesztéshez és a teszteléshez fejlesztői jogosultság kell, nem elég egy készülék, mint az Androidnál. Jogosultság nélkül futtatni sem lehet a programot a telefonon vagy emulátoron.

Az SDK (Software Development Kit – szoftverfejlesztői csomag) ingyenesen letölthető, de ahhoz, hogy valaki közvetlenül az eszközön tudjon tesztelni, és szoftvert adjon ki, az Apple Developer Program tagja kell, hogy legyen, amihez az Apple engedélye is szükséges. [8]

1.3.7. Android vs iOS

Ha szélesebb körben, globálisan szeretnénk, hogy az appunk ismert legyen, akkor érdemes cross platform alapokkal dolgozni, hogy az alkalmazás mindkét eszközön elérhető legyen. Ez

mindkét rendszer ismeretét igényli, mivel a specifikus kódrészeket a saját projekt fájlukban kell testreszabni. Míg az Android nyílt forráskóddal, az iOS zárt forráskóddal rendelkezik, ami megnehezítheti a fejlesztő dolgát, mivel fejlesztői jogosultságot kell szereznie, hogy Apple készüléken is futtathassa, tesztelhesse a programot.

Az Android alkalmazásokat általában Java nyelven írják, míg az iOS hivatalos nyelve a Swift, ami egyes fejlesztők szerint egyszerűbb, mint a nyílt forráskódú, több eszközt és bővítményt használó Java. Ebből kifolyólag viszont az Apple egy sokkal zártabb rendszert hozott létre, így a fejlesztés is kevesebb eszközön lesz csak elérhető. Ezen felül az Apple App Storenak szigorúbb szabályai, minőségi elvárásai és hosszabb elbírálási ideje is van, így az alkalmazás elbírálása hosszabb időt vehet igénybe. Az Apple ennek a zárt és védett rendszerének köszönheti sikerét a vállalati piacon.

Ezzel szemben az Android alkalmazások fejlesztése több rugalmasságot biztosít. Mivel nyílt forráskódú, így nagyobb mozgásteret kínál az alkalmazás testreszabásában. Ez ugyanakkor azt is jelenti, hogy az Android ezáltal jobban ki van téve a rosszindulatú szoftverek veszélyének. [9]

1.3.8. Fejlesztői környezet

Processzor	Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
Memória	8 GB
Tárhely	250 GB SSD
Grafikus kártya	Intel UHD Graphics 630
Operációs rendszer	Windows 10 Pro 64 bites

1. táblázat: Fejlesztői környezet hardveres része

Integrált fejlesztői környezet (IDE)	Microsoft .NET Framework (verzió: 4.8.03752) Microsoft Visual Studio Community 2019 (verzió: 16.2.3)
Programnyelv	C#

2. táblázat: Fejlesztői környezet szoftveres része

Modell	Redmi Note 6 Pro
Android verziója	8.1.0
CPU	Nyolcmagos (1.8 GHz)
RAM	4 GB

3. táblázat: A fejlesztéshez használt Androidos okostelefon adatai

Modell	iPhone X
iOS Verzió	11.3

4. táblázat: A fejlesztéshez használt iOS emulátor adatai

2. Felhasználói dokumentáció

2.1. Az applikációról

A *RecipeApp* egy receptek összegyűjtésére, tárolására létrehozott telefonos applikáció. Mind Android, mind pedig Apple termékeken is futtatható. Használatával könnyedén összegyűjthetjük, visszakereshetjük, valamint frissíthetjük kedvenc receptjeinket. Mivel a recepteket az applikáció a telefon tárhelyére menti, nincsen szükség internetkapcsolatra ahhoz, hogy betöltse a már elmentett recepteket vagy újakat tudjon elmenteni.

2.2. Rendszerkövetelmények

A minimum Android verzió 5.0, a target (cél) pedig az Android 9.

Alkalmazás mérete	~15.19 MB
Felhasználói adat	hozzáadott receptek mennyiségétől függ
Memóriaigénye	~130 MB

5. táblázat: A program rendszerkövetelménye

2.3. Használata

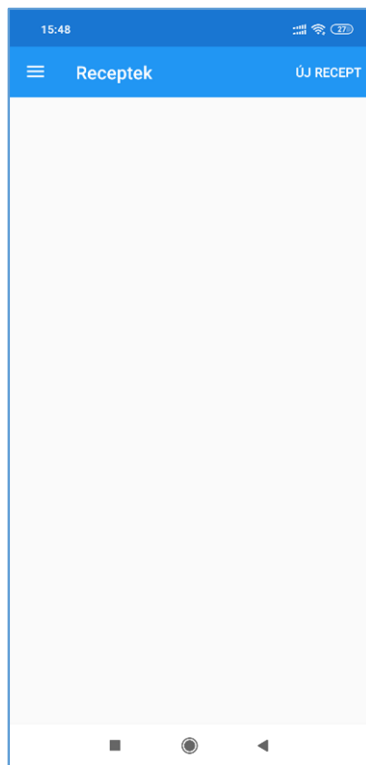
2.3.1. Telepítés és indítás

Két lehetőség is van az applikáció telepítésére. Az első, hogy megnyitjuk Visual Studioban a forráskódot és lefuttatjuk a telefonunkra – miután az eszköz fejlesztői beállításában bekapcsoljuk az USB-hibakeresés módot, illetve a „Telepítés USB-ről” beállítást és végül a készüléket csatlakoztattuk a számítógéphez.

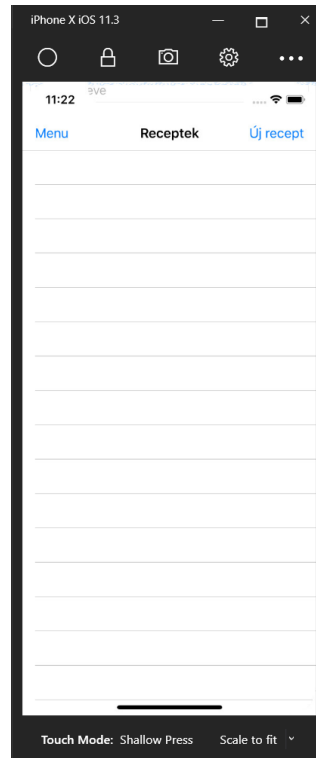
A másik lehetőség, hogy a programfájlt átmásoljuk a készülékünk tárhelyére. Ezt megtehetjük a mellékelt CD-ről. Ha ez megtörtént, a készülékünk fájlkezelőjében meg kell keresnünk a *RecipeApp.apk* nevű telepítőt, amelyre rányomva azt telepíthetjük.

2.3.2. Első lépések, kezdőoldal

Amikor megnyílik az applikáció, a *Receptek* oldal tölt be, ami az első használatkor még üres lesz, mivel még nincsen hozzáadott receptünk. A felső részen egy toolbar található, aminek bal oldalán a menü és az oldal neve, jobb oldalon pedig az új recept gomb található.



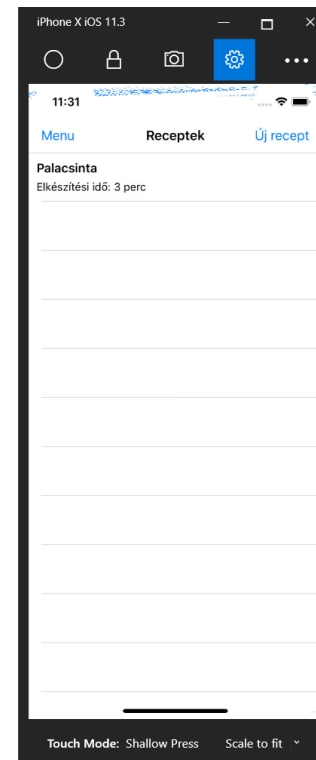
1. ábra: Android kezdő oldal – recept nélkül



2. ábra: iOS kezdőoldal – recept nélkül



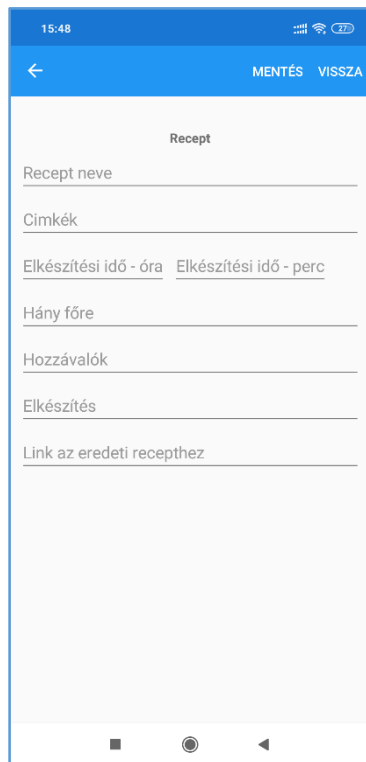
3. ábra: Android kezdőoldal – receptekkel



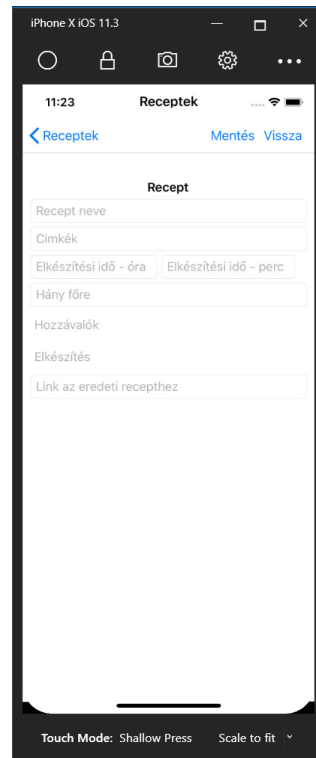
4. ábra: iOS kezdőoldal - recepttel

2.3.3. Új recept hozzáadása

Új receptet az új recept gombra kattintva tudunk hozzáadni. Ekkor az új recept hozzáadásához szükséges űrlap tölt be, ahol a következő adatokat tudjuk megadni a receptünkhöz: recept neve, címkék, elkészítési idő (óra és perc), hány főre, hozzávalók, elkészítés, valamint az eredeti recept linkje. Ezeket az adatokat manuálisan kell beírni, vagy az eredeti oldalról kimásolva is be tudjuk illeszteni.

The screenshot shows the 'Recept' form in an Android application. The form has a blue header bar with a back arrow and the text 'MENTÉS VISSZA'. The form fields are: 'Recept neve', 'Címkék', 'Elkészítési idő - óra' and 'Elkészítési idő - perc' (two separate input fields), 'Hány főre', 'Hozzávalók', 'Elkészítés', and 'Link az eredeti recepthez'. The bottom of the screen shows the Android navigation bar.

5. ábra: Android – új recept hozzáadása

The screenshot shows the 'Recept' form in an iOS application. The form has a white header bar with a back arrow and the text 'Mentés Vissza'. The form fields are: 'Recept neve', 'Címkék', 'Elkészítési idő - óra' and 'Elkészítési idő - perc' (two separate input fields), 'Hány főre', 'Hozzávalók', 'Elkészítés', and 'Link az eredeti recepthez'. The bottom of the screen shows the iOS home indicator bar.

6. ábra: iOS – új recept hozzáadása

Ezek közül a recept neve, a hozzávalók és az elkészítés kötelezően kitöltendő. Ezek megadása nélkül nem tudjuk elmenteni a receptet. Ha mégis megpróbáljuk, akkor a mezők alatt megjelenik egy piros üzenet: „Kötelező mezők: Recept neve, Hozzávalók, Elkészítés”.

7. ábra: Android – kötelező mezők

8. ábra: iOS – kötelező mezők

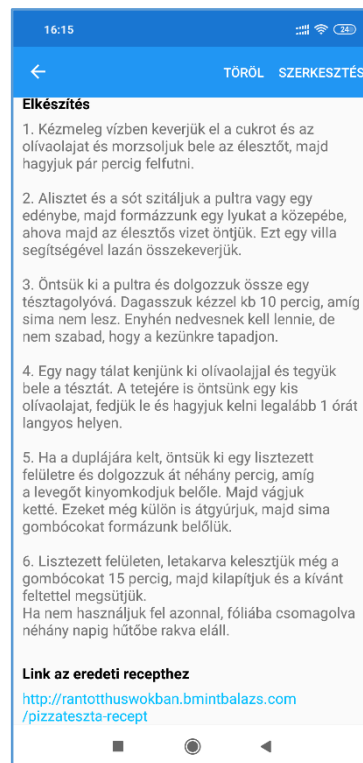
Miután beírtuk a recept adatait, a mentés gombra kattintva menthetjük a receptet, de ha meggondoltuk magunkat, a vissza gombra nyomva elvethetjük az esetlegesen már beírt adatokat és visszakerülünk a receptek oldalára, ahol az eddig beírt receptjeinket találjuk.

2.3.4. Receptek megtekintése

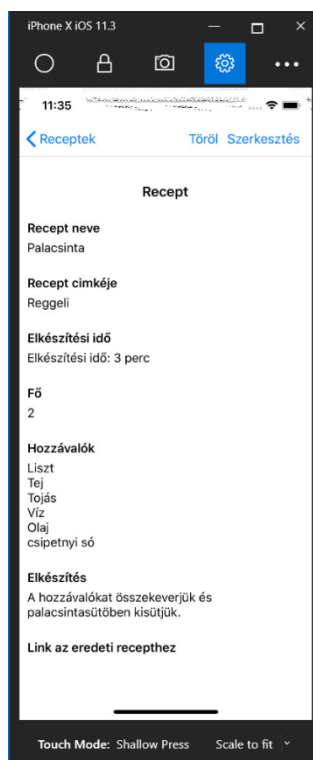
Amennyiben a kezdőoldalon lévő listában megtaláltuk a megnyitni kívánt receptet, egyszerűen csak rá kell nyomnunk és megjelenik a recept adatlapja. Itt lehetőségünk van elolvasni a receptet, ami ha túl hosszú és nem fér ki a képernyőre, akkor az ujjunk képernyőn való fel és le mozgatásával tudjuk görgetni. Amennyiben van link megadva a recepthez, rá tudunk kattintani és az eredeti recept oldala megnyílik a telefon böngészőjében. Fent a toolbarban bal oldalon található egy vissza gomb (nyíl), ami visszavisz a kezdőoldalra, a jobb oldalon pedig a törlés és a szerkesztés gomb található. A törlésre kattintva el tudjuk távolítani a receptet, míg a szerkesztésre nyomva lehetőségünk van átírni.



9. ábra: Android – recept adatlapja 1



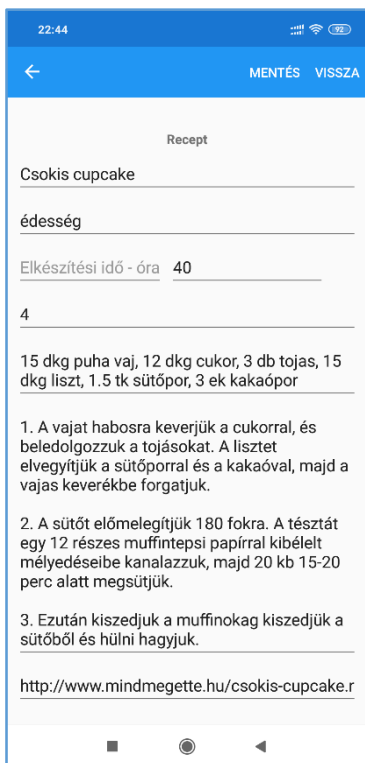
10. ábra: Android – recept adatlapja 2



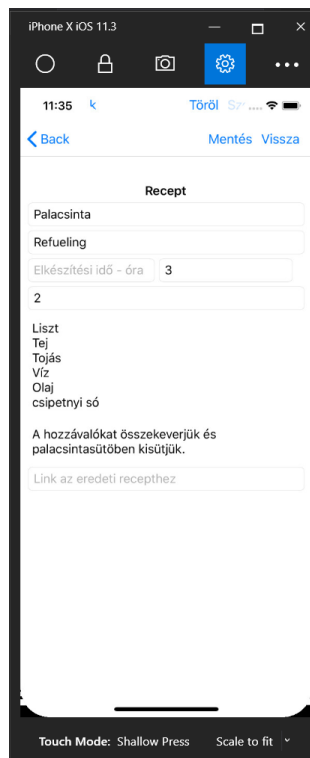
11. ábra: iOS – recept adatlapja

A szerkesztésre kattintva az új recept hozzáadása fejezetben bemutatotthoz hasonló űrlap jelenik meg kitöltve a recept korábban megadott adataival. Itt lehetőségünk van menteni a

toolbar jobb oldalán található mentés gombra nyomva – ami visszavigáz minket a recept adatlapjára – valamint, ha meggondoltuk magunkat és mégsem szeretnénk módosítani a receptet, a vissza gombra nyomva elvethetjük a módosításokat és szintén a recept adatlapjára lépünk vissza.



12. ábra: Android – recept szerkesztése

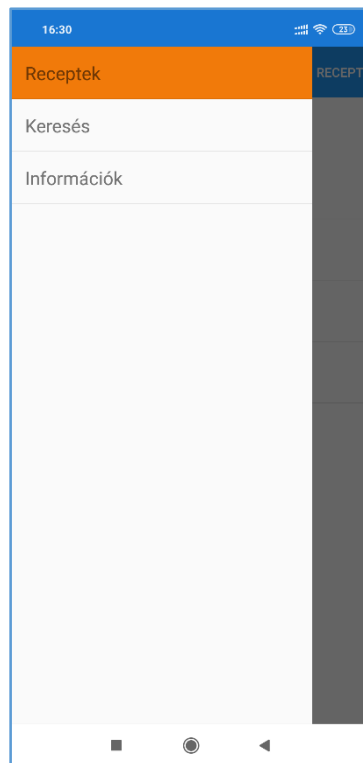


13. ábra: iOS – recept szerkesztése

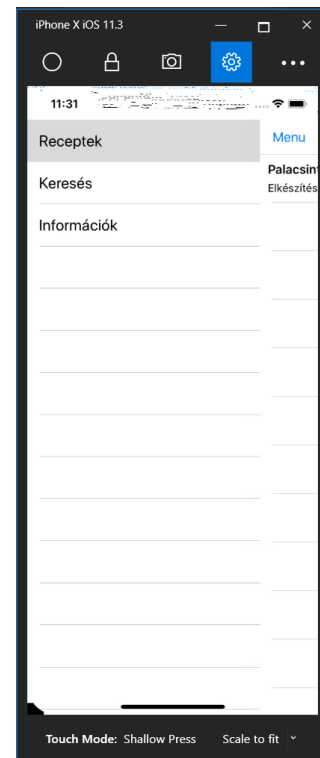
2.3.5. Menü

A kezdőoldalon a toolbar bal oldalán található a menü, amire ha rákattintunk, balról beúszik a menü, amiben a következő menüpontok közül választhatunk:

- Receptek: ez a kezdőoldal, ez tölt be az applikáció elindításakor, itt találjuk a már mentett receptjeink listáját abc sorrendben
- Keresés: itt részletesen tudunk keresni a receptjeink között
- Információk: a jelenlegi funkciókról, valamint a várható fejlesztésekről ad információt



14. ábra: Android menü

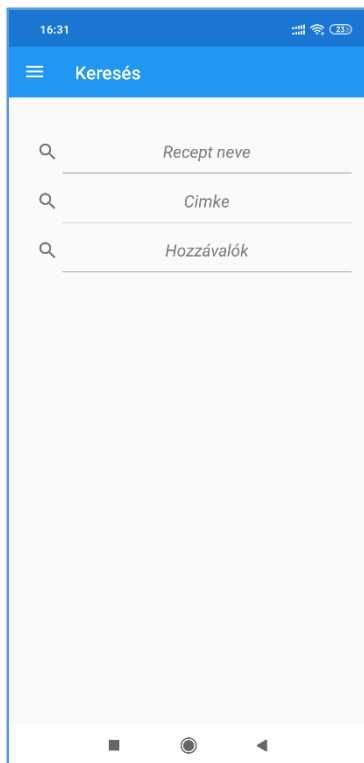


15. ábra: iOS menü

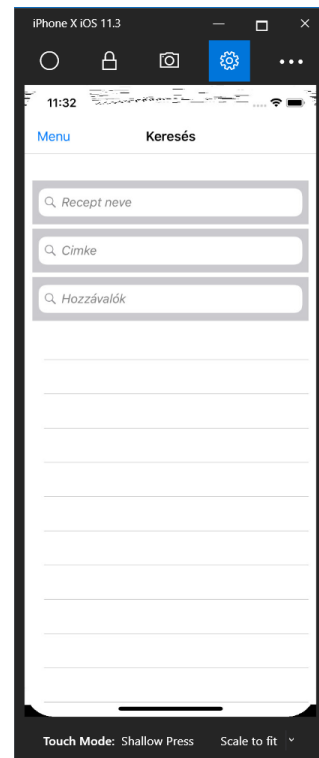
Amennyiben szeretnénk a menüt bezárni, elég a mellette lévő elhomályosított részre kattintanunk vagy a menüt balra húzva is visszacsúszthatjuk.

2.3.6. Keresés

A menüben a Keresésre kattintva egy részletes keresésre nyílik lehetőségünk. Itt tudunk a recept neve, címkéje, illetve a hozzávalók szerint keresni. Ezek külön-külön is hozhatnak fel találatot, de bármilyen kombinációban tudjuk ezeket a mezőket együttesen is használni.

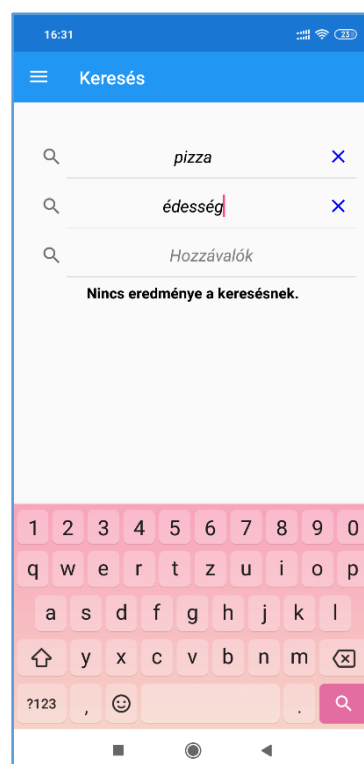


16. ábra: Android – Keresés

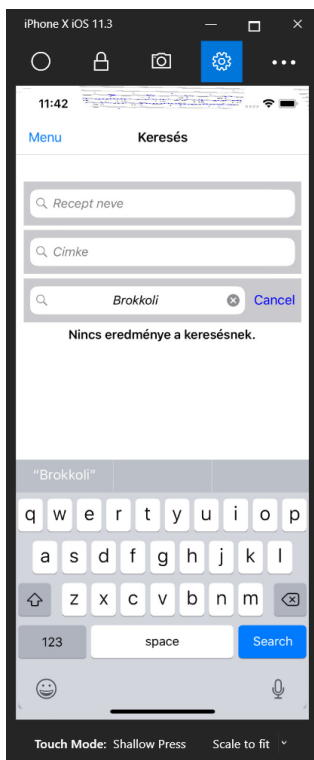


17. ábra: iOS – Keresés

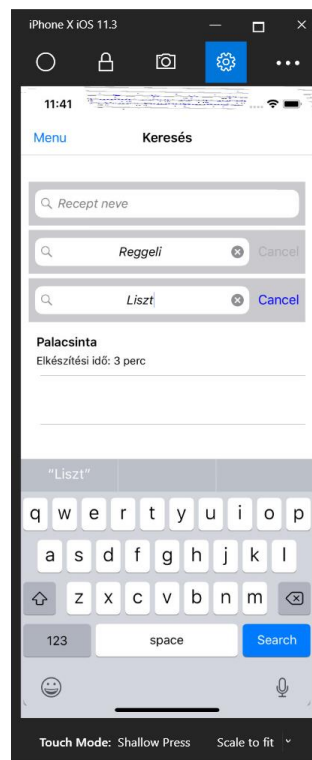
Amennyiben nincsen a beírtaknak megfelelő találat, a keresőmezők alatt a „Nincs eredménye a keresésnek” üzenet jelenik meg.



18. ábra: Android – nincs találat az összetett keresésnek



19. ábra: iOS – nincs találat a keresésnek



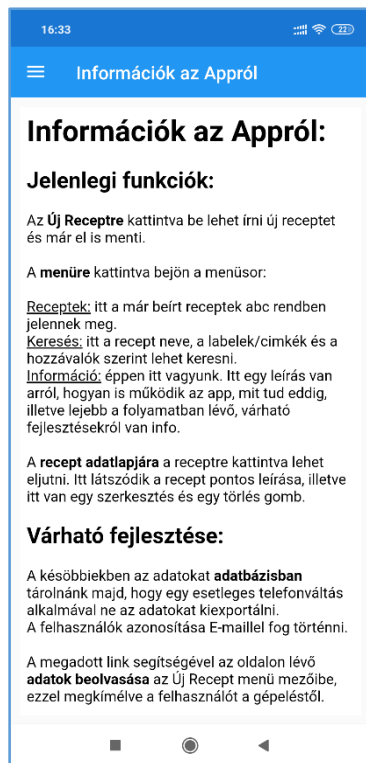
20. ábra: iOS – összetett keresés

Mind a három keresőmezőre igaz, hogy a beírt szöveget a már eltárolt receptek megfelelő mezőiben a teljes szövegben keresi.

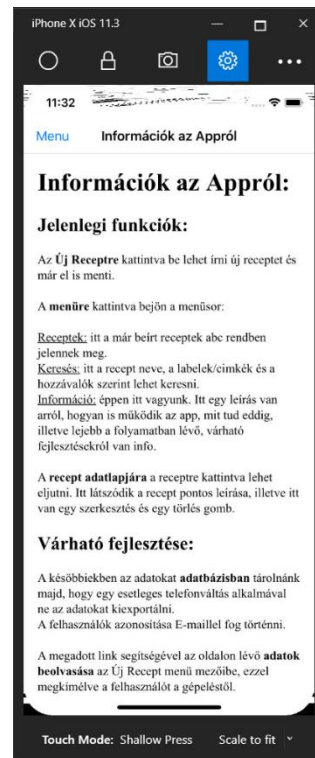
A találatokat a keresőmező alatt listában jeleníti meg – a kezdőoldalhoz hasonlóan. A listában a receptre kattintva a recept adatlapjára navigálja a felhasználót a program.

2.3.7. Információk

Az információk a jelenleg már működő funkciókat írja le segítségként, valamint a várható fejlesztésekről ad információt. Ezen az oldalon keresztül tud a fejlesztő kommunikálni a felhasználókkal, itt tudja tájékoztatni őket az estleges változásokról.



21. ábra: Android – Információs oldal



22. ábra: iOS – Információs oldal

Az oldal alján található a kontakt pont, aminél egy e-mail cím van megadva. A felhasználók ezt az e-mail címet használva jelezhetik az esetleges problémákat, illetve esetleges ötleteiket a fejlesztőnek.



23. ábra: Android – Információs oldal/Kontakt

3. Fejlesztői dokumentáció

A fejlesztéshez a C# programozási nyelvet, valamint a Visual Studio 2019 Community verziót használtam.

3.1. Az alkalmazás projekt

Mobiltelefonra fordítható és azon futtatható projektet is létre tudunk hozni a Visual Studio segítségével, miután telepítettük a fejlesztőkörnyezet megfelelő bővítményét, név szerint a Xamarin.Forms-t.

Az újonnan létrehozott solution három alprojektet foglal magába:

- Az első a közös kódbázist tartalmazó alprojekt - melynek neve alapértelmezés szerint megegyezik a solution nevével – tartalmazza a közös kódbázist, amely mind Androidon, mind iOS-en is működik, minden kellemetlenség nélkül (a hivatalos dokumentációt követve).
- A második a <Solution neve>.Android névvel ellátott alprojekt, amely alapból tartalmaz egyes, Android alapú eszközökön való futtatáshoz szükséges alap kódbázist, az erőforrások mappáit (például.: képek, OS specifikus layout-ok, egyéni betűtípusok).
- A harmadik a <Solution neve>.iOS nevű alprojekt. Ezen belül találjuk az iOS alapú eszközökön való futtatáshoz szükséges alap kódbázist és az erőforrások mappáit (szintén ide lehet betenni a képeket, OS specifikus layout-okat, egyéni betűtípusokat).

Minden olyan kód, amely csak az adott operációs rendszeren értelmezett (legyen az egy UI-t érintő elem, vagy az eszköz speciális hardverét vezérlő algoritmus), az OS specifikus alprojektbe kell, hogy kerüljön.

A továbbiakban kizárólag a közös kódbázist tartalmazó alprojektre fogok hivatkozni (a továbbiakban **projekt**), mivel a szakdolgozatomhoz elkészített szoftver nem tartalmaz OS specifikus algoritmust és kódot, mindent sikeresen implementáltam a projekten belül.

3.2. A dolgozat során használt fő cross-platform elemek

- Page

A Page objektumok egy cross-platform alapú mobil alkalmazás képernyőit reprezentálják.

- Navigation

Az alkalmazás során a különböző oldalakat a `NavigationPage` segítségével lehetséges könnyedén lecserélni, ami egy last-in, first-out (LIFO) alapú verem megoldás.

- XAML

A Microsoft által fejlesztett deklaratív XML alapú leíró nyelv. Egyszerűvé teszi a Xamarin.Forms alapú applikációkhoz való UI fejlesztést.

- Events, Sender, EventArgs

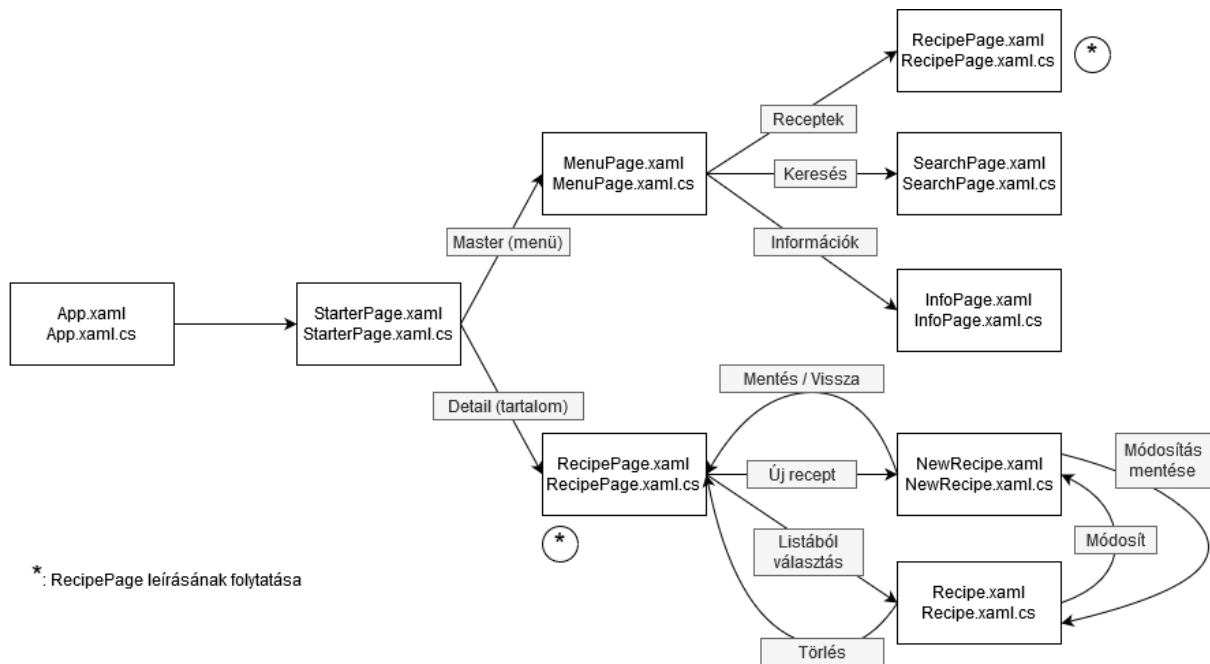
Az implementált applikációnak képesnek kell lennie különféle interakciókat kezelni, legyen az egy gombnyomás, vagy gépelésre keresés.

Az eseménykezelés egy módja függvények hozzákötése objektumok eseményeihez. A gomboknak létezik `Clicked` nevű eseménye, míg a `SearchBar` objektumnak van `TextChanged` eseménye. Az eseményeket nevezzük Event-eknek. Minden olyan függvénynek, amit egy Event-hez kötünk, kötelezően 2 paramétert kell tartalmaznia. Az első paraméter a `Sender` objektum, amelyik megadja, hogy melyik elemre fut le az esemény. A második paraméter pedig az `EventArgs`, ami tartalmazza az esemény paramétereit.

3.3. A projekt szerkezete

A projekt az MVVM mintára épül, azaz a Model-View-ViewModel elvet követi:

- Model: Az alkalmazásnak egy modellje van, amely egy recept leírását adja meg (`MRecipe.cs` fájl)
- View: A grafikus felületet `.xaml` kiterjesztésű fájlokkal lehetséges leírni
- ViewModel: Minden grafikus felületet leíró `.xaml` fájlhoz tartozik egy fájl (ezek neve `.xaml.cs`-re végződik), melyekben a `ViewModel`-t van lehetőség megvalósítani – a `ViewModel` implementálását akár egy külön osztállyal is meg lehet tenni igény szerint. Ezek a fájlok biztosítják a grafikus felületet leíró xml attribútumokhoz tartozó értékek halmazát, illetve a `xaml` fájlok inicializálását is elvégzik az `InitializeComponent()` függvény segítségével.



Text

24. ábra: Felépítés ábra

3.4. Az alkalmazás életútja és a projekt osztályai

3.4.1. App

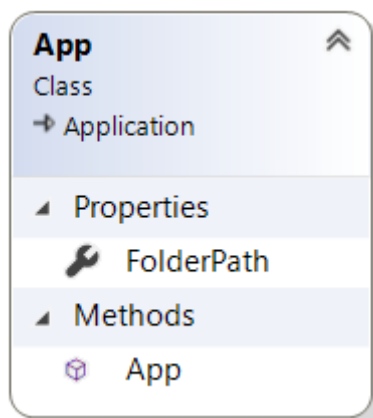
Az alkalmazást elindítva elsőnek az App nevű fájlok töltődnek be, melyeknek a feladata:

App.xaml	Az alkalmazás futása alatt elérhető erőforrásokot (például: értékek, színek) helyezhetjük el az <Application.Resources> xml tag-en belül.
App.xaml.cs	A konstruktorban van lehetőség megadni, hogy melyik View legyen az alkalmazás kezdő képernyője. A megvalósított alkalmazásom esetén ez a StarterPage, melyet az app indulásakor betölt a kód.

Az App.xaml.cs függvényei:

- App() konstruktor:

Az alkalmazás inicializálásáért felelős speciális függvény. Továbbá beállítja a recepteket tartalmazó speciális rendszermappa elérési útvonalát, illetve az alkalmazás főoldalát, a StarterPage-et.



25. ábra: App.xaml.cs

3.4.2. StarterPage

A StarterPage felelős a fő nézet megvalósításáért:

StarterPage.xaml	<p>Az alkalmazás fő nézetét az úgynevezett Master-Detail minta jellemzi. A Master a bal oldalról behúzható menüt, a Detail pedig a képernyőt kitöltő tartalmi részt (Page objektum) jelenti az implementált alkalmazásom esetén. A menüben navigálva mindig különböző tartalom jelenik meg választástól függően.</p> <p>A Master tartalma állandó, minden esetben a MenuPage tartalma tűnik elő három menüponttal, amikor a menüt a képernyő bal oldaláról behúzzuk.</p> <p>A Detail tartalma az alkalmazás elindulásakor minden esetben a RecipePage-ben meghatározott értékeknek megfelelően kerül betöltésre – A receptek listája és az új recept hozzáadása gomb jelenik meg. A Detail tartalma minden esetben egy NavigationPage-be kerül bele, ezáltal könnyítve a navigálást az egyes aloldalak között.</p>
------------------	--

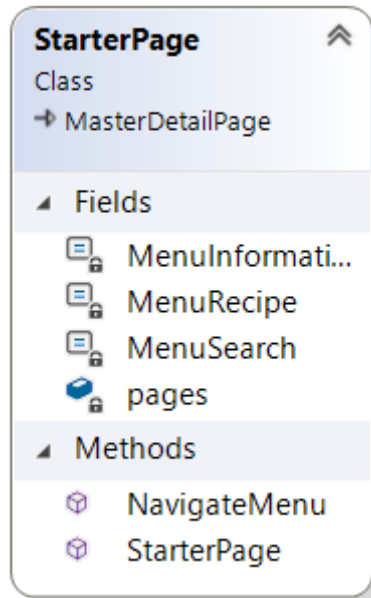
StarterPage.xaml.cs	<p>Ez az osztály végzi a menüben (MenuPage) történt interakció (esemény) hatására szükséges képernyő cserét.</p> <p>Továbbá implementáltam egy cache mechanizmust, hogy navigáció esetén ne hozzon létre minden esetben egy új példányt az egyes oldalakból (Page).</p>
---------------------	---

A StarterPage.xaml.cs változói:

- MenuRecipe, MenuSearch, MenuInformation:
A menü elemeinek a sorszámait jelzik.
- pages:
Cache objektum: ebbe kerülnek a gyorsítótárazott Page objektumok.

A StarterPage.xaml.cs függvényei:

- StarterPage() konstruktor:
A MasterDetail Page inicializálásáért felelős konstruktor, továbbá inicializálja az alkalmazás egyes oldalait tároló, cache-elésért felelős Dictionary<int, NavigationPage> típusú változót.
- NavigateMenu(int id) függvény:
A paraméterként átadott id változóhoz tartozó Page objektum betöltéséért felelő függvény. Amennyiben talál a cache objektumban az id változóhoz rendelt értéket, úgy azt tölti be a Detail objektumba, ellenkező esetben létrehozza a megfelelő Page objektumot, elhelyezi a cache objektumban, majd betölti a Detail objektumba is a Page-t.



26. ábra: StarterPage.xaml.cs

3.4.3. MenuPage

A MenuPage felelős a menü kezeléséért.

MenuPage.xaml	<p>A Master (menü) kinézetét leíró xaml elemeket tartalmazó leíró fájl.</p> <p>Az alkalmazás három menüpontot tartalmaz:</p> <ul style="list-style-type: none"> • Receptek (RecipePage) • Keresés (SearchPage) • Információk (InfoPage)
MenuPage.xaml.cs	<p>A menü inicializálásáért felelős osztály. Továbbá a menüben kiváltott esemény kezeléséért felel.</p>

MenuPage.xaml.cs függvényei és változói:

- MenuPage() konstruktor:

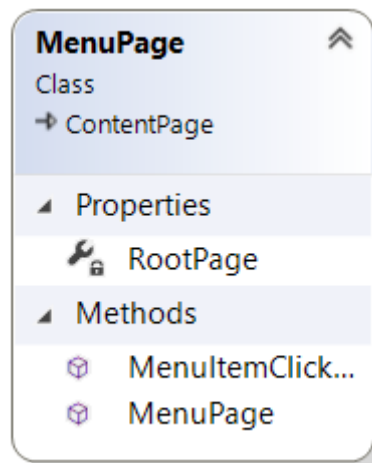
A MenuPage inicializálásáért felelős konstruktor. Továbbá ezen belül van lehetőség megadni a menüpontokat egy string objektumokat tároló listaként. Hozzárendeli az előbb említett listát a MenuPage.xaml-ben definiált RecipeMenuList nevű listanézethez. Legvégül kijelöli a menü alapértelmezettként kiválasztott elemét, amely alapértelmezetten az első eleme a listának („Receptek” menüpont).

- MenuItemClicked(object sender, SelectedItemChangedEventArgs e) függvény:

Egy listaelem kiválasztásakor bekövetkező ItemSelected esemény kezelését végző függvény, amely megkapja paraméterként a küldő objektumot (ebben az esetben egy lista nézet) és az esemény paramétereit. A függvény ellenőrzi, hogy valóban történt-e kiválasztás a listában: amennyiben igen, a kiválasztott elem indexét felhasználva meghívja a RootPage NavigateMenu(id) függvényét.

- RootPage osztályváltozó:

Ez a változó az alkalmazás aktuálisan beállított MainPage objektumára mutat, amely az App.xaml.cs fájlban kerül beállításra, és minden esetben egy StarterPage típusú objektumot tárol.



27. ábra: MenuPage.xaml.cs

3.4.4. RecipePage

A RecipePage tölti be a recepteket, illetve irányít át gombnyomásra az új recept aloldalra:

RecipePage.xaml	A receptek listáját leíró xaml fájl. Ezen kívül tartalmazza az oldalon megjelenő Toolbar-ban szereplő új recept hozzáadása gombot is.
RecipePage.xaml.cs	Azon kívül, hogy inicializálja a hozzá tartozó xaml objektumot, betölti az eszköz tárhelyére elmentett recepteket, illetve kezeli az új recept gombra kattintás és a listában való elemre kattintás eseményeket.

A RecipePage.xaml.cs változói:

- allRecipes:

Ebben a statikus listában tároljuk az összes receptet, amelyet a LoadRecipes() függvény frissít. Továbbá ebben a listában történik a keresés a SearchPage-en belül.

A RecipePage.xaml.cs függvényei:

- RecipePage() konstruktor:

A RecipePage inicializálásáért felelős konstruktor.

- OnAppearing() függvény:

Egy olyan beépített függvény, amely minden esetben lefut, amikor előtérbe kerül a RecipePage objektum. Ez a függvény felelős azért, hogy betöltse a recepteket és feltöltse a RecipePage.xaml-ben definiált RecipeList lista nézetet. A recepteket abc szerint rendezi az OrderBy függvénnyel. Továbbá frissíti a keresésnél használt statikus allRecipes listát is - a listán hívott Clear függvénnyel eltávolítja az összes eltárolt receptet, majd újra hozzáadja, ezzel elkerülve az esetleges duplikációkat.

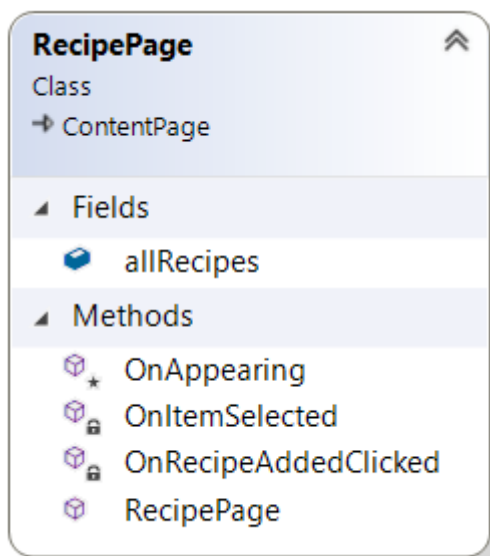
- OnRecipeAddedClicked(object sender, EventArgs e) függvény:

Az Új recept szövegű gomb megnyomásakor történő esemény kezelését végzi ez a függvény. Paraméterként megkapja a küldő objektumot (ebben az esetben ez egy gomb) és az esemény paramétereit.

Inicializál egy NewRecipe típusú Page objektumot és átnavigál ugyanerre a Page-re a Xamarin.Forms által biztosított Navigation objektum használatával.

- OnItemSelected(object sender, SelectedItemChangedEventArgs args) függvény:

Egy listaelem kiválasztásakor bekövetkező ItemSelected esemény kezelését végző függvény, amely megkapja paraméterként a küldő objektumot (ebben az esetben egy lista nézet) és az esemény paramétereit. A függvény ellenőrzi, hogy valóban történt-e kiválasztás a listában: amennyiben igen, a kiválasztott elemet felhasználva átnavigál (Navigation.PushAsync(new Recipe(item))) a Recipe (a recept részleteit tartalmazó aloldal) oldalra, paraméterként átadva a kiválasztott recept elemet. Legvégül eltávolítja a kiválasztást a recepteket tartalmazó listából, hogy ne maradjon feleslegesen színezve a kiválasztás, miután egy felhasználó bezárja a Recipe oldalt.



28. ábra: RecipePage.xaml.cs

3.4.5. NewRecipe

A NewRecipe oldalon van lehetőség hozzáadni új receptet a receptgyűjteményhez.

NewRecipe.xaml	Az űrlapot és annak kinézetét leíró xaml fájl. Label-ek és Entry-k segítségével állítjuk be az oldal kinézetét. Az Entry-k a beviteli mezőket jelenítik meg, míg a Labelek keretbe foglalják és egységekben jelenítik meg a szövegeket és beviteli mezőket. Továbbá tartalmazza a Toolbar-ban a mentés és a vissza gombot.
NewRecipe.xaml.cs	Ennek az osztálynak két konstruktora van. Az első nem vár változót és az új receptek létrehozásáért felel, míg a második egy MRecipe típusú változót vár és a szerkesztendő receptek betöltéséért felel. Valamint ez az osztály kezeli a recept mentését és a visszalépést az oldalról.

A NewRecipe.xaml.cs változói:

- beforeModRecipe:

Ebben a változóban tárolódik el a recept arra az esetre, ha mégsem szeretné a felhasználó módosítani a receptet. Innen tudja visszaállítani a szerkesztés előtti állapotot a program, ha a felhasználó a vissza gombra kattint.

A NewRecipe.xaml.cs függvényei:

- NewRecipe() konstruktor:

Az új recept létrehozásáért felelős.

- NewRecipe(MRecipe toModRecipe) konstruktor:

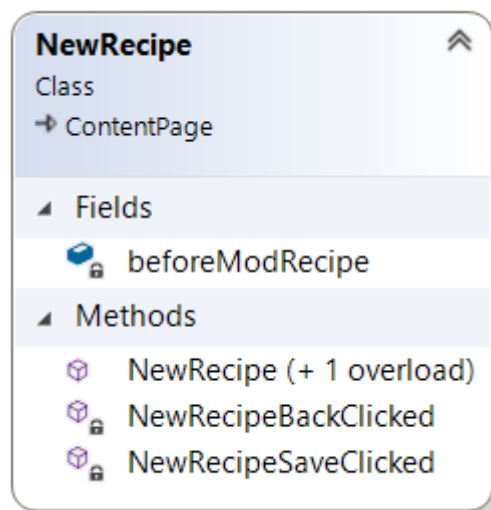
MRecipe típusú változót vár és a szerkesztendő recept betöltéséért felel.

- NewRecipeSaveClicked(object sender, EventArgs e) függvény:

Amikor a felhasználó a mentés gombra kattint, ez a függvény ellenőrzi, hogy a recept neve, hozzávalói és az elkészítése ki lett-e töltve. Amennyiben legalább a kötelező mezőket kitöltötte a felhasználó, a recept sikeresen mentésre kerül. Ellenkező esetben figyelmeztető hibaüzenetet ír ki a program, hogy mik a kötelezően kitöltendő mezők.

- NewRecipeBackClicked(object sender, EventArgs e) függvény:

Amennyiben a felhasználó a vissza gombra kattint ez a függvény fut le. Abban az esetben, ha szerkesztettük a receptet, a függvény visszaállítja a szerkesztés előtti állapotát a receptnek, amit a beforeModRecipe változóban tárol el a szerkesztés idejére, majd visszalép az előző oldalra, pontosabban a szerkesztő oldalt tünteti el. Abban az esetben, ha új recept létrehozása folyamatban van, akkor pedig visszalép az előző oldalra, bezárva a szerkesztőt.



29. ábra: NewRecipe.xaml.cs

3.4.6. Recipe

A Recipe oldal jeleníti meg egy recept részletes leírását, ide töltődik be minden adat, amelyet a NewRecipe oldalon írt be a felhasználó.

Recipe.xaml	A recept adatlapjának kinézetét leíró xaml fájl. A StackLayout-on belül Label-ek segítségével jelenítem meg és formázom a recept adatait. Ez a ScrollView tagek között helyezkedik el. A ScrollView tagekre azért van szükség, hogy a recept oldala görgethető legyen, amennyiben a recept túl hosszú lenne, és nem férne ki a képernyőre. Továbbá tartalmazza a Toolbar-ban a töröl és a szerkesztés gombot.
Recipe.xaml.cs	Egy konstruktora van ennek az osztálynak, ami vár egy MRecipe típusú változót, vagyis egy receptet kell neki átadni, mivel ennek az osztálynak minden esetben egy receptet kell megjelenítenie. Amennyiben van link a recepthez megadva, a program megnyitja a böngészőben. Továbbá lehetőség van a recept törlésére és szerkesztésére.

A Recipe.xaml.cs változói:

- item:

A konstruktorban kapott MRecipe típusú változót reprezentáló osztályváltozó. El kell tudni érni az osztály többi függvényében is, mivel bizonyos változóinak az értékét használják.

A Recipe.xaml.cs függvényei:

- Recipe(MRecipe item) konstruktor:

Egy MRecipe típusú változót, vagyis egy receptet kell neki átadni, mivel ennek az osztálynak minden esetben egy receptet kell megjelenítenie.

- OnLinkClickHandler függvény:

A TapGestureRecognizer beépített eseménykezelő segítségével teszi kattinthatóvá a linket. A függvény kezeli, hogy a linket minden esetben http:// kezdettel tölti be a böngésző, ezzel megelőzve az esetleges betöltési hibákat.

- EditingRecipeClicked() függvény:

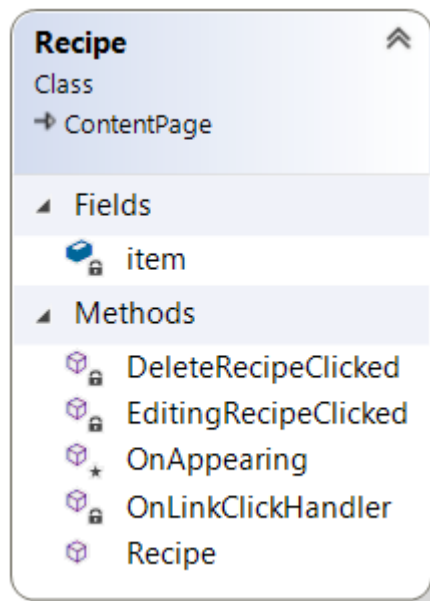
Ez a függvény az aktuális receptet átadja a NewRecipe osztály MRecipe típusú változót váró konstruktorának és a recept a szerkesztőben töltődik be.

- OnAppearing() függvény:

Az oldal megjelenítéséért felel. Minden esetben lefut, amikor a képernyő megjelenik és frissíti a recept adatlapját.

- DeleteRecipeClicked() függvény:

A recept törléséért felel. Ellenőrzi, hogy létezik-e ez a file még és ebben az esetben a Delete utasítással törli, majd a PopAsync visszavigál arra az oldalra, ahonnan korábban megnyitottuk a receptet.



30. ábra: Recipe.xaml.cs

3.4.7. SearchPage

A SearchPage oldalon a felhasználó a három keresőmező segítségével összetett és szimpla keresést is végezhet.

SearchPage.xaml	A kereső képernyő kinézetét leíró xaml fájl. Tartalmaz három SearchBar komponenst, amelyek Placeholderekkel jelzik, hogy mire lehet rákeresni. Itt van a kinézetük is formázva. Alattuk Labelben jelenik meg, ha a keresésnek nincs találat, majd alatta
-----------------	--

	van egy lista nézet, amelybe az esetleges keresési eredmények kerülnek.
SearchPage.xaml.cs	A komponens inicializálása mellett érzékeli a karakterleütéseket a keresési mezőkben, illetve a találati listában való elemre kattintás eseményeket.

A SearchPage.xaml.cs változói:

- `filteredRecipes`:
Egy `MRecipe` objektumokat tároló lista. Ebbe kerülnek a keresési eredmények.

A SearchPage.xaml.cs függvényei:

- `SearchPage()` konstruktor:
A `SearchPage` inicializálásáért felelős konstruktor. A paraméteres változatát Unit tesztelésnél használom. Amikor kap bármilyen értéket a paraméter, akkor az `InitializeComponent()` függvény nem fog lefutni – ez azért hasznos, mert Unit tesztelés közben nem fut az alkalmazás, így nincs mit inicializálnia a rendszernek.
- `OnTextChanged(object sender, EventArgs e)` függvény:
Ez a függvény minden karakterleütésnél újra lefut és ellenőrzi a találatokat. Ez a függvény meghívja a `SearchInList()` és a `HandleSearchResultOnScreen()` függvényeket.
- `SearchInList()` függvény:
Kizárólag opcionális paraméterek találhatók a paraméterlistában, hogy Unit tesztelésnél kényelmesen lehessen használni. A képernyőn való keresés esetén paraméter nélkül hívódik.
Itt egy hosszabb `if – else if – else` fut le és vizsgálja meg a logikai kapcsolatot a megadott keresési feltételek és az `allRecipes` lista elemei között hét különböző lehetőségre kitérve. A legszigorúbb és legtöbb feltételt tartalmazó esetektől haladunk a kevesebb feltételt tartalmazó esetek felé. Először mindhárom mező kitöltöttségét vizsgáljuk meg – ebben az esetben a recept neve, a címkék és a hozzávalók is ki vannak töltve. A második, harmadik és a negyedik eset azt vizsgálja, amikor csak két mező van kitöltve különböző variációkban – a recept neve és a címkék, majd a recept neve és a hozzávalók, majd a

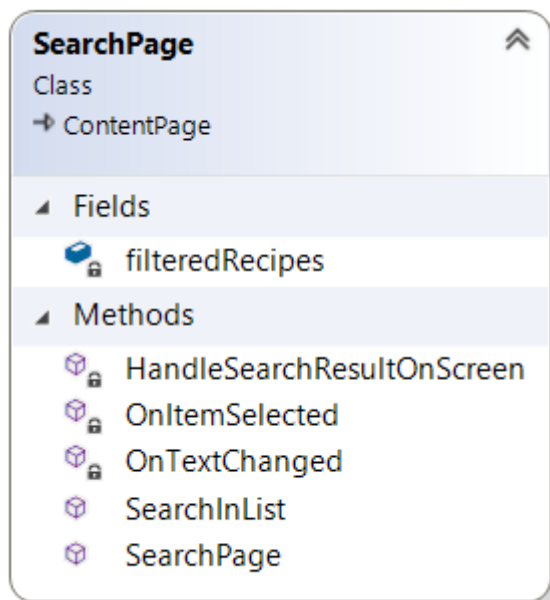
címkék és a hozzávalók. Utolsóként a mezők kitöltöttségét egyesével is megvizsgálja az ötödik, hatodik és a hetedik esetben.

- HandleSearchResultOnScreen() függvény:

A keresési eredmény alapján a keresési lista találatait frissíti.

- OnItemSelected(object sender, SelectedItemChangedEventArgs args)

A keresési lista egy elemére kattintva fut le a függvény. Ellenőrzi, hogy valóban történt-e kiválasztás a listában: amennyiben igen, a kiválasztott elemet felhasználva átnavigál (Navigation.PushAsync(new Recipe(item))) a Recipe (a recept részleteit tartalmazó aloldal) oldalra, paraméterként átadva a kiválasztott recept elemet. Legvégül eltávolítja a kiválasztást a recepteket tartalmazó listából, hogy ne maradjon feleslegesen színezve a kiválasztott elem, miután egy felhasználó visszalép a Recipe oldalról.



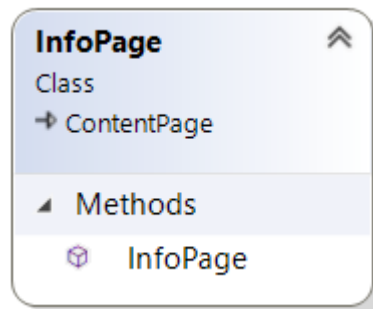
31. ábra: SearchPage.xaml.cs

3.4.8. InfoPage

Az InfoPage oldalon az applikáció jelenlegi funkcióiról és a várható fejlesztésekről olvashat a felhasználó. Az oldal alján található a kontakt, így az esetleges hibákat tudja jelenteni.

InfoPage.xaml	A xaml a WebView-t tartalmazza. Ez egy olyan nézet, ami webes és HTML tartalom megjelenítéséért felel.
---------------	--

InfoPage.xaml.cs	<p>Ebben az osztályban nincs függvény, csak egy konstruktor. Itt csak az információk kiírása történik meg. A Toolbar tartalmazza a menü gombot.</p> <p>A HtmlWebViewSource-nak köszönhetően a megjelenítendő szöveget manuálisan tudtam beírni html kódok/tagek közé és a formázást is itt tudtam elvégezni.</p>
------------------	--



32. ábra: InfoPage.xaml.cs

3.4.9. Az MRecipe modell

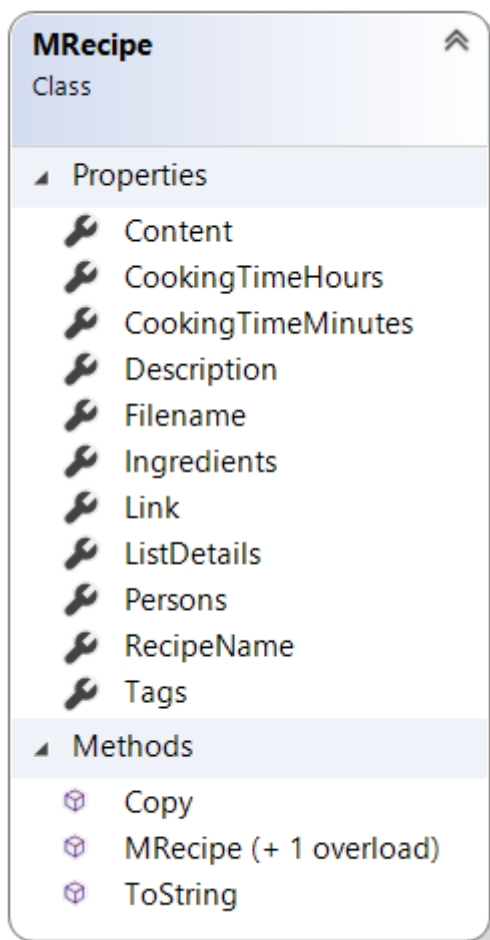
Az MRecipe osztály egy példánya egy receptet reprezentál a tulajdonságaival (osztályváltozók) és a tárolásával együtt. Mind a RecipePage, NewRecipe, Recipe, SearchPage komponensek mögött megtalálható az osztálynak egy, vagy több példánya is.

Az MRecipe.cs függvényei:

- Két konstruktort tartalmaz. A paraméter nélküli konstruktort új létrehozására, míg a paraméteres konstruktort másolásra lehet használni. A másolás a szerkesztő nézetnél nagyon hasznos, mivel ezáltal a szerkesztés előtt egyszerűen lehet készíteni egy biztonsági mentést a szerkesztendő MRecipe objektumról, majd ha történt is változtatás a szerkesztés képernyőn (és a BindingContext miatt meg is változott az eredeti recept tartalma), könnyedén vissza lehet állítani a lemásolt objektumból az eredetit.

- Copy(MRecipe other) függvény:

Ez a függvény felel azért, hogy a paraméterként kapott MRecipe típusú változónak lemásolja az értékeit.



33. ábra: `MRecipe.cs`

4. Tesztelés

4.1. Black Box tesztelés

A tesztelés során a Black Box testing (fekete doboz tesztelés), vagyis a funkcionális tesztelés elméletét alkalmaztam. Ez a tesztelési módszer az applikáció funkcionalitását vizsgálja anélkül, hogy megvizsgálná a belső kódszerkezetet, annak részleteit - tehát a függvényeket figyelmen kívül hagyja. Ez a típusú tesztelés teljes mértékben a szoftverkövetelményeken és a specifikáción alapszik. A Black Box tesztelés során a szoftver bemeneteire és kimeneteire összpontosítunk, anélkül, hogy foglalkoznánk a szoftver belső ismereteivel. Ilyenkor helyes és hibás bemenetekkel egyaránt tesztelni kell a programot, hogy megbizonyosodjunk arról, hogy valóban minden esetben helyesen működik a program. Ezzel ellenőrizzük, hogy a helyes bemenetet megfelelően fel tudja-e dolgozni, illetve az érvénytelen bemenetet képes-e észlelni. [10]

4.1.1. A kezdőképernyő tesztelése

A kezdőképernyőn a már mentett recepteket látjuk abc sorrendben. Fent a Toolbar-ban pedig a menü és az új recept hozzáadása gomb van. Három továbblépési lehetőségünk van, így most mindhárom esetet tesztelni fogjuk.

- **A recept adatlapjának betöltésének tesztelése:**

Bármelyik receptre kattintva be kell töltnie a recept adatlapjának. A recept adatlapjának működését egy következő pontban részletesebben tesztelem.

A tesztelés eredménye:

Teszteléskor az történt, amit vártunk, betöltődött a kiválasztott recept adatlapja.

- **Új recept hozzáadásának tesztelése:**

Az új recept gombra kattintva egy kitöltendő űrlapnak kell megjelennie nyolc beviteli mezővel, a Toolbar-ban pedig a mentés és a vissza gombok jelennek meg.

A tesztelés eredménye:

A tesztelés folyamán az új recept gombra kattintva az történt, amit vártunk - betöltődött az űrlap. Az új recept hozzáadásának részletes tesztelését egy következő pontban részletezem.

- **Menü gomb tesztelése:**

A menüre kattintva balról be kell úsznia a menünek, amiben a következő menüpontokat kell látnunk és ezekre kattintva új content oldalnak kell betöltődniük: Receptek, Keresés, Információ. Amennyiben mégsem szeretnénk a menüből új oldalt választani, akkor a menüt balra húzva vagy mellé kattintva tudjuk eltüntetni.

A tesztelés eredménye:

A tesztelés folyamán az történt, amit vártunk. A menü kattintásra megjelent, megnyitotta a menüpontokhoz tartozó új oldalakat és balra húzva vagy mellé kattintásra eltűnt.

4.1.2. Az új recept hozzáadásának tesztelése

Ezen a képernyőn nyolc beviteli mezőt, valamint a Toolbar-ban a mentés és a vissza gombokat láthatjuk.

A következő beviteli mezőknél bármilyen input helyes lehet: Recept neve, Címkék, Hozzávalók, Elkészítés, Link az eredeti recepthez. Ennek az az oka, hogy mindenki saját igénye szerint szabadon nevezheti el a receptjét, adhat hozzá címkét, valamint a hozzávalók, az elkészítés folyamata leírása során, valamint a link megadásakor is szükség lehet különböző karakterekre. Az elkészítési idő (óra, perc) és a hány főre esetében viszont csak szám elfogadott inputként. A szerkesztő nem is enged mást beírni, mivel csak a számokat ajánlja fel beviteli lehetőségként.

16:01

← MENTÉS VISSZA

Recept

Pizzatészta

olasz

Elkészítési idő - óra Elkészítési idő - perc

Hány főre

Hozzávalók

Elkészítés

Link az eredeti recepthez

1 2 3 -

4 5 6 _

7 8 9 ✕

, 0 . ✓

34. ábra: Input megszorítás - Numerikus billentyűzet

A Hozzávalók és az Elkészítés beviteli mezők mérete a beírt függvény hosszához igazodva dinamikusan követi a szöveg méretét.

15:57

Recept

Csokis cupcake

édesseg, muffin

Elkészítési idő - óra 40

4

15 dkg puha vaj, 12 dkg cukor, 3 db tojas, 15 dkg liszt, 1.5 tk sütőpor, 3 ek kakaópor

1. A vajat habosra keverjük a cukorral, és beledolgozzuk a tojásokat. A lisztet elvegyítjük a sütőporral és a kakaóval, majd a vajas keverékbe forgatjuk.
2. A sütőt előmelegítjük 180 fokra. A tésztát egy 12 részes muffintepsi papírral kibélelt mélyedéseibe kanalazzuk, majd 20 kb 15-20 perc alatt megsütjük.
3. Ezután kiszedjük a muffinokat, és megszórjuk a...

1 2 3 4 5 6 7 8 9 0

q w e r t z u i o p

a s d f g h j k l

↑ y x c v b n m ✕

?123 , ☺ . ←

35. ábra: A szöveg méretéhez igazodó beviteli mezők 1

16:02

← MENTÉS VISSZA

Recept

Pizzatészta

olasz

1 50

2

300 ml víz
3 evőkanál olívaolaj
500 g liszt
2.5 dkg élesztő
1 tk cukor
1 tk só

1 2 3 4 5 6 7 8 9 0

q w e r t z u i o p

a s d f g h j k l

↑ y x c v b n m ✕

?123 , ☺ . ←

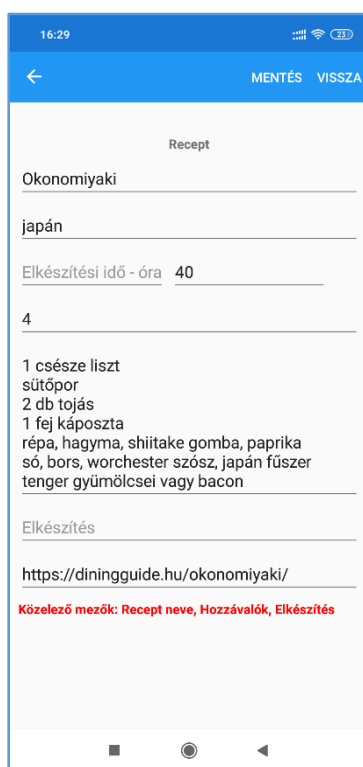
36. ábra: A szöveg méretéhez igazodó beviteli mezők 2

A mentés gombra kattintva a receptünket elmenthetjük. Ezután a program visszavigál minket a kezdő oldalra, ahol már a listában láthatjuk az előzőleg elmentett receptet.

A vissza gombra a recept nem kerül mentésre és a program visszavisz minket a kezdőoldalra a receptek listájához, ahol most a listában nem látunk változást.

Helytelen input beadása:

A receptet megpróbáljuk csak névvel elmenteni. Ezt nem kéne engednie, mivel három kötelező jelleggel kitöltendő mező van - Recept neve, Hozzávalók, Elkészítés. Ha ezeket üresen hagyva vagy csak szóközt írva próbáljuk meg elmenteni, akkor egy piros figyelmeztető üzenetnek kell a beviteli mezők alatt megjelennie: „Kötelező mezők: Recept neve, Hozzávalók, Elkészítés”

A screenshot of a mobile application interface for managing recipes. The screen is titled "Recept" and has a blue header bar with a back arrow, "MENTÉS" (Save), and "VISSZA" (Back) buttons. The form contains several input fields: "Okonomiyaki" (Recipe name), "japán" (Origin), "Elkészítési idő - óra" (Preparation time in hours) with the value "40", and "4" (Number of servings). Below these is a list of ingredients: "1 csésze liszt", "sütőpor", "2 db tojás", "1 fej káposzta", "répa, hagyma, shiitake gomba, paprika", "só, bors, worchester szósz, japán fűszer", and "tenger gyümölcsei vagy bacon". There is also a field for "Elkészítés" (Preparation instructions) containing the URL "https://diningguide.hu/okonomiyaki/". At the bottom, a red error message reads: "Kötelező mezők: Recept neve, Hozzávalók, Elkészítés". The status bar at the top shows the time "16:29" and signal strength.

37. ábra: Kötelezően kitöltendő mezők és hibaüzenet

A tesztelés eredménye:

A tesztelés során az történt, amit elvártunk. Az elkészítési idő és a személyek megadásánál csak a számok bevitele volt engedélyezett, míg a többi esetben szabadon nevezhettük el a receptet és írhattuk be a szöveges utasításokat a mezőkbe. A receptet a Recept neve, Hozzávalók és az Elkészítés mezők kitöltése nélkül nem lehetett elmenteni.

4.1.3. A recept adatlapjának tesztelése

Ezen az oldalon a recept összes adata megjelenik. A felhasználó által bevitt adatok szürke színben, míg a címkék kiemelve fekete színben jelennek meg. A toolbarban a törlés, a szerkesztés és a vissza (nyíl) gombok láthatóak.

A recept adatlapján a következő funkcióknak kell működniük:

- Az oldal görgetése, amennyiben a recept nem fér el a képernyőn.
- A recept törlése a törlés gombra kattintva.
- A receptet a szerkesztés gombra kattintva tudnunk kell szerkeszteni. Ez a funkció átnavigál egy - az új recept hozzáadásánál már megismert – oldalra, ahol egy űrlapba töltődnek be az adatok.
- Vissza gomb (nyíl), amivel visszaléphetünk a receptek listájához.

A tesztelés eredménye:

Az oldal a fent leírtak alapján jelent meg és a tesztelés során a funkciók a felsorolásban leírtaknak megfelelően működtek.

4.1.4. A menü tesztelése

A menüre kattintva balról beúszik a menü, amiben három menüpont található – Receptek, Keresés, Információk. Ezekre kattintva a menüpontoknak megfelelő oldalaknak kell betölteniük. Amennyiben mégsem szeretnénk a menüből új oldalra navigálni, két lehetőség áll rendelkezésünkre a menü bezárásához. A menüt balra húzva, illetve a mellette lévő elhomályosított részre kattintva is be lehet zárni.

A tesztelés eredménye:

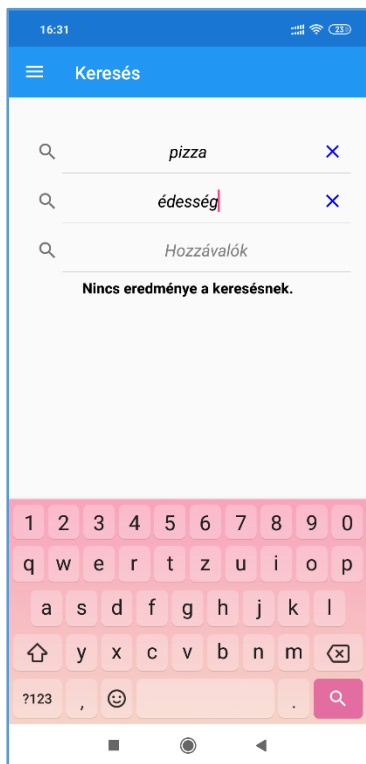
A tesztelés során a funkciók hibátlanul működtek.

4.1.5. A keresés tesztelése

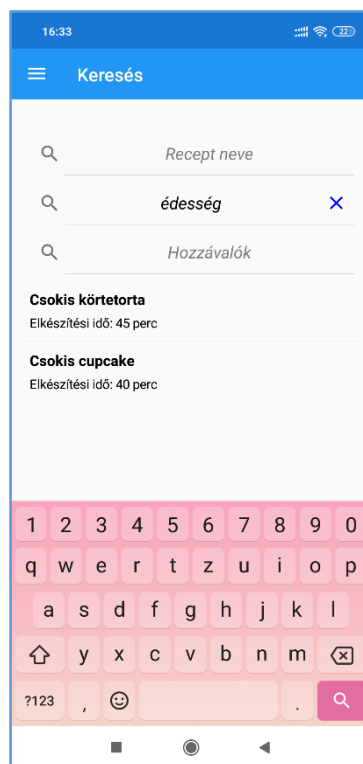
A keresés oldalán három bemeneti mezőt látunk. A toolbarban a menü gomb található, aminek segítségével elnavigálhatunk az oldalról.

Mindegyik kereső mező placeholder segítségével tájékoztatja a felhasználót, hogy mire tud az adott mezőben rákeresni. Minden egyes karakterleütés után a program frissíti a keresési eredményeket, így már az első esetleges elgépelésnél észrevehetjük, hogy nem jól írtunk be valamit vagy tényleg nincs találat a keresésünkre. Lehet a keresési mezőket külön-külön is

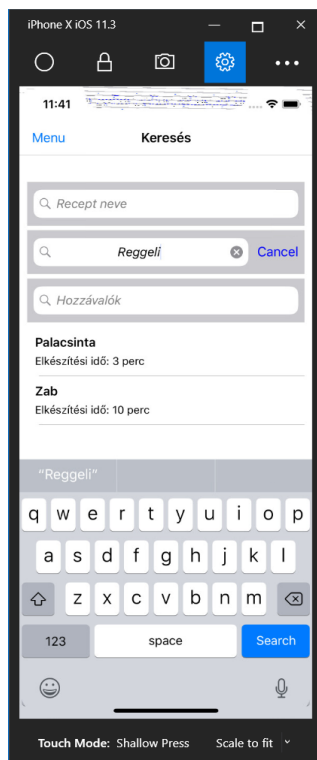
használni, de lehetőségünk van összetett keresésre is. Amennyiben nincs találat, a kereső mezők alatt a „Nincs eredménye a keresésnek” üzenet jelenik meg.



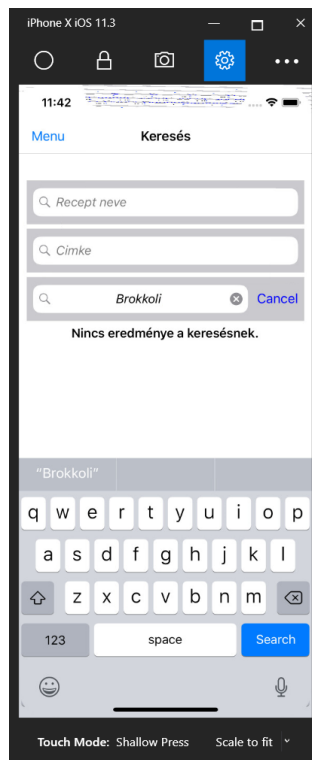
38. ábra: Android - Összetett keresés - nincs találat



39. ábra: Android - Keresés címke szerint - több találat



40. ábra: iOS – Keresés címkére - több találat



41. ábra: iOS – Keresés hozzávalóra - nincs találat

A tesztelés eredménye:

A keresés az elvártaknak megfelelően működött.

4.1.6. Az információs oldal tesztelése

Az információs oldal egy statikus oldal, nincsenek funkciói. A szöveg szerkesztve, tagolva jelenik meg rajta. Amennyiben a szöveg túl hosszú és nem fér ki a képernyőre, az ujjunk mozgatásával lehet fel-le görgetni. A bal felső sarokban a menü gomb található, amivel el lehet navigálni az oldalról.

A tesztelés eredménye:

A tesztelés során az oldal tagolva megjelent, görgethető volt, valamint a menü gomb is működött.

4.2. Unit tesztelés

A Unit teszt (egység teszt) segítségével tesztelhetjük, hogy a kód a várt módon működik-e. Egység tesztnek hívják, mert a megfelelően részekre bontott kódot vizsgálja, amelyeket önálló egységeként tesztel. A Visual Studioban a Unit Test Framework segítségével lehet unit teszteket készíteni, futtatni és ellenőrizni a tesztek eredményeit. Amennyiben változás történik a kódban, érdemes a tesztet újra futtatni, hogy továbbra is helyesen működik-e. A Unit tesztelésnek van a legnagyobb hatása a kód minőségére, amennyiben szerves része a szoftverfejlesztési munkafolyamatnak.

A legcélravezetőbb, ha akkor van lehetőségünk a Unit teszt megírására, amikor egy új függvényt vagy egyéb blokkot hoztunk létre, így igazolva annak helyes működését. [11]

4.2.1. Unit Tesztelés menete

A teszteléshez egy MSTest Test Project típusú alprojektet hoztam létre. A UnitTest alprojektben belül a UnitTest1.cs osztályban végeztem el a legösszetettebben működő egység funkció, a keresés tesztelését.

Az inSearch listába generáltam meghatározott számú recepteket.

Az InitSearchTest függvény inicializálja a teszteléshez szükséges erőforrásokat.

A Cleanup függvény eltünteti az esetleges korábbi tesztelések által lefoglalt erőforrásokat.

A [TestMethod] parancsszavak alatt találhatóak a különböző tesztesetek.

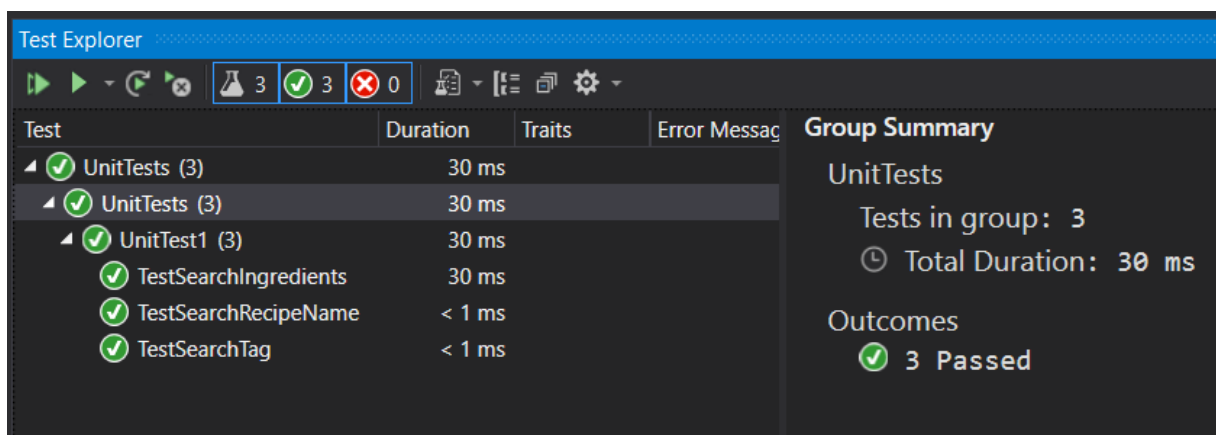
A **TestSearchRecipeName** függvény végzi a keresést a random generált receptek nevein és három különböző esetet vizsgálva és ezek eredményeit három különböző listában tárolja. Az első listában, a searchResult1-ben azokat az eredményeket tárolja, amik egy darab recept nevére való keresés után keletkeztek. A második esetben az összes "recipeName"-et tartalmazó receptre keres és az eredményt a searchResult2 listában tárolja. A harmadik esetben az összes "recipeName"-et tartalmazó receptre keres, vagyis a receptnév egy részletére, amely bárhol lehet a recept nevében belül és az eredményeket a searchResult3 listában tárolja. Mindegyik eredményt az Assert.AreEqual(tervezett eredmény mennyisége, aktuális találat mennyisége) függvénnyel értékeltem ki.

A **TestSearchTag** függvény a random generált receptek címkéin végez el három különböző tesztelést és az eredményeket szintén három különböző listában tárolja. Az első esetben egy darab tag-re keres és az eredményt a searchResult1 listában tárolja el. A második esetben az

összes "tag" kifejezést tartalmazó tag-re keres és az eredményt a searchResult2 listában tárolja el. A harmadik esetben az "a" betűre keres a tag-ekben, vagyis a címkék egy részletére, ami bárhol előfordulhat a címkékben. Ennek az eredményét a searchResult3 címkében tárolja el. Mindegyik eredményt az Assert.AreEqual(tervezett eredmény mennyisége, aktuális találat mennyisége) függvénnyel értékeltem ki.

A **TestSearchIngredients** függvény az összetevők szerinti keresést vizsgálja. Az első esetben egy konkrét hozzávalóra keres és az eredményt a searchResult1 listában tárolja el. A második esetben az összes hozzávalóra rákeres és az eredményt a searchResult2 listában tárolja el. A harmadik esetben az összes "gred"-et tartalmazó hozzávalóra keres, vagyis a hozzávaló egy részletére, amely bárhol lehet a hozzávalón belül és az eredményt a searchResult3 listában tárolja. Mindegyik eredményt az Assert.AreEqual(tervezett eredmény mennyisége, aktuális találat mennyisége) függvénnyel értékeltem ki.

Az MRecipe típusú és egy intet váró **GenerateRecipes** függvény a random receptek generálásáért felel, amelyeket a generatedRecipes listában tárol el, majd ad vissza.



42. ábra: Unit teszt futási eredmény

5. Összegzés

Elmondhatom, hogy sikerült egy olyan alkalmazást implementálnom, amit a jövőben is szívesen használok saját és egyéb forrásokból származó receptek összegyűjtésére cross platform alapokon. Az alkalmazás fejlesztése nagy kihívást jelentett, de remek gyakorlási- és továbbfejlődési lehetőség is volt számomra. Örülök, hogy ezt a témát választottam, mert így nem csak a C# ismereteimet mélyítettem el, hanem megismerkedtem a Xamarinnal és a cross platform alapú alkalmazásfejlesztés lehetőségeivel is. A fejlesztés során nagy segítséget jelentettek a különböző fórumok és felhasználói dokumentációk. Ezeken a platformokon szinte minden problémára lehet megoldást találni. Sok esetben a platformra látogatók több megoldás közül választhatják ki a számukra legmegfelelőbbet.

Az elkészített alkalmazás fennakadás nélkül működik, de véglegesnek és tökéletesnek nem mondanám. Már most vannak terveim a további fejlesztéseket illetően –amiket az információs felületen fel is tüntettem-, valamint a folyamatosan fejlődő informatika világában állandó fejlesztésekre és újításokra van szükség, hogy lépést lehessen tartani az igényekkel.

Jelenleg ezeket a fejlesztéseket tervezem az applikáción elvégezni:

- Az adatok adatbázisban való tárolása, hogy egy esetleges telefonváltás alkalmával a tárolt adatok ne vesszenek el. A felhasználók azonosítása email címmel fog történni.
- A megadott link segítségével az oldalon lévő recept beolvasása – néhány weboldalról - a megfelelő beviteli mezőkbe, ezzel megkímélve a felhasználót a gépeléstől, illetve a másolás-beillesztés folyamatától.

6. Irodalomjegyzék

[1] <https://theappsolutions.com/blog/development/ios-vs-android/>

Megtekintés dátuma: 2020.02.25. 10:33

[2] <https://dotnet.microsoft.com/learn/dotnet/what-is-dotnet>

Megtekintés dátuma: 2020.02.25. 10:59

[3] <https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>

Megtekintés dátuma: 2020.02.25. 11:24

[4] <https://docs.microsoft.com/en-us/visualstudio/get-started/visual-studio-ide?view=vs-2019>

Megtekintés dátuma: 2020.02.25. 12:10

[5] Reiter István - C# programozás lépésről lépésre (2018) Kiadó: Jedlik Oktatási Stúdió

[6] <https://dotnet.microsoft.com/apps/xamarin>

Megtekintés dátuma: 2020.02.26. 11:56

[7] [https://hu.wikipedia.org/wiki/Android_\(oper%C3%A1ci%C3%B3s_rendszer\)](https://hu.wikipedia.org/wiki/Android_(oper%C3%A1ci%C3%B3s_rendszer))

Megtekintés dátuma: 2020.02.26. 12:23

[8] <https://hu.wikipedia.org/wiki/IOS>

Megtekintés dátuma: 2020.02.26. 13:45

[9] https://medium.com/@the_manifest/android-vs-ios-which-platform-to-build-your-app-for-first-22ea8996abe1 (<https://theappsolutions.com/blog/development/ios-vs-android/>)

Megtekintés dátuma: 2020.02.26. 14:50

[10] <https://www.guru99.com/black-box-testing.html>

Megtekintés dátuma: 2020.03.02. 14:36

[11] <https://docs.microsoft.com/en-us/visualstudio/test/unit-test-basics?view=vs-2019>

Megtekintés: 2020.03.05. 20:40

7. Ábrák és táblázatok jegyzéke

Ábrák

1. ábra: Android kezdő oldal – recept nélkül	2. ábra: iOS kezdőoldal – recept nélkül	11
3. ábra: Android kezdőoldal – receptekkel	4. ábra: iOS kezdőoldal - recepttel	11
5. ábra: Android – új recept hozzáadása	6. ábra: iOS – új recept hozzáadása	12
7. ábra: Android – kötelező mezők	8. ábra: iOS – kötelező mezők	13
9. ábra: Android – recept adatlapja 1	10. ábra: Android – recept adatlapja 2	14
11. ábra: iOS – recept adatlapja		14
12. ábra: Android – recept szerkesztése	13. ábra: iOS – recept szerkesztése	15
14. ábra: Android menü	15. ábra: iOS menü	16
16. ábra: Android – Keresés	17. ábra: iOS – Keresés	17
18. ábra: Android – nincs találat az összetett keresésnek		17
19. ábra: iOS – nincs találat a keresésnek	20. ábra: iOS - összetett keresés	18
21. ábra: Android – Információs oldal	22. ábra: iOS – Információs oldal	19
23. ábra: Android – Információs oldal/Kontakt		19
24. ábra: Felépítés ábra		22
25. ábra: App.xaml.cs		23
26. ábra: StarterPage.xaml.cs		25
27. ábra: MenuPage.xaml.cs		26
28. ábra: RecipePage.xaml.cs		28
29. ábra: NewRecipe.xaml.cs		29
30. ábra: Recipe.xaml.cs		31
31. ábra: SearchPage.xaml.cs		33
32. ábra: InfoPage.xaml.cs		34
33. ábra: MRecipe.cs		35
34. ábra: Input megszorítás - Numerikus billentyűzet		38
35. ábra: A szöveg méretéhez igazodó beviteli mezők 1	36. ábra: A szöveg méretéhez igazodó beviteli mezők 2	38
37. ábra: Kötelezően kitöltendő mezők és hibaüzenet		39
38. ábra: Android - Összetett keresés - nincs találat	39. ábra: Android - Keresés címke szerint - több találat	41
40. ábra: iOS – Keresés címkére - több találat	41. ábra: iOS – Keresés hozzávalóra - nincs találat..	41
42. ábra: Unit teszt futási eredmény		44

Táblázatok

1. táblázat: Fejlesztői környezet hardveres része	8
2. táblázat: Fejlesztői környezet szoftveres része	8
3. táblázat: A fejlesztéshez használt Androidos okostelefon adatai	9
4. táblázat: A fejlesztéshez használt iOS emulátor adatai	9
5. táblázat: A program rendszerkövetelménye	10

8. Mellékletek

- CD
- Konzultációs lap

9. CD lemez tartalma

A mellékelt CD lemez a következőket tartalmazza:

- A forráskódot RecipeApp.zip néven, becsomagolva
- A szakdolgozatot pdf formátumban MatyusAnnamariaZarodolgozat.pdf néven
- Android-ot futtató eszközhöz való telepítő fájlt RecipeApp.apk néven

10. Nyilatkozat

Nyilatkozat a záródolgozat készítésére vonatkozó szabályok betartásáról

Alulírott **Mátyus Annamária** jelen nyilatkozat aláírásával kijelentem, hogy a

Mobilalkalmazás Cross Platform alapokon

című záródolgozatot magam készítettem.

Mindezek alapján jelen záródolgozat önálló munkám, annak elkészítésekor betartottam a szerzői jogairól szóló 1999. LXXXVI törvény szabályait, valamint az intézmény által előírt, a dolgozat készítésére vonatkozó szabályait, különösen a hivatkozások és idézetek tekintetében.

Kijelentem, hogy az elektronikus és a papír alapú dokumentum mindenben megegyezik.

Fentiekén kívül kijelentem, hogy az önállóságra vonatkozóan, a dolgozat készítése közben konzulensemét és a feladatot kiadó oktatót nem tévesztettem meg.

Budapest,

.....

Mátyus Annamária