

SAPI DINER

- Dokumentáció -

Készítette:

Balázsai Anita

Bíró Alida

Tartalom

1.Bevezető	3
1.1 A projekt témája	3
1.2 A projekt célja	3
2. Követelmények	4
2.1 Felhasználói követelmények	4
2.2 Rendszerkövetelmények	5
2.3 Alkalmazott rendszerkomponensek	5
3.Tervezés.....	6
3.1 Osztálydiagram.....	6
3.2 Use case diagram	6
4. Architektúra	7
5. Az alkalmazás működési elve	8
5.1 Bejelentkezési oldal	8
5.2 Regisztrációs oldal	9
5.3 Étlap szerkesztési oldal	10
5.4 Étlap megtekintési oldal.....	12
6. Továbbfejlesztési lehetőségek	13

1.Bevezető

1.1 A projekt témája

Az általunk választott projekt, vagyis a Sapi Diner telefonos applikáció témáját a Sapientia Egyetem hallgatói, illetve tanárai által naponta tapasztalt probléma ihlette, ami nem más mint az egyetem ebédlőjében tapasztalható túlszűfolttság az ebédszünetekben. Úgy a diákok mint a tanárok gyakran szembesülnek azzal a kellemetlen szituációval, hogy a szünetek nagy részét sorban állva kénytelenek tölteni az ebédlőben, arra várva, hogy leadhassák a rendelésüket.

Meglátásunk szerint a legnagyobb problémát az ebédelésre szánt, önmagában is kevés idő rossz beosztása, illetve felhasználása okozza, amely optimalizálásában akár egy telefonos applikáció is sokat segíthetne.

1.2 A projekt célja

Ahhoz, hogy a fent említett problémát megoldjuk és hogy hozzájáruljunk egy kellemesebben eltölthető ebédszünet élményéhez, megterveztük a Sapi Dinernek nevezett applikációt, amely legfőbb célkitűzései az előbb említettek.

Az alapötlet magában foglalta, hogy az applikáció olyan funkcionálisokkal rendelkezzen, ami lehetővé teszi a felhasználók számára, hogy rendeléseiket az appon keresztül adják le, és értesítést kapjanak annak készenlétéről, így teljesen kiküszöbölve a fölösleges várakozási időt. Az értesítés elküldése SMS formájában történik és ezen módszert alkalmazva indokolatlanná válik, hogy bárki is az SMS beérkezése előtt keresse fel az ebédlőt.

További célja az applikációnak, hogy az ebédlőben dolgozók munkáját is segítse, illetve megkönnyítse, mivel használatával a lehető legegyszerűbben tudják publikálni az aktuálisan fogyasztható ételek listáját a többi felhasználó számára.

2. Követelmények

2.1 Felhasználói követelmények

A Sapi Diner alkalmazásunk funkcionalitásainak megtervezésekor szem előtt kellett tartanunk néhány alapkövetelményt, amely az applikáció hatékonyságában játszik jelentős szerepet. Ezen követelmények a következők:

1. Regisztráció

- Az adatbázisba való bekerülés feltétele azok számára akik az applikációt használni szeretnék
- Minden beregisztrált felhasználó 0-s típussal jön létre az adatbázisban, azaz nincsenek adminisztrátori jogai
- Az adminisztrátorok számára nincs szükség regisztrációra, az ő regisztrálásuk kódból történik, az adatbázisban megjelenő típusuk 1-es

2. Bejelentkezés

- Email cím és jelszó alapján történik
- Csak beregisztrált felhasználóknak lehetséges a bejelentkezés
- A felhasználók autentifikációjában játszik szerepet, ellenőrzi, hogy a bevitt adatok léteznek-e az adatbázisban
- Az itt bevitt adatok alapján különíti el az applikáció azokat akik adminisztrációs jogosultságokkal rendelkeznek a többi, átlagos felhasználótól

3. Rendelés leadása

- Amennyiben a bejelentkezett személy átlagos felhasználó, lehetősége van megtekinteni az éppen aktuális étlapot és rendelést leadni

4. Rendelés megtekintése

- Átlagos felhasználók számára saját rendelésük megtekintése és jóváhagyása

- Adminisztrátorok számára az összes beérkező rendelés megtekintésének lehetősége

5. Étlap szerkesztése

- Adminisztrátorok számára készült funkcionalitás
- Lehetőség van új ételek hozzáadására képfeltöltéssel együtt
- A már nem aktuális ételek törlésének lehetősége is adott az étlapról

6. Értesítés küldése

- Szintén adminisztrátori jogosultság szükséges az igénybevételéhez
- SMS formájában értesíthető a kliens arról, ha átvételre kész a rendelése

2.2 Rendszerkövetelmények

1. Android operációs rendszerrel rendelkező eszköz
2. Minimum SDK: Android 4.1 (API 16)
3. 12 MB tárhely
4. Internet hozzáférés

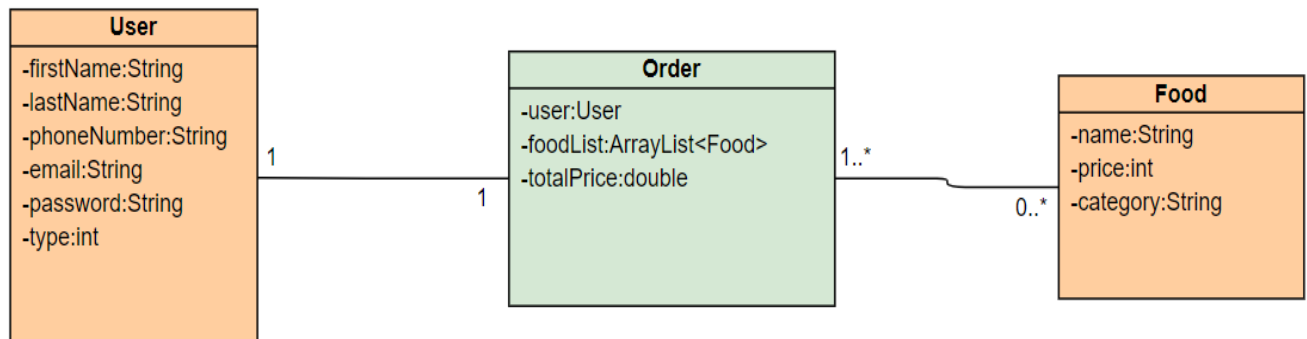
2.3 Alkalmazott rendszerkomponensek

1. Adatbázis: Firebase
2. Android:
 - Activity
 - Fragment
 - Adapter
 - RecyclerView

3.Tervezés

3.1 Osztálydiagram

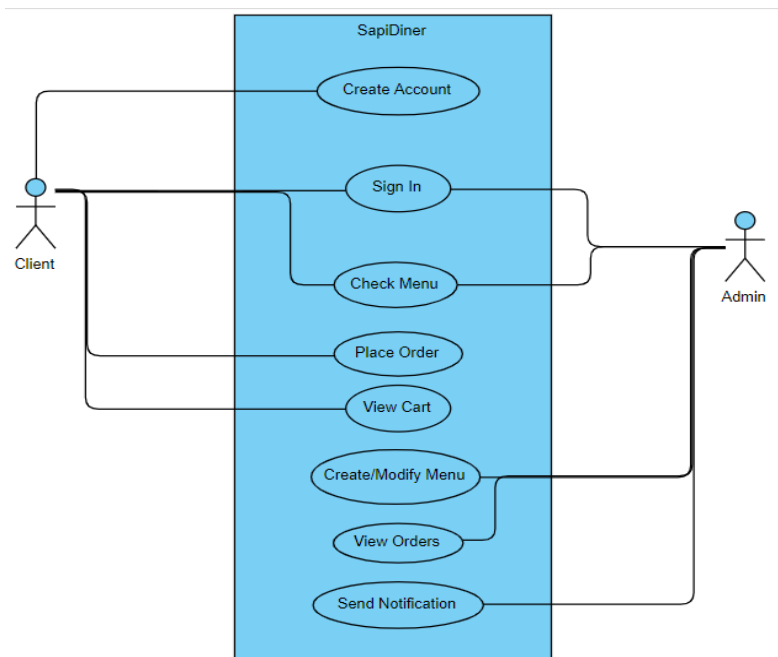
A szükséges adatok adatbázisban való eltárolására 3 osztályt hoztunk létre: User, Food és Order osztályokat, az ezek közötti kapcsolatot az alábbi osztálydiagram szemlélteti:



A fenti osztályok az alábbi módokon járulnak hozzá az applikáció megfelelő működéséhez

1. User osztály
 - A regisztráció során bevitt adatok eltárolása
 - A felhasználó típusának lekérése is ebből az osztályból történik
2. Food osztály
 - Az adminisztrátor által bevitt ételek nevét, kategóriáját és árát tárolja
3. Order osztály
 - Tárolja a megrendelő kliens összes adatát
 - A rendelt étel(ek) összes adatát
 - A rendelés összárát

3.2 Use case diagram

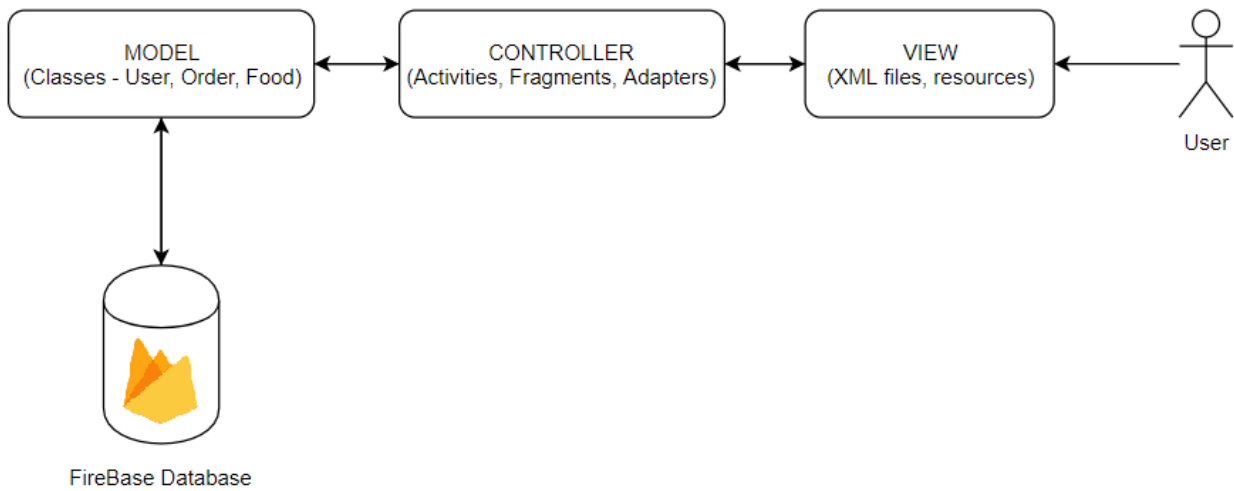


4. Architektúra

Alkalmazásunk az MVC (Model-View-Controller) architektúra alapján készült el, az adatokat, a felhasználói felületet valamint a vezérlő szerkezeteket elkülönítettük egymástól.

A felhasználónak csak View részhez tartozó layoutokkal (XML fájlok) van közvetlen kapcsolata, ezek határozzák meg a felhasználói felületet valamint biztosítják, hogy a felhasználó és az alkalmazás között létrejöhessen a kommunikáció. A Controllerhez tartozó Activityk, Fragmentek és Adapterek teszik lehetővé, hogy a Model és View rész között adatcsere jöhhessen létre: feldolgozzák a bejövő adatot, módosításokat végeznek a Modelen, valamint kimenő adatokat biztosítanak a View számára. A Model részhez tartoznak az osztályok, amelyek az alkalmazás által használt adatokat írják le.

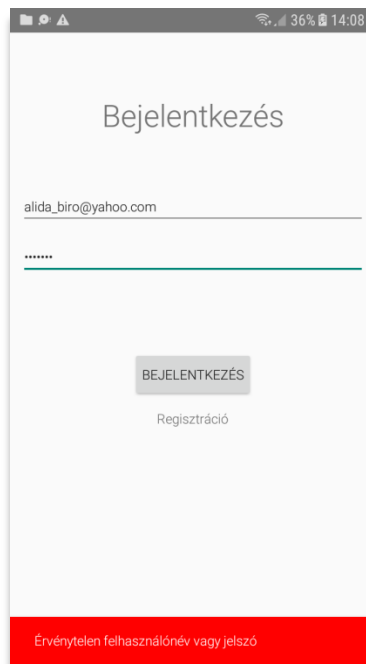
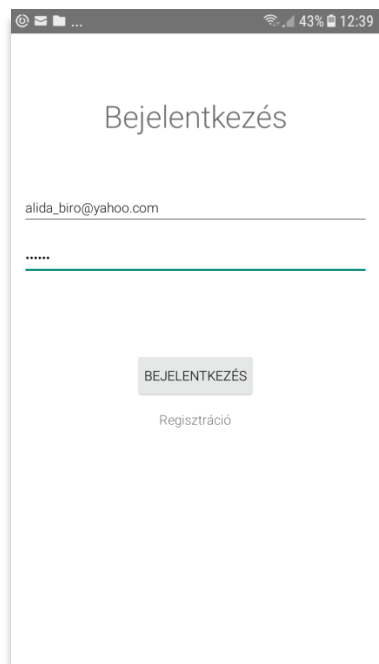
A Sapi Diner architektúráját az alábbi diagram szemlélteti:



5. Az alkalmazás működési elve

5.1 Bejelentkezési oldal

A felhasználó a már adatbázisban meglévő email címével és jelszavával bejelentkezhet az applikációba. Amennyiben a felhasználó hibás jelszót vagy email címet ad meg, egy figyelmeztető ablak jelenik meg az applikáció alján.



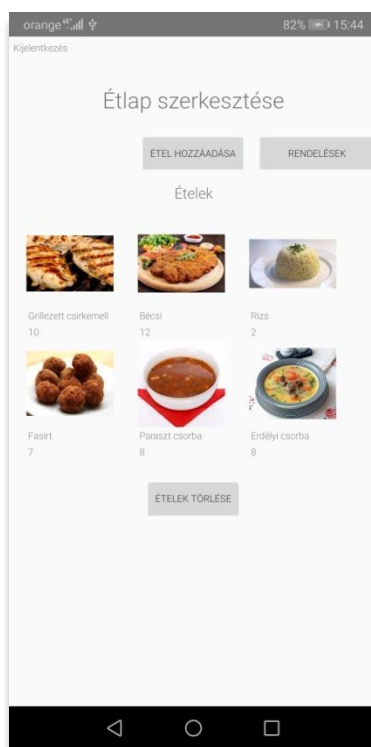
5.2 Regisztrációs oldal

Amennyiben a felhasználó még nem rendelkezik saját fiókkal, a bejelentkezés gomb alatt levő regisztráció felíratra kattintva lehetősége van létrehozni egyet, ez a felirat a regisztrációs oldalra irányítja a felhasználót.

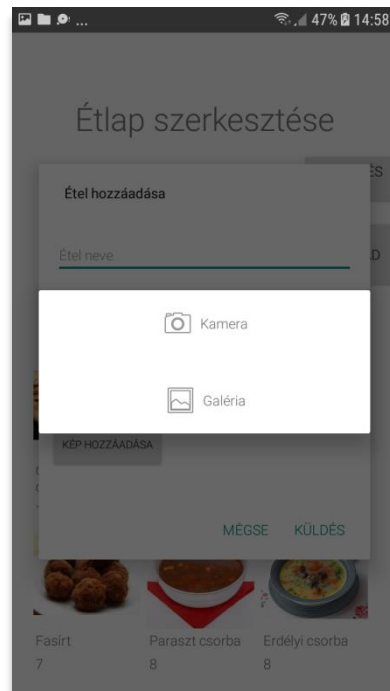
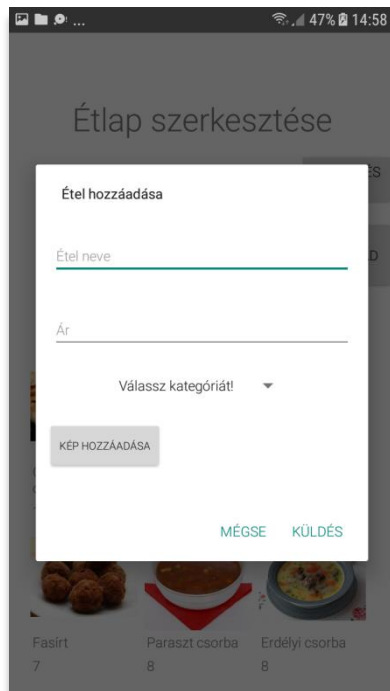
A mobile app registration screen titled "Regisztráció". It features a status bar at the top with icons for signal, Wi-Fi, and battery (43%), and the time 12:39. Below the title, there are five text input fields labeled "Vezetéknév", "Keresztnév", "Telefonszám", "Email", and "Jelszó". A grey button labeled "REGISZTRÁCIÓ" is positioned at the bottom of the form.

5.3 Étlap szerkesztési oldal

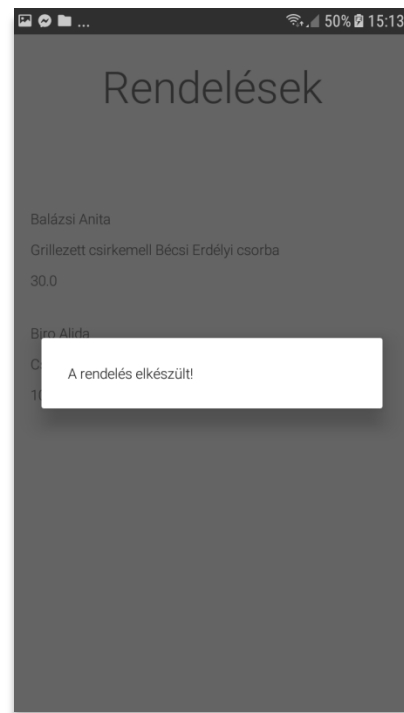
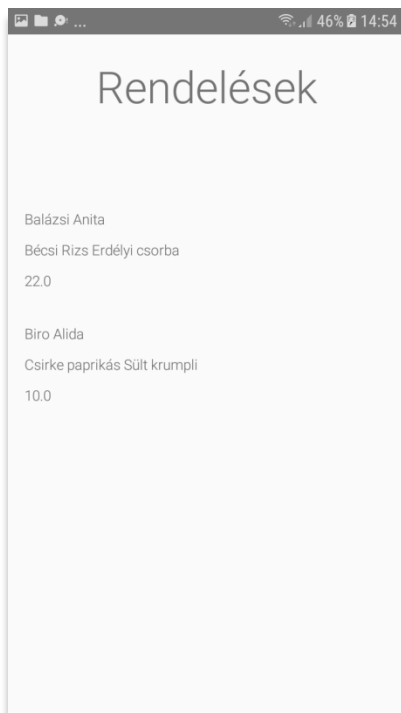
Ha a bejelentkezett felhasználó egy adminisztrátor a következő oldalon megjelenik az aktuális étlap és annak szerkesztésének lehetősége.



Ha törölni kíván egy adott ételt az étlapról, annak kijelölésével és a törlés gombra való kattintással egyszerűen megteheti azt. Ha új ételt szeretne hozzáadni az ÉTEL HOZZÁADÁSA gombra kattintva egy dialog box jelenik meg amelyben meg kell adnia az étel nevét, árát és kategóriáját. Ugyanitt képet is rendelhet az adott ételhez a KÉP HOZZÁADÁSA gombbal.

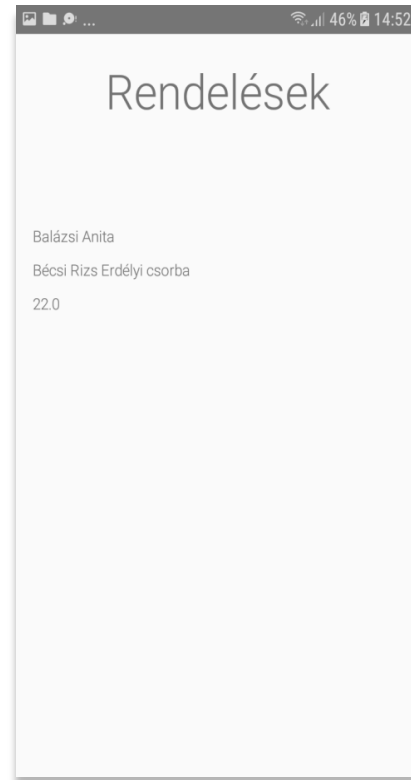
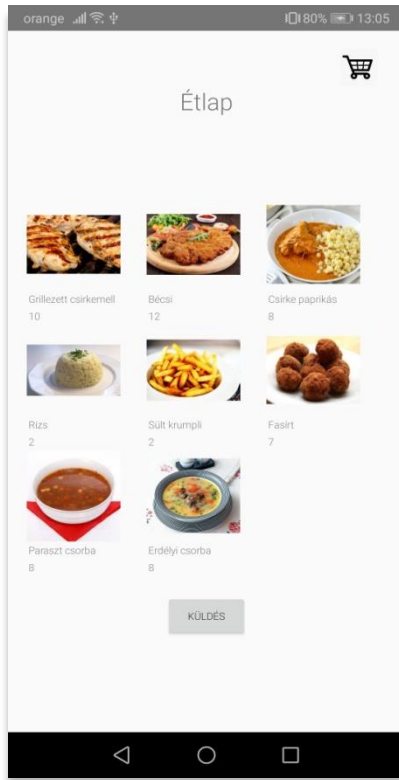


Egy további funkcionalitás az adminisztrátor részen az összes beérkező rendelés megtekintése. Ha egy rendelés elkészült, a rendelésre való kattintással megjelenik egy dialog box, amelyre kattintva egy SMS küldődik az adott kliensnek arról, hogy átveheti a rendelését.

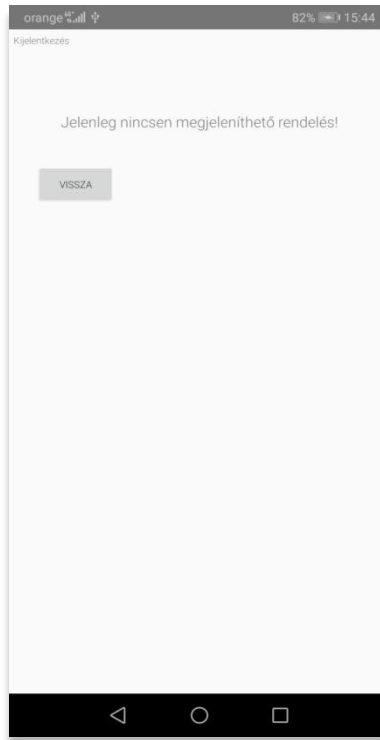


5.4 Étlap megtekintési oldal

Kliens oldalon a bejelentkezést követően megtekinthető az étlap. Rendelés leadásához a megfelelő ételekre kell kattintani, ezt követően pedig a küldés gombra. Ha a kliens a saját rendelését szeretné megtekinteni a jobb felső sarokban levő kosárra kattintva teheti ezt meg.



Amennyiben a rendelés megtétele előtt tekinti meg a bevásárló kosarát a következő üzenet jelenik meg:



6. Továbbfejlesztési lehetőségek

A továbbfejlesztésnél elsősorban a felhasználói felületet optimalizálnánk, egy letisztultabb, a felhasználók számára könnyen használható designnal. Az alkalmazás használóinak jelenleg egyidejűleg csak egy aktív rendelésük lehet, a továbbfejlesztésnél szeretnénk ezt kibővíteni és lehetőséget biztosítani arra, hogy a kliensek több rendelést is elküldhessenek. Emellett a várakozási időt jelentősen csökkentenénk, ha az alkalmazáson belül a rendelést ki is lehetne fizetni a bankkártya adatok megadásával.