



SAPIENTIA  
ERDÉLYI MAGYAR  
TUDOMÁNYEGYETEM

# Price-Tracker



Marosvásárhely,  
2019-12-23

Szerzők: Palfi Szabolcs  
Sólyom Botond

# Tartalomjegyzék

<b>1. Bevezető .....</b>	<b>3</b>
<b>2. Célkitűzések.....</b>	<b>3</b>
<b>3. Felhasználói Követelmények.....</b>	<b>3</b>
<b>4. Rendszerkövetelmények .....</b>	<b>5</b>
4.1 Backend.....	5
4.2 Frontend.....	5
4.3 Adatbázis.....	6
<b>5. Architektúra .....</b>	<b>6</b>
<b>6. Implementálás .....</b>	<b>7</b>
6.1 Backend.....	7
6.2 Frontend .....	8
<b>7. Alkalmazás Bemutatása .....</b>	<b>8</b>
7.1 Regisztráció.....	8
7.2 Bejelentkezés .....	9
7.3 Kezdő oldal .....	10
7.4 Termék információi .....	11
7.5 Statisztikák.....	11
<b>8. Következtetés.....</b>	<b>12</b>
<b>9. Bibliográfia .....</b>	<b>13</b>

## 1. Bevezető

A mai rohanó világban a bevásárlások egyre növekvő százaléka az interneten történik, mindez lehetőséget nyújtva a vásárlóknak, hogy egy bizonyos terméket több, akár hazai akár külföldi, oldalról is megvásárolhasson. Az e-commerce-el foglalkozó cégek rohamos fejlődésnek indultak az utóbbi évtizedben mely maga után vonja az érdekesebbnél érdekesebb marketing fogásokat, melyekkel a célközönséget próbálják vásárlásra bírni.

Valószínűleg mindenki hallott már a “Black Friday” az-az „Fekete Péntek” -nek nevezett jelenségről amely inspirációként szolgált az alkalmazás megvalósításához. Ez a kifejezés legelőször az 1800-as években fogalmazódott meg, amikor is Jay Gould és James Fisk az amerikai arany árak manipulálása által 20%-os esést okoztak a részvényt piacon melynek következtében az árucikkek értéke felére csökkent. A 20. század közepe fele ez már egészen más jelentéssel bírt, ugyanis a Hálaadás ünnepét követő napon, az-az pénteken vette kezdetét a karácsonyi árleszállítás, mely sok cég esetében életmentő volt, hiszen ekkor kerültek át a veszteséges állapotból melyet pirossal jelöltek, a jövedelmezőbe, amit már fekete írószerral jegyeztek fel. Ebben az időszakban a megszokottnál jóval nagyobb és több árleszállítással vonzották az embereket.

Mint azt sokan tudjuk, országunkban is nagy népszerűségnek örvend ez a jelenség, habár eléggé távol áll az eredeti koncepciótól. Nagyon sok mesterséges árleszállítással próbálják becsapni az embert, melyet legtöbb esetben jól kitervelt ár ingadozással oldanak meg. Ugyanakkor, nem kizárólag ebben az periódusban lehet észre venni az úgymond „hamis” kedvezményeket ezért szükségét láttuk egy olyan alkalmazás kifejlesztésének, amely nyomon tudja követni egy megadott termék árat, illetve annak ingadozását.

## 2. Célkitűzések

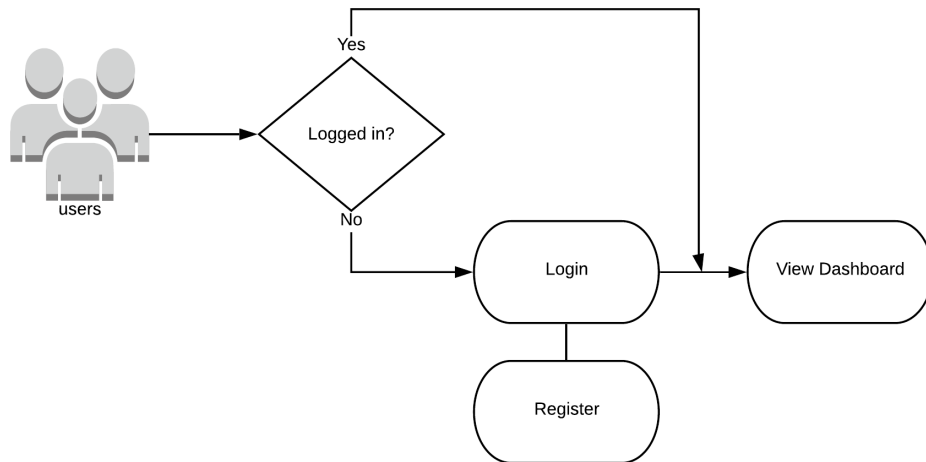
Az alkalmazás célja, hogy segítsen a vásárlónak reális képet alkotni egy termék áráról, illetve annak változásáról. Fontosnak tartottuk, hogy az alkalmazást egyszerűen és intuitívan lehessen kezelni, azaz bármilyen funkció pár érintésre legyen a felhasználótól.

Az árak változását a felhasználó számára vizuális formában szeretnénk volna ábrázolni a könnyebb átláthatóság érdekében. Lehetőséget szeretnénk volna adni a felhasználónak hogy adott kategóriákba sorolhassa a megadott termékeket majd e szerint jelenítsük meg őket.

## 3. Felhasználói Követelmények

A felhasználónak mindenekelőtt be kell tudnia jelentkezni, ahhoz, hogy elérje a terméklistáját, ez azért fontos, mivel eszköz váltás esetén nem szeretnénk elveszíteni az addig követett termékeket. A bejelentkezést megelőzően lehetséges a „ Remember Me ” opció kipipálása, amely lehetővé teszi, hogy ne keljen minden alkalommal újra bejelentkezni, amikor elindítjuk az alkalmazást. A bejelentkezéshez szükséges adatok a regisztrációkor megadott felhasználónév, illetve jelszó. Mindezt az 1.ábra illusztrálja.

#### Login Activity

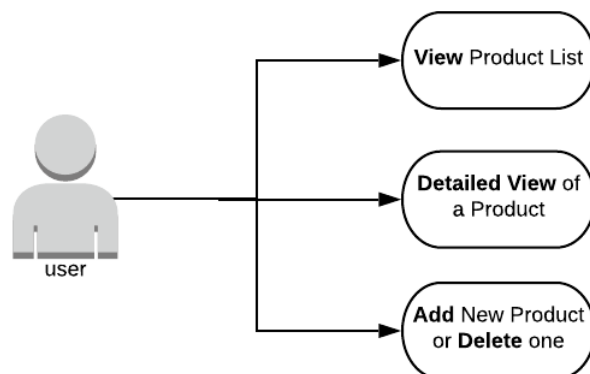


( 1.ábra )

Abban az esetben, ha a felhasználó nincs regisztrálva, megteheti ezt a „Not registered? Click here! ” szövegre kattintva. A regisztrációhoz szükséges egy felhasználónév, e-mail cím, valamint jelszó megadása. Sikeres regisztrálás esetén a felhasználó vissza kerül a bejelentkező ablakra, ahol be tud lépni.

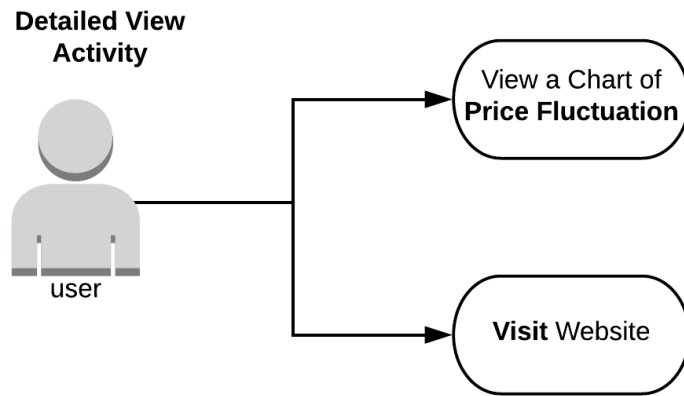
A bejelentkezést követően a felhasználó a főoldalra kerül, ahol a követett termékek listáját tekintheti meg. A Listában minden egyes elemnek látható a megnevezése, egy a terméket ábrázoló kép, illetve az adott termék aktuális ára. Továbbá ezen az oldalon lehetősége van a felhasználónak új termékeket hozzá adni a listához. ( 2.ábra )

#### Dashboard Activity



( 2.ábra )

Egy a listában lévő elemre kattintva, az alkalmazás átvisz egy másik oldalra, ahol több információt kapunk a követett termékről. A „ View Statistics ” gombra kattintva, egy grafikonon tekinthetjük meg a termék árának változását a hozzáadás napjától az aktuális dátumig. Az „ Open Link ” gombra kattintva az alkalmazás megnyitja a terméket tartalmazó linket az eszköz alapértelmezett böngészőjét használva. ( 3.ábra )



( 3.ábra )

## 4. Rendszerkövetelmények

### 4.1 Backend

Ahhoz, hogy a szoftver működőképes legyen szükséges egy rendszer, amelyen futtatni tudjuk. A rendszernek képesnek kell lennie Python 3-as kódot futtatnia, fel kell legyen telepítve a Python 3.8.0, valamint szükséges egy stabil azaz mindig aktív internet kapcsolat ( minimális fel-le töltési sebesség ne essen 10 Mb/s alá ) mivel a rendszer ezen keresztül kéri le, illetve tölti fel az információkat az adatbázisba.

### 4.2 Frontend

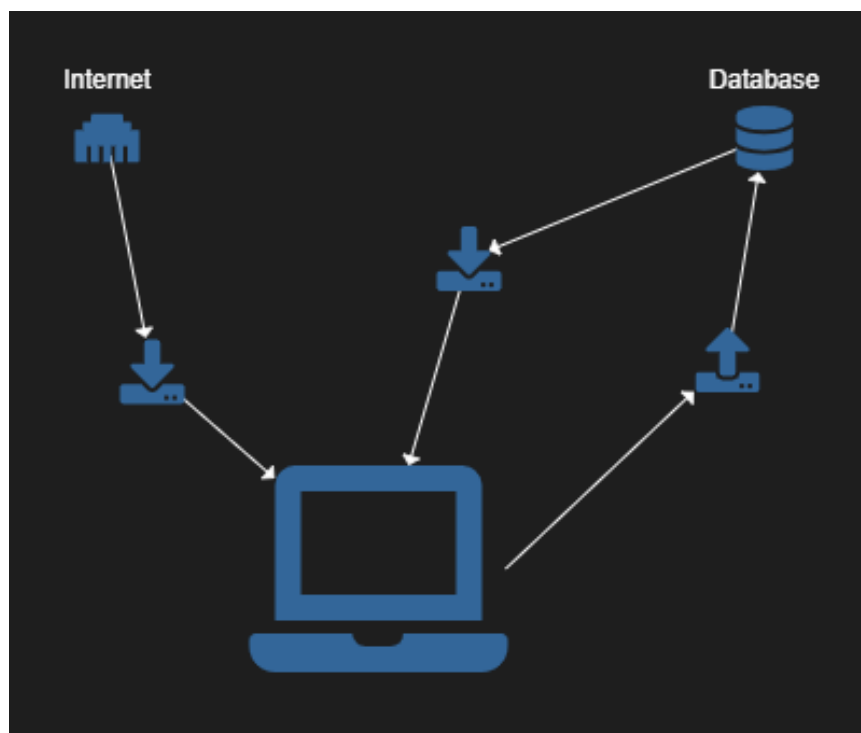
Az alkalmazás hibátlan működése érdekében a készülék minimum 5.0 –ás Lollipop Android operációs rendszerrel (Level API 21) kell rendelkezzen. A telepítéshez szükséges 12MB szabad tárhely, illetve az eszköznek rendelkeznie kell minimum 1 GB RAM memóriával az applikáció zökkenőmentes futása érdekében. Továbbá a használatához szükség van internet kapcsolathoz, ugyanis ennek használata elengedhetetlen az adatbázis és az applikáció kommunikálása érdekében. Internet kapcsolat nélkül már a bejelentkezés, illetve regisztráció sem lehetséges, ugyanis az applikáció nem képes hozzáférni az adatbázisban tárolt felhasználók adataihoz, valamint nem képes újabb felhasználókat hozzáadni sem. Tehát az internet kapcsolat követelmény megelőlegezi a zökkenőmentes kommunikációt az alkalmazásunk, és az adatbázisunk között, biztosítva ezáltal a felhasználónak egy naprakész, friss adatokat tartalmazó applikációt.

Alkalmazásunk megalkotása és tesztelése végett különböző technológiák felhasználásait vettük igénybe, melyek segítségével sikerült létrehozunk egy üzemképes és felhasználóbarát rendszert. A projektünk frontend részének kivitelezését az Android Studio integrált fejlesztő környezetében valósítottuk meg, teszteléséhez pedig saját Android rendszerrel rendelkező mobil eszközünket használtuk. Az így létrejött felületen a felhasználó által bevitt adatokat a Firebase nevű adatbázis segítségével tároltuk, illetve a megjeleníteni kívánt adatokat is ezen adatbázisból kérjük le.

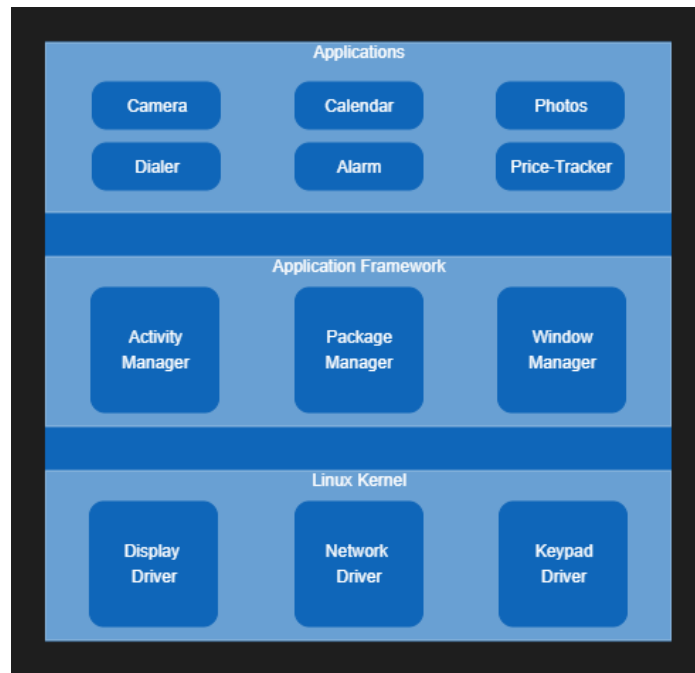
### 4.3 Adatbázis

A Firebase mobil és webes applikációt fejlesztő platformot választottuk adatbázisunk létrehozása érdekében, ugyanis ez kitűnőnek minősült célunk elérésére. Segítségünkre szolgált az, hogy ez nem egy megszokott adattáblákkal rendelkező adatbázis, hanem egy nosql, valós idejű rendszer, mely lehetővé teszi a felhasználó számára a pillanatnyi adatok láthatóságát, amit annak köszönhetünk, hogy az adat fel és letöltése valós időben történik.

## 5. Architektúra



( Backend )



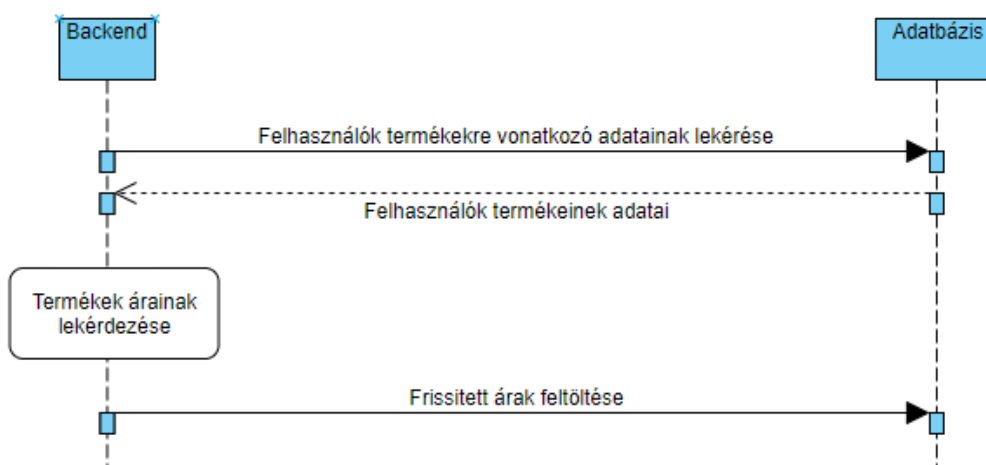
( Frontend )

## 6. Implementálás

### 6.1 Backend

A backend a felhasználó számára rejtett, nincs beelátása. Ez a rész felelős a frontend által ábrázolt, megjelenített tartalomért melyet egy bizonyos logika alapján állít elő. Implementálása Python 3 nyelven történt, PyCharm 2019.2.5 (Community Edition) fejlesztői felület segítségével. Megvalósítása során több függvény könyvtárat is használtunk, ezek közül kiemelkedő szerepük volt a „Beautiful Soup”, „requests” illetve „firebase” könyvtáraknak.

Egy adott felhasználó termékeinek ellenőrzésére több lépés szükséges. Első sorban az adatbázisból le kell kérdezni az összes terméket a velük megfelelő információkkal. Következő lépésben minden egyes termék esetében lekérjük a megadott url HTML kódját melyből ki tudjuk nyerni a termék árát. Ezt követően, a friss adatokat feltöltjük az adatbázisba. Az előbb leírt folyamatot a következő szekvencia diagramm illusztrálja.



Implementálás során szükség volt több osztály létrehozására, melyek sokkal átláthatóbbá tették a kódot. Ezek szükségesek voltak az adatok adatbázisból való lekérdezéséhez, mivel a szolgáltatott adatokat valamilyen módon parsolni kellett. Parsolás után minden adat belekerül a megfelelő osztály példányába, nagy mértékben megkönnyítve a velük való munkát.

Az adatbázisban található adatok naponta kerülnek frissítésre. Ezt a frissítést manuálisan kell végezni, a megfelelő függvény lefuttatásával.

## 6.2 Frontend

Az alkalmazásunk frontend része a felhasználóhoz tartozó legközelebbi rész, amelyben a teljes grafikus felület kivitelezése látható. Esetünkben Java nyelven íródott kód formájában nyilvánul meg a grafikus felület, avagy a „user interface” kivitelezése. Segítségünkre szolgált az Android Studio integrált fejlesztő környezete, melynek köszönhetően sikerült elérnünk egy interaktív vizuális felület megalkotását. Alkalmazásunk különböző funkcióinak a megjelenítését az activity-k használata teszi lehetővé. A felhasználónak lehetősége van navigálni a különböző activity-ken, látva azon adatokat, melyeket megjelenítenek, és használni azon funkciókat, melyeket ezen szolgáltatás nyújt nekik. Kezdetben megalkottuk az activity-k kinézetét, majd kiviteleztük a különböző funkciók működését. Az egyéb funkciók működésének alapkövét az adatbázissal való kapcsolat kialakítása képezte. Használatkor már az első lépésekben az applikáció kapcsolatba lép az adatbázissal, ugyanis a kezdő felületet egy bejelentkezési ablak képezi. A bejelentkezési ablak kapcsolatban áll a regisztrációs felülettel, bejelentkezés nélkül a felhasználó képtelen tovább lépni.

## 7. Alkalmazás Bemutatása

Alkalmazásunknak öt különálló felhasználói oldala van: Bejelentkezés (Login), Regisztráció (Register), Kezdőoldal, irányító pult (Dashboard), Termék információi (Product detail), illetve a Statisztikák (Statistics) oldalak. Az alkalmazás futtatása előtt szükséges bekapcsolni az internetet ugyanis ez a működésnek egy alapfeltétele. Az alábbiakban sor kerül ezen oldalak részletes bemutatására.

### 7.1 Regisztráció

Első és egyben legfontosabb eseménye alkalmazásunk használatának a regisztráció. Mint számos alkalmazás, így az általunk készített applikáció is megköveteli a felhasználó beregisztrálását a használathoz. Erre azért van szükség, hogy az applikációt bármely felhasználó az általa kívánt termékek árának követésére használja. A regisztrációs oldalt a felhasználó a “ Not registered? Click here! ” (Login oldalon) szövegre kattintva elérheti, amennyiben még nem regisztrált be. Ennek következtében az alábbi oldal fogadja felhasználóinkat.

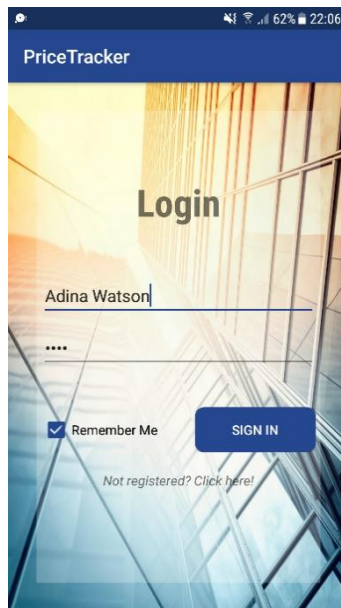




Regisztráció során felhasználóinknak négy mezőt kell kitölteniük, megadva ezáltal egy felhasználó nevet ( username ), melyet használnak majd a bejelentkezés során, meg kell adniuk továbbá e-mail címüket, illetve jelszavukat. A jelszót 2 -szer kell megadni annak érdekében, hogy ellenőrizhessük, hogy a felhasználó valóban mind a két esetben ugyanazt a jelszót adta meg. A regisztrációs oldal összes mezőjét kitöltve lehetőség adódik a tovább lépéshez, a “Confirm” gomb megnyomása által. Abban az esetben, ha valamely mező üresen maradt volna, a felhasználó az alábbi üzenettel szembesül: “ Please fill all fields of data! ”, illetve nem kerül sor a tovább lépésre sem. Amennyiben a regisztráció az elvárásoknak megfelelően zajlott az alkalmazás átlépik a következő oldalra, a Bejelentkezés (Login) oldalra.

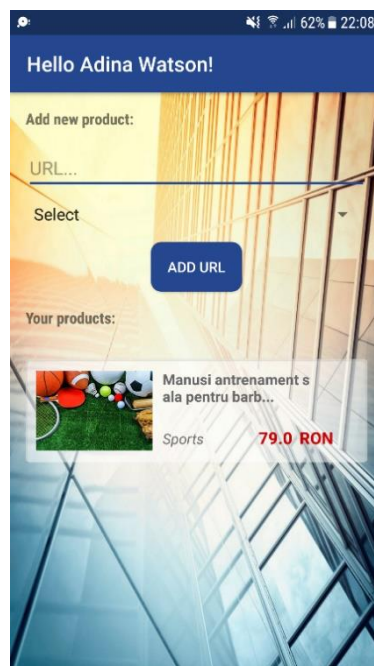
## 7.2 Bejelentkezés

A regisztrációt a bejelentkezés követi, melynek hiányában a felhasználó képtelen tovább lépni, ennek következtében a két oldal, a regisztrációs és a bejelentkezés oldal kapcsolatban állnak egymással, ezért aki még nem regisztrált, a bejelentkezési oldalról elérheti a regisztráció lehetőségét. Amennyiben a felhasználó sikeresen be regisztrált, lehetősége nyílik a bejelentkezésre, amelyet megtehet két mező kitöltésével. Kezdetben meg kell adnia a felhasználó nevét (amivel regisztrált), illetve a jelszavát. Fontos, hogy az adatokat helyesen adja meg. Alkalmazásunk lehetőséget nyújt arra, hogy amennyiben a felhasználó szeretné, mentse le a bejelentkezéshez szükséges adatait, mely kényelmesebbé teszi az applikáció használatát. Adatainak lementését megteheti a “ Remember Me ” négyzet bepipálásával. Végso soron a tovább lépést a “SIGN IN” gomb teszi lehetővé, melynek lenyomásával a felhasználó tovább léphet a kezdőoldalra, ahol követni kívánt termékeit adhatja hozzá listájához, és kiválaszthatja a már megadott termékeket.



### 7.3 Kezdő oldal

A bejelentkezést követően a felhasználó a kezdőoldalra lépik, ahol lehetősége nyílik arra, hogy a már meglévő termék listáját követni tudja, illetve amennyiben új terméket kíván hozzáadni a listához, azt ezen oldalon szintén megteheti. Az új termék hozzáadását az “ Add new product: ” alatt elhelyezkedő mezőbe az adott termék adott oldalon követni kívánt URL címének a bemásolásával megteheti. Itt szükség van a URL cím bemásolására, illetve a termék kategóriába sorolására, majd az “ ADD URL ” gombra kattintva sikerül véglegesíteni az újabb követni kívánt termék hozzáadását. Az új termék hozzáadási menüpont alatt található a felhasználó termék listája (“ Your products: ”). Minden egyes termék ablakocská tartalmazza a nevet, azt, hogy milyen kategóriába tartozik és a jelenlegi árát a terméknek.



A listából egy adott termékre kattintva az alkalmazás tovább lépik a következő oldalra, amely az adott termék részletesebb információit tartalmazza.

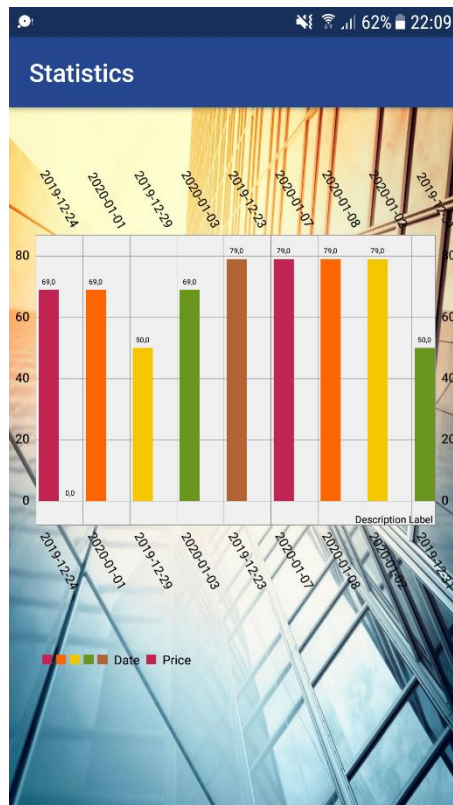
## 7.4 Termék információi

Ez az oldal fényt derít a kiválasztott termék információira. Kezdetben megjelenik egy az adott kategóriát szemléltető kép, majd ezt követően a termék neve, kategóriába sorolása (“Category: “), a termék linkje, melyre kattintva az alkalmazás tovább navigálja a felhasználót az adott oldalra, melyről megvásárolhatja a terméket (“ Open link: ”), végül pedig látható a jelenlegi ár, az adott pénznemben. Az információk sorozata alatt egy “ View Statistics ” nevű gombot láthatunk, mely elnavigálja felhasználónkat a Statisztikákat tartalmazó oldalra.



## 7.5 Statisztikák

A “ View Statistics ” gomb életbe léptetése során a statisztikákat tartalmazó oldal nyílik meg a felhasználó előtt. Ez az oldal egy diagramot tartalmaz, mely a termék árának adott dátum szerinti változását hivatott szemléltetni. Ez az applikáció tulajdonképpen összegző oldala, melynek tanulmányozásával a felhasználó nyomon követheti azokat a dátumokat, mikor érdemes megvásárolni az adott terméket zseb barát, vagy kedvezőbb áron.



## 8. Következtetés

Alkalmazásunk szinte teljességében az előre eltervezett módon működik. A felhasználóknak lehetőségük van be jelentkezni, regisztrálni továbbá hozzáadni termékeiket a követési listájukhoz. Meg is tekinthetik ezeket, tudva, hogy naponta frissülnek az adatok. A termékek áráról megjelenített grafikon minden ellenőrzés adatát tartalmazza, az adott termék hozzáadásának pillanatától az aktuális dátumig.

Sok szempontból lehetne fejleszteni az alkalmazáson úgy a backend mint a frontend részen is. Mivel jelenleg az adatok frissítése „manuálisan” történik, ezért a jövőbeli tervek között első helyen szerepel az adatok feldolgozásának automatizálása, illetve egy esemény alapú válasz kifejlesztése, hogy a felhasználó mindig valós időben a legfrissebb adatokat lássa.

U.I. szempontjából modernebbé és aktuálisabbá szeretnénk tenni az alkalmazást, apró forma béli változásokkal, simításokkal.

## **9. Bibliográfia**

- Head-First Python, 2nd edition - Paul Barry (O'Reilly, 2016)
- <https://www.python.org/doc/>
- <https://firebase.google.com/>
- <https://developer.android.com/>