UNIVERSITATEA SAPIENTIA DIN CLUJ-NAPOCA
FACULTATEA DE ȘTIINȚE TEHNICE ȘI UMANISTE, TÎRGU-MUREȘ
PROGRAMUL DE STUDII ...

Price Monitor

PROIECT DE DIPLOMĂ

Coordonator ştiinţific:
Dr. Szántó Zoltán

Absolvent:
Palfi Szabolcs

2021

UNIVERSITATEA "SAPIENTIA" din CLUJ-NAPOCA **Viza facultății:**
Facultatea de Științe Tehnice și Umaniste din Târgu Mureș
Specializarea: **...**

LUCRARE DE DIPLOMĂ

Coordonator științific:	Candidat: Anul absolvirii:

a) Tema lucrării de licență:

b) Problemele principale tratate:

c) Desene obligatorii:

d) Softuri obligatorii:

e) Bibliografia recomandată:

f) Termene obligatorii de consultații: săptămânal

g) Locul și durata practicii: Universitatea Sapientia,
Facultatea de Științe Tehnice și Umaniste din Târgu Mureș

Primit tema la data de:
Termen de predare:

Semnătura Director Departament **Semnătura coordonatorului**

Semnătura responsabilului programului de studiu **Semnătura candidatului**

Declaraţie

Subsemnatul/a … , absolvent al specializării …, promoţia … cunoscând prevederile Legii Educaţiei Naţionale 1/2011 şi a Codului de etică şi deontologie profesională a Universităţii Sapientia cu privire la furt intelectual declar pe propria răspundere că prezenta lucrare de licenţă/proiect de diplomă/disertaţie se bazează pe activitatea personală, cercetarea/proiectarea este efectuată de mine, informaţiile şi datele preluate din literatura de specialitate sunt citate în mod corespunzător.

Târgu Mureș,

Data:

Extras

Extract

Cuvinte cheie:

SAPIENTIA ERDÉLYI MAGYAR TUDOMÁNYEGYETEM
MAROSVÁSÁRHELYI KAR
SZÁMÍTÁSTECHNIKA SZAK

Price Monitor

DIPLOMADOLGOZAT

Témavezető:
Dr. Szántó Zoltán

Végzős hallgató:
Palfi Szabolcs

2021

Kivonat

Napjainkban az emberek nagy többsége a vásárlásaikat az online térben bonyolítják le. Rengeteg webáruház létezik, szinte állandóan vannak kedvezmenyék vagy ajánlatok. Viszont sok esetben a feltüntetett árak ingadoznak, vagy a kedvezmény csak látszólagos. Ezekre akkor lehet felfigyelni, ha napi szinten követjük az árak alakulását, viszont ez időigényes és repetitív folyamat. Ugyanez elmondható akkor is, ha egy terméket megszeretnénk vásárolni kedvező áron. A dolgozatban bemutatunk egy árukövető rendszert, mely ezt a folyamatot automatizálja. A rendszer meghatározott idő pillanatokban felébred, az Interneten elérhető publikus adatokat bányássza, és az eredményeket elmenti. A felhasználónak egy böngésző kiegészítőt biztosítunk, mely segítségével hozzá adhat termékeket a követéshez a támogatott weboldalakról, illetve grafikonokon az ár változását is követheti. Ugyanakkor, mivel a vásárlások egyre nagyobb része zajlik telefonos alkalmazásokon keresztül, ezért ilyen platformra is elérhetővé tesszük a szolgáltatást egy Flutter segítségével készült alkalmazáson keresztül.

Kulcsszavak: webscraping, böngésző kiegészítő, Flutter alkalmazás

Abstract

Abstract

Keywords:

Tartalomjegyzék

Ábrák jegyzéke

1. fejezet

Bevezető

A mai rohanó világban a bevásárlások egyre növekvő százaléka történik az Interneten, 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 részvénypiacon melynek következtében az árucikkek értéke felére csökkent [1]. 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 írószerrel jegyeztek fel [2]. 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 em-

[1] https://en.wikipedia.org/wiki/Black_Friday_(1869)

[2] https://en.wikipedia.org/wiki/Black_Friday_(shopping)

bert, melyet legtöbb esetben jól kitervelt ár ingadozással oldanak meg[3]. Ugyanakkor, nem kizárólag ebben a periódusban lehet észrevenni az úgymond „hamis” kedvezményeket ezért szükségét láttuk egy olyan alkalmazás kifejlesztésének, amely nyomon tudja követni egy megadott termék árat, illetve annak ingadozását.

Mivel az Interneten publikus adatok találhatók, ezek felhasználásával semmiéle kár nem keletkezik az adott weboldalak számára. Egy internetes oldal betöltése során, mi, mint felhasználók, egy kérést intézünk egy szerver fele a böngészőnkön keresztül, ami majd a kapott válasz alapján felépíti és megjeleníti számunkra a megtekinteni kívánt oldalt. Ezt a műveletet természetesen legtöbbször ahogy az előbbiekben is említettem, böngészőn keresztül végezzük, viszont ez nem egy szükséglet, inkább egy eszköz, számos más módon is intézhetünk kéréseket egy adott szerver fele. Az általunk megvalósítani kívánt alkalmazás ezt a tulajdonságot hivatott kihasználni, ezáltal nyilvánosan elérhető adatok begyűjtésével, feldolgozásával és elemzésével szeretne foglalkozni.

[3] https://cavaleria.ro/tepele-de-black-friday-2020/

2. fejezet

Célkitűzések

Az alábbi fejezetben összefoglaljuk az alkalmazás fontosabb célkitűzéseit. A fő cél, mely érdekében létrejön a szoftver, az, hogy segítsen egy potenciális vásárlónak követni egy adott termék árának időbeli változását. Ennek megfelelően, a következő célok fogalmazódtak meg:

- Rövid használati útmutató, szöveges formában
- Támogatott oldalak listázása
- Regisztrálási, bejelentkezési lehetőség biztosítása, hogy különböző eszközökön is, mint például mobilos vagy webes, elérhetőek legyenek a követett termékek információi
- Felhasználói fiók jelszavának változtatási lehetősége, felhasználói fiókból való ki jelentkezés valamint annak törlése.
- A termékek ábrázolása listaszerűen történjen
- Az árak változását a felhasználó számára grafikus formában szeretnénk ábrázolni a könnyebb átláthatóság érdekében
- Az árak időben való változását könnyen átlátható vonal diagramon ábrázolni
- Egyszerű átirányítás a termék oldalára
- Telefonos alkalmazás

- Kiegészítő, Chrome alapú böngészőkre

3. fejezet

Szakirodalom áttekintése

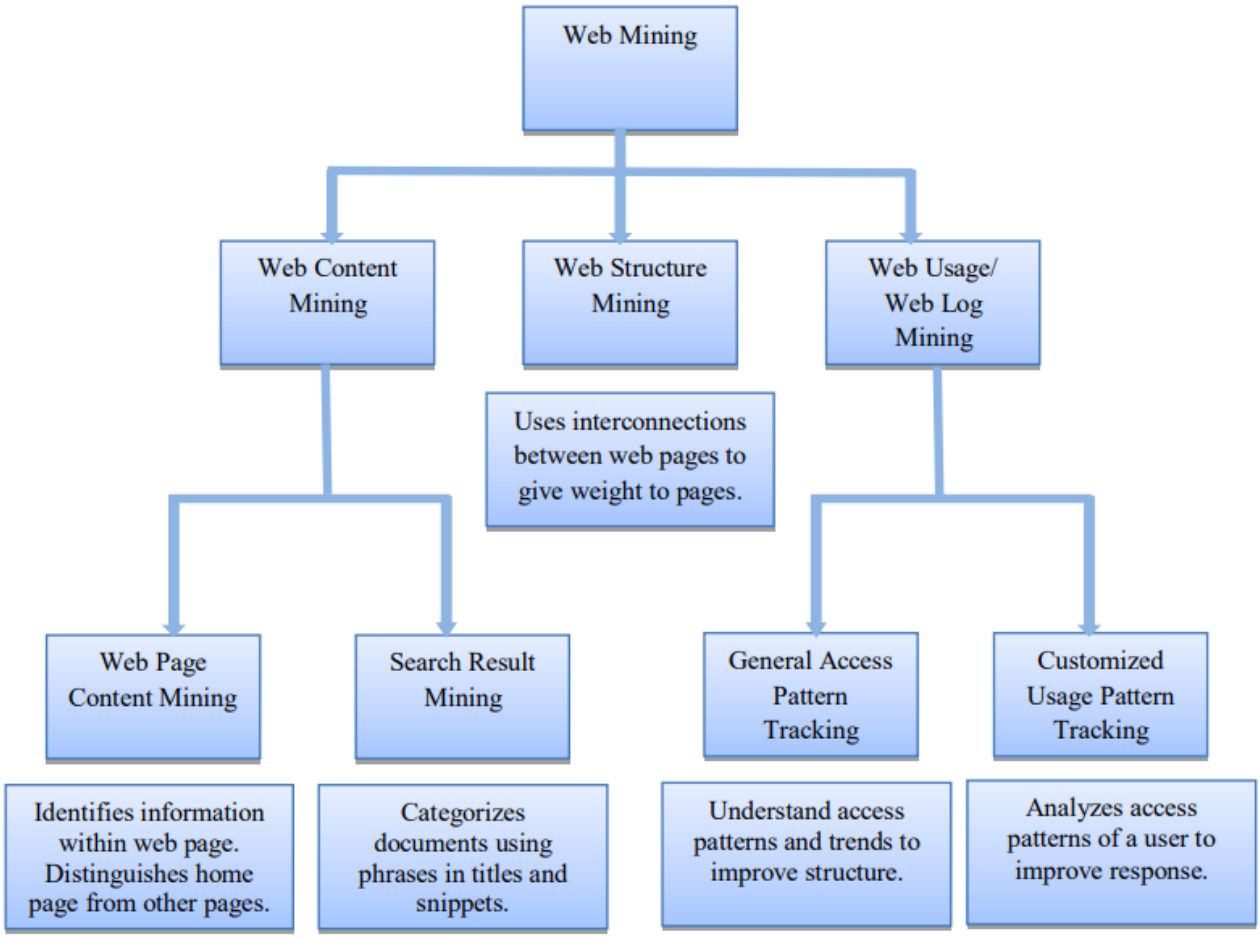
3.1. Web bányászat

Web bányászat, vagy angol kifejezésben Web mining, alatt azt a folyamatot értjük, amely által új, eddig nem ismert, de hasznos információt fedezünk fel az Interneten található adatok között. Mindezt a cégek arra használják, hogy új értékes információkat gyűjtsenek, ezeket feldolgozzák, majd ezek által a felhasználókat vagy fogyasztókat jobban megismerjék. Ez a folyamat az úgynevezett adat bányászat technikát alkalmazza ahhoz, hogy automatikusan kinyerje az adatokat az Internetről [2].

Számos más technikát is alkalmaztak már új információk kinyerésére, az így is hatalmas és egyre növekvő adat mennyiségből, mint például az Information retrieval, Information extraction illetve gépi tanulás. Az Information retrival működési elve az, hogy a szöveg indexelésé után nyeri ki a hasznos információt. Az Information Extraction arra fokuszál, hogy csak a lényeges információt nyerje ki, míg az előbb említett inkább hasznos dokumentumokat jelöl meg. A gépi tanulásos módszer nem kötődik direkt módon a Web Scraping-el viszont segítséget nyújt a szövegek osztályozási folyamatában. A web bányászatot három fő kategóriába soroljuk, ahogy az látható az 3.1 ábrán is.

A Web Content Mining vagy magyarul webes tartalom bányászat, olyan tartalmakra fekteti a hangsúlyt, mint például szöveg, kép, audió, video, metaadatok, hiperlinkek. Ezek tanulmányozása, segít nekünk megérteni a felhasználók, vásárlók viselkedését mely által a weboldalak teljesítményét növelni lehet a későbbiekben, hogy azok jobban, illetve hatékonyabban működjenek.

Mivel a webes tartalom bányászat megvizsgálja úgy a kereséseket, mint azoknak a konkrét tartal-



3.1. ábra. A Web Bányászat rendszertana [1]

mát ezért ezt is tovább lehet osztani két kategóriába, Keresési Eredmény, illetve Weboldal Tartalom bányászat. Nevükből adódóan, ezek kiegészítik egymást, mivel a tartalom bányászat azon oldalakon történik, melyeket korábban megvizsgált és ígéretesnek talált a Keresési eredmény általi elemzés. A Web Structure Mining egy olyan ágazat, mely struktúrák bányászásával foglalkozik, mint például HTML vagy XML címkék, amely által weboldalak közötti kapcsolatot tud felismerni, ezáltal súlyokat rendelve azokhoz. A Web Usage Mining által lehet megérteni a különböző használati mintákat, amelyeket a felhasználók követnek, ezeket főként naplózás, felhasználók profiljai, sütik, könyvjelzők által, de ugyanide tartoznak a különböző egér mozdulatok vagy görgetési adatok is.

3.2. Web Struktúra Bányászat

Rengeteg új adat generálódik az Interneten, napról napra az adat mennyiség exponenciálisán növekszik. Bőségesen állnak rendelkezésünkre szolgáltatások, illetve információk, elektronikus áruházak, elektronikus újságok, közösségi oldalak formájában, hogy csak párat említsünk. Habár ezen adatok fogyasztás céljára lettek szánva, sok időt el lehet tölteni az adatok kinyerésével és elemzésével. Továbbá, a weboldalak adatai HTML, illetve más webre szánt formátumban vannak jelen, amely megnehezíti az automata feldolgozást. Ez lett a mozgató rugója az ezen a téren zajló kutatásnak, mint például a Web Scraping.

A Web Structure Mining, vagy ismertebb nevén Web Scraping, az a folyamat, amely által hasznos információkat nyerünk ki egy weboldal HTML kódjából, amely az Internet fő formázási eszköze [3]. Az egyik metodológia megállapította, a rendezett annotációk, melyek nem mások, mint gépek számára is értelmezhető leíró információk az adott oldalra tekintve, egy külön szemantikus rétegben vannak tárolva, elválasztva a weboldaltól, ezáltal megkönnyítve és felgyorsítva a bányászási folyamatot, mivel először ezeket a fontos metaadatokat dolgozza fel, majd csak ezt követően magát a weboldalt.

A HTML struktúráját tekintve, két fő adat típust lehet bányászni belőle. Az egyik a felhasználó által generált - a másik pedig a metaadat. A felhasználó által generált adat bármi olyan típusú adatra vonatkozik, amelyet a felhasználó hozott létre vagy adott hozzá a weboldalhoz, akár személyesen akár egy közösségi platform hozzácsatolásával. A metaadat definíció szerint olyan adat, amely leír egy másik adatot [4], ezek általában minden weboldalon megtalálhatók, fontosabb leíró adatokat tartalmazva, mint például szerző, cím, cikkek eseten megjelenés időpontja, kulcsszavak. Ezek általában nem láthatóak a felhasználó számára, viszont kiolvashatóak az adott oldal HTML kódjából [4].

3.2.1. Szemelyre szabott információ lekérése példával illusztrálva

Tegyük fel, hogy egy személy fárasztónak találja, hogy a nap végen a fontos vagy számára értékes hírek után keressen, vagy előkeresse a kedvenc sport csapata elért eredményeit. Egy intelligens Web Scraper a tökéletes eszköz ebben az esetben, mivel az időközönként végig tudja böngészni az Internetet a felhasználó által megszabott témakörökben található információk után kutatva, vagy akár specifikus kulcsszavakat tartalmazó híreket előkertivé. Ez egy előre weboldalakat tartalmazó listán

menne végig, melyet a felhasználó határoz meg, hogy a számára hiteles információt kapja meg [3].

Egy másik példa egy Web Scraper felhasználására az, amikor például egy vásárló több terméket is kinézett magának, több különböző weboldalon. Ha esetleg nem szeretné azonnal megvásárolni a terméket, hanem csak követni annak árát, akkor esettől függően, több oldalra is be kell jelentkeznie, több felhasználóval oldalankent, de legjobb esetben is számos oldalt kellene naponta megfigyeljen és lejegyezzen. Egy Web Scraper abban könnyítené meg a felhasználó dolgát, hogy például egy böngésző kiegészítő keretein belül, a vásárló hozzá tudja adni egy listához a terméket és attól a pillanattól, a program naponta akár többször is le tudja kérni a termék árát, anélkül, hogy a felhasználó bármit is tenne. Ezáltal egy helyen lenne a vásárló több terméke, és pár kattintással tisztább kepét alkothat a termékek árára vonatkozóan

3.3. A Web Scraping gyakori alkalmazásai

- Online ár összehasonlítás – ugyanazon termék arának összehasonlítása több weboldalon, pl. https://www.compari.ro/, https://www.price.ro/
- Contact Scraping – általában email címeket gyűjtenek, marketing céljából
- Időjárással kapcsolatos adatok gyűjtése
- Weboldal változásainak figyelése
- Több forrásból származó adat egyesítése
- Kedvezmeny kuponok pl. pouch – chrome extension
- Álláshirdetések összesítése
- Brand monitoring – egy bizonyos márkához tartozó adatokat gyűjtik, általában az ahhoz társított véleményre kíváncsiak
- Piac tanulmány – Egy adott termék piacon való elhelyezkedését, illetve potenciális sikerességet próbáljak megjósolni a bányászott adatok átvizsgálásával.

3.4. Web Scraping Technikák

Számos technika áll rendelkezésünkre melyekkel az adatgyűjtést végezhetjük, ezeket mindig az adott helyzetnek megfelelően kell kiválasztani, főként a hatékonyságot tartva szemelőt. Ebben a részben a Web Scraping egy pár technikája kerül röviden ismertetésre.

Copy-paste - Időközönként valaki kézzel történő adatgyűjtést, valamint vizsgálatot végez. Adott helyzetekben ez a leghatékonyabb módszer, viszont nagyon hajlamos a hibákra, sok időt és fáradságot vesz igénybe az ember részéről, amíg a nagy adathalmazokat feldolgozza.

Reguláris kifejezések - Ez egy egyszerű és erőteljes megközelítése az információ gyűjtésnek. A UNIX vagy más programozási nyelv által használt reguláris kifejezés illesztésen alapszik.

HTML Parsing – Félig strukturált lekérdező nyelvek segítségével elemezni, illetve módosítani a weboldalak tartalmát.

DOM Parsing – A böngészőkbe beépített kezelő programok segítségével, az erre a célra fejlesztett alkalmazások lekérhetik a kliens oldalon dinamikusan létrejött tartalmakat is, mellyel utána fel tudják építeni a DOM fát, ebben pedig könnyebben lehet specifikus adatok után keresni.

Web Scraping Szoftver – Számos szoftver áll rendelkezésünkre, amelyeket személyre szabott keresésre lehet használni.

Mesterséges Intelligencia – Több helyen is kísérleteznek gépi tanulásos adatbányászattal, melynek az a célja, hogy a gépek megtanulják úgy értelmezni a weboldalakat, ahogy azt az emberek tennék.

3.5. Web Scraping Szoftverek

A Web Scraping szoftverek rendkívül fontos szerepet játszanak ezen a téren, mivel automatizálják és rendkívül felgyorsítják az adat-gyűjtő, valamint feldolgozó folyamatot. Számos ilyen szoftver található a piacon, a maga előnyeivel és hátrányaival. Ezeknek az ára a funkcionalitásuk függvényében, valamint a támogatás és frissítési időszakok függvényében változik.

Visual Web Ripper[1] – Az egyik legfejletteb web scraping szoftver, melyet a Sequentum csoport fejlesztett 2006-tól kezdődően. Weboldalakról gyűjtött információk bányászására használják, úgy egy-

[1] http://visualwebripper.com/

szerű weboldalak, mint e-commerce oldalak esetében, mint például eBay, Amazon, magento, azonban titkosított tartalmak eseteben is segítségünkre lehet. A bányászott adatokat kimenthetjük adatbázisba vagy CSV, illetve XML formátumban is. Előnye, hogy vizuális felülettel rendelkezik, ezért rendkívül egyszerűen ki lehet választani, hogy pontosan mit is szeretnénk. Egyszeri fizetéssel lehet megvásárolni a szoftvert, $349.00 áron[2], viszont fontos szempont, hogy ezen alkalmazás elveszti a gyártó általi támogatottságát 2021-gyel kezdődően, kivételt képezve azon esetek, ahol a vásárlóval karbantartási szerződés van érvényben, mely túlhaladja ezt az időpontot.

Web Content Extractor[3] – Nagyon jó automatizálási lehetőséget nyújt, rendkívül egyszerűen használható, pár kattintással meg lehet adni a kívánt mintát, ami szerint majd adatokat fog gyűjteni. A program rugalmas, abból a szempontból, hogy nem próbálja túlokoskodni a felhasználót, hanem egy előlnézetet ad az eredményről, majd a felhasználó maga végezheti el a szükséges módosításokat, amennyiben szükség van erre. Ezt a szoftvert, bérlet alapú előfizetéssel lehet megszerezni, több változat is elérhető, az árak $30 - $150 / hónap[2] között mozognak.

Mozenda[4] – Az egyik legegyszerűbben használható szoftver ezen a téren, ami lehetővé teszi a kevésbé technikailag hozzáértő személyeknek is, az egyszerű bányászásokat. Fő különbsége más alkalmazásokhoz kepést, hogy maga az adatbányászati folyamat a felhőben történik és nem a felhasználó erőforrásait felhasználva, amely hatalmas előnyt jelenthet. Elérhető egy 30 napos próba csomag is, ami után $250/hónap-tól[2] kezdődően változnak az árak, a választott csomag függvényében.

Screen-Scraper[5] – Nagyon fejlett web scraping alkalmazás, amelyet több változatban is el lehet érni. Az alapszínűt verzió ingyenes, ezzel egyszerűbb adatok után lehet bányászni, könnyen kezelhető, nem kell sok technikai tudás hozzá. Más változatok, mint például a profi vagy vállalkozás szintű verziók már sokkal komplexebbek, több lehetőséget nyújtanak. Nagy előnyé, hogy más rendszerekkel könnyen összeférhető pl. Java, ezért fel lehet használni más, nagyobb szintű programokban is.

Természetesen sok más program is rendelkezésünkre áll, melyek hasonló funkcionalitásokkal rendelkeznek, minden esetben az adott alkalmazásnak megfelelőt és legjobban illőt érdemes választani a nagyobb hatékonyság érdekében. Egyéb web scraper szoftverek [5]: WebHarvy, Easy Web Extract,

[2] Az árak aktualitásáért lásd a szolgáltató weboldalát

[3] https://www.webcontentextractor.com/

[4] https://www.mozenda.com/

[5] https://www.screen-scraper.com/

WebSunDew, FMiner, Scrapy, import io.

3.6. Legális és Etikai keretek

Ebben a fejezetben a legális valamit etikai kérdésekről lesz szó, amely elég megosztó, illetve nem teljesen egyértelmű terület. Egyenesen a Web Scraping-et nem szabályozza semmilyen törvény, de több területen is problémába lehet ütközni, ilyen például a védett tartalom, az úgynevezett szerződésszegés vagy GDPR. Attól függően, hogy a bányászat milyen országhoz tartozó területen zajlik, figyelembe kell venni az ottani törvénykezést is, ezért is nehéz konkrétan jellemezni legalitási szempontból. A legpontosabb jellemzés talán az lenne, hogy van, ami igen és van, ami nem.

3.6.1. Felhasználói feltételek

Ugyanúgy, mint egy szoftver vagy szolgáltatás esetén, amikor egy weboldalt használunk el kell fogadnunk bizonyos felhasználói feltételeket, melyek leggyakrabban kis felrúgó ablakkent jelennek meg, amikor először látogatunk a weboldalra, vagy ezeket a regisztrálás folyamán tudatosítják velünk. Amennyiben valamilyen módon megszegjük ezeket a feltételeket, érvénybe lép a fentebb említett szerződésszegés. Mivel ez csak akkor érvényes, ha a felhasználó explicit elfogadja a feltételeket, jogi szempontból nem nehéz kizárni a Web Scrapinget.

3.6.2. Szerzői jogok

Bányászni vagy újra publikálni olyan adatokat vagy információkat, amelyeknek a szerzője explicit módon fenntartja a szerzői jogokat, legális szempontból szerzői jogsértésnek minősül. Ugyanakkor egy weboldal nem minden esetben rendelkezik a felhasználói által generált adatokkal, vegyünk példának egy film értékelő oldalt, ahol a felhasználók kifejtik a véleményüket [6]. Más ebbe a kategóriába tartozó tartalmak például a videók, kepék, zenék, adatbázisok, cikkek. Maga a bányászása ezeknek az adattoknak nem teszi illegálissá, a felhasználási módjuk határozza meg azt, hogy milyen kategóriába soroljuk azt.

3.6.3. GDPR

Az Európai Unió által érvénybe léptetett Általános Adatvédelmi Rendelet[6] alapján nem tiltott az adatbányászat, kivételt kepézve, ha ez a tevekénység nem tartalmaz személyes adatokat. Ilyennek minősül a név, lakcím, email cím, telefonszám, bankkártya adatok, banki adatok, IP cím, születési dátum, foglalkoztatási információk, orvosi adatok, személyes fotók vagy videók.

3.6.4. Weboldal Károsítása

Ha bármilyen tevekénység által, amelyet a Web Scraper végez, túlterheljük az adott weboldal szervereit vagy bármilyen módon sértjük, gátoljuk annak működését bűncselekménynek számít és legális következményei lehetnek. Ehhez azonban a kárnak anyaginak kell lennie, valamint könnyen bizonyíthatónak bíróság elött, ahhoz, hogy bármiféle kártérítési kérelemre legyen jogosult a weboldal [6].

A fentieket figyelembe véve tehát, nem lehet egyértelmű választ adni arra, hogy a Web Scraping legális-e vagy sem. Helyette a válasz az, hogy helyzettől függ. Maga a Web Scraping nem illegális, viszont akár az is lehet a következő három dolog függvényében [7].

- Hogyan lett bányászva az adat?
- A bányászott adat típusa
- Hogyan lesz felhasználva a bányászott adat?

Ahhoz, hogy eldöntsük az esetünk legalitását meg kell vizsgálni, hogy az adat, amit szeretnénk bányászni publikusan elérhető-e vagy sem. Ha az adat eléréséhez nem szükséges bejelentkezni egy adott oldalra akkor a felhasználói feltételek nem érvényesek ezért legálisan lehet bányászni, mivel ez az adat publikusnak számít. Ha bejelentkezésre van szükség a bányászni kívánt adat eléréséhez, akkor a felhasználói feltételek tanulmányozásával el kell dönteni, hogy legális-e vagy sem, mivel azok elfogadásával legálisan alkalmazhatóvá vált számunkra, azaz a megszegésük jogi következményeket vonhat maga után.

A bányászott adat típusa szerint két formájára kell nagyon odafigyelni, Személyes Adatok és Szerzői Jogokkal rendelkező adatok, ezek fentebb részletesebben is tárgyalva voltak.

[6] https://gdpr-info.eu/

Az adatok felhasználása során figyelembe kell venni, hogy az illegális módon vagy csalás által történő adatgyűjtést minden állam bünteti, tehát ez bűncselekménynek minősül. Amennyiben olyan tartalmakat bányászunk melyek bizalmas vagy bármi féle módon védett információt tartalmaznak, ezeknek felhasználása ugyancsak törvényszegést jelent és az erényben levő büntető eljárások érvényesek. Természetesen, legtöbb esetben már maga az törvénysértő lehet, hogy ezekhez az adatokhoz jogosultság nélkül fértünk hozzá, ezért egyértelműen nem felhasználhatóak ezek az adatok.

Leegyszerűsítve, három alapvető kérdésre kell választ adni, ahhoz, hogy eldöntsük legális-e az adat bányászat, amelyet végre szeretnénk hajtani:

- Bejelentkezés által elérhető adatot bányászok?
- Személyes adatot bányászok?
- Szerzői jogokkal rendelkező adatot bányászok?

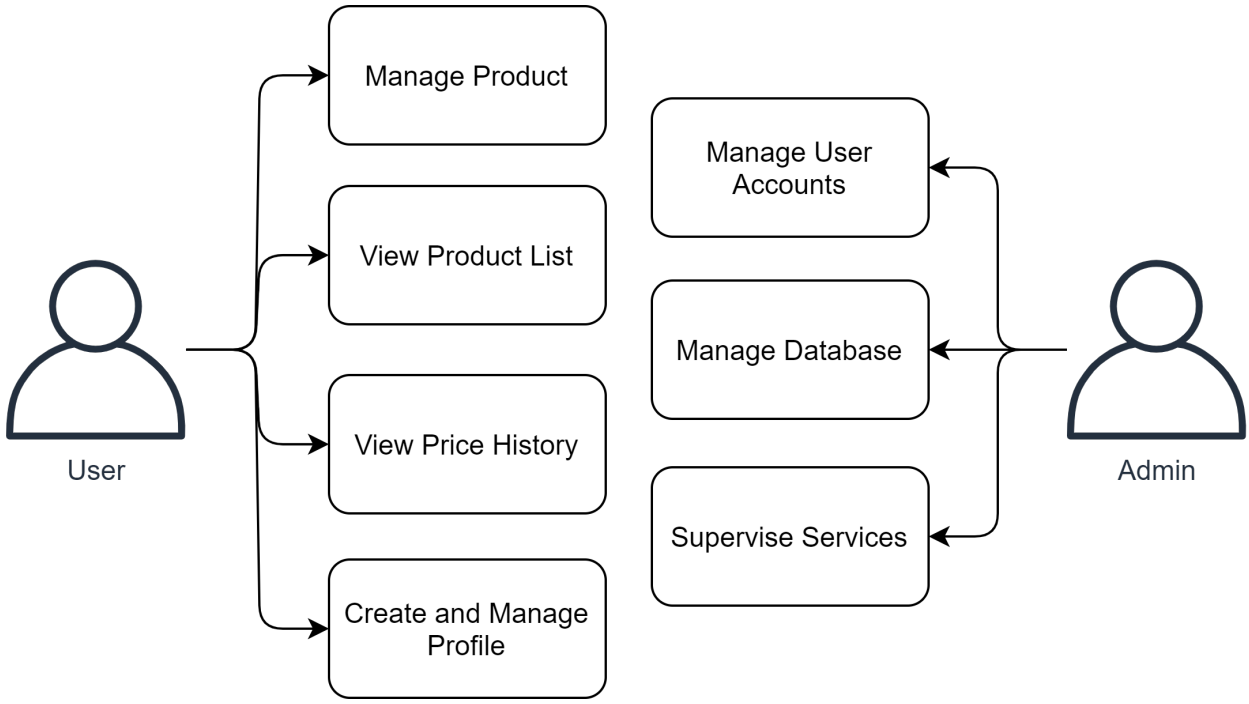
Amennyiben mindhárom kérdésre nemleges a válasz, legálisnak bizonyul adott esetben a Web Scraping. Azonban, ha bármelyik kérdésre pozitív választ adunk nagy valószínűséggel újra kell gondolni és figyelmesebben megvizsgálni az eset legális mivoltát. Ugyanakkor egyik esetben sem szabad megfeledkezni arról, hogy minden ország másképp kezelheti ezen kérdéseket, ezért ilyen értelemben is vizsgálódnunk kell.

4. fejezet

Rendszer specifikációi

A jelen dolgozatban egy árakat figyelő alkalmazás készítéséről van szó, mellyel különböző termékek árát lehet követni, bizonyos webshopokon. Az alkalmazás több platformon is elérhető, hogy minél könnyebben eljusson a kívánt információ a felhasználóig. A következőkben tárgyalva lesznek a felhasználói, valamint rendszer követelmények is.

4.1. Felhasználói Követelmények



4.1. ábra. Use Case diagram

Amint a fenti 4.1 ábra mutatja, a rendszer használata során két alapvető szerepet lehet elkülöníteni, az egyik a tulajdonképpeni felhasználó, aki használja a rendszert, igénybe veszi a szolgáltatást, a másik pedig egy adminisztrátor szerepkört betöltő személy. A következőkben e két szerepet betöltő személy követelményei lesznek bemutatva.

Felhasználó:

- Menedzselni tudja a profilját, vagyis regisztrálni, bejelentkezni tud, illetve, lehetősége van a jelszavának módosítására vagy adott esetben a profil törlésére
- Megtekintheti a termékeket tartalmazó listáját
- Kezelheti a termékeket tartalmazó listáját, vagyis új termékeket adhat hozzá, esetleg törölhet belőle

- Részletes reprezentálást kaphat a követett termékek árának változásáról, a követes pillanatától kezdődően

Adminisztrátor:

- Kezeli a felhasználók fiókjait, az azokkal közbejövő problémákat
- Kezeli az adatbázist
- Felügyeli a rendszerek helyes működését, karbantartja a rendszert

4.2. Rendszer Követelmények

4.2.1. Funkcionális követelmények

A rendszernek mindenek elött, egy bejelentkezési, illetve, regisztrálási felülettel kell rendelkeznie. Regisztrálás után, a felhasználónak egy ellenőrző email-t kell kapnia, amivel igazolja, hogy ő a cím tulajdonosa. A bejelentkezés nem lehetséges, abban az esetben, ha a felhasználó nem igazolta vissza az előbb említett email-ben a címét. A cím igazolása egy linkre való kattintással történik.

A felhasználónak lehetősége van a jelszavának módosítására melyet a bejelentkezési felületről ér el. Miután a felhasználó beírta az email címét, egy levelet fog kapni az adott címre, amelyen keresztül új jelszót tud beállítani.

Bejelentkezést követően, a felhasználó egy felületet lát, melyen bizonyos műveleteket végezhet. Megtekintheti a profiljához tartozó email címét, valamint törölheti a felhasználóját. Ugyanakkor, lehetősége van kijelentkezni az alkalmazásából melynek hatására újra a bejelentkezési oldalra kerül.

Ugyancsak a főoldalról a felhasználónak lehetősége van az alkalmazás használatával kapcsolatos információk megtekintésére mely tartalmaz egy listát is. A lista bizonyos weboldalkát tartalmaz, melyeket kiválasztva, az alkalmazás átirányít az adott elem oldalára.

A felhasználónak lehetősége van termékeket hozzáadni és kitörölni a listájából, valamint görgetni a lista tartalmában. Amennyiben frissíteni szeretné a lista tartalmát, ez úgy lehetséges, hogy a lista tetején tartózkodva, annak tartalmát megpróbálja görgetni (swipe down). A terméklistában egy elemet

kiválasztva, részletes reprezentációt kap az adott elem tárolt adatairól, mint például aktuális ár, annak időbeli változása, hozzáadás időpontja, termék megnevezése.

Amennyiben egy új termék került hozzáadásra vagy a termék törlése következett be, a rendszernek ezt fel kell ismernie és elvégeznie a szükséges lepéseket annak érdekében, hogy a felhasználó számára mindig a legfrissebb adatok legyenek láthatóak.

A rendszer back-end részének, mely a termékek hozzáadásáért és az adatbázis periodikus frissítéséért felelős, autonóm módon kell működni, minimális beavatkozással. Amikor egy felhasználó hozzá szeretne adni egy terméket, a rendszer azonnal reagáljon erre a kérésre. A termékek időszakos ellenőrzése is autonóm módon történjen, a megadott időszakokban.

4.2.2. Nem-Funkcionális követelmények

A rendszer backend része, mely a felhasználok által hozzáadott teremkék árainak ellenőrzését, frissítését, hozzáadását végzi, megszakítás nélkül kell működnie, napszaktól függetlenül. A rendszernek képesnek kell lennie Python 3-as kódot futtatnia, fel kell legyen telepítve a Python 3.8.0. Létfontosságú, hogy egy stabil Internetkapcsolattal rendelkezzen, vagyis a kapcsolatban ne legyenek megszakítások, valamint a fel-letöltési sebesség ne csökkenjen 10 Mb/s alá, annak érdekében, hogy megfelelő sebességgel lehessen az adatfeldolgozást, valamint az adatok adatbázisba való fel-, letolteset elvégezni. Amennyiben a kapcsolat megszakad, a rendszer azonnal próbáljon újrakapcsolódni, mindaddig amig ez a művelet nem sikers.

A rendszer vizuális felülete két alapvető platformon kell elérhető legyen. Az egyik egy böngésző kiegészítő, mely bármilyen Chromium alapú böngészőre telepíthető, asztali, valamint hordozható számítógépek eseteben is, legalább 58-as verziójú Chrome-ot tamogatva. Természetesen ebben az esetben is elengedhetetlen az Internethez való csatlakozás.

A másik egy telefonos alkalmazás, mely Android operációs rendszerrel felszerelt készülékeken legyen elérhető. Az alkalmazásnak legalább 7.0 verziójú Androiddal felszerelt telefonokon kell működnie, mely rendelkezik stabil Internetkapcsolattal. Rendelkeznie kell Light illetve Dark móddal is, melyek közötti váltást automatikusan végzi, az operációs rendszer beállításaihoz igazodva. A felhasználó számára, a követett termékeit egy görgethető listában kell megjeleníteni. Egy listaelem

tartalmazza a termék megnevezését, aktuális árát, illetve egy képet róla. A termék ára mellett egy nyíl található, mely azt jelzi a felhasználó számára, hogy az aktuális ár hogyan változott a korábbi ellenőrzéshez képest. A termék árának csökkenését, egy lefele irányuló, a növekedését egy felfele irányuló, amennyiben pedig az nem változott, egy vízszintesen irányított nyíl jelzi. Ezeket a változásokat színekkel is kell jelezni, mely az árra és az előbb említett nyílra hat ki. Csökkenés esetén zöld, növekvés esetén piros, valamint, ha változatlan, akkor fehér vagy fekete színnel kell jelölni ezeket.

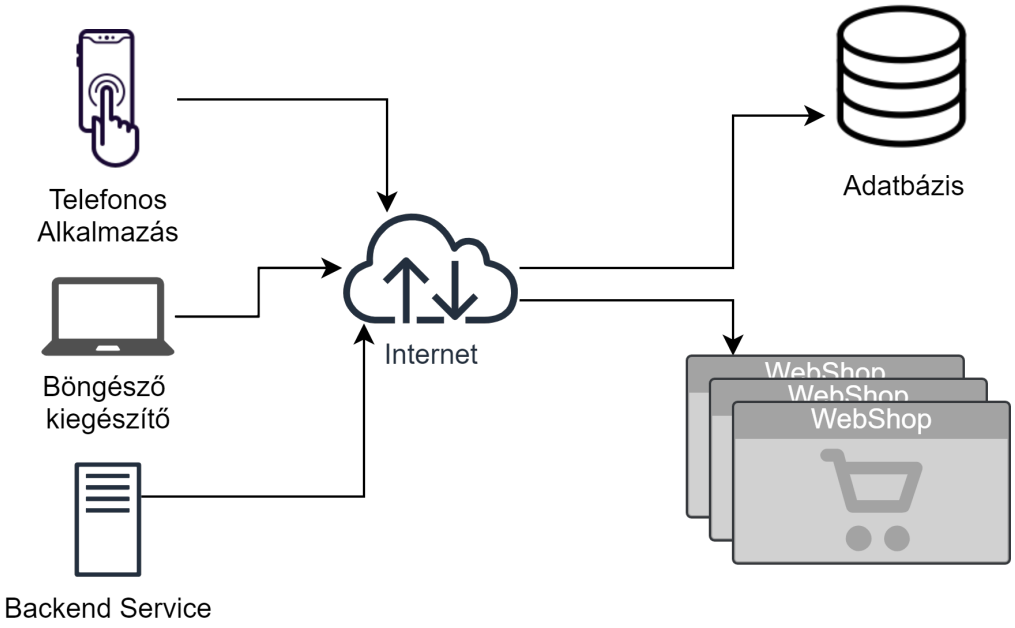
A rendszer a Firebase által biztosított Realtime Database nevű adatbázist kell használnia az adatok tárolására. A felhasználó bejelentkezését és a fiókjához tartozó műveleteket szinten a Firebase által biztosított Authentication szolgáltatás végezze, mivel ez biztonságos módon tárolja a szükséges információkat, ugyanakkor, a jelszavakat titkosítva kezeli. Továbbá, ezen keresztül lehessen új jelszót beállítani, a felhasználót törölni, ugyanakkor a regisztrálás után szükséges email visszaigazolása is ezen a szolgáltatáson keresztül történjen, mivel ezekre egyszerűen használható, ugyanakkor hatékony és biztonságos megoldást nyújt.

Az alkalmazás könnyen használható kell legyen, a felhasználónak minden funkciót három kattintáson belül el kell érnie.

Amikor a felhasználó egy új terméket szeretne hozzáadni a listájához, ne keljen több mint 5 másodpercet várnia ahhoz, hogy az új termék megjelenjen a listájában.

5. fejezet

A Rendszer Architektúrája



5.1. ábra. A rendszer architektúrája

A rendszer alapvető részét képezi az adatbázis, amely az adatok tárolását, illetve az azokat elérő API-t szolgáltatja. Ehhez az adatbázishoz két alapvető típusú készülék csatlakozik. A felhasználó oldali, amely lehet akár böngésző kiegészítő vagy telefonos alkalmazás, illetve a szerver vagy logika oldali, amely az adatok feldolgozását biztosító logikát és erőforrást tartalmazza, utóbbi a 5.1 ábrán Backend Service-ként van jelölve.

A böngésző kiegészítő szükséges, mivel sokkal gyorsabban el lehet érni a megtekinteni kívánt adatokat, valamint sokkal egyszerűbbé teszi a termékek hozzáadását azzal, hogy a követni kívánt termék oldalát egyáltalán nem kell elhagyni. Továbbá fontos, hogy az alkalmazás tudja azt, hogy a felhasználó épp milyen oldalon tartózkodik, ahhoz, hogy a megfelelő URL-t kapja meg, mindezt a kiegészítő könnyen és megbízhatóan el tudja végezni.

Mivel azonban nem mindig vagyunk laptop vagy asztali gép közelben, ezért lehetőséget nyújtunk arra, hogy telefonos applikáció segítségével is el lehessen érni a követett termékeket, valamint minden ehhez tartozó műveletet, akárcsak az előbb említett kiegészítő esetén. Egy termék hozzáadása ezúttal az operációs rendszer megosztási menüjén keresztül történik, ami által az alkalmazás megkapja a követni kívánt termék elérhetőségét.

Ahhoz, hogy az eszközök kommunikálni tudjanak egymással, egyértelmű, hogy szükség van valamiféle összeköttetésre, amely a mi esetünkben az internet lesz. Ez a funkció létfontosságú, mivel minden adatot fel, illetve le kell tölteni az adatbázisból, függetlenül az eszközök tartózkodási helyétől, ugyanakkor a szolgáltatást biztosító rendszer is ezen keresztül éri el a termékek weboldalát.

Az architektúra fontos részét képezik a webshop-ok is, mivel ezeket úgy a termék hozzáadásakor, mint azoknak periodikus ellenőrzésekor el kell érni, az adatok begyűjtése érdekében. A támogatott webshop-ok főként népszerűségüket tekintve lettek kiválasztva, mint például az Emag, viszont ezen kívül két más webáruház is támogatott, ezek a Flanco és QuickMobile. Mivel minden weboldal másképp épül fel, mindegyik oldal struktúrájából másképp kell kinyerni az adatokat, ezért ezt a folyamatot személyre szabottan kell végezni. Ugyanakkor változhatnak is idővel ezek a struktúrák, ezeket figyelni kell.

5.1. A modulok megvalósítása

5.1.1. Chrome Extension

A böngésző kiegészítő vagy angolul browser extension, egy a böngésző környezetén belül futó alkalmazás, ami olyan funkciókat hivatott hozzáadni a felhasználói felülethez, melyek megkönnyítik vagy jobbá teszik a felhasználóii élményt. Előnye, hogy alkalmazások ezrei állnak a felhasználók

rendelkezésére, melyeket par kattintással telepíthetnek is. Ezek már szinte minden böngészőn megtalálhatók valamilyen formában, a legnépszerűbbek esetében, mint például, Google Chrome, Firefox ez a funkció régóta jelen van.

Ezen kiegészítők működése valószínűleg sokak számára ismert, amolyan lenyíló ablakként jelennek meg a böngészőben, anélkül, hogy hatással lennének az éppen megjelenített tartalomra. Ennek tudatában, ez a megközelítés tűnt a legmegfelelőbbnek a dolgozatban tárgyalt szoftver felhasználói felületének elkészítése során. Mivel ezek a funkciók csak asztali gépeken érhetőek el, ezért szükséges volt egy telefonos interface kifejlesztése is, mely a későbbiekben kerül bemutatásra.

A kiegészítő megvalósítása során, JavaScript, HTML, CSS programozási nyelvek voltak felhasználva. Mivel korábban nem volt tapasztalatom böngészőhöz való kiegészítők fejlesztésében, ezért az implementálási folyamat információ gyűjtéssel kezdődött. Utánanéztem hogyan is zajlik egy ilyen kiegészítő fejlesztése, mik a követendő lepések illetve fázisok.

Első lépésben szükséges egy manifest.json file létrehozása, mely a kiegészítő alapvető információit tartalmazza, mint például verziószám, név, rövid leírás, felhasznált függőségek, engedélyezett műveletek stb... . Ezek után, a fejlesztés hasonló egy hagyományos weboldal elkészítéséhez.

A fejlesztés során több könyvtár került felhasználásra, különböző funkciók ellátására, ezek a Bootstrap[1], sweetalert2[2], firebase[3], amcharts[4]. A bootstrap egy ingyenes, nyílt forráskádul CSS framework, mely segítségével interaktívabbá, szebbé tehetjük a weboldalunkat, előre definiált mintákat biztosit nekünk, gombok, navigáció vagy más komponensek esetében. A sweetalert2 egy úgynevezett riasztásokért felelős könyvtár, olyan esetekben került használatra, amikor egy felugró ablak segítségével szeretnénk megerősítést kérni a felhasználótól egy bizonyos művelet elvégzésére, mint például termékek törlése vagy kijelentkezés során, viszont egyéb, információ közlési célokra is felhasználva lett, mint például egy művelet sikeres elvégzésének visszajelzése. A firebase könyvtárakat bejelentkezési, valamint adatbázis kezelő funkciók miatt volt szükséges használni. Az árak időbeli változását ábrázoló diagramok eseteben több könyvtárat is kipróbáltam, viszont a végső választás az amcharts nevűre eset, mivel ez volt a leg megfelelőbb az adott esetben.

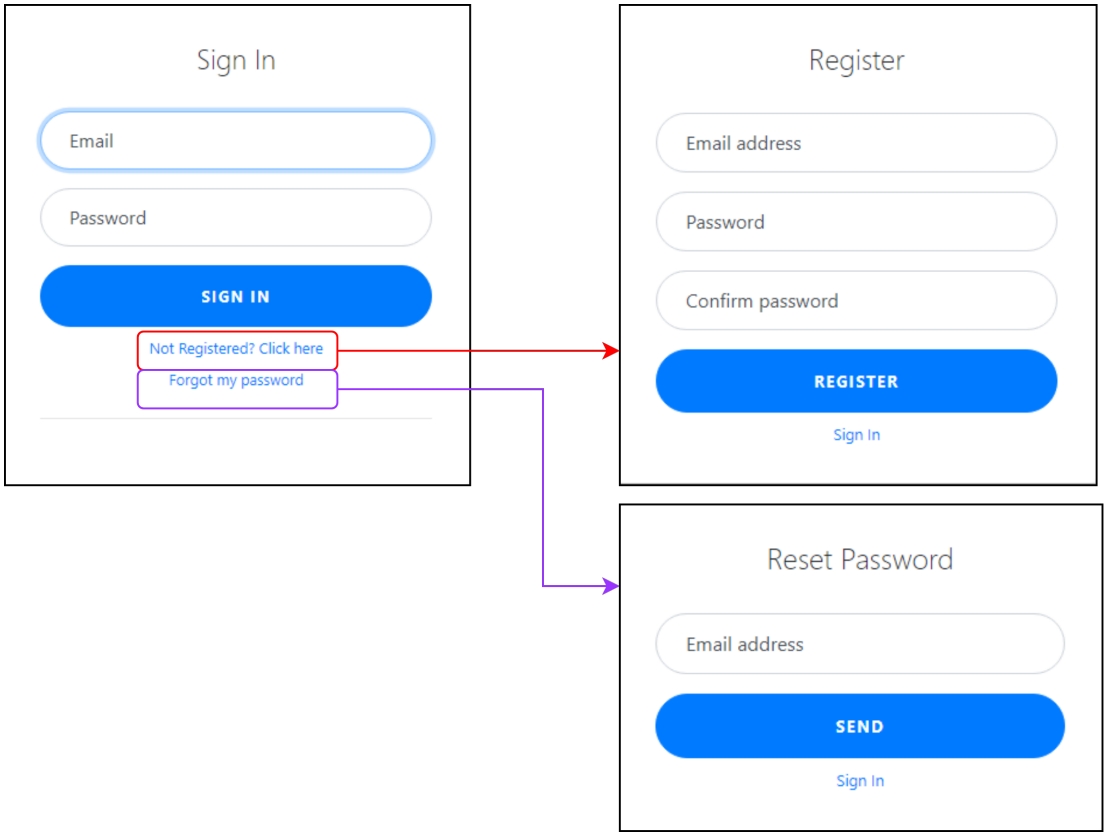
[1] https://getbootstrap.com/

[2] https://sweetalert2.github.io/

[3] https://firebase.google.com/

[4] https://www.amcharts.com/

Amikor a felhasználó először használja a kiegészítőt, egy login oldal jelenik meg számára, melyen be tud jelentkezni, valamint regisztrálni tud 5.2.

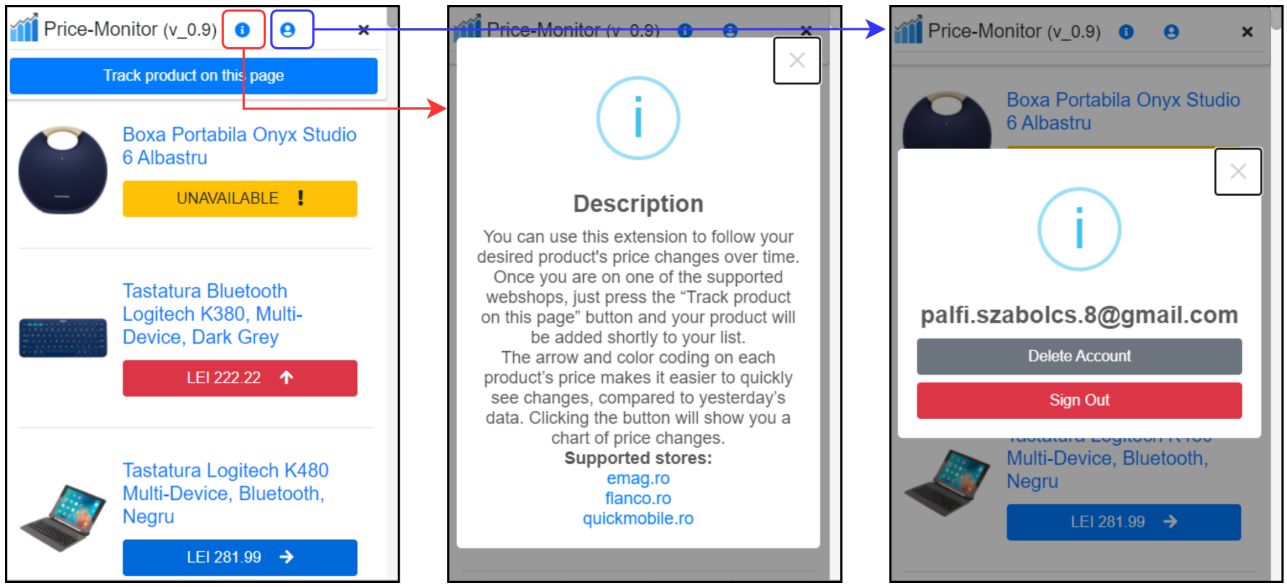


5.2. ábra. Kiegeszito bejelentkezési/regisztrálási felülete

Bejelentkezés után a felhasználó a főoldalon találja magát, ahol megtekintheti a követett termékeit, ugyanakkor az alkalmazás használatával kapcsolatos információkat is megtalálja, a fejlécben található "i" szimbólummal jelölt gombra kattintva, valamint ugyanitt vannak felsorolva a támogatott oldalak, melyekre kattintva, az adott weboldalra navigálhatunk. A jelenleg támogatott webshopok az Emag[5], Flanco[6] és QuickMobile[7]. Továbbá, szintén a fejlécben található a felhasználó fiókjához tartozó információkat elérő gomb, melynek hatására egy felrúgó ablakban tekintheti meg az email címet, amivel bejelentkezett, illetve itt tud adminisztrációs műveleteket is végezni. Az előbb tárgyaltokat 5.3 ábrán láthatjuk.

[5] https://www.emag.ro/
[6] https://www.flanco.ro/
[7] https://www.quickmobile.ro/

Új termék hozzáadásakor, az ablak tetején található „Track product on this page” gombra kattintva (5.3 ábra) tehetjük ezt meg, melyek után pillanatokon belül látható majd a listában az adott termék. Amennyiben nem az alkalmazás által nem támogatott oldalon tartózkodunk, ez az opció nem elérhető, a gomb egyszerűen nem jelenik meg.

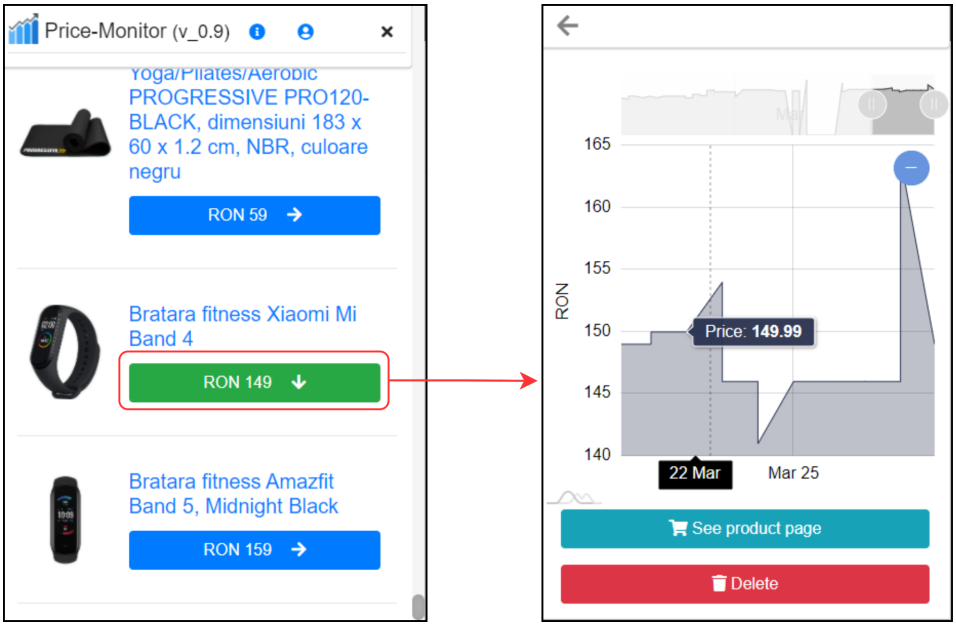


5.3. ábra. Főoldal, adminisztrációs rész

Amint a 5.3 ábrán is láthatjuk, a termékek gombjai tartalmazzák a termék aktuális árát, tőlük jobbra egy nyilat, valamint ezek különböző színűek. A nyilak, illetve, színkódolás azért van, hogy a felhasználónak visszajelzést adjuk arról, hogy a termék drágult, olcsóbb lett, nem változott az ára, esetleg jelenleg nem elkéretó. Amennyiben a termék drágul, ezt egy felfele mutató nyíl és piros szín jelzi, csökkenés esetén a nyíl lefele mutat és a gomb színe zöldre vált, az utóbbi az 5.4 ábrán látható. Amikor egy termék ára nem változik, ezt egy vízszintesen irányuló nyíl és a kék szín jelzi, valamint, ha a termék nem elérhető, akkor ezt sárga szín és az „UNAVAILABLE” szöveg mutatja.

Ha egy termék gombjára kattintunk, akkor megnézhetjük az adott termék árának változását egy diagramon, a követés pillanatától az aktuális dátumig. Amint a 5.4 ábra mutatja, a diagramon szépen látható az arák változása, ami sok esetben naponta akár többször is változhat. Mivel az adatmennyiség idővel nagyon nagy lehet, ezért a diagramot mozgatni lehet, hogy egy adott intervallumot vizsgáljunk, vagy az időskálát növelhetjük, illetve csökkenthetjük, igénynek megfelelően, a diagram tetején talál-

ható csúszka segítségével. Láthatjuk továbbá azt is, hogy ha az egeret a diagram egy adott pontján tartjuk, akkor az levetíti nekünk az adott dátumon a termék árát. Az ablak alján található a törlés gomb, ezzel törölhetjük a terméket a listánkból, ha esetleg már nem vagyunk érdekeltek az adott árucikk követesében. Az előbb említett törlés gomb fölött helyezkedik el még egy gomb, mely megnyomásával az adott termék weboldala nyílik meg számunkra a böngésző egy új ablakában.



5.4. ábra. Termék árváltozása

6. fejezet

Összefoglalás

6.1. Összefoglalás

Irodalomjegyzék

[1] F. Johnson and S. K. Gupta, „Web content mining techniques: a survey," *International Journal of Computer Applications*, vol. 47, no. 11, 2012.

[2] F. Gorunescu, *Data Mining: Concepts, models and techniques*, vol. 12. Springer Science & Business Media, 2011.

[3] B. G. Dastidar, D. Banerjee, and S. Sengupta, „An intelligent survey of personalized information retrieval using web scraper," *International Journal of Education and Management Engineering*, vol. 6, no. 5, pp. 24–31, 2016.

[4] R. N. Landers, R. C. Brusso, K. J. Cavanaugh, and A. B. Collmus, „A primer on theory-driven web scraping: Automatic extraction of big data from the internet for use in psychological research.," *Psychological methods*, vol. 21, no. 4, p. 475, 2016.

[5] D. S. Sirisuriya *et al.*, „A comparative study on web scraping," 2015.

[6] V. Krotov and L. Silva, „Legality and ethics of web scraping," 2018.

[7] D. Ni, „Is web scraping legal?."

A. függelék

Függelék