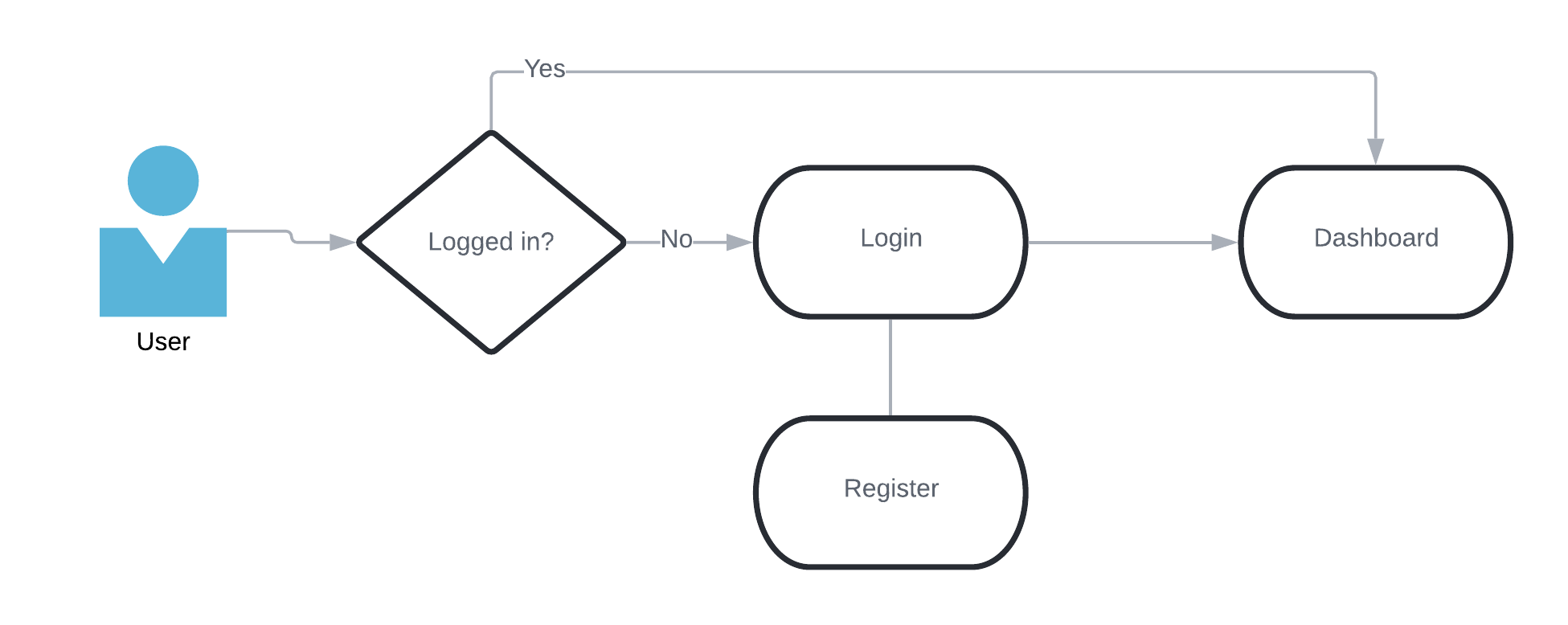
# **3. Felhasználói Követelmények**

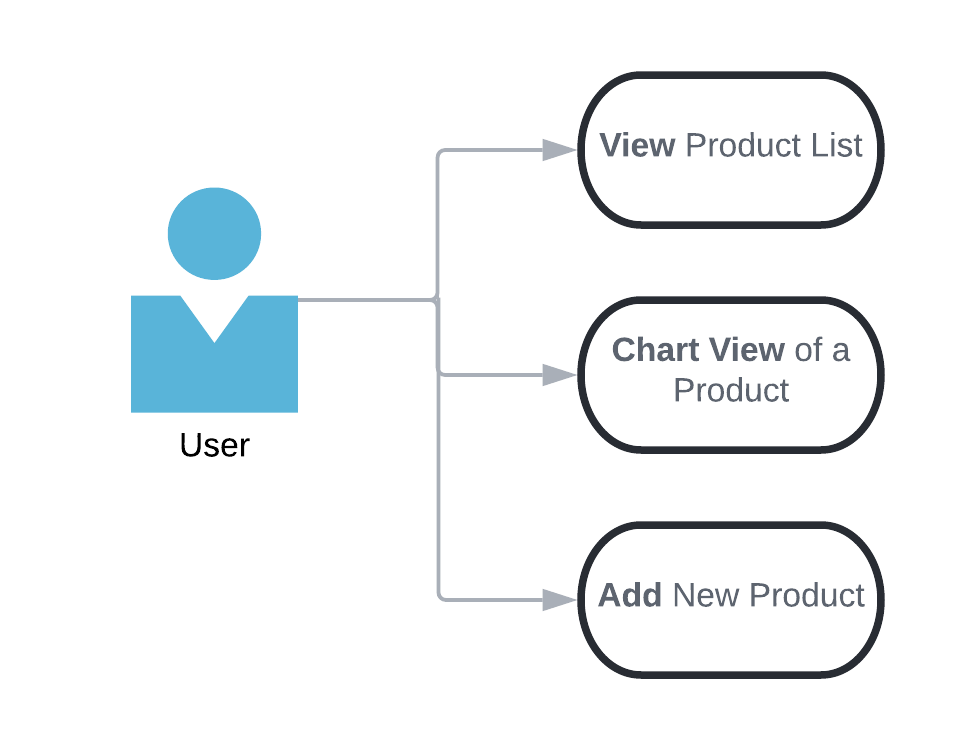
A felhasználónak mindenekelőtt be kell tudnia jelentkezni, ahhoz, hogy elérje a terméklistáját, ez azért fontos, mivel eszköz váltás esetén nem szeretnénk elveszíteni az addig követett termékeket. A bejelentkezéshez szükséges adatok a regisztrációkor megadott email cím, illetve jelszó. Mindezt az 1.ábra illusztrálja.



( 1.ábra )

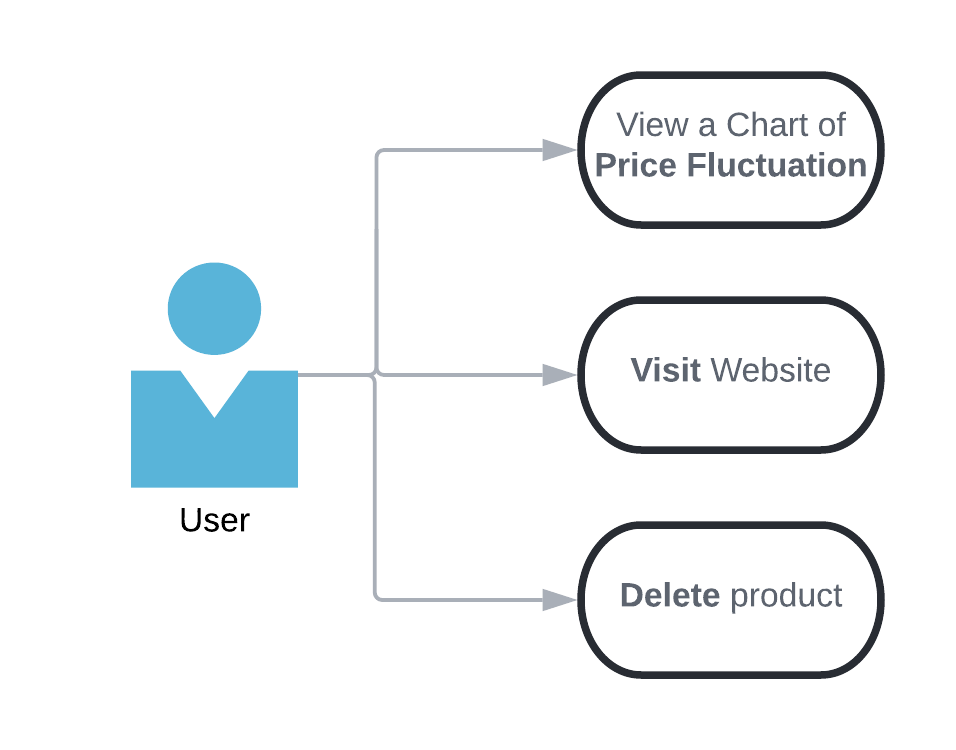
Abban az esetben, ha a felhasználó nincs regisztrálva, megteheti ezt a „ Not registered? Click here! ” szövegre kattintva. A regisztrációhoz szükséges egy érvényes e-mail cím, valamint jelszó megadása. Sikeres regisztrálás esetén a felhasználó vissza kerül a bejelentkező ablakra, ahol be tud lépni, azzal a feltétellel, hogy az automatikusan küldött levél által visszaigazolta email címét.

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 )



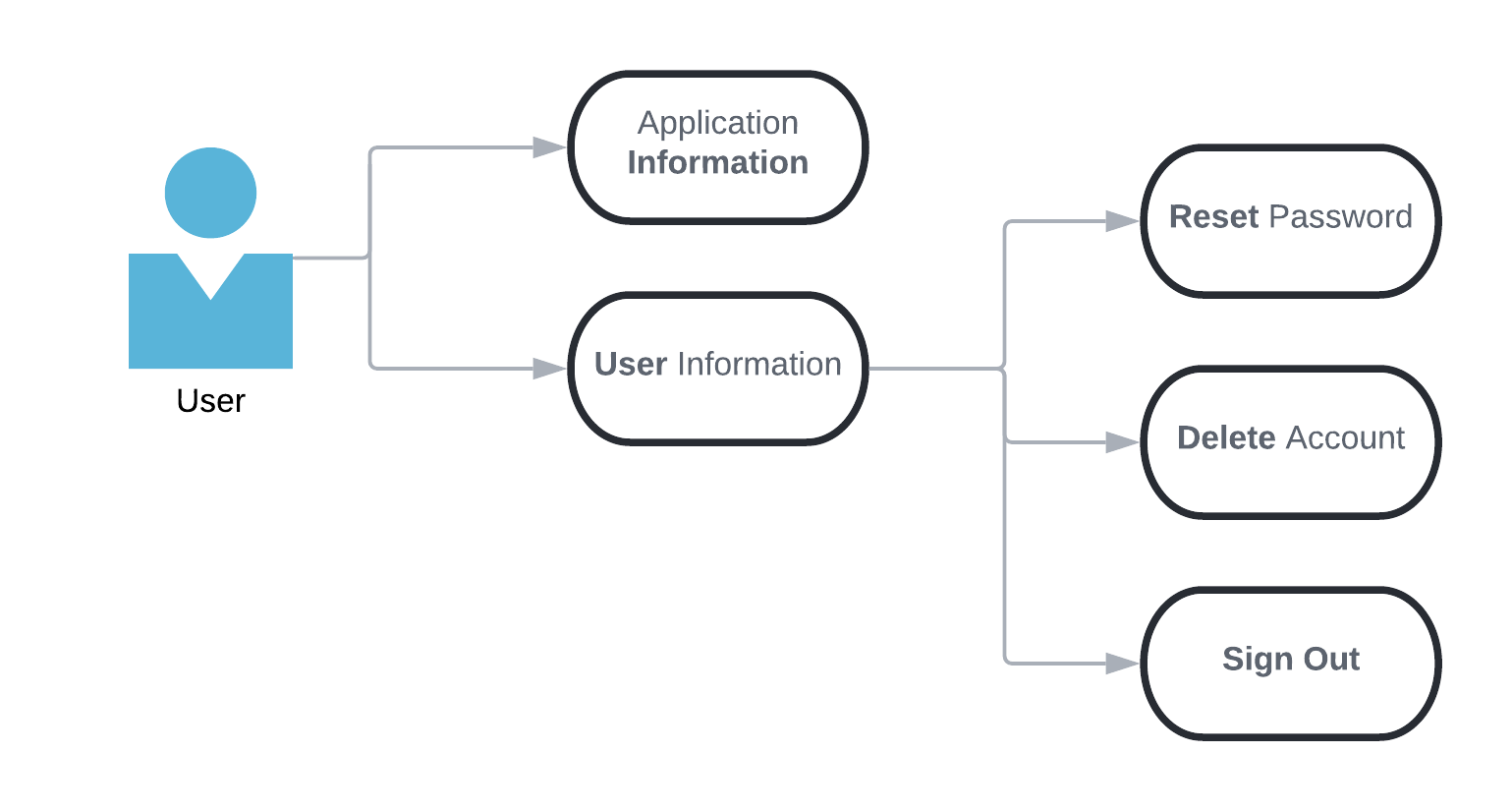
( 2.ábra )

Egy a listában lévő elemre kattintva, az alkalmazás átvisz egy másik oldalra, ahol több információt kapunk a követett termékről. A termék árát tartalmazó gombra kattintva, egy grafikonon tekinthetjük meg a termék árának változását a hozzáadás napjától az aktuális dátumig. Az „ See product page ” gombra kattintva az alkalmazás megnyitja a terméket tartalmazó weboldalt. Ugyanitt található a törlés gomb, melyre kattintva a termek törlésre kerül a listából és nem fogjuk tovább követni ( 3.ábra )



( 3.ábra )

A felhasználó rendelkezésére áll továbbá két menü. Egy információs, amely röviden leírja az alkalmazás használatát, illetve tartalmazza az általa támogatott weboldalak listáját. A másik a felhasználó fiókjával kapcsolatos információkat és funkciókat tartalmaz. Itt tekintheti meg a felhasználó, hogy milyen email címmel jelentkezett be, megváltoztathatja az aktuális jelszavát, törölheti a fiókját, illetve kijelentkezhet az alkalmazásból. ( 4.ábra )



( 4.ábra )

# **4. A rendszer leírása**

## 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.

A Python kód futtatásához, szükséges néhány, a beépített függvény könyvtárat kiegészítő, specifikus könyvtár. Ezek nélkül a program nem fog helyesen lefutni, ezért a következő elemeket kell hozzáadni: bs4, dacite, firebase, gcloud, requests, schedule, time. A requests könyvtár segítségével tudunk HTTP kéréseket intézni, amely ahhoz szükséges, hogy megkapjuk egy adott termék linkjéhez tartozó HTML struktúrát. A visszakapott nyers kódban nehezen lehetne kereséseket végrehajtani, ezért ezt parse-oljuk a bs4 által megvalósított BeautifulSoup nevű függvénnyel, amely egy kereshető struktúrát hoz létre több hasznos funkcióval, a neki megadott HTML kódból. Mikor mindez megvan és feldolgoztuk a szükséges adatokat, a firebase illetve gcloud könyvtárak segítségével tudjuk elérni a használt adatbázist, melyre feltöltjük az adatokat a megfelelő formában. Adatok lekérésé során szükséges a dacite könyvtár által megvalósított dictionary parser, amely az adatbázis által visszatérített dict struktúrából, egy általunk megvalósított osztályt felhasználva hoz létre, egy, az osztályhoz tartozó példányt, mellyel ezek után dolgozni tudunk. A schedule illetve time könyvtárak az árak illetve termékek időszakos ellenőrzésére szükségesek.

Az adatfeldolgozás megkönnyítésé érdekében, több saját osztály is szükséges volt létrehozni. Ezek közül fontosabbak a ProductData, amely egy termekről tárolt információkat tartalmazza, NewData, amely egy új link hozzáadásakor használandó, illetve UtilProduct, mely a fentebb említett parsolaskor fontos.

## 4.2 Frontend

## 4.2.1 Böngésző kiegészítő

A kiegészítő használatához szükséges egy kompatibilis böngésző, ami chromium alapú. Egy rövid regisztrálás majd bejelentkezés után használható is az alkalmazás. A felhasználó információi, legyenek azok a fiókjához vagy a termék listájához tartozóak, az adatbázisban tárolódnak el, ezért létfontosságú, hogy a megfelelő internetkapcsolat álljon rendelkezésünkre.

A felhasználói felület elkészítéséhez is felhasználásra került pár külső könyvtár: firebase-app, firebase-auth, sweetalert2, fontawesome, bootstrap. Adatbázis kapcsolatra a firebase-app könyvtárat kellett használni, amelynek egyszerűen az adatbázis elérhetőségét kellett megadni, onnantól a létrejött példányon keresztül lehetett végezni a műveleteket. A firebase-auth könyvtár tette lehetővé, hogy gyorsan és egyszerűen lehessen megoldani a regisztrálást, illetve bejelentkezést. Regisztrálást követően a felhasználó egy megerősítő email-t kap, ahol a visszaigazolás után, aktívvá válik a fiókja és be tud jelentkezni. Fontos, hogy amíg a megerősítés nem történt meg, addig a felhasználónak nincs lehetőségé bejelentkezni. A felület vizuális felülete bootstrap, fontawesome, valamint sweetalert2 elemek segítségével került megvalósításra, ahol a bootstrap a fő struktúrákért, míg a sweetalert, a beugró ablakokért felelős. A fontawesome az ikonok vagy betűtípusok formázásában kapott szerepet.

## 4.2.2 Telefonos alkalmazás

## 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.