

# THE SNAIL SALE



## RENDSZERTERV

### 1. A rendszer célja

A The Snail Sale egy olyan desktop applikáció, amivel a felhasználók könnyen, és gyorsan tudnak vásárolni.

A rendszer célja egy olyan applikáció létrehozása, ami egy online adatbázishoz csatlakozva, szofisztikált keresőrendszert alkalmazva segíti a felhasználóit, a megfelelő termékek megvásárlásában. A kívánt termék megkeresése után, a termékről a program megjeleníti a különböző, az adatbázisban tárolt adatokat a felhasználó számára. Ezeket az adatokat a termékről az ablak közepén elhelyezett, információs részlegben tekintheti meg a felhasználó.

A kiválasztott, megvételre szánt termékeket a kosárhoz adhatja user, amit bármikor megtekinthet. Miután kiválasztotta a megvásárolandó terméket, a rendszer összegyűjti és tematikusan bemutatja, hogy mely termékeket választotta ki a felhasználó. Ekkor még lehetősége van a felhasználónak, arra, hogy az esetlegesen nem kívánt termékeket kivegye a kosarából a megfelelő gombra kattintva.

Ezután a program folytatva a vásárlás folyamatát, megkéri a felhasználót, hogy írja be a vásárláshoz szükséges személyes adatokat. A megfelelő adatok beírása után, a felhasználót, még egy, utolsó adategyeztetés választja el a sikeres vásárlástól. Miután meggyőződött arról, hogy az adatait helyesen írta be, és a megfelelő termékeket választotta ki, konfirmálja a vásárlás helyességét. Ezután a program generál egy úgynevezett "tracking number" amivel a vásárlást tudja követni, illetve a megfelelő adatokat rögzíti az adatbázis megfelelő táblájába.

A vásárlás befejezése után a program a kereső oldalra irányítja a felhasználót, további vásárlásra adva lehetőséget.

A rendszernek nem célja, a felhasználók által eladni kívánt termékek felvétele az adatbázisba, így arra nem is kínál lehetőséget a program. Nem célja a termékek szöveges értékelése se, így csak egy 1-től 5-ig, egész számokat tartalmazó skálán értékelhetik a regisztrált felhasználók a termékeket.

## 2. Üzleti folyamatok modellje

A szoftver abból a célból készül, hogy a felhasználók (vásárlók) könnyen elérhessék a számukra szükséges termékeket, a lehető leggyorsabb és legkényelmesebb módon. Maga az üzleti folyamat része a szoftver használata. Ehhez a szükséges szereplők a felhasználók, és az adminisztrátorok, valamint egy megfelelően működő és kezelhető adatbázis, a felhasználók fiókjaival, valamint a termékek listájával.

A programot a felhasználó eléri, majd elindítja, ezzel indul a folyamat. Ha a felhasználó először akarja használni a szoftvert, akkor regisztrálnia kell. A megadott adatok az adatbázisban tárolódnak, ezért a továbbiakban elegendő bejelentkeznie a korábban megadott adatokkal.

A bejelentkezés egy adatbázisban való keresés segítségével végzi el a program. A megfelelő felhasználó és jelszó kombinációjának beírásával ez a folyamat sikeres, így eléri a további folyamatokat, amik a vásárláshoz szükségesek. A másik üzleti szereplő az adminisztrátor.

Ahhoz, hogy a vásárló böngészhessen a termékek között, a programot adatokkal kell feltölteni, amit a felhasználó olvashat és láthat. Az adatbázis feltöltése és módosítása az adminisztrátor dolga. Mivel a termékek száma és elérhetősége változó, így ezt fontos karbantartani. Ezután a felhasználó megtekintheti a feltöltött termékeket, kiválaszthatja, hogy számára melyik szükséges, majd megvásárolhatja. A termék kiválasztása után a felhasználó kitölt egy szállítással és fizetéssel kapcsolatos oldalt, majd a vásárlásra kattint. Az igény pedig az adminisztrátorhoz érkezik be, ami a felhasználó által kitöltött adatokkal és a

kiválasztott termékekkel együtt érkeznek be. Az entitás maga a SnailSale vásárlásra kitalált program, a benne felsorakoztatott termékekkel.

Összefoglaló:

Üzleti szereplők: felhasználó, adminisztrátor, (adatbázis)

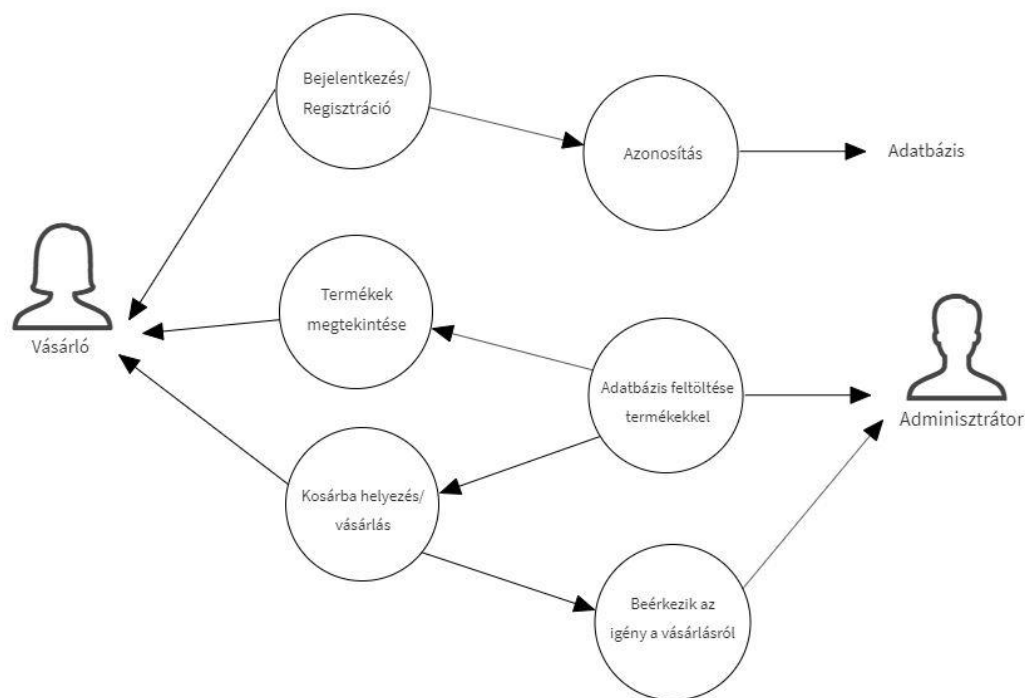
Üzleti folyamatok: bejelentkezés, regisztráció -> azonosítás

adatbázis feltöltése -> termékek böngészése, vásárlás

vásárlás -> beérkezik az igény a vásárlásról

Üzleti entitás: SnailSale és annak termékei

Összefoglaló képformájában:



### 3. Követelmények

Funkcionális követelményekhez tartozik az egyszerűen használható, felhasználóbarát felület, ahol az elemek érthetően vannak megfogalmazva és egyértelmű az elemek funkciója. A program gyorsan letölthető, könnyen elérhető és internet kapcsolat mellett jól működik. A folyamat és a működés célja az, hogy a felhasználó hozzájuthasson a kívánt termékhez. A termékek listája és az adatbázis folyamatosan frissülő és mindig aktuális. Az alkalmazás mindig elérhető legyen (a karbantartás időszaka képezhet kivételt), valamint a használata gördülékeny és gyors.

A regisztrációval mindig tudjon a felhasználó profilt létrehozni, de ne lehessen két egyforma felhasználó nevű vásárló. A bejelentkezés helyes megadott adatok esetén legyen sikeres. A szoftver felismerje, ha még nem regisztrált felhasználó próbálja használni és bejelentkezni. A felhasználó nevet és jelszót megfelelően és törvényeknek megfelelően kell kezelni. Ezt a felhasználók és az ott dolgozók nem tekinthetik meg adatvédelmi okokból.

Bejelentkezett felhasználó esetén a termékek kosárba tehetőek és megvásárolhatóak. A személyes és szállításhoz szükséges információkkal kitöltött oldal továbbításra kerül a kosárba rakott termékekkel együtt. Így az adminisztrátor segítségével továbbításra kerül mik a szállítandó termékek és hova kell szállítani, valamint komplikáció esetén hol lehet értesíteni a felhasználót.

Nem funkcionális követelmények szempontjából a középpontban az egyszerűen kezelhető felület áll, hogy a felhasználók előszeretettel használhassák a szoftvert. A gombok egyértelműen használhatóak, és elérhetőek az elhelyezésük szempontjából is.

Az elrendezés is segíti a vásárlót a gördülékeny használathoz. A szövegek és a különböző karakterek jól olvashatóak, eltérnek a háttér színétől, és egyszerűen észrevehetőek. Az alkalmazásban nem megengedett a sértő tartalom megjelenítése, szabadon kitölthető szöveges mezőket a felhasználóknak is a kért tartalomnak megfelelően kell kitölteni. Nem megfelelő használat esetén az adminisztrátoroknak, illetve a készítőknak lehetőségük van az adott felhasználót eltávolítani az adatbázisból.

#### 4. Funkcionális terv

A feladat egy számítógépen futó The Snail Sale nevű program megtervezése.

A program kérje le egy "csiga" nevű adatbázisból a keresési és kiválasztási feltételeknek megfelelő terméket, mikor a felhasználó éppen a szoftvert használja. Továbbá a feladatai között van az értékelés, pl. egy felhasználó megrendeli a számára megfelelő termékeket, és kiszállítás után van lehetősége értékelni a csomagot. 1-től 5-ig tudnak értékelni. Alapvető funkciói közé tartozik a keresés különböző szempontok szerint (ár szerint, növekvő-csökkenő sorrendben stb.)

Egy vásárlás ezzel a szoftverrel : A felhasználó bejelentkezés vagy regisztráció után megkeresi a számára fontos termékeket, elolvashatja az adatait majd berakhatja a kosárba.

A korárba maximum 3 terméket rakhat, ennyit lehet összesen egyszerre vásárolni. A kosárban is megtekintheti a megrendelni kívánt termékeket az adataikkal együtt. Többszörös megerősítés és adategyeztetés után lehet megrendelni, majd kiszállítás után véleményezni és értékelni. A rendszert három szereplőre lehet felbontani, a megrendelőre, felhasználókra és a programozókra.

A fejlesztők megtervezik és implementálják a megrendelő igényeit. Majd a továbbiakban a felhasználók igényei és észrevételei szerint fogják a programozók fejleszteni a felületet. A felület egyszerű , mégis több egyszerű menü és gomb van a programban.

## 5. Fizikai környezet

Az applikációt a Java programozási nyelvnek a használatával fogjuk elkészíteni. A 11-es verziót használva, de előre láthatólag 100%-ban csak a 8-as verzió funkcióit felhasználva, azaz teljes mértékben kompatibilis lesz a 8-as verzióval.

A Java programozási nyelvnek köszönhetően, így a program, használható lesz mind Windows, illetve Mac gépeken is.

A programnak szüksége lesz internet kapcsolatra, az online adatbázis használatából kifolyólag. Az internetkapcsolat sebességének minimálisan el kell érnie az 1Mb/s sebességet, élvezhető felhasználás érdekében.

A java nyelv sajátosságaiból kifolyólag, és a gördülékeny felhasználás érdekében, az éppen szóban forgó operációs rendszer, valamint a fejlesztőkörnyezet, és egyéb segédprogramokhoz (ld. később) szükséges minimális rendszerkövetelményein felül ajánlatos egy 2 magos processzor, és legalább 1Gb RAM használata.

Fejlesztői környezetnek az IntelliJ IDE-t fogjuk használni, a minél könnyebb fejlesztés érdekében.

Az online adatbázis kezeléséhez pedig phpMyadmint.

Az adatbázis szerverhez például wamp vagy xamp szerver is használható.

## 6. Absztrakt domain modell

Minden oldalról közvetetten vagy közvetlenül vissza lehet navigálni a főoldalra.

Az alkalmazáson belül az oldalak és panelek megfelelően összekapcsolódnak, és gördülékenyen átvált egyik oldalról a másikra. Ezek főként gombok segítségével



valósulnak meg. Például a bejelentkezés oldal a főoldaltól elérhető egy be/kijelentkezés gomb segítségével, erről az oldalról pedig tovább mozoghatunk a regisztrációs panelra.

A szoftver adatbázisból dolgozik, innen jeleníteni meg a termékeket és a hozzá tartozó adatokat. Majd a kiválasztott termékeket átvihetjük a kosár oldalára. Ezután a véglegesítés során a személyes infókat is fel kell vinnünk, és ezeket az adatokat a termékekkel együtt kell eljuttatnunk az ezzel foglalkozó munkatárshoz, aki felveszi a rendeléseket. Ez adatbázis segítségével fog történni, így az adminisztrátorként dolgozó munkatárs is könnyedén áttekintheti melyik termékből és kinek kell kiküldeni.

## 7. Architektúrális terv

Az alkalmazás alapvetően egy főoldalból, bejelentkező, regisztrációs oldalból, a kosár, a kosár tartalmának véglegesítése, a vásárlás véglegesítése és az ezekhez szükséges személyes infók panelből áll. A szoftver struktúráját tekintve nem funkcionális elemek módosítására folyamatosan van lehetőség. A vágyálom rendszerben korábban felvetett ötletek megvalósítása a továbbiakban van mód és lehetőség, ha az igényt megkapjuk.

A változtatások, különböző módosítások a felületen várhatóak. A változtatásokon kívül a program bővíthető elemekkel. A képernyőtervek alapján készülne el a program.

A főoldal a tervek szerint balról jobbra egy keresősávból, és egy sorrendet módosító részből áll, amit egy keresés gombbal tuunk működtetni. Az olddal középső részén találhatóak a termékek az adataikkal, felette a rendezés feltétele, és mellette a kosár gomb, amivel a kosár oldalára navigálhat a felhasználó. A képernyő közepén, ahol az adott termék látható, megjelenik egy kosárba gomb is, amivel kosárba lehet helyezni az éppen megtekintett terméket. A termékeket az előző és a következő gombokkal lapozhatjuk.

A főoldal jobb oldalán egy be/kijelentkezés gomb található, alatta a termék értékelése rész, és egy vásárláshoz gomb, ami a vásárláshoz irányít tovább.

A keresés gombbal szűkíthetjük a megjelenő termékek listáját és ezeket sorrendbe is állíthatjuk. (ugyanúgy a főoldalon maradunk)

A kosár gomb átirányít minket a kosár oldalra, ahol továbbugorhatunk a vásárlás oldalra egy gombbal, vagy vissza a főoldalra. A kosárba rakott termékek kilistázódnak, amiket az eltávolítás gombbal eltüntethetünk a listából.

A bejelentkezés oldalra ezzel a gombbal irányíthatjuk át magunkat. Itt csak bejelentkezésre vagy továbbugrani a regisztrációhoz van lehetőség. Regisztrációnál felhasználónevet, email címet és jelszót kérünk be. Ha a mése gombra kattintunk, visszakerülünk a főoldalra.

Majd ha a vásárlás gombra kattintunk ki kell töltenünk a személyes információkkal egy új lapot. Ezzel továbbugrunk a véglegesítéshez, ahol a kosárban lévő elemeket és a személyes infókat még egyszer áttekintheti a vásárló. A véglegesítés gombbal leadja a rendelést.

A rendszer bővíthető további lapokkal. A meglévő lapokon belül bővíthető funkcionális és nem funkcionális elemekkel. Több helyen megjeleíthető kép vagy a logó, amennyiben újabb igény érkezik be ezzel kapcsolatban.

## 8. Adatbázis terv

Programunk a tárolandó adatokat, online adatbázisban fogja tárolni.

Online felületnek a phpmyadmint fogjuk használni. Tesztelési fázisban, egyelőre mindenki offline, saját gépén elérhető adatbázist használjon, amit egy központi adatbázisból fogunk exportálni, így elkerülendő bárminemű, nem kívánt módosítás a központi adatbázisban.



Az adatbázis felépítésénél ügyeljünk a megrendelő kérésére. Azaz legyen elkülönítve a regisztrált felhasználók, a vásárlások, illetve az áruk adatai, egy-egy különálló táblában.

A táblák a következők legyenek:

- "users",
- "goods",
- "orders".

A "users" táblában fogjuk tárolni a regisztrált felhasználók adatait. Tárolandó adatok: felhasználó neve, e-mail címe, és jelszava titkosítva.

A tábla oszlopai:

- "username"-felhasználó név, elsődleges kulcs, külső kulcs az "order" táblához, egyedi, nem lehet üres, varchar2, 64 karakter hosszú.
- "email"-e-mail cím, varchar2, egyedi, nem lehet üres, 64 karakter hosszú.
- "password"-jelszó, nem lehet üres, varchar2 64 karakter hosszú, MD5 titkosított.

A "goods" táblában fogjuk tárolni, az áruk adatait.

Tárolandó adatok: elérhető mennyiség, név, ár, cikkszám, értékelés.

- "quantity"-elérhető mennyiség, number, 4 karakter hosszú, nem lehet üres.
- "name"-név, egyedi, nem lehet üres, varchar2 32 karakter hosszú.
- "price"-ár, nem lehet üres, number 6 karakter hosszú.
- "number"-cikkszám, elsődleges kulcs, egyedi, nem lehet üres, varchar2 64 karakter hosszú.

- "rating"-értékelés, number 1 karakter hosszú.

Az "orders" táblában fogjuk tárolni a leadott rendeléseket. Tárolandó adatok: megrendelő neve, felhasználó neve, telefonszáma, szállítási címe, e-mail címe, fizetési módszere, szállítási módszer, rendelt áruk, rendelés száma.

- "name"-név, nem lehet üres, varchar2, 64 karakter hosszú.

- "username"-felhasználó neve, külön kulcs a "users" táblához, nem lehet üres, varchar2 64 karakter hosszú.

- "phone"-telefonszám, number 11 karakter hosszú, nem lehet üres.

- "address"-szállítási cím, varchar2 64 karakter, nem lehet üres.

- "email"-e-mail, varchar2 64 karakter hosszú, nem lehet üres.

- "payment\_method"-fizetési módszer, nem lehet üres, varchar2 32 karakter hosszú.

- "shipment\_method"-szállítási módszer, nem lehet üres, varchar2 32 karakter hosszú.

- "goods"-rendelt áruk, nem lehet üres, number 324 karakter hosszú. Az áruk cikkszámait tárolja ", "-vel elválasztva, maximum 5 áru tárolására alkalmas.

- "tracking\_number"-rendelés száma, egyedi, elsődleges kulcs, nem lehet üres, number 6 karakter hosszú.

Az adatbázisról megtalálható egy diagram az alábbi címen:



## 9. Implementációs terv

A fejlesztés során, objektum orientált programozással fogjuk elkészíteni a The Snail Sale programot. Ehhez Java programozási nyelvet fogunk használni, a 11-es verziót. A grafikus felületet JavaFx segítségével valósítjuk meg. A program osztályait 4 csoportba lehet besorolni.

- A grafikus felületekért felelős osztályok
- A grafikus felületek inputját lekezelő függvényeket tartalmazó osztályok
- Az adatbázist használó és kezelő függvények osztálya
- A main függvény szerepét játszó indító osztály

Az egyes csoportba tartozó osztályok és megnevezéseik:

-GUI osztályok:

- login.java
- registration.java
- mainPage.java
- cart.java

- cartOverview.java
- personalInfo.java
- orderConfirmation.java

-GUI input kezelő osztályok:

- loginController.java
- registrationController.java
- mainPageController.java
- cartController.java
- cartOverviewController.java
- personalInfoController.java
- orderConfirmationController.java

-Az adatbázist kezelő osztályok:

- database.java

Az indító osztály:

- launcher.java

Az egyes osztályok és tartalmuk:

-login.java: Ebben az osztályban kerül megvalósításra a belépés képernyőnek a megírása.

Kérje be a program a belépéshez szükséges információkat a felhasználótól. Sikeres bejelentkezés esetén a mainPage.java megfelelően megváltoztatott ablakát hozza fel.

A még nem regisztrált felhasználóknak adjon lehetőséget a regisztrációra a registration.java-ban megírt ablak felhozásával.

-registration.java:Ebben az osztályban kerül megvalósításra a regisztráció képernyőnek a megírása. Kérje be a program a regisztrációhoz szükséges információkat a felhasználótól. Sikeres regisztráció esetén a login.java ablakát hozza fel.

-mainPage.java:Ebben az osztályban kerül megvalósításra a böngészés képernyőnek a megírása. Továbbá, ez a főképernyő, ezt indítja a launcher.java main függvénye. Ezen a képernyőn helyezkedik el a kosár, a bejelentkezés, a vásárlás ablakhoz vezető gomb is.

-cart.java:Ebben az osztályban kerül megvalósításra a kosár képernyőnek a megírása. Sorakoztassa fel a felhasználó által a kosárba tett termékeket, információval együtt.

-cartOverview.java:Ebben az osztályban kerül megvalósításra a vásárlás előtt a kosár tartalmát felmutató képernyőnek a megírása. Tudjon a felhasználó törölni nem kívánt árucikkeket, vagy mennyiségeket. A vásárlás folytatásához a megfelelő gomb megnyomása után a personalInfo.java osztályban megírt ablakot hozza fel.

-personalInfo.java:Ebben az osztályban kerül megvalósításra a vásárlásnál a személyes információk bekérésének képernyőjének a megírása.

-orderConfirmation.java: Ebben az osztályban kerül megvalósításra a vásárlás végleges visszaigazolásának képernyőjének a megírása.

-loginController.java: a login.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-registrationController.java: a registration.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-mainPageController.java: a mainPage.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-cartController.java: a cart.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-cartOverviewController.java: a cartOverview.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-personalInfoController.java: a personalInfo.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-orderConfirmationController.java: a orderConfirmation.java GUI osztályhoz tartozó függvényeket tartalmazó osztály

-launcher.java: Ebben az osztályban található a main függvény, ami elindítja a mainPage.java start függvényét.



-database.java: Az adatbázist használó és kezelő függvényeket tartalmazó osztály

## 10. Tesztterv

A tesztelés során a teszt tárgya lehet a rendszer azon része, amelyet tesztelünk, ez lehet az egész rendszer is. A mi esetünkben teszt eseteket fogunk megnézni, és manuálisan tesztelni, ezzel lefedve az egész rendszert. Tesztbázis azon dokumentumok összessége, amelyek a teszt tárgyára vonatkozó követelményeket tartalmazzák: Követelmény specifikáció és funkcionális specifikáció.

A tesztelés során ellenőrizzük, hogy a felhasználói felület megfelelően betöltődött, a keresés, a gombok és a kattintható felületek megfelelően működnek.

Ha a felület betöltődött, akkor gombok(Kosár,Regisztráció,Belépés stb), keresés és egyéb kattintható felületet vizsgáljuk, megjelenik-e minden opció, amiket a fejlesztés során implementáltak.

Az eddig elkészített program teszt esetei:

A program megfelelően, hibamentesen elindul, majd a felület betölt.

A felület a képernyőterv szerint jelenik meg, kivéve a Kosár(javítás alatt áll).

Az elindítás utáni felületen több gomb és keresési mező is látható.

A gombok kattinthatóak. A keresés manuális tesztelése még nem lehetséges, mivel a program még nincs teljesen kész, fejlesztés alatt áll.

A Kosár gombra kattintva behozza a kosár tartalmát(még üresen), ez még nincs teljesen kész.Innen visszalehet ugrani a fő felületre, vagy pedig a pénztárhoz, de előtte mégfelugrik egy ablak, a személyes információkért.

A feladat egyik fő része a regisztráció és a bejelentkezés.

A Belépésre kattintva átvált a erre az oldalra és kér egy felhasználónevet, jelszót. Még nem lehet belépni, vagy beregisztrálni.

Ha még nincs fiók akkor tovább kell lépni a regisztrációhoz.

Itt majd meg kell adni felhasználónevet, email címet és jelszót, amikell később majd be tudnak jelentkezni. Ezek a tesztesetek működőképeseek, a gombok kattinthatóak, a program mostani állása szerint, ha rányomunk egy adott gombra, akkor átvisz minket a megfelelő felületre. A program hiányossága miatt a keresés, értékelés, belépés és regisztráció még nem lehetséges, de ezt a későbbiekben korrigáljuk, fejlesztés alatt van. A fenti tesztelések manuálisan történtek.

## 11. Telepítési terv

Meghatározzuk, hogy mikor és milyen módon lesz elérhető a szoftvertermék a végfelhasználók számára. Szükséges az ügyfelekkel való együttműködés és az előkészületek is. Gyakran a szoftver fejlesztésétől független tényezők határozzák meg egy szoftvertermék bevezetését, például a hardver infrastruktúra hiánya, vagy a nem megfelelően képzett felhasználók.

A gördülékenység érdekében telepítési csomagot kell létrehozni. A telepítési csomag a kifejlesztett szoftver mellett azokat a szoftvereket is tartalmazza, amelyek az installáláshoz és a használathoz szükségesek. A telepítő szoftver biztosítja a szoftver komponensek hozzárendelését a megfelelő hardver elemekhez.

A későbbiekben a program megrendelésének és letöltésének lehetővé tétele az interneten keresztül. A program első verziójában a szoftver csak lokálisan működne, így a felhasználó a telepítés után az éppen használt eszközön tudná igénybe venni a szolgáltatást. Majd a webes adattárolás segítségével a felhasználók internetkapcsolat segítségével több eszközről (amin a program telepítve van) is hozzáférhetnek az adatokhoz.

A program fizikai környezetének információi fentebb olvashatóak.

## 12. Karbantartási terv

A szoftver internetkapcsolat segítségével a telepítési hellyel megegyező helyre kerülnének ki a frissítések, és az új verziók. A tesztervek az új igények alapján készülnek el. A regressziós tesztet minden frissítés és új verzió alkalmával elvégezzük. Ezek a a tesztervek menüpontban olvashatóak.

A tapasztalatok gyűjtése, dokumentálása segítséget nyújt majd a későbbi fejlesztések során/folyamatfejlesztéshez, hogy milyen újítások legyenek, milyen hibák vannak, vagy karbantartási céllal is készülhetnek.

Továbbá a gyakori frissítések megnövelik a karbantartás költségeit.