A program telepítése:  
A ***weblap*** könyvtárt a webszerverre kell másolni.  
A backend rendszert a ***gs-rest-service\initial*** könyvtárból az InteliJ IDEA rendszerben futtattuk.  
Az adatbázist a ***vizsga.sql*** fájlból a localhost-on levő adatbázisba kell importálni. A használt sql login a root és az üres jelszó.

A program frontend részét egy nyílt forráskódú keretrendszer, a Bootstrap segítségével készítettük el.

A weboldal neve 'Pénz-kereső', melynek HTML és CSS kódjait a Visual Studio Code alkalmazásával szerkesztettük.

Célja, hogy összehozza egymással azon emberek csoportját, akik nem banki hitelt, vagy befektetést keresnek. Lehetővé tesz egy későbbi csereügyletet. Amihez a leendő partnerek itt megtalálhatják egymást. Az egyszerűség és könnyebb követhetőség érdekében adott összegeket és futamidőket „cserélhetnek”. Ezeket megjelölve tudja az oldal használója kiválasztani, hogy mekkora összeget és milyen futamidőt keres. Előkalkulációt végezhet arra, hogy várhatóan mennyi lesz a kamata, vagy arra, hogy mennyi hitelkamatot szükséges megfizetnie a lejáratkor. A kamatszámítás alapját egy adatbázis adja, ahol a jegybanki alapkamathoz igazodva kerül meghatározásra a kamatláb. Ez heti rendszerességgel manuálisan frissítésre kerül.

Regisztráció után tudják rögzíteni az igényüket. Ennek során eltárolásra kerülnek az adatok: email-cím és jelszó

Majd a két oldali igény egymásra találása, azaz „egyezés” esetén, összerendeljük az információkat és erről visszajelzést adunk.

A weboldal felépítése:

A navigációs sávon található fülek, egymással összekapcsolt HTML oldalak.

A következő elnevezést és tartalmat kapták:

**Főoldal** - az oldal lehetőségeinek ismertetése és a működés felvezetése. Innen is elérhetők a konkrét érdeklődést vezető aloldalak: a „Befektetés érdekel” és a „Hitel érdekel”.

**Befektetés érdekel** - Leírást tartalmaz a befektetésekről és az inflációról, a befektetni kívánt összeget és a futamidőt rádiógombok segítségével választhatjuk ki ,majd a 'Kamatszámítás' gombra nyomva láthatjuk mennyi lesz a hozammal növelt pénzösszeg, majd ha szimpatikus amit látunk, a 'Rögzítés' gomb által leadhatjuk jelentkezésünket.

**Hitel érdekel** - Itt egy rövidebb tájékoztató elolvasása után, szintén rádiógombokkal választhatjuk ki az összeget és a futamidőt illetve jelezhetjük igényünket.

**Regisztráció** - Email cím, név és jelszó megadásával regisztrálhatunk

**Belépés** - A regisztráció során megadott adatokkal beléphetünk és leadhatjuk igényünket befektetésre vagy hitel felvételhez. Belépés után ez a menüpont lecserélésre kerül a kilépés menüpontra.

A backend rendszert java nyelven írtam mysql adatbázis támogatással. A REST szerver a spring framework használatával készült.

A 4 db rest hívást az adatbázis lekérdezéssel szolgálja ki.

A rest hívások:

**registerUser:**

Egy felhasználót regisztrál

Megkapja az e-mail címet, jelszót, és a nevet, majd ezt letárolja az adatbázisban.

Ha már ilyen e-mail címet regisztráltak vagy valami egyéb hiba van hibaüzenettel tér vissza a hívás.

**loginUser:**

A beléptetés intézi

Megkapja az email címet és a jelszót, és ha ez benne van az adatbázisban akkor visszaadja a nevet és az id-t.

Ha nem szerepel az adatbázisban vagy valami hiba lépett fel akkor egy hibakóddal és hibaüzenettel tér vissza.

**kamat:**

Az aktuális kamattal tér vissza.

Ez egy lekérő lekérdezés még nincsen igazán használva.

**befektetes:**

egy tranzakciót rögzít a rendszerve(befektetés, hitel)

megkapja a befektető adatait (név, id) és a tranzakció adatait (osszeg, futamidő, trakcíó iránya(befektetés.hitel))

Ellenőrzi, hogy a felhasználó létezik-e és ha igen akkor rögzíti a tranzakciót és keres hozzá egy párt, aminek ugyanazok az adatai csak a tranzakció iránya fordított, majd mind a 2 tranzakcióhoz bejegyzi a másik azonosítóját és az összerendelés dátumát.

Visszatér azzal, hogy van-e összerendelés. Ha valami hiba lépett fel akkor a hibával tér vissza.

Tesztelést az ezt meghívó weblapon keresztül végeztük.

Tesztek:

**Regisztráció**:

A felület a következő címen érhető el: <http://localhost/regisztracio.html>

A teszteléseket minden fajta hibalehetőséget megvalósító adatokkal végeztük. Egy üres adatbázissal kezdjük és a következő adatokat tartalmazó kéréseket próbáljuk elküldeni.

|  |  |  |  |
| --- | --- | --- | --- |
| Email cím | Password | Teljes név | Eredmény |
|  |  |  | Hibás email cím! A jelszó legalább 8 karakter! Nincs név! |
| [proba@proba.hu](mailto:proba@proba.hu) | 123 | Próba | A jelszó legalább 8 karakter! |
| proba | 12345678 | Próba | Hibás email cím! |
| [proba@proba.hu](mailto:proba@proba.hu) | 12345678 |  | Nincs név! |
| [proba@proba.hu](mailto:proba@proba.hu) | 12345678 | Próba | Sikeres Regisztráció! |
| [proba@proba.hu](mailto:proba@proba.hu) | 12345678 | Próba2 | Ez az email cím már szerepelt |

Ezeknek a hibáknak nagy részét a javascript kezeli le.

**Belépés:**

A felület a következő címen érhető el: <http://localhost/belepes.html>

A tesztelést az előbb használt adatbázissal fogjuk elvégezni.

|  |  |  |
| --- | --- | --- |
| Email cím | Jelszó | eredmény |
| [proba@proba.hu](mailto:proba@proba.hu) | 123 | Nincs ilyen felhasználó jelszó páros! |
| proba | 12345678 | Nincs ilyen felhasználó jelszó páros! |
|  |  | Nincs ilyen felhasználó jelszó páros! |
| [proba@proba.hu](mailto:proba@proba.hu) | 12345678 | Belép az oldalra, lecseréli a belépés menüt és visszalép a főoldalra (a belépett felhasználó adatait eltárolja) |

A belépő oldal csak akkor elérhető ha nincs még senki belépve.

A **befektetés érdekel** és a **Hitel érdekel** weblapokat együtt érdemes tesztelni mert összefüggenek és a felépítésük és szerkezetük megegyezik.

Tesztelésnél az összes lehetőséget végig próbáljuk.

|  |  |  |
| --- | --- | --- |
| összeg | futamidő | kamatszámítás |
| 1 millió Ft | 3 hónap | 1032500 |
| 2 millió Ft | 3 hónap | 2065000 |
| 5 millió Ft | 3 hónap | 5162500 |
| 1 millió Ft | 6 hónap | 1065000 |
| 2 millió Ft | 6 hónap | 2130000 |
| 5 millió Ft | 6 hónap | 5325000 |

A beküldésnél meg a *Rögzítés* illetve *Hitel érdekel* gombok megnyomásával bekerül az adatbázisa tranzakció. Az eredmény meg attól függ van-e még olyan paraméterű ellentétes szabad tranzakció az adatbázisba.

A válasz minden esetben a *Nincs összerendelés* vagy a *Sikeres összerendelés* attól függően hogy van-e nem kezelt párja az adatbázisban vagy nincs.