BMSZC Bláthy Ottó Titusz Informatikai Technikum

1032. Budapest, Bécsi út 134.

1. Záró dolgozat

A programom címe

Konzulens tanár: Készítette:

Dr. Péter-Szabó Richárd Lovas Botond,

Rafael Dávid Bertalan, Szabó Milán Péter

Tartalom

[1 Bevezetés 3](#_Toc163563244)

[1.1 Feladat leírás 3](#_Toc163563245)

[1.1.1 Általános ismertetés 3](#_Toc163563246)

[2 Felhasználói dokumentáció 4](#_Toc163563247)

[2.1 A program általános specifikációja 4](#_Toc163563248)

[2.2 Rendszerkövetelmények 4](#_Toc163563249)

[2.2.1 Hardver követelmények 4](#_Toc163563250)

[2.2.2 Szoftver követelmények 4](#_Toc163563251)

[2.3 3. A program telepítése 4](#_Toc163563252)

[2.4 A program használatának a részletes leírása 5](#_Toc163563253)

[2.5 GYÍK, FAQ (Gyakran Ismételt Kérdések) 5](#_Toc163563254)

[3 Fejlesztői dokumentáció 6](#_Toc163563255)

[3.1 Az alkalmazott fejlesztői eszközök 6](#_Toc163563256)

[3.1.1 Adatmodell leírása 6](#_Toc163563257)

[3.2 Részletes feladatspecifikáció, algoritmusok 6](#_Toc163563258)

[3.3 Tesztelési dokumentáció 7](#_Toc163563259)

[4 Összegzés 8](#_Toc163563260)

[Továbbfejlesztési lehetőségek 8](#_Toc163563261)

[5 Ábrajegyzék 9](#_Toc163563262)

[6 Felhasznált irodalom 10](#_Toc163563263)

# Bevezetés

## 1.1 Feladat leírás

### Általános ismertetés

Témaválasztás indoklása A pénzügyi tudatosság és az okos pénzügyi döntések meghozatala alapvető fontosságú mindenki számára. Ezért választottuk a pénzügyeket projektünk témájának, hogy segítsünk az embereknek megérteni és hatékonyan kezelni pénzüket. Az oktató oldalunk a Laravel keretrendszerre épül, ami lehetővé teszi, hogy interaktív és felhasználóbarát tanulási élményt nyújtsunk a pénzügyi témákban.

A Laravel egy modern, kifinomult és könnyen tanulható keretrendszer, amely kiválóan alkalmas webalkalmazások fejlesztésére. Rugalmas kialakításának köszönhetően lehetőséget nyújt a hatékony programozásra és a gyors fejlesztésre. Ennek eredményeként az oktató oldalunkon könnyen és gyorsan létrehozhatunk interaktív tanulási modulokat, gyakorlati feladatokat és egyéb tanulási eszközöket, amelyek segítik az embereket abban, hogy megtanulják és gyakorolják a pénzügyi készségeket.

Célunk az, hogy elősegítsük az emberek pénzügyi függetlenségét és sikerét, miközben megkönnyítjük számukra a Laravel keretrendszer használatát. Azáltal, hogy kombináljuk a pénzügyi tudatosságot és az informatikai ismereteket, olyan erőforrásokat kínálunk, amelyek segítenek az embereknek felkészülni a jövő pénzügyi kihívásaira és lehetőségeire, és egyúttal fejlesztik digitális készségeiket is.

Oldalunknak a célja hogy a diákoknak olyan oktatási platformot nyújtsunk, amely célja, hogy segítse a őket a pénzügyi ismeretek megszerzésében és fejlesztésében. Az oldal célkitűzése, hogy könnyen hozzáférhető és átfogó tartalmakat kínáljon, amelyek segítenek a fiataloknak megérteni és kezelni pénzügyeiket. Az alábbiakban részletezem a weboldal általános specifikációit és funkcióit.

# Felhasználói dokumentáció

## 2.1 A program általános specifikációja

### Felhasználói Felület

**Fiatalos és vonzó dizájn**: Az oldal modern, fiatalos megjelenésű, amely könnyen vonzó a diákok számára. A színek, a betűtípusok és az elrendezés fiatalos és interaktív.

**Könnyen áttekinthető menü**: Az egyszerű navigáció segít a diákoknak gyorsan megtalálni a keresett információkat és anyagokat.

### Oktatási Tartalmak

**Tananyagok**: Az oldal átfogó tananyagokat kínál a pénzügyi ismeretek terén, beleértve az alapfogalmakat, mint például a költségvetés-készítés, a megtakarítás, a befektetés, a hitel és a biztosítás.

**Interaktív feladatok**: Az oldal számos feladatot és gyakorlatot kínál, amelyek lehetővé teszik a diákok számára, hogy alkalmazzák a tanultakat és megerősítsék tudásukat.

**Kvízek és tesztek**: Az oktatóanyagok mellett kvízek és tesztek is rendelkezésre állnak, hogy a diákok ellenőrizzék és mérjék a tudásukat.

## Rendszerkövetelmények

### 2.2.1 Hardver követelmények

#### Asztali Számítógép vagy Laptop

**Operációs Rendszer:**

Minimális követelmény**: Windows 7, macOS X 10.11, Ubuntu 16.04**

Ajánlott követelmény**: Windows 10/11, macOS Catalina, Ubuntu 20.04**

**Processzor:**

Minimális követelmény**: Dual-core 2 GHz vagy hasonló**

Ajánlott követelmény**: Quad-core 2.5 GHz vagy jobb**

**RAM:**

Minimális követelmény**: 4 GB**

Ajánlott követelmény**: 8 GB vagy több**

**Tárhely:**

Minimális követelmény**: 500 MB szabad lemezterület**

Ajánlott követelmény**: 1 GB szabad lemezterület**

**Böngésző:**

Minimális követelmény**: Modern böngészők legfrissebb verziói, mint a Google Chrome, Mozilla Firefox, Safari, vagy Microsoft Edge**

Ajánlott követelmény**: A böngésző legfrissebb stabil verziója, hogy biztosítsa a teljes kompatibilitást és biztonságot.**

#### Mobiltelefon vagy Tablet

**Operációs Rendszer:**

Minimális követelmény**: Android 7.0 vagy iOS 11**

Ajánlott követelmény**: Android 11 vagy iOS 15**

**Processzor:**

Minimális követelmény**: Quad-core 1.5 GHz vagy hasonló**

Ajánlott követelmény**: Nyolc magos (octa-core) processzor 2 GHz vagy jobb**

**RAM:**

Minimális követelmény**: 2 GB**

Ajánlott követelmény**: 4 GB vagy több**

**Tárhely:**

Minimális követelmény**: 500 MB szabad tárhely**

Ajánlott követelmény**: 1 GB szabad tárhely vagy több**

**Böngésző:**

Minimális követelmény**: Modern mobilböngészők legfrissebb verziói, mint a Google Chrome, Mozilla Firefox, Safari, vagy Samsung Internet**

Ajánlott követelmény**: A böngésző legfrissebb stabil verziója a teljes kompatibilitás és biztonság érdekében.**

**Internetkapcsolat**

Minimális követelmény**: 2 Mbps letöltési sebesség, stabil internetkapcsolat**

Ajánlott követelmény**: 10 Mbps vagy gyorsabb letöltési sebesség, stabil és megbízható internetkapcsolat**

### Szoftver követelmények

#### Operációs Rendszerek

A weboldal böngészőalapú, így bármilyen operációs rendszeren futtatható, amely támogatja a modern webes böngészők legújabb verzióit:

**Windows**: Windows 7 vagy újabb verziók, beleértve a Windows 10/11-et.

**macOS**: macOS X 10.11 vagy újabb verziók.

**Linux**: A legtöbb modern Linux disztribúció, mint például Ubuntu, Fedora, Debian.

**Android**: Android 7.0 vagy újabb verziók.

**iOS**: iOS 11 vagy újabb verziók.

#### Böngészők

A weboldal kompatibilis a modern webes böngészők legfrissebb verzióival:

* **Google Chrome**
* **Mozilla Firefox**
* **Apple Safari**
* **Microsoft Edge**
* **Opera**

Ajánlott a böngészők legfrissebb stabil verziójának használata a legjobb felhasználói élmény és biztonság érdekében.

#### Egyéb Szoftverkomponensek

**PHP**: A weboldal PHP-t használ a szerveroldali logika és a dinamikus webes tartalmak kezelésére. Ajánlott a legfrissebb PHP-verzió használata.

**Laravel**: A platform Laravel keretrendszert használ a fejlesztéshez és a hatékony szerveroldali logika megvalósításához. Ajánlott a legfrissebb stabil verzió használata a legjobb teljesítmény és biztonság érdekében.

**XAMPP**: A fejlesztők számára a XAMPP fejlesztői csomagot ajánljuk, amely egy Apache-webszervert, MySQL/MariaDB adatbázist és PHP-t kínál egy integrált környezetben. Ajánlott a legfrissebb XAMPP-verzió használata.

**Composer**: A Composer egy csomagkezelő, amely segíti a szoftverfejlesztők munkáját a függőségek és a csomagok kezelésében. Ajánlott a legfrissebb Composer-verzió használata.

**JavaScript és CSS**: A weboldal használ JavaScriptet és CSS-t a felhasználói élmény javítására és a modern webes szabványok támogatására. Ezért a böngészőknek támogatniuk kell a JavaScriptet és a modern CSS szabványokat.

**Adatbázis-szerver**: A weboldalhoz adatbázis-szerver szükséges a felhasználói adatok és tartalmak kezeléséhez. Az ajánlott választások közé tartozik a MySQL vagy a MariaDB, melyek mindegyike támogatja a Laravel alkalmazásokat.

## 2.3. A program telepítése

Képekkel illusztrált, részletes leírás a program telepítésének a menetéről.

* A leírás alapján a felhasználónak hiba nélkül telepíteni kell tudni a programot. A leírásnak ki kell térnie a telepítés során kiválasztható opciókra is.
* Ha esetleg nincs telepítőprogram, akkor kellő részletességgel le kell írni, hogy mely fájlokat, pontosan hova kell felmásolni, és hogy lehet a programot futtatni.

Ajánlott terjedelem: 2 -4 oldal, ábrákkal együtt. Sok képpel, halálpontos leírással, de értelemszerűen ne túl bőbeszédűen. Weboldalnál XAMPP telepítés, a program fájlok lehúzása, composer.json install (ha nincs fent). Röviden, ha egy üres gépre a programotok átkerül egy pendrive-ról, mit kell csinálni.

## 2.4 A program használatának a részletes leírása

Aranyszabály: Ha soha életemben nem láttam a programot, akkor is tudjak egy másfél órás előadást tartani az egyetemen arról, hogy mit csinál a program és ne tudjanak tőlem olyat kérdezni, amire itt nincs válasz.

Mindenre kiterjedő, részletes leírás a program használatáról. Alapszabályok:

* Amit leprogramoztál, azt a dokumentációban írd is le, ne legyenek eltitkolt funkciók.
* Minden pontosan, „szájbarágósan” legyen leírva. A dokumentáció alapján a teljesen kezdő, vagy laikus felhasználóknak is használniuk kell tudni a programot.
* A stílus legyen pontos és közérthető, vedd figyelembe, hogy a felhasználói dokumentáció nem szakembereknek készül.
* Alkalmazz ábrákat, screenshot-okat , de a ne legyen túlzott a képek aránya a szöveghez képest, kb. oldalanként egy ábra megfelelő.
* Logikusan építsd a szerkezetet, az első indítástól.
* Nyugodtan használjatok alfejezeteket.
* Legyen benne, hogy mikor mi történik optimális esetben és mi mehet félre. (Nem kód érdekel és nem a miértje, hanem pl.: nem megy a regisztráció, mert nem adtál e-mailt.)

Ajánlott terjedelem: 10-15 oldal, ábrákkal együtt vagy több. Ábrát szemléltetésre, nem helykitöltésre használunk.

## GYÍK, FAQ (Gyakran Ismételt Kérdések)

Írj pár általános triviális kérdést, zárásnak jó lesz.

# Fejlesztői dokumentáció

*A fejlesztői dokumentáció célja, hogy a segítse program logikájának, illetve a program kódjának a megértését, illetve a program továbbfejlesztését. Szakemberek számára készül, elvárás tehát a pontosság és a szakmai jellegű stílus. (EZT MAJD TÖRÖLD KI)*

## Az alkalmazott fejlesztői eszközök

Fel kell sorolnod az összes olyan szoftver eszközt amelyet a program fejlesztéséhez, illetve a dokumentáció készítéséhez felhasználtál. (Pl. programozási nyelv, fejlesztői környezet, adatbázis-kezelő rendszer, kép-, szöveg-, zeneszerkesztő program, modulok, pluginok, keretrendszerek stb.)

* Ha használtál mások által kifejlesztett modulokat, akkor azt is le kell írnod, a forrás pontos megjelölésével.
* Tájékozódj ezeknek a moduloknak a jogszerű felhasználásáról!

Ajánlott terjedelem: ½ -1 oldal.

### Adatmodell leírása

A feladatban alkalmazott adatbázis vagy adatszerkezet részletes leírása

* Adatbázis esetén az adattáblák leírása, a közöttük lévő kapcsolatok megadása, lehetőleg diagram is legyen. (Max A4 oldal terjedelem, ha olvashatatlan ez van, de törekedjünk, és legyen egy szöveges leírás is (kb. mint érettséginél) )
* A program fő adatszerkezeteinek a specifikációja, célszerű diagramot is rajzolni
* UML osztálydiagram, értelmesen, nem kell maximális részletezés (lásd: https://miro.com/diagramming/what-is-a-uml-diagram/)

Ajánlott terjedelem: a feladat jellegétől függően 2-5 oldal.

## Részletes feladatspecifikáció, algoritmusok

A program lényeges függvényeinek, az osztályok metódusainak a specifikációja (mit valósít meg az adott függvény, illetve metódus, milyen paraméterei vannak, mi a visszatérési érték). MINDEN EGYES NEM MAGÁTÓL ÉRTETŐDŐ ELJÁRÁS! Magától értetődő pl laravelnél a resources (CRUD) funkciók. ALSÓ HANGON LEGALÁBB 10-15, de inkább 20-25 oldal.

pl.:

A képen szöveg, képernyőkép, Betűtípus, szoftver látható

Automatikusan generált leírás

* Lehet viszont a nyomtatott dokumentációban a fontosabb kódrészeket magyarázattal szerepeltetni

## Tesztelési dokumentáció

1. Legalább 3 különböző teszteset részletes bemutatása.

* különböző felhasználó tevékenységek esetén hogyan reagált a program
* milyen üzeneteket kaptunk
* mi a teendő az egyes üzenetek esetében

1. Normál teszteset, extrém teszteset (*foolproof* tesztelése)
2. A tesztelés során kiderült hibák felsorolása A tesztelési dokumentációból derüljön ki, hogy ismered a különböző tesztelési módszereket (pl. fekete doboz, fehér doboz módszer)

Ajánlott terjedelem: a feladat jellegétől függően 2-5 oldal.

# Összegzés

## Továbbfejlesztési lehetőségek

* Olyan ötletek, amelyeket meg akartál valósítani, de nem sikerült, vagy nem fért bele az időbe
* Olyan ötletek, amelyeket még érdemes a jövőben megvalósítani
* Joker rész, minden amivel még jobb lehet a projekt majd, amivel még eladhatóbb, még többet tudhat majd stb.
* Itt azt kell prezentálni, hogy érted a piaci trendeket, elvárásokat.

Ajánlott terjedelem: 1-2 oldal

# Ábrajegyzék

Minden ábrának adjatok feliratot (jobb click az ábrán, felirat beszúrása).

Az ábra típust kiválasztva később ide beszúrható a hivatkozások, ábrajegyzék beszúrása menüvel.

# Felhasznált irodalom

Minden olyan forrás pontos megadása, amelyet a szakdolgozatodban felhasználtál.

A forrás lehet pl.

Könyv. Meg kell adnod a következőket: szerző(k), cím, kiadó, kiadás éve.

Weboldal. Meg kell adnod a linket, az oldal címét

Elektronikus dokumentum. Meg kell adnod a szerzőt, a letöltés helyét

Ajánlott terjedelem: Amennyi kell.

--------------------------------------------------------------------------------------------------------

További hasznos formai elemek:

III. A dokumentáció formai követelményei

**A fedőlapon fel kell tüntetned a következő adatokat: (Still in progress)**

* Az oktatási intézmény megnevezése
* szakképesítés neve és OKJ száma
* SZAKDOLGOZAT
* A dolgozat címe
* A szakdolgozat készítő neve, osztálya
* A témavezető (konzulens) neve
* A benyújtás helye (Budapest)
* A benyújtás éve

**Fejezetek, alfejezetek alkalmazása**

* A dokumentációt tagolni kell, lehetőleg két-három szinten.
* A fejezetek, alfejezetek többszintű sorszámozással legyenek ellátva.
* A fejezetcímeket Címsor1, Címsor2, … stílusokkal formázd, hogy ez alapján tudj majd tartalomjegyzéket generálni!

**Karakter és bekezdés formátum**

o Az alap betűméret: 12-es. Címek 18, 16 vagy 14 alap word stílus formázás.

* Times New Roman.
* Vastag és dőlt betűt alkalmazhatsz, de csak mértékkel, a fontos részek kiemeléséhez.
* A bekezdések igazítása sorkizárt legyen
* A bekezdések első sorának legyen behúzása
* 1,5-es sorköz a Kódrészletek esetében javasolt a Courier New, vagy más, fix szélességű betűtípust alkalmazni, és a kódot egy szegéllyel elkülöníteni a szövegtörzstől, vagy kód plugin, vagy egységes kép.
* Ha felsorolsz, alkalmazz egy, vagy többszintű felsorolást.

**Oldal formátum**

* Az oldalak számozva legyenek, a számozás a láblécbe kerüljön középre.
* Fejléc jobb oldali sarok, projekt neve.
* 2,5 margó jobb oldalt és alul felül, BAL oldalt pedig 3.5 kötésmargó.

**Tartalomjegyzék**

* Kötelező szerepeltetni, és a dokumentáció elejére tenni.
* A tartalomjegyzéket érdemes a dokumentáció elkészültével, a fejezet és alfejezet címek alapján automatikusan generálni.

**Ábrák, táblázatok**

* Az ábráknak, képeknek, diagramoknak legyen sorszáma, a szövegben legyen rájuk hivatkozás. (Ha a jobb click, felirat beszúrásával csinálod, így lesz.
* Ha nem saját készítésű az ábra, a forrást meg kell jelölni.
* A táblázatok használata nagyon javasolt, pl. az adatmodell leírásánál, a tesztelési dokumentációban.

**Helyesírás-ellenőrzés, elválasztás alkalmazása**

* Nagyon nagy hibának számít a szakdolgozatot helyesírási hibákkal telezsúfolva beadni, Kapcsold be az ellenőrzést a dokumentáció írásakor.
* Az elválasztás alkalmazása célszerű, mert szebb lesz az oldalkép, nem lesznek nagy hézagok a sorokon belül.
* Terjedelem
* A vizsgadolgozat (mellékletek, táblázatok, grafikonok, forráskód nélküli) szövegrészének terjedelme legalább 30.000 karakter legyen.
* A szokásos szakdolgozat terjedelem, képekkel, diagramokkal, táblázatokkal együtt kb. 25-30 oldal, de inkább 40-50.

**Kötés**

* Ha kell, akkor elegendő spirálozni, az elejére egy átlátszó műanyag lapot tenni, a végére egy fekete karton. (COPYGURU).