



Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Automatizálási és Alkalmazott Informatikai Tanszék

Hanicz Tamás

FÁJLMEGOSZTÓ KERETRENDSZER FEJLESZTÉSE RASPBERRY PI ESZKÖZRE

KONZULENS

Dr. Asztalos Márk

BUDAPEST, 2017

Tartalomjegyzék

Összefoglaló	4
Abstract.....	5
1 Bevezetés	6
1.1 Téma ismertetése	6
1.2 A rendszerrel szemben támasztott elvárások	6
2 Technológiák	8
2.1 Frontend	8
2.2 Backend	8
2.2.1 Flask.....	8
2.2.2 Passlib	8
2.2.3 SQLAlchemy	8
2.2.4 Pyjwt	9
2.2.5 SQLite	9
2.3 Raspberry PI	9
3 Tervezés és fejlesztés.....	10
3.1 Adatbázis	10
3.1.1 User.....	10
3.1.2 File	11
3.1.3 Folder	11
3.1.4 File share.....	11
3.1.5 Role.....	11
3.1.6 Credential store	11
3.1.7 Log	11
4 Irodalomjegyzék.....	12
Függelék.....	13

HALLGATÓI NYILATKOZAT

Alulírott **Hanicz Tamás**, szigorló hallgató kijelentem, hogy ezt a diplomatervet meg nem engedett segítség nélkül, saját magam készítettem, csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző, cím, angol és magyar nyelvű tartalmi kivonat, készítés éve, konzulens(ek) neve) a BME VIK nyilvánosan hozzáférhető elektronikus formában, a munka teljes szövegét pedig az egyetem belső hálózatán keresztül (vagy hitelesített felhasználók számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik. Dékáni engedéllyel titkosított diplomatervek esetén a dolgozat szövege csak 3 év eltelte után válik hozzáférhetővé.

Kelt: Budapest, 2017. 04. 29

.....
Hanicz Tamás

Összefoglaló

Abstract

1 Bevezetés

1.1 Téma ismertetése

1.2 A rendszerrel szemben támasztott elvárások

A feladat egy olyan alkalmazás elkészítése, mely a felhasználók által feltöltött fájlokat tárolja és megosztja lehetőség szerint. A felhasználók bármikor letölthetik, törölhetik felülírhatják az általuk feltöltött fájlokat. A fájlokhoz a tulajdonosuk engedélye nélkül senki sem férhet hozzá.

A felhasználók a böngészőjük segítségével érhetik el az alkalmazást. Először egy login oldal köszönti őket, ahol a korábban beregisztrált adataikkal tudnak belépni. A még nem regisztrált felhasználóknak itt van lehetőségük regisztrálni. Regisztráció után a megadott email címre érkezik egy megerősítő email benne egy aktivációs linkkel. A felhasználó accountja csak ezen link meglátogatása után használható. Bejelentkezés során a jelszót háromszor lehet elrontani, utána rendszer kizárja a felhasználót és csak email-en keresztül lehet jelszót változtatni.

Belépés után a felületen az eddig a felhasználó által feltöltött fájlokat lehet letölteni, felülírni vagy esetleg újat feltölteni. Továbbá lehet feljegyzéseket, szövegeket elmenteni az oldalon keresztül egy beépített jegyzetömb segítségével. A fájlok fájlrendszerbe rendezhetők tetszőleges módon. Törlés után 14 napig még minden fájl visszaállítható utána azonban már nincs rá lehetőség. A felhasználók limitált tárolási területtel rendelkeznek. Ez jelenleg 1 GB, a későbbiekben az rendszer bővítése után nőni fog. Az oldalon elérhető egy logfile amiben az elmúlt napok történéseit lehet megtekinteni. Minden felhasználó csak a hozzá tartozó fájlokról lát logbejegyzéseket. A fájlok között keresni is lehet.

Egy adott fájl megosztásáról a tulajdonosa rendelkezhet. Tetszés szerint megoszthatja bárkivel egy publikus link segítségével vagy pedig adott felhasználókkal név vagy email segítségével. A jegyzetömb segítségével megosztott fájlokat több módon is meg lehet osztani:

- Read

- Write
- Delete (Magában foglalja az előző kettőt)

Lehetőség van a fájlok verziókövetett tárolására is, ami azt jelenti, hogy egy fájl korábbi verziói is elérhetőek, ezt a funkciót azonban külön be kell kapcsolni fájlanként.

A rendszerhez egy felhasználó típus tartozik, ami az egyszerű felhasználót jelöli. Admin jogosultságra nincs szükség, mivel más adataihoz alapvetően nem lehet hozzáférni.

Az alkalmazás képes egy másik nagyobb fájlmeegosztóval szinkronizáltan működni, ami azt jelenti, hogy bizonyos fájlok az felhasználó kérésére feltöltődnek oda is vagy pedig onnan letöltésre kerülnek.

2 Technológiák

Ebben a fejezetben a rendszer elkészítéséhez felhasznált technológiák kerülnek bemutatásra, valamint, hogy az egyes könyvtárak, kiegészítők milyen pluszt adtak hozzá a szoftverhez.

2.1 Frontend

2.2 Backend

Az alkalmazás backend-je Python programozási nyelv felhasználásával készült. Az alábbiakban a fejlesztéshez használt framework-ök bemutatása olvasható.

2.2.1 Flask

A backend alapja egy REST API, mely a Flask felhasználásával készült el. A Flask egy webes keretrendszer, mely Python alapú. Könyvtárakat, technológiákat nyújt ahhoz, hogy minél egyszerűbben és hatékonyabban lehessen webes alkalmazásokat elkészíteni. Önmagában nem nyújt adatbázis absztrakciós réteget, így azt a fejlesztőnek kell megvalósítania, vagy felhasználnia valamilyen kiegészítőt.

2.2.2 Passlib

A Passlib egy jelszó hash-elő könyvtár Pythonhoz. Ennek a segítségével hash-elem a jelszavakat és tárolom el a kapott értéket az adatbázisban. A könyvtárat a belépés és regisztráció funkcióknál használtam fel, ezek a folyamatok a későbbiekben kerülnek bemutatásra.

2.2.3 SQLAlchemy

Az SQLAlchemy egy Python SQL eszköz, mely ORM tulajdonsággal rendelkezik. Nagy segítséget nyújtva ezzel a fejlesztéshez.

2.2.4 Pyjwt

A Pyjwt egy könyvtár, ami kódol és dekódol JSON web tokeneket. A könyvtár az autentikáció során kerül felhasználásra. Belépéskor a kliens kap egy kódolt token-t, melyet minden kérésnél el kell küldenie a szerver felé.

2.2.5 SQLite

Az alkalmazás működéséhez elengedhetetlen valamilyen adatbázis használata az felhasználói adatok, logolás és a felhasználókhöz tartozó fájlok elérésének tárolásához. A választásom az SQLite-ra esett annak gyorsasága és egyszerűsége miatt.

2.3 Raspberry PI

Az alkalmazás egy Raspberry Pi 3 B modellen fog futni. Mivel az erőforrásai végesek ezért olyan megvalósításra kell törekedni, mely számításba veszi az alábbiakban felsorolt paramétereket.

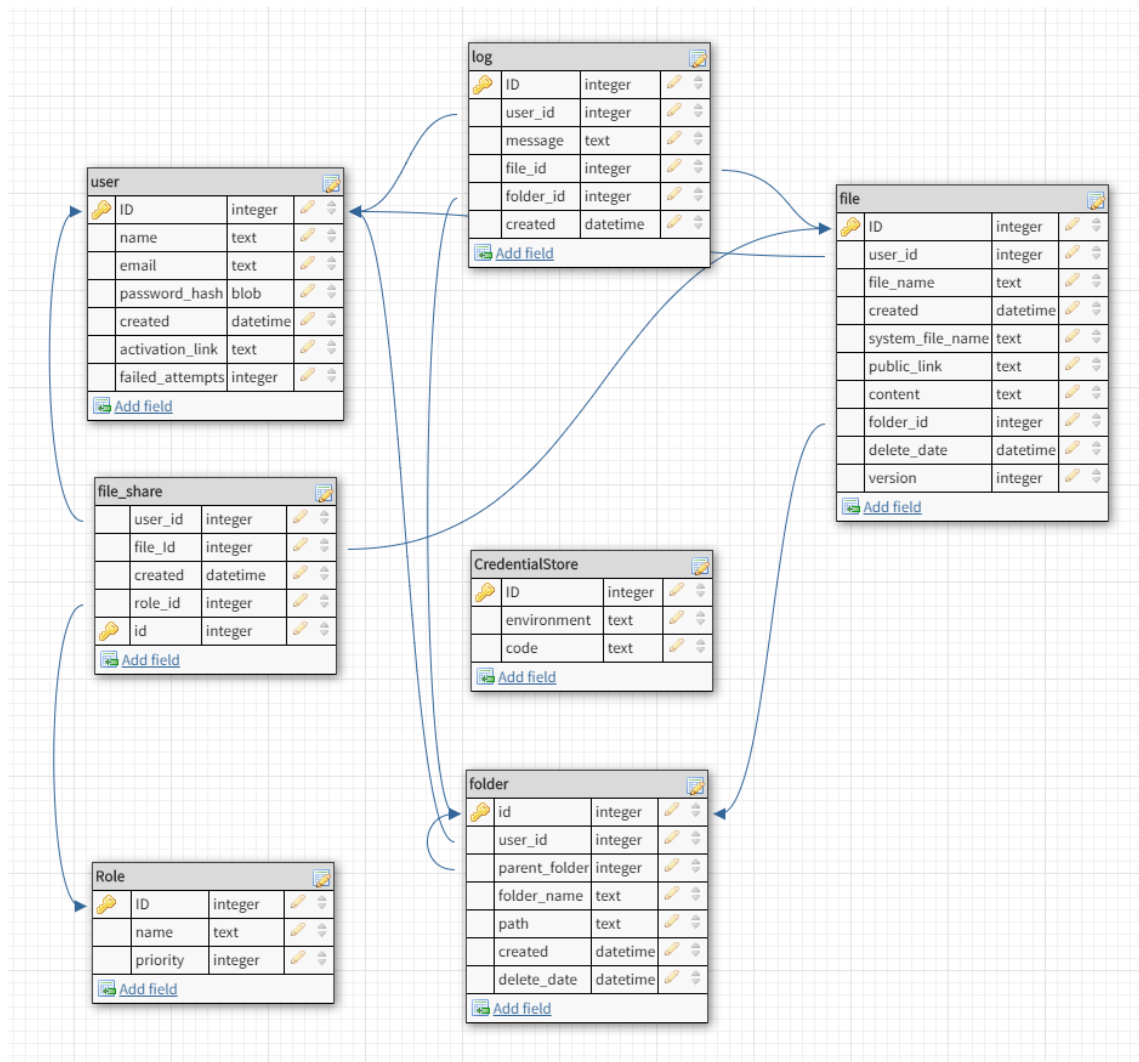
Az eszköz legfontosabb fizikai paraméterei:

- Quad Core 1.2GHz BCM2837 CPU - 64bit
- 1GB SDRAM
- 16GB-os C10-es microSD kártya (későbbiekben bővíthető külső HDD-vel)
- Raspbian OS

3 Tervezés és fejlesztés

3.1 Adatbázis

A rendszer helyes működéséhez elengedhetetlen az adatok perzisztálása. Az rendszer adatbázis struktúrája és annak kialakítása kerül bemutatásra az alábbi fejezetben. Továbbá az egyes táblák szerepe és a közöttük lévő kapcsolatok.



3.1.1 User

A felhasználók adatait tárolja, hogy a rendszer azonosítani tudja őket felhasználónév – jelszó páros segítségével. Emellett segít megkülönböztetni a még nem

aktivált felhasználókat az aktiváltaktól és felfüggesztettektől, továbbá tárolja a rontott jelszavak számát. A jelszó hash-elve kerül eltárolásra növelve ezzel a rendszer biztonságát.

3.1.2 File

A File tábla a felhasználók által feltöltött fájlokról tárol információkat.

3.1.3 Folder

3.1.4 File share

3.1.5 Role

3.1.6 Credential store

3.1.7 Log

4 Irodalomjegyzék

[1]

[2]

[3]

Függelék