**Fejlesztői Dokumentáció**

**Program neve: Határidőnapló Felhasználókezeléssel**

**Áttekintés**

Ez a Python program egy alapvető felhasználókezelési rendszert és határidőnapló-alkalmazást valósít meg. A program képes felhasználói regisztrációra, bejelentkezésre, határidők hozzáadására, törlésére, listázására és készre állítására. Az adatok fájlokban kerülnek tárolásra, biztosítva a perzisztenciát.

**Fő Funkciók**

**1. Felhasználókezelés**

* **Regisztráció:**
  + Felhasználónevet, e-mail címet és jelszót kér a program.
  + Ellenőrzi az egyediségüket (felhasználónév és e-mail).
  + Az adatokat egy users.py fájlba menti Python szótár formátumban.
* **Bejelentkezés:**
  + Ellenőrzi a felhasználónevet és jelszót.
  + Speciális „Admin” bejelentkezés, amely megjeleníti az összes regisztrált felhasználó adatait.

**2. Határidőnapló-kezelés**

* **Adatstruktúra:**
  + A napló adatai JSON fájlban tárolódnak, a fájlnév formátuma: hataridonaplo\_{username}.json.
  + Az adatstruktúra egy szótár, amely napok szerint tárolja az óránkénti bejegyzéseket.
* **Funkciók:**
  + Új bejegyzések hozzáadása.
  + Bejegyzések törlése.
  + Bejegyzések listázása (pl. ABC sorrendben vagy elvégzendő feladatok szerint).
  + Bejegyzések készre állítása.

**3. Adminisztrátori Funkciók**

* Az Admin felhasználó megtekintheti az összes regisztrált felhasználó adatait.

**A Főbb Osztályok és Függvények**

**1. Load\_users és save\_users**

* **Feladata:** A regisztrált felhasználók adatainak betöltése és mentése.
* **Tárolási formátum:** Python szótár a users.py fájlban.

**2. register**

* **Feladata:** Új felhasználók regisztrálása.
* **Főbb ellenőrzések:**
  + A felhasználónév nem lehet már regisztrált.
  + Az e-mail cím formátuma helyes legyen.
  + A jelszavaknak meg kell egyezniük.

**3. HataridonaploApp osztály**

* **Feladata:** Felhasználóspecifikus határidőnapló-kezelés.

**Főbb metódusok:**

* \_\_init\_\_: Inicializálja a felhasználó adatait és betölti a meglévő naplót.
* adatok\_betoltes: Betölti a JSON fájlból a napló adatait.
* adat\_mentes: Elmenti az aktuális naplót JSON fájlba.
* hataridonaplo\_kiir: Megjeleníti a napló tartalmát heti bontásban.
* uj\_bejegyzes: Új bejegyzést ad hozzá egy adott naphoz és időponthoz.
* torles: Törli az adott nap és időpont bejegyzését.
* keszre\_allitas: Egy bejegyzést készre állít.
* feladatok\_listazasa: Feladatokat listáz különböző szűrők alapján (pl. ABC sorrend).
* fut: A felhasználói menü kezelése.

**4. login**

* **Feladata:** Felhasználók hitelesítése és beléptetése.
* **Speciális funkció:** Admin jogosultsággal rendelkező felhasználó esetén külön menü jelenik meg.

**5. main**

* **Feladata:** A program főmenüjének kezelése.
* **Lehetőségek:**
  + Regisztráció.
  + Bejelentkezés.
  + Kilépés.

**Továbbfejlesztési Lehetőségek**

1. **Biztonság:**
   * Jelszó hash-elése (pl. bcrypt használatával).
   * Felhasználói adatok titkosítása.
2. **Interfész:**
   * Grafikus felhasználói felület (GUI) fejlesztése.
3. **Értesítések:**
   * Értesítések hozzáadása a közelgő határidők esetén.
4. **Tesztelhetőség:**
   * Automata tesztek írása az unittest vagy pytest könyvtárral.
5. **További funkciók:**
   * Bejegyzések ismétlődése.
   * Részletesebb szűrési lehetőségek.

**Követelmények**

* **Python verzió:** 3.6 vagy újabb
* **Könyvtárak:**
  + json
  + os

**Jogok**

A programot szabadon módosíthatod és terjesztheted, amennyiben megőrzöd a fejlesztői kreditációt.

**Fejlesztők: Szilágyi Zsombor, Kocsis Roland Milán, Fejes Máté Csaba**  
**Dátum:** 2025. január 5.