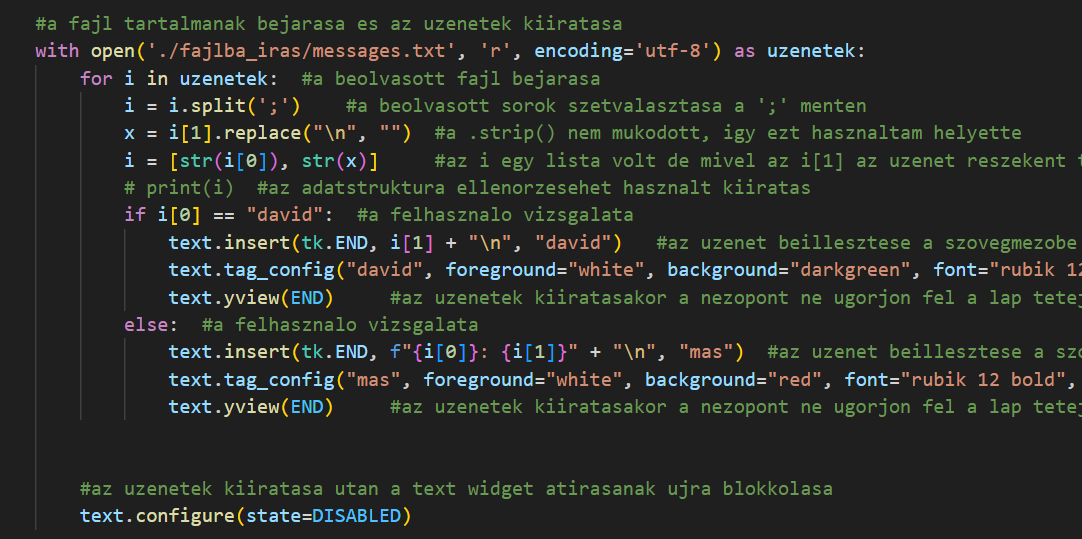
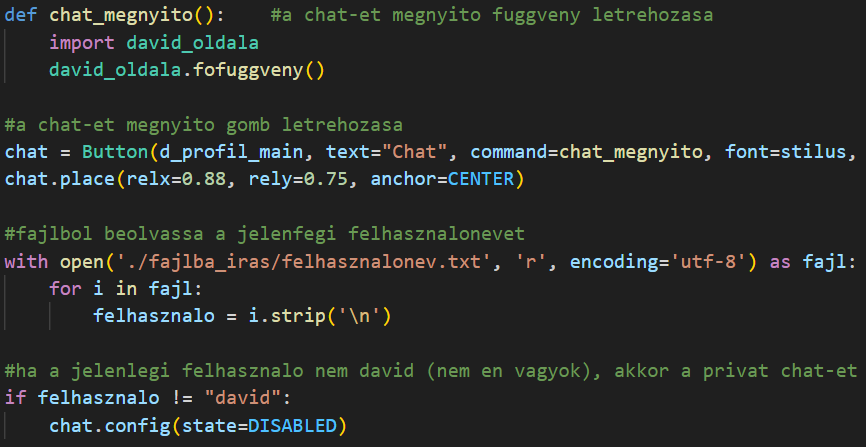
* Program célja: a program funkciói:

1. A felhasználók be tudnak regisztrálni, majd ezt követően lépni a programba, ahol a program mindenkit a ‘rangjának’ megfelelő helyre enged belépni.
2. A felhasználók tudnak egymás között üzeneteket küldeni és fogadni.
3. A felhasználók megtekinthetik a davidgaleria és a patrikgaleria képeit, amelyek között tudnak navigálni két gomb segítségével.

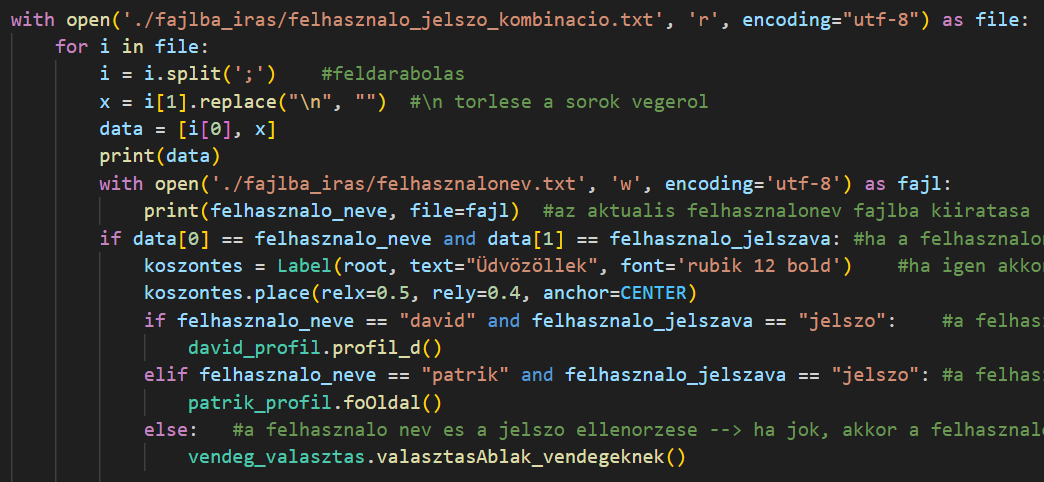
* Megvalósított technológia: A fejlesztés közben sok probléma adódott, de véleményem szerint mindet vagy majdnem mindet sikerült megoldani (például.: belépésnél az új felhasználó felvétele, hogy ne lehessen többször felvenni ugyanazt a felhasználót; a chat funciónál több probléma is volt, meg kellett oldani, hogy a chat szövege a megnyitaskor jól, rendezve jelenjen meg, és , hogy az üzeneteket az elküldés után már ne lehessen átírni a text-box-ban).
* Kód:



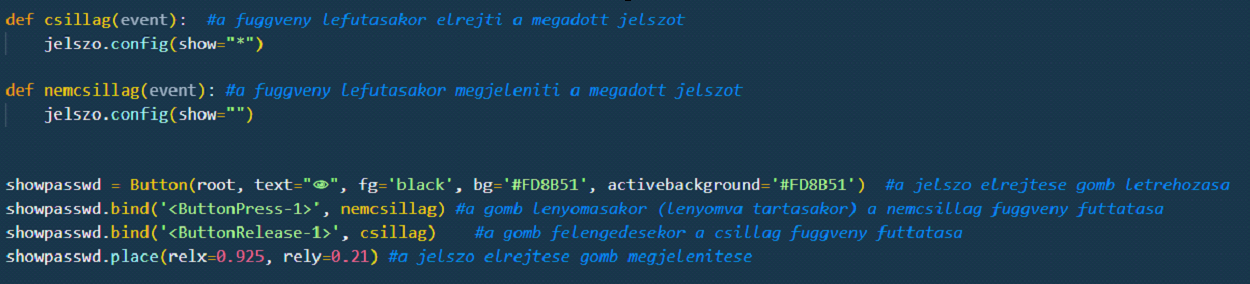
* + Dávid Kódelemzés: Ez a kódrészlet a david\_oldala nevű .py kiterjesztésű állományból származik, pontosabban annak a programnak a kulcsfontosságú része. A programrészlet egy másik felhasználótól származó üzenetet és annak küldőjét jeleníti meg a bal oldalon, a jobb oldalon pegid az aktuális felhasználó által elküldött üzenetek. A program az oldalra rendezés mellett még színekkel is szétválasztja az üzeneteket, mivel a beérkezőket pirossal, a sajátokat pedig zölddel jelenteti meg. A kódrészlet első sora megnyitja a felhasználókat és a hozzájuk tartozó üzeneteket tartalmazó fájlt uzenetek néven. A második sorban megtörténik a változó tartalmának a bejárása , ahol a ciklusváltozó (i) a fájlban (.txt) lévő sorokat veszi fel. A következő lépés a sorok felbontása egy megadott karakter mentén, ami jelen esetben a ‘;’. Miután megtörtént ez a felbontás a cikluson belül az i értéke megváltozik. Az i egy két elemes lista lesz, amelynek első eleme a felhasználó (küldő) neve, a második pedig maga az üzenet (mind a kettő string típusú). A második elem, (az üzenet) végén található egy új sort jelző karakterpáros (\n), amelyet el kell tüntetni a megfelelő kiíratáshoz. Erre a műveletre én a replace() metódust használtam, melynek zárójelei között kell megadni a lecserélendő értéket, és hogy mire cserélje le azt. Az így létrejött szöveget x változónéven, a felhasználónévvel együtt (i) visszatöltöm az i listába. Utána leellenőriztem, hogy a lista első eleme (felhasznalo) david-e. Ha igen akkor a text-box aljara kiirtam az uzenetet jobb oldalra rendezve, zöld színnel kiemelve. Az ezt követő sor arra, szolgál, hogy egy új üzenet kiíratásakor a nézőpont ne ugorjon a text-box tetejére, hanem maradjon annak az alján. Ha a felhasznalo nem david, akkor a program kiírja a küldő felhasználónevét és az üzenetet ‘:’-tal elválasztva, a bal oldalra, zöld színnel kiemelve. Majd néhány sorral ezt követőleg megtörténik a text-widget szerkeszthetőségének átállítása átírhatatlanra (mert ha kihagynánk ezt a lépést, akkor a felhasználó, igaz, hogy csak saját képernyőjén, de át tudná írni az üzeneteket és a felhasználóneveket), mivel azt előlegesen már feloldottuk a képernyő törlésére és az üzenetek kiíratására.



* + Dávid Kódelemzés: Ez a kódrészlet a david\_profil.py fájlból származik, ahol a felhasznalok ennek köszönhetően képesek vagy nem képesek megnyitvi a felhasznaloi profilt ami, a köztem és a patrik közötti üzenetváltásokat tartalmazza. A kép elején egy függvény definiálása látható chat\_megnyito néven. Ebben a függvényben történik meg a david\_oldala-nak importálása és az abban található fofuggveny nevű függvény meghívása. A függvény egy gomb segítségével lehet lefuttatni, amelyet néhány sorral lejjebb chat néven hoztam létre. Majd az ablak méretéhez alkalmazkodó módon (relx, rely) elhelyeztem. A következő lépés a felhasználónév beolvasása volt, amelyet a felhasznalonev.txt-ből olvastam be. Megnyitottam a fájlt fajl néven, ezt követően pedig bejártam (nem volt igazi bejárás mivel, a fajl-nak csak egy eleme volt). Az igy kapott érték egy string volt a végén \n karakterekkel, amelyeket itt a .strip() metódussal tüntettem el (a másik kódrészletben ez a metódus nem működött, így ezt helyettesítettem a replace metódussal). Ezt már csak egy egyszerű vizsgálat követi, amelynek a feladata az, hogy ha a felhasználónév nem dávid, akkor letiltsa a chat gomb működését. Ezzel lehet elérni, hogy akármelyik felhasználó ne érhesse csak úgy el a privát beszélgetésünket. Ez a programrész csak egyszer fut le, az ablak megnyitásakor, és csak a függvény futhat le többször, ha a chat gomb lenyomásra kerül.



* + Dávid kódelemzés: A fent látható kódrészlet a belepes.py fajlbol szarmazik. A kód nagyvonalakban annyit csinál, hogy beolvassa a felhasználóneveket és a jelszavakat a fájlból, majd összehasonlítja azokat a megadott felhasználónévvel és jelszóval. Ha a páros mindkét tagja megtalálható egy sorban (a fájlon belül), akkor belépteti a felhasználót. Az első sorban a felhasználóneveket és jelszavakat tartalmazó szöveges fájl beolvasása történik, amit ‘beletolt’ a file változóba. Utána egy for ciklussal bejártam a vátozó tartalmát (majdnem az egész kód a for ciklus belsejében van). Az i felveszi minden egyes sor értékeit, ezt még fel kell darabolni a ‘;’ mentén és el kell távolítani belőle a ‘\n’ sortörést, majd vissza kell tolteni a data listába. Utána megtörténik az aktuális felhasználó nevének egy másik fájlba való kiíratása. Ezt követi egy feltétel, ami ellenőrzi, hogy az aktuális felhasználó neve szerel-e a nyilvántartásban. Ha igen akkor a felhasználónévnek megfelelően beengedi. A két fejlesztőt (david, patrik), máshová dobja be (saját profil), mint a vendég felhasználókat (vendég chat).



* + Patrik kódelemzés: Apróság ugyan, de szerintem egy egész jó funkció a jelszó elrejtése/megjelenítése, melyet a belepes.py fájl tartalmaz. Az alábbi kódrészletből látható, hogy ez 1 gomb és 2 függvény segítségével lett megvalósítva. A csillag függvényben a jelszo.config(show=”\*”) sor (ahogy a kommentből is látható) a függvény lefutásakor kicsillagozza a beírt jelszót. Ez a függvény mindig lefut, hiszen a <ButtonRelease-1> eventhez lett rendelve, ami alap esetben folyamatosan fut, amíg a <ButtonPress-1> eventtel meg nem szakítjuk (gombnyomásra), ami így lefuttatja a nemcsillag függvényt, ami megjeleníti az addig csillagozott jelszót.
* Cél: A kitűzött célt szerintem túlléptük, mivel mindent sikerült kivitelezni, amit szerettünk volna. Működnek az alapfunkciók, mint a belépés, a chat, és galériák, sőt még egy plusz funkcióval is kibővítettük a programot, ami az új felhasználók felvétele. A felhasználó ha belépésekor egy hibaüzenetet kap vissza, amiben a program jelzi neki, hogy még beregisztrálva vagy nem jól adta meg a felhasználónévből és jelszóból álló párost. Ezt az üzenetet látva a felhasználó a Regisztráció gombra kattintva regisztrálhat egy felhasználónevet és jelszót megadva. A program ellenőrzi, hogy a felhasználónév már nem foglalt-e, és ha nem akkor elmenti a párost és közli a felhasználóval, hogy most már a Belépés fülön be tud lépni a programba (a jelszó nincsen ellenőrizve mivel több, eltérő felhasználónévhez is tartozhat ugyanaz a jelszó). Bár véleményem minden előre eltervezettet sikerült megvalósítani, lenne még néhány új, összetettebb funkció, amivel jobbá tehetnénk az alkalmazásunkat (például.: videókat is képes legyen megjeleníteni).
* Továbbfejlesztési lehetőségek: Szerintem egy programon mindig van valami amit lehetne még tovább fejleszteni, de van egy ésszerű határ. A mi programunkon is bőven van még mit fejleszteni, de az idő és a kreativitás hiányában csak eddig jutottunk. Saját véleményem szerint már sok funkció megfelelően működik, de az alkalmazás design-ján, kinézetén még nagyon sokat lehetne javítani. A galéria bővíthetőségét is lehetne még fejleszteni, mert jelenleg csak előre meghatározott számú képpel működik, az új kép felvétele esetén nemcsak új változóval kell felvenni a képet, hanem még a cikluson is változtatni kell egy keveset. Továbbá amit még nagyon fontos lenne megoldani, az a felhasználónevekhez tartozó jelszavak titkosítása. Ugyanis így aki hozzáfér a felhasznalo\_jelszo\_kombinacio nevű .txt állományhoz az könnyedén el tudja olvasni vagy át tudja másolni a jelszavakat, amelyeket elvileg még a készítők sem láthatnának.
* Illusztráció:



