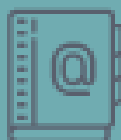


# VÁLAS(S)Z



**Készítette:** Sárkány Illés-Sámuel

Mikló József-Péter

Bartha Róbert-Dezső

**Tantárgy:** Szoftverrendszerek tervezése

**Tanévyszerkezet:** 2022-2023

## Tartalom

|  |    |
|--|----|
| 1.Bevezető .....   | 3  |
| 2.Projekt célja .....                                    | 3  |
| 3.Architektúra diagram .....                             | 3  |
| 4.Követelmények .....                                    | 4  |
| 4.1.Felhasználói követelmények.....                      | 4  |
| 4.1.1.Regisztráció: .....                                | 4  |
| 4.1.2.Felhasználói bejelentkezés: .....                  | 4  |
| 4.1.3.Admin bejelentkezés:.....                          | 4  |
| 4.1.4.Kérdés kategóriák: .....                           | 4  |
| 4.1.5.Kérdések szerkesztés: .....                        | 4  |
| 4.1.6.Kvíz indítása: .....                               | 5  |
| 4.2.Rendszer követelmények.....                          | 5  |
| 4.2.1.Funkcionális követelmények:.....                   | 5  |
| 4.2.2.Nem funkcionális követelmények:.....               | 5  |
| 5.UML diagramok.....                                     | 6  |
| 5.1.Viselkedés diagrammok: .....                         | 6  |
| 5.1.1.Használati eset diagram - Use case diagram.....    | 6  |
| 5.1.2.Aktivitás diagramok - Activity diagrams.....       | 7  |
| 6.Az adatbázis .....                                     | 10 |
| 6.1.A rendszer kezdetleges adatbázisa (SQL).....         | 10 |
| 6.2.A rendszer Végleges FireBase adatbázisa (NoSQL)..... | 11 |
| 7.Wireframe .....  | 12 |
| 8.Útmutató az alkalmazáshoz .....                        | 14 |
| 9.Felhasznált felületek/programozási nyelvek .....       | 15 |
| 9.1.Python:.....   | 15 |
| 9.2.Firebase:.....                                       | 15 |
| 9.3.Tkinter(Tool kit interface): .....                   | 15 |
| 10.Következtetések:.....                                 | 15 |
| 11.Tovább fejlesztési lehetőségek: .....                 | 15 |

## 1.Bevezető

Az ember nap mint nap tanul valami újat és ez segít minket jobban megérteni a világot, amiben élünk. Ezt a tudást szerezhettük egy tudás átadásra létrehozott intézményben, de a saját vagy mások tapasztalataiból is. Sok ember szereti felmérni a tudását, ezért is vannak a vetélkedők, feladványok de a kvízzjátékok is.

Napjainkban megannyi kvízzjáték található számtalan tematikában. Nem mondhatjuk, hogy valami újat találtunk fel, azonban elkészítettük a saját verzióinkat. Egy olyan kvízzjátékot, ahol az alapvető kategóriák mellett, mint a Sport vagy a Tudományok a játékosok a közösségi („Felhasználói”) kategóriában olyan kérdésekre tudnak válaszolni, amit ők maguk vagy más játékosok szerkesztettek.

Sok utána böngészés és új dolgok megtanulása árán fejlődött a játék a jelenlegi állapotára. A későbbiekben sok újdonsággal bővíthető, de mindezek mellett nagyon tanulságos volt rájönni, kitapasztalni, hogy hogyan is működik egy kvízzjáték program-szerkezete.

A projekt elején egyikünk sem tudta, hogy hogyan kell Pythonban programozni, de azt sem, hogy hogyan lehet egy FireBase adatbázist létrehozni és összekötni egy Python projekttel, viszont végül sok mindent megtanultunk ezekből.

Egy idézet a tanulásról:

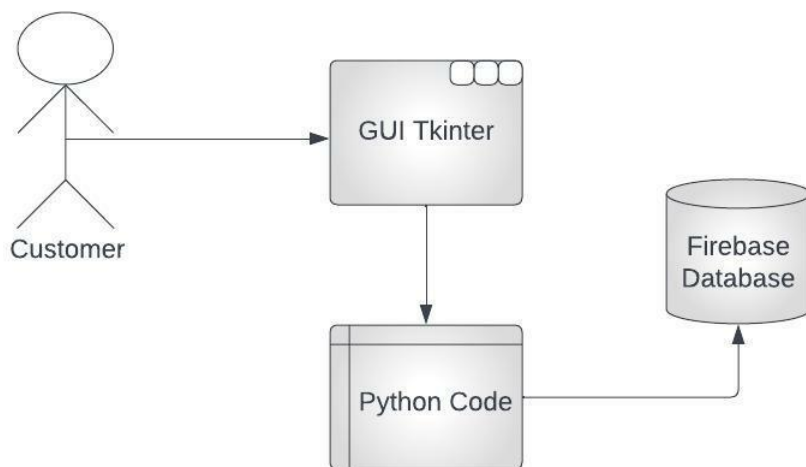
„A tanulásnál mindig az jelenti a legnagyobb akadályt, amiről az ember azt hiszi, hogy nem képes megtanulni.” - Ruth Werner

## 2.Projekt célja

Egy alkalmazás, amely segítségével az emberek fel tudják mérni a tudásukat különböző kategóriájú kérdésekben, illetve ha van egy jó kérdés ötletük akkor ezt hozzá is tudják adni az alkalmazáshoz.

Minden felhasználó első sorban kell regisztráljon és ezzel létrejön egy fiók, ahová a pontszámait gyűjti az illető felhasználónak illetve, hogy hány teljesített kvíze van. A projekt bővíthető a felhasználó által létrehozott kvízekkel is.

## 3.Architektúra diagram



3.1 ábra – Architektúra diagram

Az architektúra diagram bemutatja a felhasznált komponenseket.

## 4.Követelmények

### 4.1.Felhasználói követelmények

4.1.1.Regisztráció: A felhasználó kitölti a regisztrációs ablakban lévő mezőket

- a) Validálási hiba esetén hibaüzeneteket jelenít meg, amely figyelmezteti és segíti a felhasználót, hogy helyesen töltsse ki a mezőket
- b) Sikeres regisztráció esetén egy üzenetet kap a felhasználó, hogy sikeresen regisztrált és a megadott e-mail címen is értesítve lesz.

4.1.2.Felhasználói bejelentkezés: egy létező felhasználó adataival a rendszer beléptet a felhasználó fiókjába. Itt látja az eddigi pontszámait, tud kérdéseket létrehozni és játékot indítani.

4.1.3.Admin bejelentkezés: az admin képes játszani és minden kategóriában kérdéseket létrehozni.

4.1.4.Kérdés kategóriák: Minden kérdés egy-egy különböző kategóriába sorolható

4.1.5.Kérdések szerkesztés: Az admin kérdés létrehozása esetén az admin kiválasztja a kérdésnek megfelelő kategóriát. A felhasználó kérdés létrehozása esetén a felhasználó kérdések egy saját kategóriába kerülnek.

4.1.6.Kvíz indítása: Játék indításakor ki lehet választani kategóriákat, hogy milyen kategóriájú kérdéseket szeretnénk a játékba. Játék végén a játékos megnézheti a helyes és az általa helyesnek gondolt választ a kérdésekre, illetve az elért pontszámát a teljesített kvízen.

## 4.2.Rendszer követelmények

### 4.2.1.Funkcionális követelmények:

Az alkalmazás indításakor a kezdő oldal megjelenik, ahonnan admin vagy felhasználóként lehet bejelentkezni. Bejelentkezés után felhasználó módban a felhasználó tud kvízet indítani vagy kérdéseket létrehozni, „Felhasználói” kategóriában. Admin módban lehet kvízet indítani, kérdéseket létrehozni minden kategóriában.

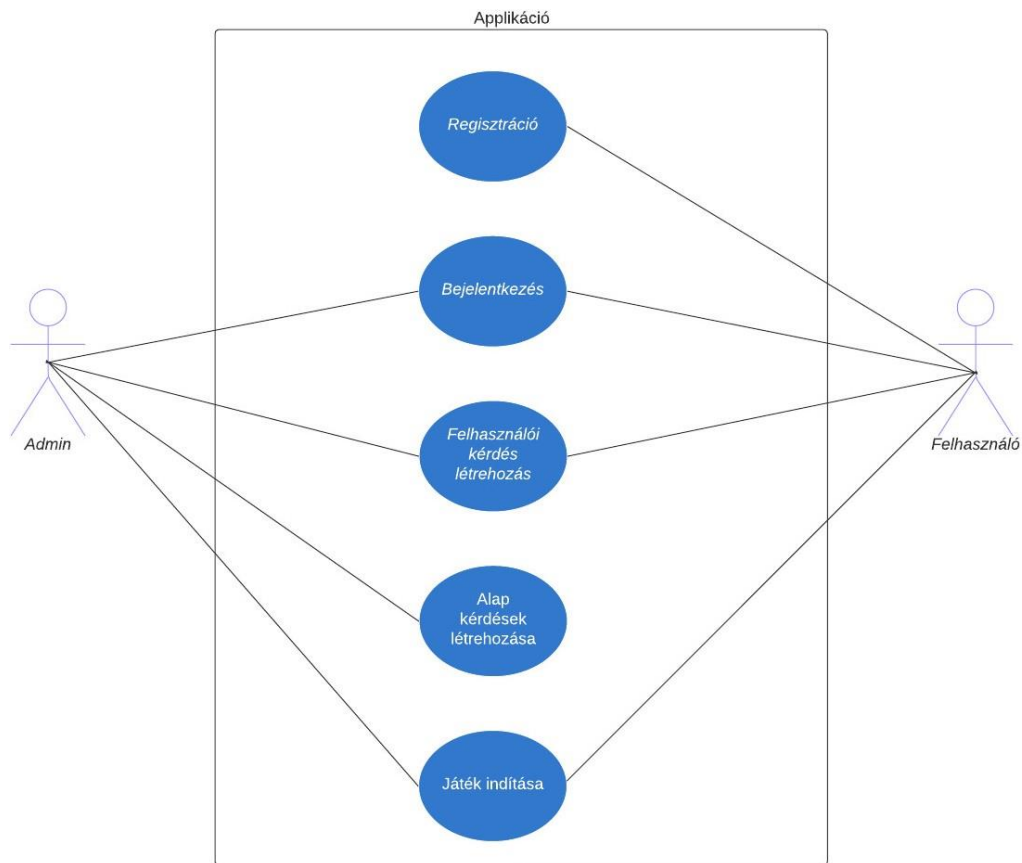
### 4.2.2.Nem funkcionális követelmények:

- Számítógép
- Python 3.9
- Windows: 10/11
- Internetkapcsolat
- Szabad tárhely: 50 Mb

## 5.UML diagrammok

### 5.1.Viselkedés diagrammok:

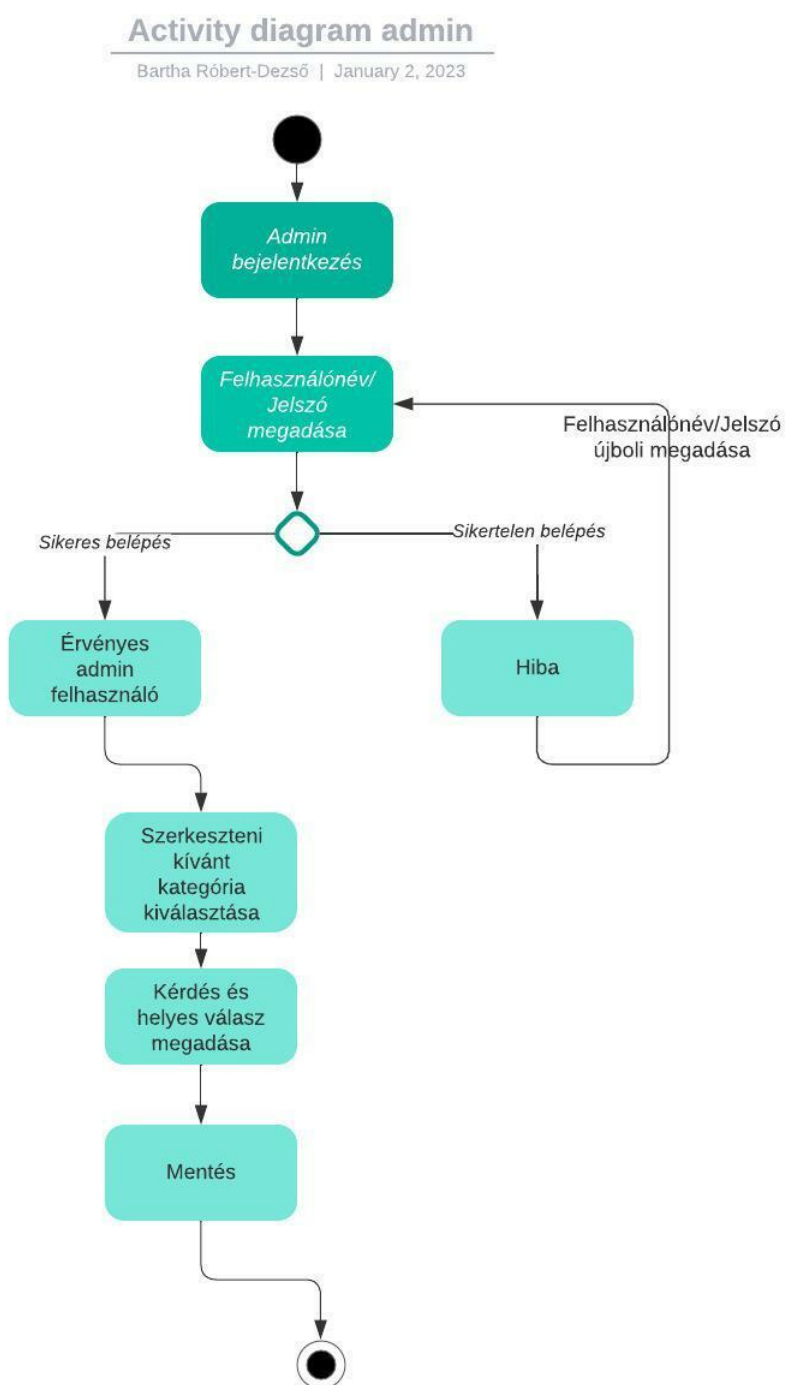
#### 5.1.1.Használati eset diagram - Use case diagram



5.1.1 ábra – Használati eset diagram

Célja a követelmény rögzítése. A projekt célja, hogy kialakítsunk egy olyan számítógépes applikációt, amin keresztül a felhasználó regisztrálás és bejelentkezés után képes játékot indítani vagy felhasználói kérdéseket létrehozni. Az adminnak nincs szüksége regisztrációra és ezen kívül ő alap kérdéseket is tud létrehozni.

## 5.1.2. Aktivitás diagramok - Activity diagrams

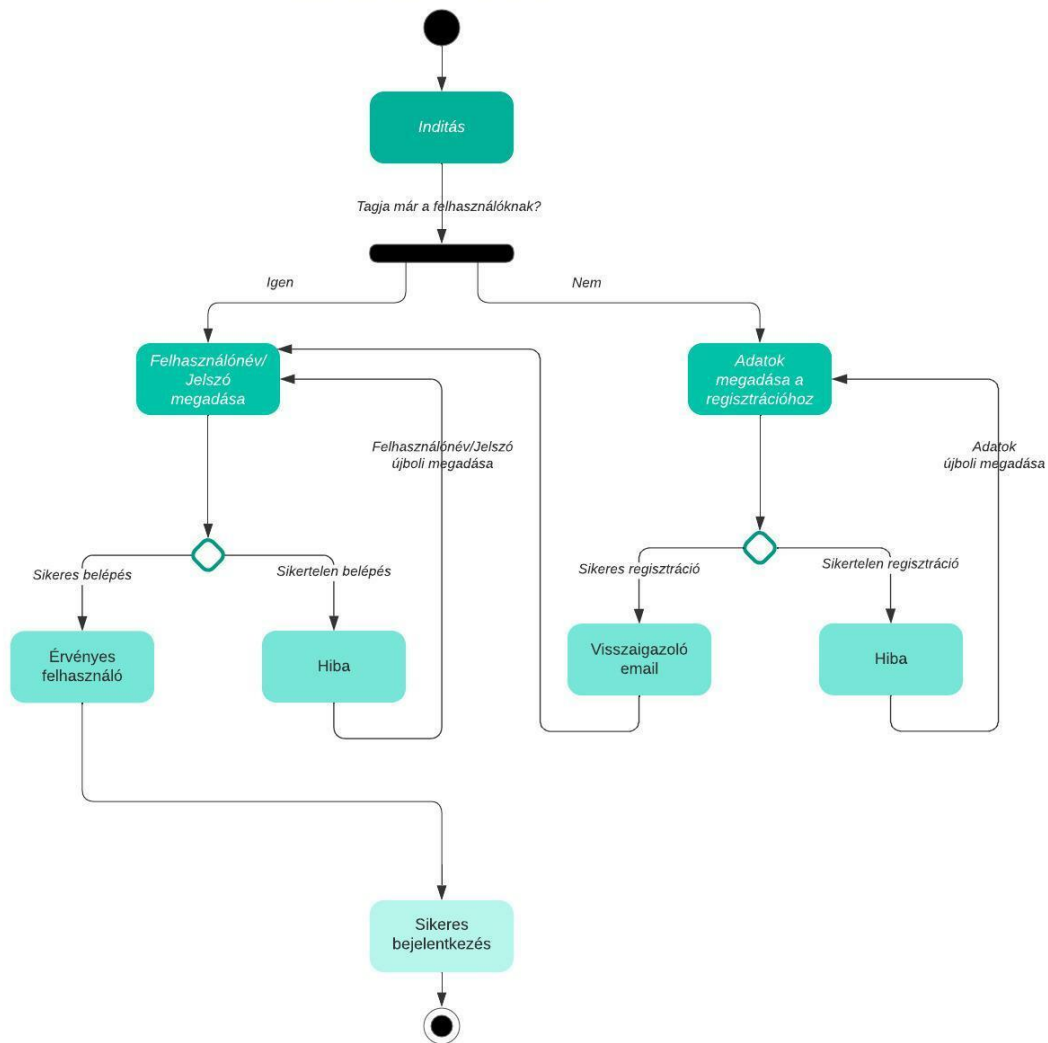


5.1.2.1 ábra – Aktivitás diagram admin felhasználó módban

Az admin felhasználó létrehozhat alapkérdéseket az alkalmazáshoz, amelyeket majd a felhasználók meg tudnak válaszolni.

### Activity diagram authentication

Bartha Róbert-Dezső | December 31, 2022



5.1.2.2 ábra – Aktivitás diagram bejelentkezésre/regisztrációra

A fenti ábrán a bejelentkezés illetve regisztráció aktivitás diagram diagramja látható.

Első esetben ha már regisztrált felhasználó akkor meg kell adnia a felhasználónevet illetve ehhez tartozó jelszót, ha netán elírta valamelyiket akkor hibaüzenetet fog kapni.

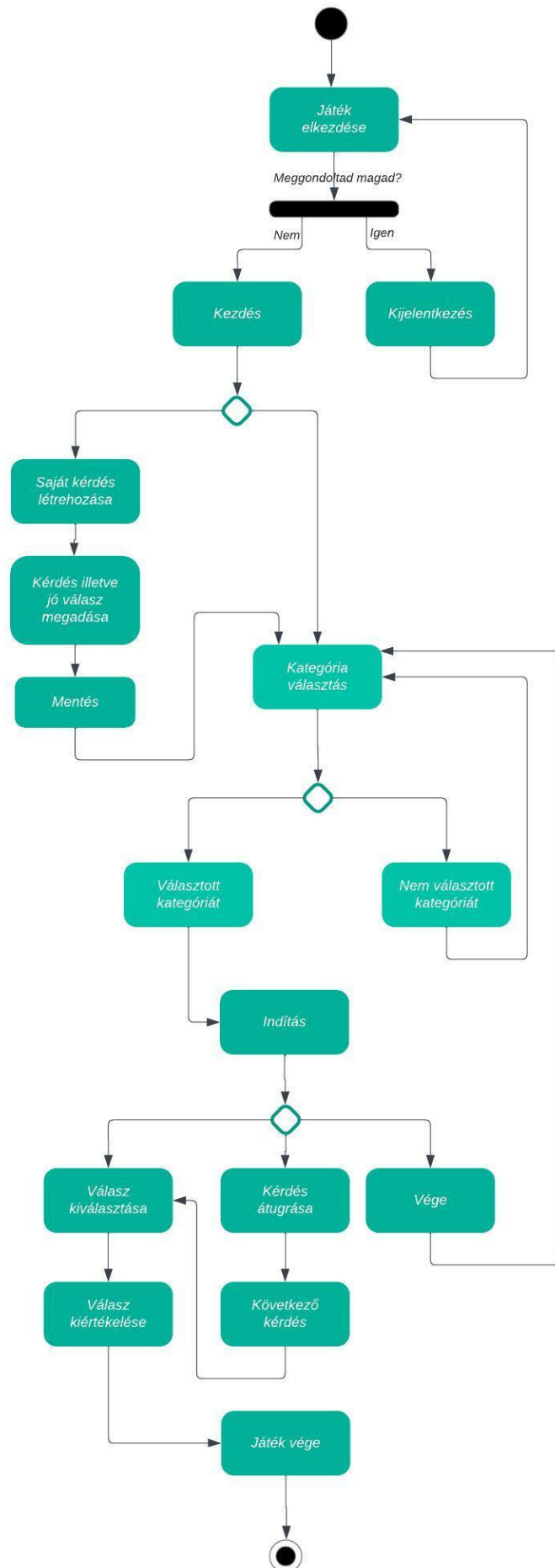
Második esetben ha még nem regisztrált felhasználó akkor regisztrálnia kell.

A regisztrációhoz szükséges adatok megadásával: vezetéknév, keresztnév, felhasználónév, jelszó, email. Ha sikeresen megadta az adatait akkor egy visszaigazoló email-t kap és bejelentkezhet.



## Activity diagram play

Bartha Róbert-Dezső | January 2, 2023

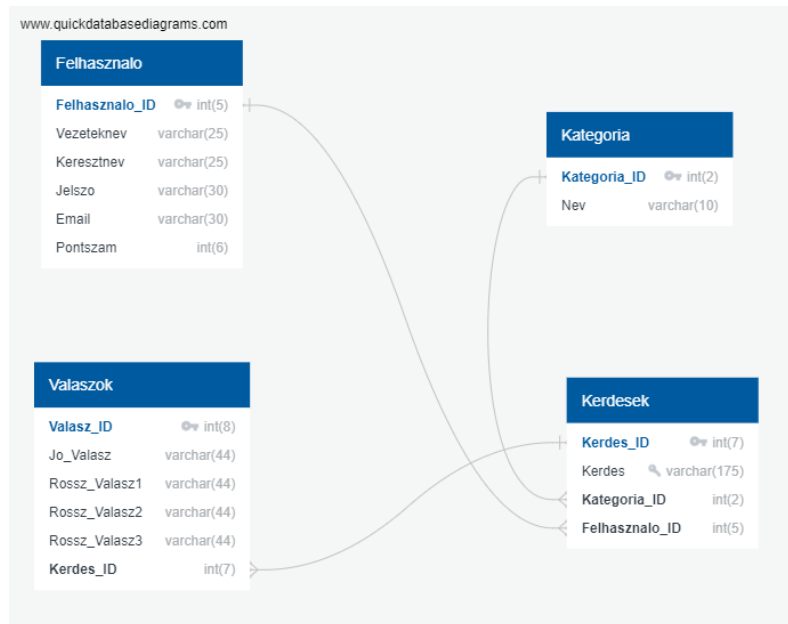


### 5.1.2.3 ábra – Aktivitás diagram kvízhez

A fenti diagramon látható maga a kvíz része. Kezdshez ki kell választani egy kategóriát, ha netán ez nem sikerült volna akkor nem lehet továbblépni és hibaüzenetet kap. Ha sikerült akkor megkapja az első kérdést és válaszolni tud rá illetve tovább is tud lépni ha esetlegesen nem tudja a választ az adott kérdésre, de azt a kérdést meg kell válaszolnia a kvíz végén. Saját kérdést is hozzá tud adni minden felhasználó, be kell írnia a kérdést és ki kell választania a jó választ ezzel elmenti a kérdést az adatbázisba.

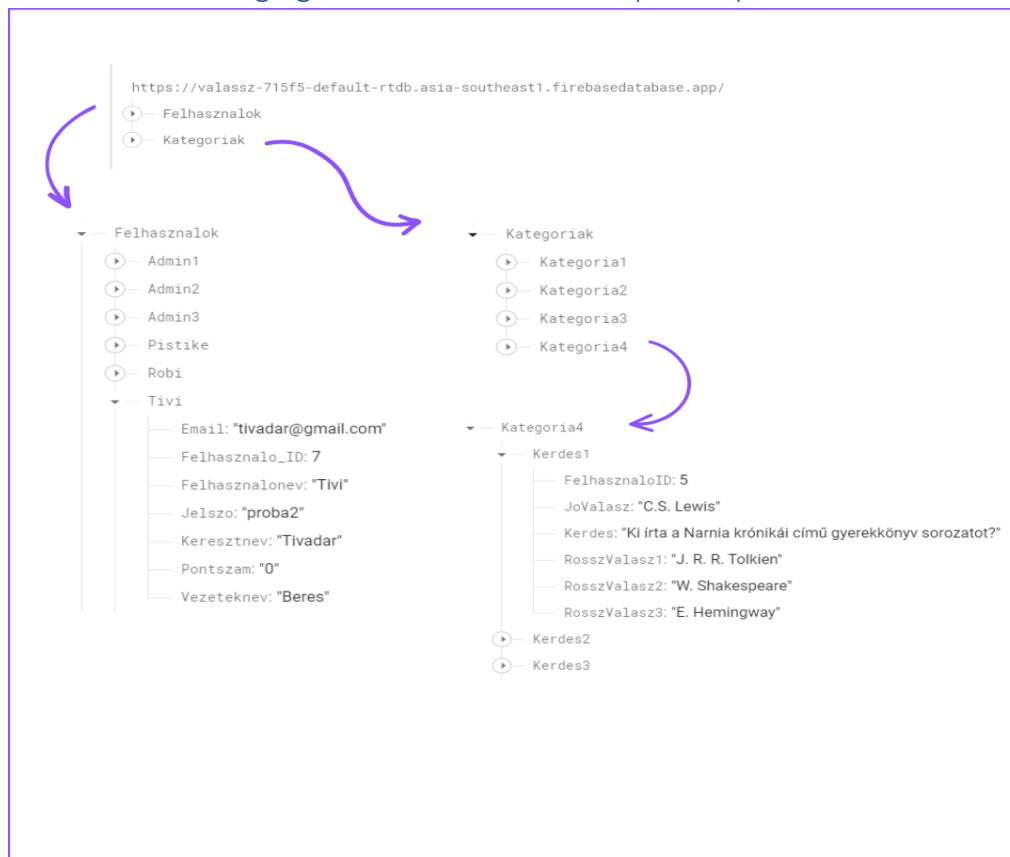
## 6.Az adatbázis

### 6.1.A rendszer kezdetleges adatbázisa (SQL)



6.1 ábra – Kezdetleges SQL adatbázis terv

## 6.2.A rendszer Végleges FireBase adatbázisa (NoSQL)



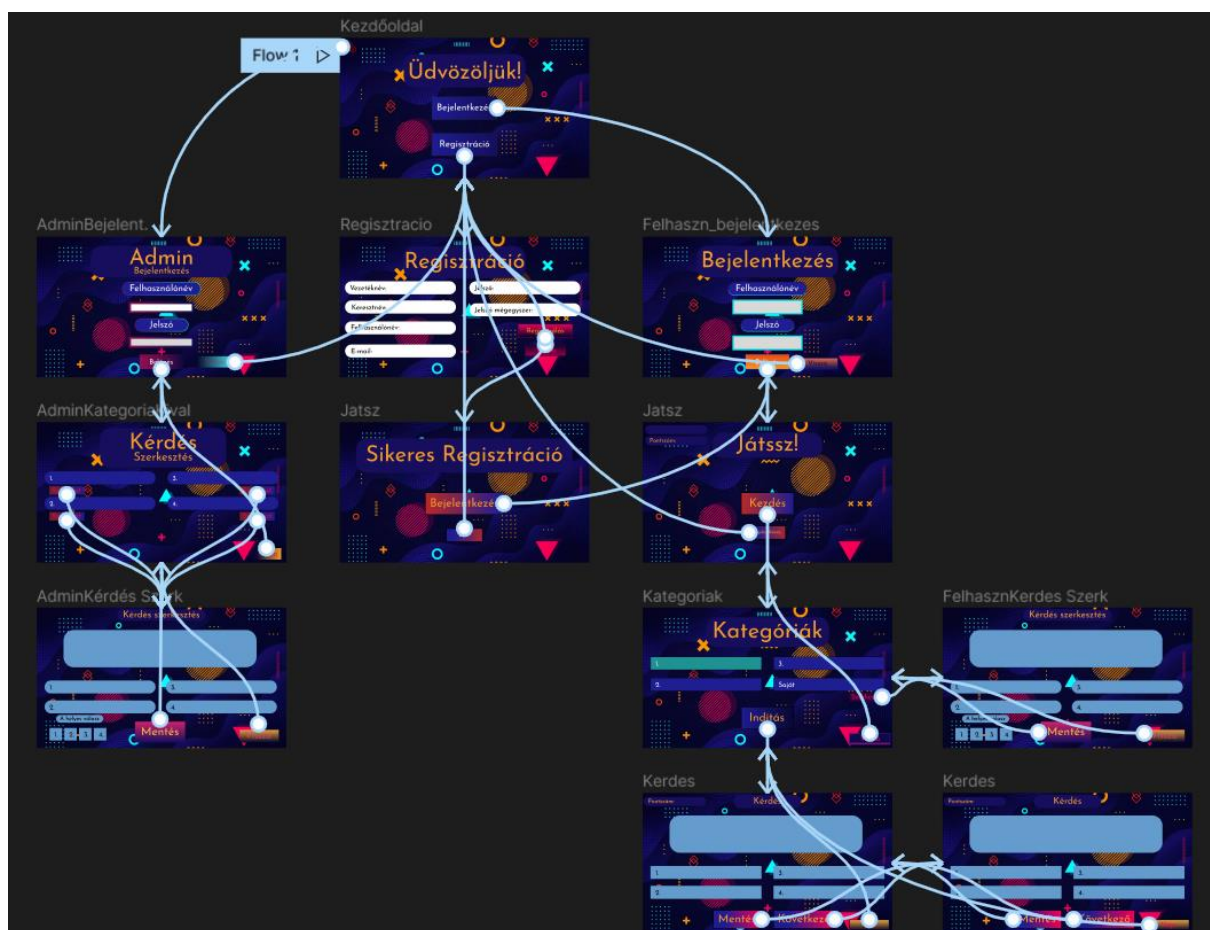
**6.2 ábra – Végleges NoSQL adatbázis séma**

Kezdetben SQL típusú adatbázisban kezdtünk el gondolkodni és ennek megfelelően készítettünk el egy tervet, amely az 6.1. ábrán látható. Azonban rájöttünk, hogy könnyebb lesz az adatbázis összekötése a kóddal, hogyha a FireBase-t használjuk, így elkezdtek az SQL-ben megalkotott tervet NoSQL-re áttervezni. A kész adatbázis úgy tűnt, hogy megfelel a célnak, azonban a későbbiekben rájöttünk, hogy a keresés az adatbázis széttagoltsága miatt nagyon lassú. Ennek okán megszületett az 6.2. ábrán látható adatbázis szerkezet, ami gyorsabbnak bizonyult az elődjénél.

## 7. Wireframe

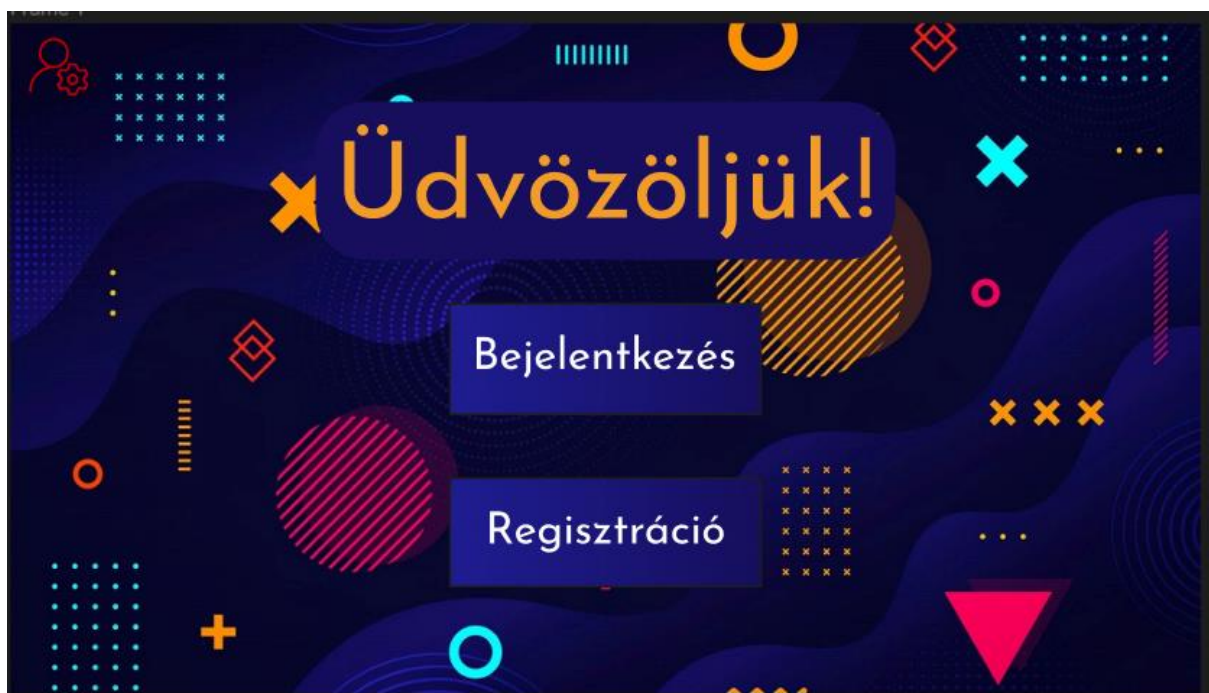
Ezen a linken szimulálható az alkalmazás wireframe-e.

<https://www.figma.com/proto/Jy1CNrLwrh8m5VDGZGRT8q/Untitled?node-id=1%3A2&scaling=min-zoom&page-id=0%3A1&starting-point-node-id=1%3A2&fbclid=IwAR2-fwd-b1OKhPsDchTYcFmeP2fbUOfRjERm2VwmIfTLMczWqTTwGIOSrdc>

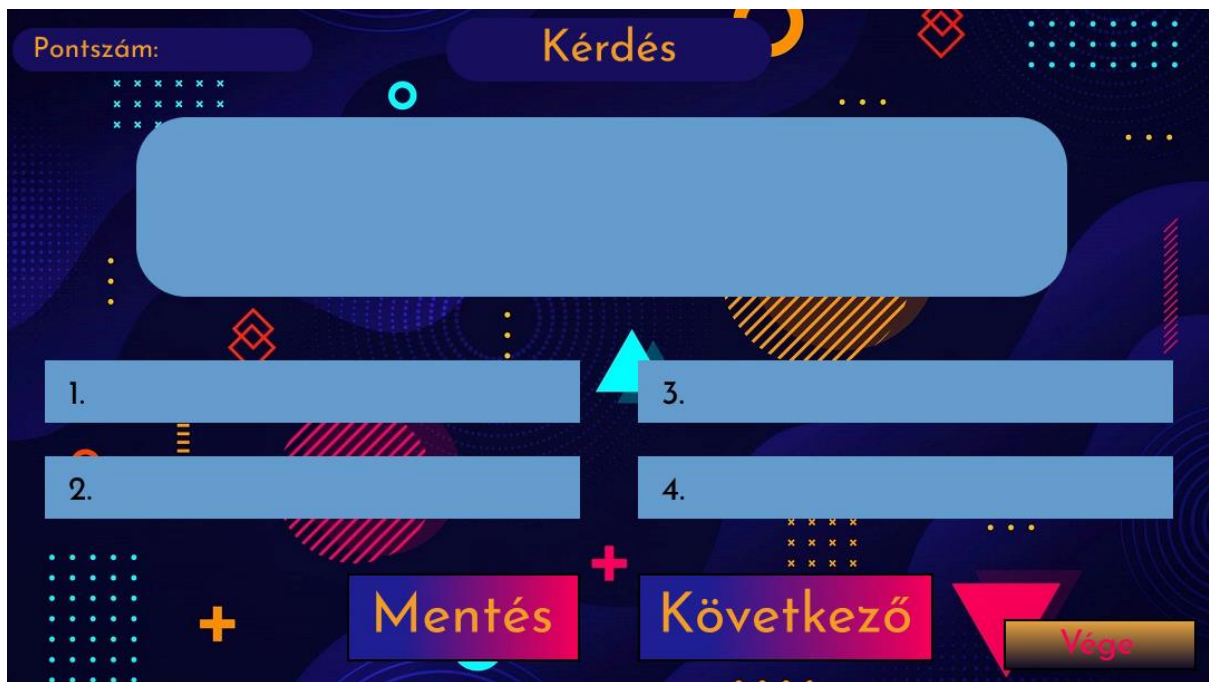


7.1 ábra – Az alkalmazás Wireframe

7.2 ábra – Regisztráció frame



7.3 ábra – Kezdőoldal frame



7.4 ábra – Kérdés megválaszolás frame

## 8. Útmutató az alkalmazáshoz

Az alkalmazás két fő részből tevődik össze: első rész a bejelentkezés/regisztráció a második pedig a maga a kvíz.

Az alkalmazás elindítása után ha már regisztrált akkor a bejelentkezésre kell kattintani, ahol szükséges a bejelentkezéshez a felhasználónév és jelszó megadása, ha hibás valamelyik akkor hibaüzenetet kap a felhasználó. Ha érvényes felhasználónevet és jelszót adott meg akkor kezdődhet a játék.

Ki kell választania egy kategóriát, ha ezt nem teszi meg hibaüzenetet fog kapni és nem tud továbblépni a kérdésmegválaszolásra, de ha sikerült dönteni a kategóriával kapcsolatban akkor az indítás gombra kell kattintania és felugrik az első kérdés, amelyet meg tud válaszolni, ha esetleg nem tudja a választ akkor tovább is léphet a következő kérdésre, de a megválaszolatlan kérdéssel ismét találkozni fog.

Ha nem regisztrált felhasználó még akkor a regisztráció gombra kell kattintania és megadni az ehhez szükséges adatokat: Vezetéknév, keresztnév, felhasználónév, e-mail, jelszó(2x) utána a regisztrálás gombra. Ha érvényes adatokat adott meg akkor egy visszaigazoló email-t kap.

Minden felhasználónak lehetősége van saját kéréseket is hozzáadni a felhasználói kategóriában.

Admin felhasználók is vannak, az adminoknak is szükséges a bejelentkezés érvényes adatokkal. Az adminok feleltek a kezdetleges kérdés adatbázis létrehozásáért.

Amit a kérdésekről érdemes tudni, hogy randomizálva vannak a kérdések és ezeknek a válaszai.



## 9.Felhasznált felületek/programozási nyelvek

- Python-programozási nyelv
- Firebase-Felhasználó adatai/Kérdések/válaszok tárolására alkalmazzuk
- Tkinter-GUI(Grafikus Felhasználói Felület)

### 9.1.Python:

Egy programozási nyelv, amelynek tervezésekor az olvashatóság illetve a programozói munka megkönnyítését helyezi előtérbe a futási sebességgel szemben. Rengeteg kiegészítő könyvtár létezik hozzá, amelynek köszönhetően rendkívül széleskörűen alkalmazható.

### 9.2.Firebase:

A Firebase egy NoSQL típusú adatbázis, tehát az adatokat nem táblákban tárolja el hanem dinamikus sémát használva.A projektben használtuk a valós idejű adatbázist és az autentikációt.

### 9.3.Tkinter(Tool kit interface):

Grafikus könyvtár a Python nyelv számára, amely lehetővé teszi grafikus interfészek létrehozását.

## 10.Következtetések:

A projekt megvalósítása során nagyon sok kihívással kellett szembenéznünk, de a célkitűzéseket sikerült teljesítenünk.

Python-t azért használtuk ebben a projektben, mert egyikünk sem használt idáig és meg akartunk ismerkedni ezzel a programozási nyelvel is illetve azért is, mert alkalmas volt a projekt megvalósítására.

Esetünkben elsőként egy SQL típusú adatbázist szerettünk volna használni, mivel ezzel a típussal ismerkedtünk meg ebben a félévben adatbázisok tantárgy keretein belül. Viszont számunkra ez nem volt annyira optimális és ezért áttértünk egy NoSQL típusú adatbázisra a Firebase-re, mivel több hasznos funkció elérhető (például: adattárolás, azonosítás email-el, stb.) és egyszerűen és gyorsan alkalmazható.

## 11.Tovább fejlesztési lehetőségek:

Az alkalmazásunkon lenne még amit bővíteni/ fejleszteni, mint például:

- Visszajelzések kezelése: hogy tetszik az alkalmazás/hiba valamelyik kérdésben vagy válaszban
- Jelszó csere
- Teljes képernyős nézet
- Kérdések újra szerkesztése illetve törlése
- Statisztika készítése minden felhasználó részére
- Kategória bővítés