

**Készítette:** Albert Benedek

Bányai Krisztián

**Tantárgy:** Szoftverrendszerek tervezése

**Tanévszerkezet:** 2023-2024

**Szak :** Automatika és Alkalmazott Informatika

# Tartalom

# 1.Bevezető

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

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 kvízjáték program-szerkezete.

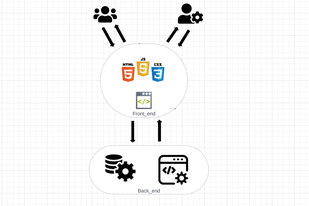
A projekt elején egyikünk sem tudta, hogy hogyan kell Pythonban programozni, de azt sem, hogy hogyan lehet egy FireBase adatbázist létrehozni és összekötni egy Python projekttel, viszont végül sok mindent megtanultunk ezekből.

# 2.Projekt célja

Egy weboldal , amely segítségével az emberek követni tudják a csapat legfrissebb híreit illetve a játékosokkal kapcsolatos újdonságokat. A platform lehetővé tesz egy online áruházat , amely segítségével a szurkolók hozzáférhetnek a csapat által kínált különböző termékekhez.

Vendégként mindenki számára nyitva áll a weboldal, ahol könnyen böngészhetnek és tájékozódhatnak a csapat legfrissebb híreiről . A regisztráció illetve a bejelentkezés csak az online áruház használatához szükséges . Ez azt jelenti, hogy a felhasználók szabadon böngészhetnek a weboldalon, megtekinthetik a csapat legfrissebb tartalmait, de amikor vásárolni szeretnének, regisztrálniuk kell.

# 3.Architektúra diagram



*1. ábra – Architektúra diagram*

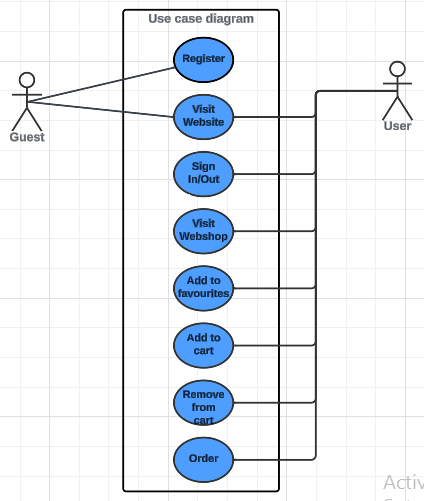
Az architektúra diagram bemutatja a felhasznált komponenseket.   
Nagyon fontos, hogy alaposan kidolgozzuk a szoftverünk architektúráját annak érdekében, hogy a tervezett funkcionalitást minél hatékonyabban megvalósíthassuk.

A webalkalmazást 3 fő alkotóelemre lehet osztani:  
1. Felhasználó  
2. Böngésző  
3.Adatbázis

# 4.Követelmények

1. 1.Felhasználói követelmények . Use Case Diagram.

Célja leírni a modellezendő rendszer és a környezete kapcsolatát.  
A rendszer olyan funkciója vagy funkció halmaza, amely a felhasználó (aktor) céljának elvégzéséhez járul hozzá.

 *2.ábra-Use case diagram*

Amint látjuk két használati esetet tudunk megkülönböztetni , éspedig vendég felhasználót illetve regisztrált felhasználót.  
A vendég felhasználó látogathatja a weboldalt bejelentkezés nélkül is , viszont a regisztrált felhasználó a weboldal keretén belül elérheti az online áruházat is .

1. 2.Rendszer követelmények

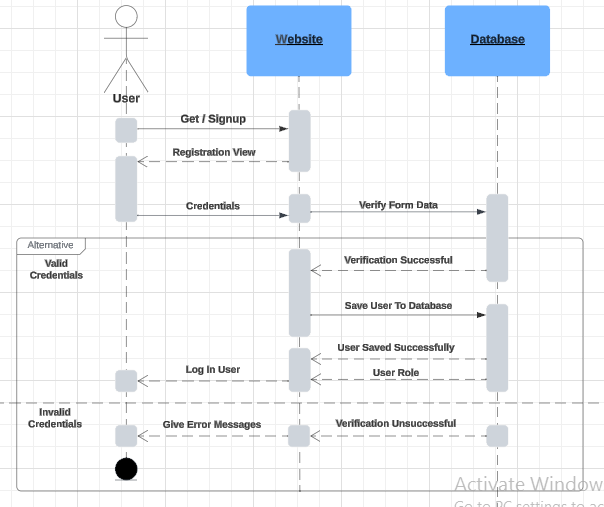
4.2.1.Funkcionális követelmények:

Az weboldal indításakor a kezdő oldal jelenik meg , ahova automatikusan vendégként lépünk be.   
Amint az online aruházat szeretnénk elérni szükségünk lesz regisztrációra illetve bejelentkezésre.

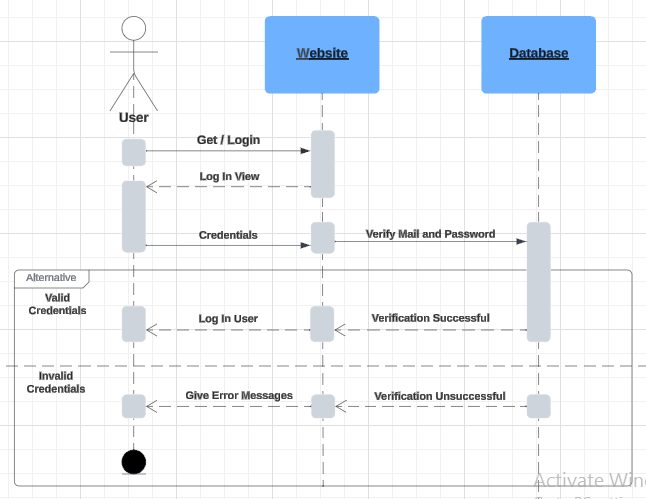
4.2.2.Nem funkcionális követelmények:

* + Számítógép
  + Okostelefon
  + Internetkapcsolat

1. 3.Fontosabb műveletek magyarázata:

Felhasználó regisztrációja  
  
 *3.ábra – Regisztráció(Szekvencia diagram)*

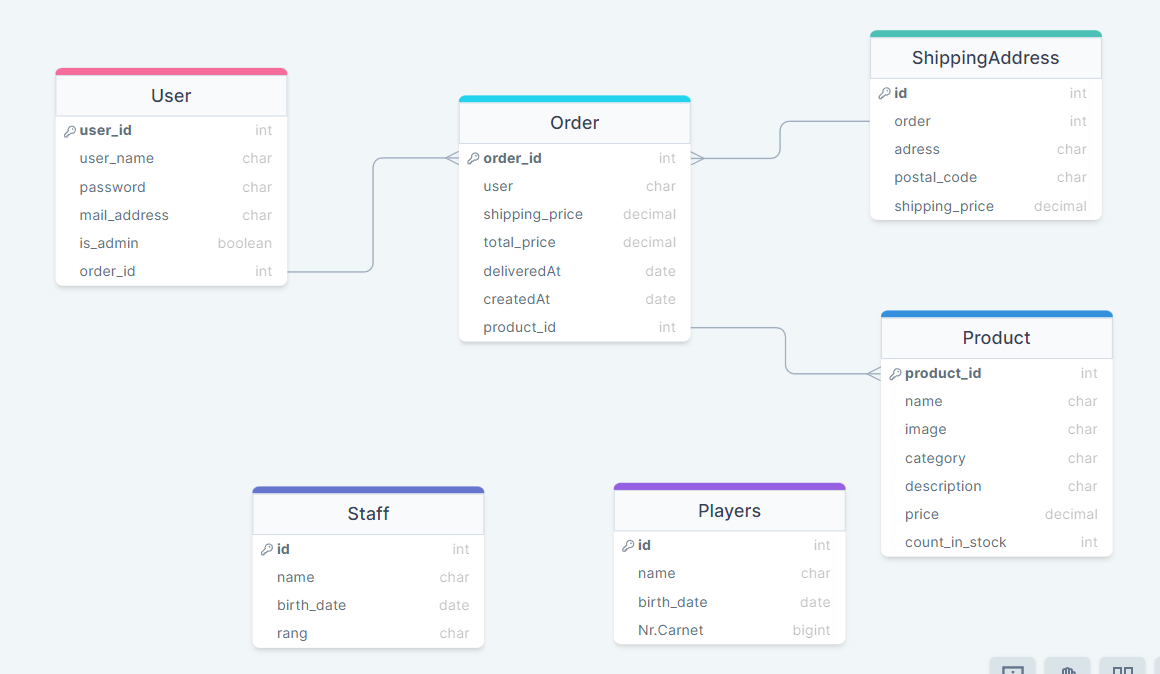
Az előbb említett ábrán látható a felhasználó regisztrációjának folyamata.  
A szükséges adatok megadása után ellenőrzésre kerül a mail cím helyessége , amennyiben ez helytelen , a felhasználó adatai nem kerülnek eltárolásra , és hibaüzenetek formájában tájékoztatjuk a felhasználót.

Felhasználó bejelentkezése  
  
  
 *4.ábra - Bejelentkezés(Szekvencia diagram)*

Az előbb említett ábrán látható a felhasználó regisztrációjának folyamata.  
Hasonlóan történik, mint a felhasználó regisztrációja. Ha helyesek a megadott mail cím és a jelszó , akkor lekérdezzük az adatbázisból a már létező felhasználót.

# 5.Az adatbázis

5.1.A rendszer kezdetleges adatbázisa



*6.1 ábra – Kezdetleges adatbázis terv*

6.2.A rendszer Végleges FireBase adatbázisa

# 6.Wireframe

# 7.Útmutató az alkalmazáshoz

# 8.Felhasznált felületek/programozási nyelvek

9.Következtetések:

10.Tovább fejlesztési lehetőségek: