**Záródolgozat**

Készítették: Macsek Levente, Kapi Richárd, Fegyverneki Tamás

Tartalomjegyzék:

1. **Bevezetés**
2. **Fejlesztési környezettek bemutatása**

2.1 Visual Studio 2022

2.2 Visual Studio Code

2.3 MySQL ­ phpmyadmin

2.4 React

2.5 ASP. NET Web API

2.6 MySql

2.7 Trello

2.8 Github

1. **Tervrajz**
2. **Forrásjegyzék**

**1. Bevezetés**

Ez a feladat egy spanyol szavakat tanító webalkalmazás fejlesztésére vonatkozik, amely a **React**, **Web API**, **MySQL** és **XAMPP** környezettek használatával készül. Az alábbiakban részletesen bemutatom, hogyan zajlik a fejlesztési folyamat, miért választottuk ezeket a környezetteket, és hogyan építhetjük fel az alkalmazást.

A projekt célja egy olyan program kifejlesztése, amely segít a felhasználóknak bővíteni spanyol szókincsüket. Az alkalmazás interaktív módon segíti a felhasználókat a spanyol szavak megtanulásában, azok írásának és jelentésének megértésében.

**2. Fejlesztési környezettek bemutatása**

**2.1 A Visual Studio**

**Visual Studio** egy integrált fejlesztői környezet (IDE), amelyet a Microsoft fejlesztett ki. Számítógépes programok fejlesztésére használják, beleértve webhelyeket, webes alkalmazásokat, webszolgáltatásokat és mobilalkalmazásokat. A Visual Studio a Microsoft szoftverfejlesztési platformjait használja, beleértve a Windows API-t, a Windows Forms-t, a Windows Presentation Foundation-t (WPF), a Microsoft Store-t és a Microsoft Silverlight-ot. Natív kódot és felügyelt kódot is képes előállítani. A Visual Studio tartalmaz egy kódszerkesztőt, amely támogatja az IntelliSense-t (a kódkiegészítő komponenst), valamint a kódrefaktorálást. Az integrált hibakereső forrásszintű hibakeresőként és gépszintű hibakeresőként is működik. Egyéb beépített eszközök közé tartozik a kódprofilozó, a grafikus felhasználói felület- alkalmazások készítésére szolgáló tervező, a webdesigner, az osztálytervező és az adatbázisséma- tervező. Elfogadja azokat a beépülő modulokat, amelyek szinte minden szinten bővítik a funkcionalitást – ideértve a forrásvezérlő rendszerek támogatását (például a Subversion és a Git), valamint új eszközkészletek hozzáadását, például szerkesztőket és vizuális tervezőket a tartományspecifikus nyelvekhez, vagy eszközkészleteket a szoftverfejlesztés egyéb aspektusaihoz. életciklus (például az Azure DevOps- ügyfél: Team Explorer). **Visual Studio 2022** a Microsoft fejlesztői környezete, amely számos hasznos funkciót kínál a webalkalmazások fejlesztéséhez, különösen, ha a back-end kódolásáról van szó.

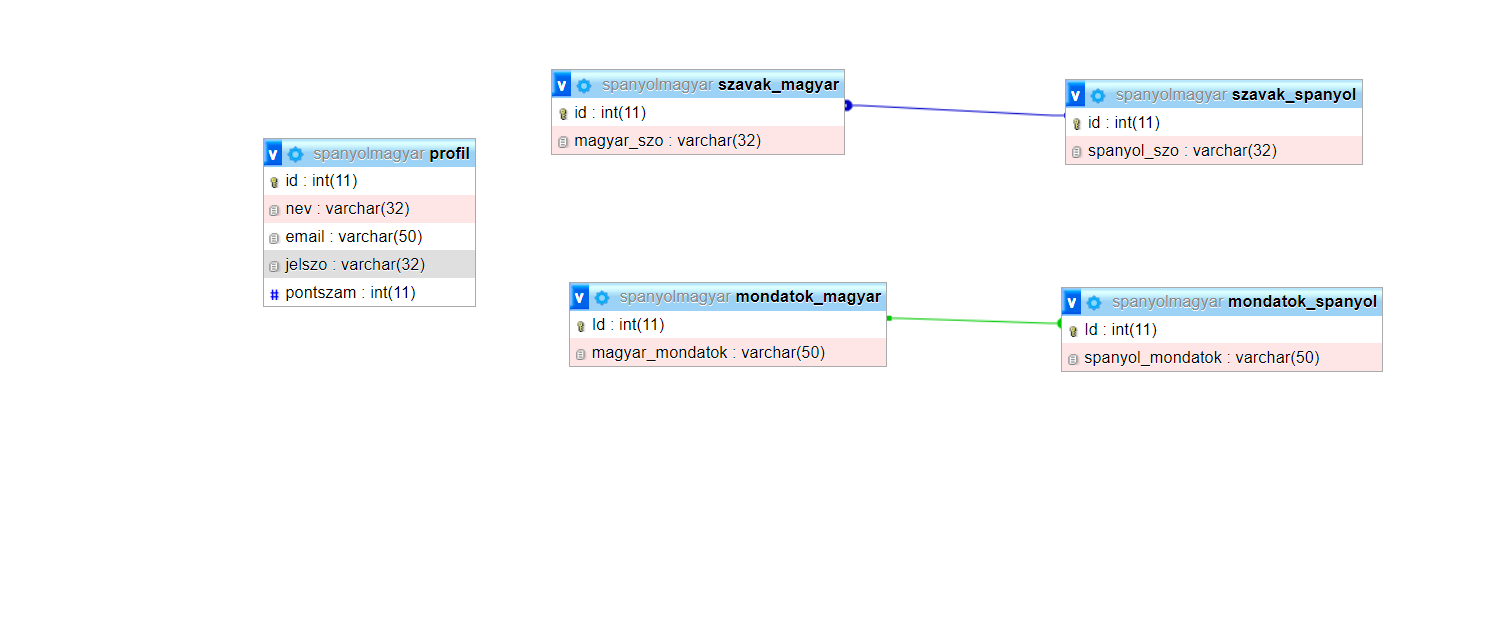
**2.2 Visual Studio Code**

**Visual Studio Code** egy könnyű, nyílt forráskódú fejlesztői környezet, amely kifejezetten frontend fejlesztéshez, például a **React** alapú alkalmazásokhoz ideális. A VS Code ideális választás a JavaScript és React fejlesztéséhez, mivel számos plugin és bővítmény érhető el, amelyek gyorsítják a fejlesztést és segítenek a hibakeresésben. Támogatja a hibakeresőket, valamint beépített Git támogatással rendelkezik, továbbá képes az intelligens kódkiegészítésre az IntelliSense segítségével. A Visual Studio Code-ban a felhasználók megváltoztathatják a kinézetet (témát), megváltoztathatják a szerkesztő gyorsbillentyű-kiosztását, az alapértelmezett beállításokat és még sok egyebet. Támogatja a kiegészítőket, melyek segítségével további funkciók, testreszabási lehetőségek érhetőek el.

**2.3. MySQL ­ phpmyadmin**

**A XAMPP** egy összeállított, platformfüggetlen webszerver­szoftvercsomag. Ez egy integrált rendszert alkot, mely lehetővé teszi webes alkalmazások készítését, tesztelését és futtatását egy csomagban. A XAMPP részeként elérhető a phpMyAdmin is, mely egy ingyenes, nyílt forráskódú adminisztrációs eszköz a MySQL adatbázisok kezelésére. Ez a szoftver könnyen telepíthető és használható, és kiválóan alkalmas az adatbázisok intuitív adminisztrációjára. A phpMyAdmin számos fontos funkciót kínál a felhasználóknak. Képes adatbázisok létrehozására, törlésére és módosítására, valamint táblák kezelésére, beleértve azok létrehozását, törlését és módosítását is. Emellett lehetőséget nyújt az egyes mezők módosítására, hozzáadására vagy törlésére, valamint SQL parancsok futtatására az adatbázisban.

Az adatbázisunkról egy terv:



**2.4 React**

**A React** egy JavaScript-könyvtár felhasználói felületek (UI) rendelésére. A felhasználói felület kis egységekből, például gombokból, szövegekből és képekből épül fel. A React segítségével újra felhasználható, egymásba ágyazható alkatrészekké egyesítheti őket. A webhelyektől a telefonos alkalmazásokig minden, ami a képernyőn látható, összetevőkre bontható. Ebben a fejezetben megtanulhatja a React összetevők létrehozását, testre szabását és feltételes megjelenítését. **A React** alkalmazások a felhasználói felület izolált részeiből, úgynevezett összetevőkből épülnek fel. A React komponens egy JavaScript-függvény, amelyet megszórhat jelöléssel. Az alkotóelemek kicsik lehetnek, mint egy gomb, vagy akár egy teljes oldal méretűek is. Minden React összetevő egy JavaScript függvény, amely tartalmazhat néhány jelölést, amelyet a React megjelenít a böngészőben. A React összetevők a JSX nevű szintaktikai bővítményt használják a jelölés megjelenítésére. A JSX nagyon hasonlít a HTML-re, de egy kicsit szigorúbb, és képes dinamikus információkat megjeleníteni.

**2.5 ASP. NET Web API**

A **Web API** segítségével RESTful szolgáltatásokat hozunk létre, amelyek lehetővé teszik az adatok kommunikációját a front-end (React) és a back-end között. Az API fogja kezelni az adatbázissal való interakciókat, mint például a spanyol szavak lekérdezését és a felhasználói kérések fogadását. A webes API egy alkalmazásprogramozási felület (API) webszerverhez vagy webböngészőhöz. Webfejlesztési koncepcióként kapcsolódhat egy webalkalmazás ügyféloldalához (beleértve a használt webes keretrendszereket is). A szerveroldali webes API egy vagy több nyilvánosan elérhető végpontból áll egy meghatározott kérelem-válasz üzenetrendszerhez, amelyet általában JSON-ban vagy XML- ben fejeznek ki egy HTTP -alapú webszerver segítségével. A szerver API (SAPI) nem tekinthető szerveroldali webes API-nak, kivéve, ha nyilvánosan elérhető egy távoli webalkalmazás által.

**2.6 MySQL**

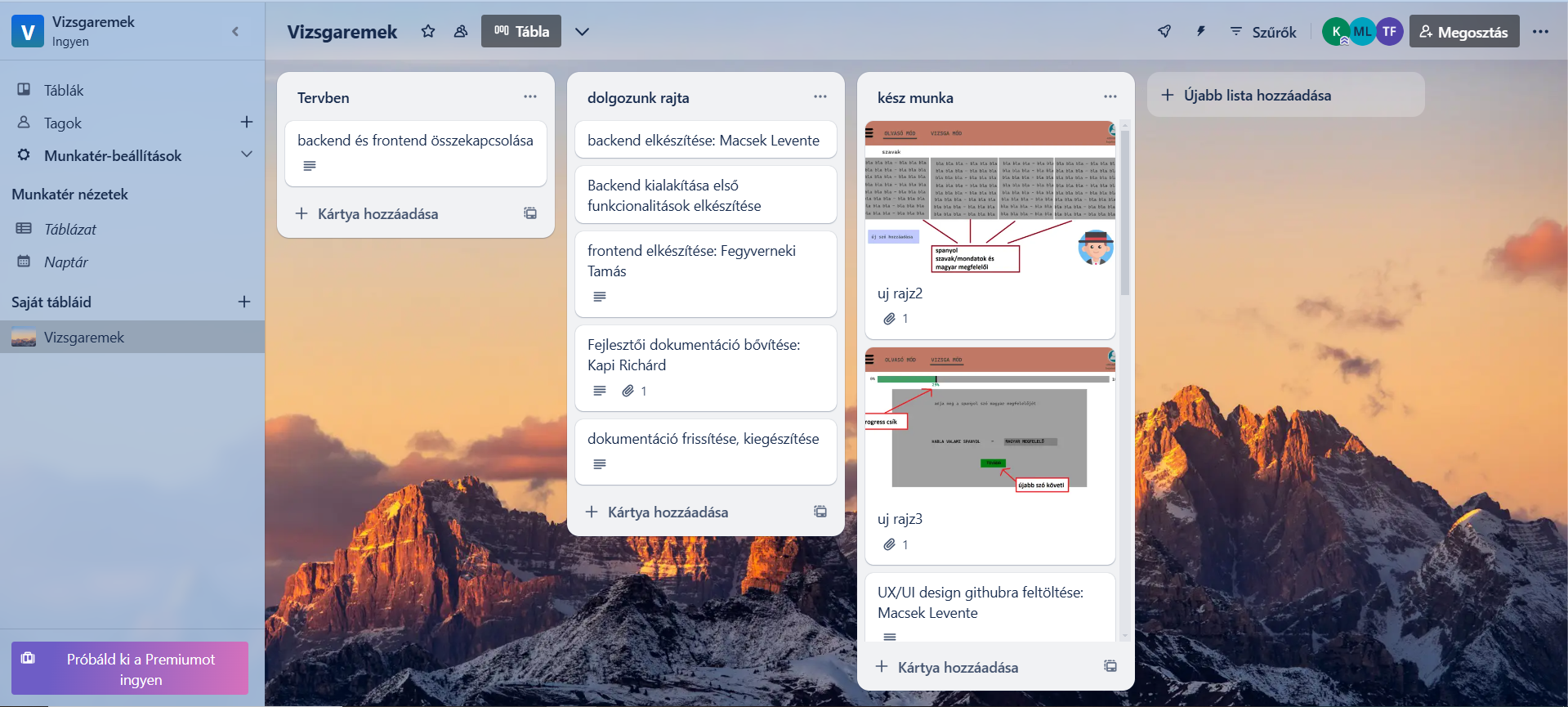
**MySQL** egy népszerű, nyílt forráskódú relációs adatbázis­kezelő rendszer, melyet gyakran használnak dinamikus webhelyek szolgáltatására. Az SQL egy programozók által használt nyelv, amely lehetővé teszi relációs adatbázisok létrehozását, módosítását, adatok lekérdezését és a felhasználói hozzáférés szabályozását. A relációs adatbázisok az adatokat egy vagy több táblába szervezik, ahol az adatok kapcsolatban lehetnek egymással, ami segíti az adatok strukturálását, és kezelését.

**2.7 Trello**

A **Trello** egy vizuális projektmenedzsment eszköz, amely segít a csapatoknak, egyéni felhasználóknak vagy cégeknek a feladatok és projektek nyomon követésében és szervezésében. A rendszer alapja a \*\*kanban módszertan\*\*, ami egy vizuális feladatkezelési rendszer, amely segít átlátni a munka áramlását, és hatékonyan nyomon követni a projekt előrehaladását.

1. Vizuális áttekintés:

- A Trello kártyák és listák segítségével könnyedén követhetjük, hogy mi történik a projektünkben. A kártyák állapotát (pl. "To-Do", "In Progress", "Done") vizuálisan is könnyen követhetjük, ami segít abban, hogy gyorsan átlássuk a feladatok állapotát.



2. Könnyen használható:

- A Trello rendkívül intuitív, nem szükséges mélyebb technikai tudás, hogy elkezdjük használni. A drag-and-drop (húzás és ejtés) funkcióval könnyen áthelyezhetjük a kártyákat, listákat és feladatokat.

3. Rugalmasság:

- Használható különféle típusú projekteknél és célokra: csapatmunkához, személyes feladatkezeléshez, üzleti projektekhez stb. Könnyedén testreszabhatók a boardok, listák és kártyák, hogy illeszkedjenek a felhasználó igényeihez.

4. Kollaboráció:

- A Trello kiválóan alkalmas csapatmunkára. A felhasználók meghívhatók a projektekbe, és közösen dolgozhatnak a feladatokon. Lehetőség van kommentelni a kártyák alatt, fájlokat csatolni, és különböző határidőket beállítani.

5. Részletes feladatok:

- Minden kártya részletes információkat tartalmazhat, például határidőket, címkéket, ellenőrök listáját, mellékleteket, kommenteket, ellenőrzőlistákat stb. Ez lehetővé teszi, hogy mindenki pontosan tudja, mit kell tenni és mikor.

- Projektmenedzsment: Nyomon követhetjük a projektek előrehaladását, kezelhetjük a határidőket és feladatokat.

- Csapatmunka: Könnyen oszthatjuk meg a feladatokat és a projekt részleteit a csapattagokkal.

- Fejlesztési projektek: A szoftverfejlesztési projektek tervezéséhez és a fejlesztési ciklusok követéséhez is kiváló.

**2.8 Github**

**Github** egy webalapú verziókezelő rendszer, amely a Gitverziókezelő eszközt használja. Alapvetően arra szolgál, hogy a fejlesztők és csapatok könnyen kezelhessék a kódot, együttműködhessenek, és biztosítsák a különböző verziók közötti nyomon követést. A Github egy központi platform, amely lehetővé teszi a kód tárolását, megosztását és együttműködést más fejlesztőkkel.

Csapatmunka és Kollaboráció

A Github ideális eszköz csapatok számára. Lehetővé teszi, hogy több fejlesztő párhuzamosan dolgozzon ugyanazon a kódon, miközben biztosítja, hogy a különböző módosítások ne ütközzenek. A pull request és merge funkciók segítenek abban, hogy mindenki munkája egyesüljön, és a csapat tagjai kódot véleményezzenek.

Nyilvános és privát tárolók:

A Github lehetővé teszi a nyilvános és privát tárolók létrehozását. Nyilvános tárolókkal a világ bármely fejlesztője hozzáférhet a kódhoz, így hasznos lehet open-source projekteknél. Privát tárolókat használhatunk, ha bizalmas vagy belső projekteken dolgozunk.

Kódmegosztás és Open Source közösség:

A GitHub központi szereplő az open source (nyílt forráskódú) közösségben. Fejlesztők milliói osztják meg a projektjeiket, hogy mások is felhasználhassák, javíthassák vagy hozzájárulhassanak hozzájuk. A GitHub a világ egyik legnagyobb kódmegosztó platformja.

Projektek és feladatok kezelése:

A GitHub nemcsak kód tárolására alkalmas, hanem projektkezelési eszközként is működik. A Issues (problémák) és Projects (projektek) funkciók lehetővé teszik, hogy feladatokat hozzunk létre, nyomon kövessük azok előrehaladását, és rendszerezzük a projekteket.

Dokumentáció:

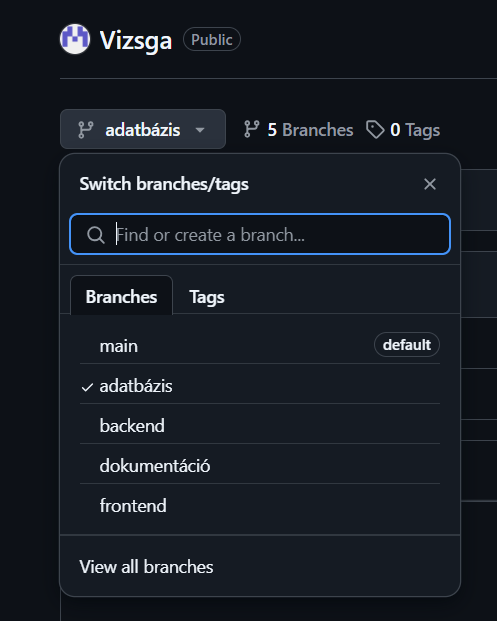
A GitHub egyébként lehetőséget ad a projektek dokumentálására is, például a README.md fájlok segítségével, amelyek tartalmazhatják a projekt leírását, telepítési utasításokat, használati példákat stb.

Biztonság és Hozzáférés-vezérlés:

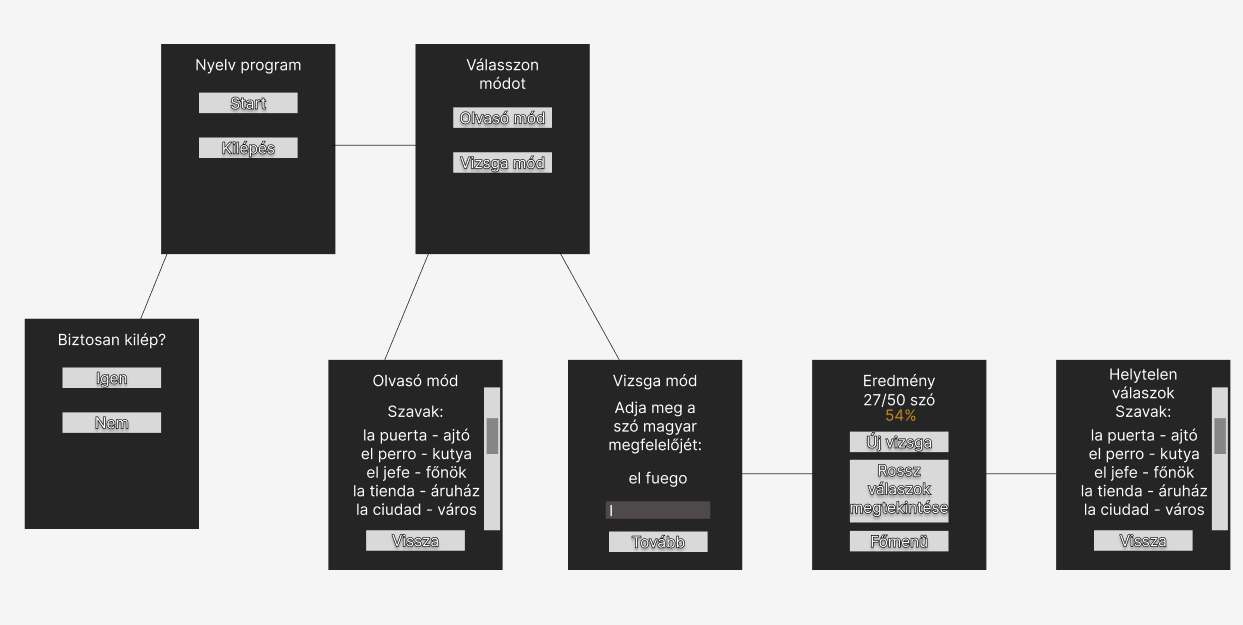
A GitHub biztosítja a biztonságos hozzáférést és a különböző szintű jogosultságokat. A csapattagok meghatározott szerepeket kaphatnak, például adminisztrátorok, fejlesztők vagy csak olvasók, biztosítva ezzel a kód védelmét.

Kódminőség javítása:

A pull request funkciók és az automatikus tesztelés segítenek abban, hogy a kód mindig magas színvonalú legyen. A fejlesztők egymás munkáját át tudják nézni, és biztosíthatják, hogy minden új kód frissítése a legjobb gyakorlatok szerint készüljön el.



**3. Tervrajz:**



**4 Forrásjegyzék:**

**Visual Studio 2022**

<https://en.wikipedia.org/wiki/Visual_Studio>

**Visual Studio Code**

<https://hu.wikipedia.org/wiki/Visual_Studio_Code>

**React**

<https://react.dev/learn/describing-the-ui>

**Web API**

<https://en.wikipedia.org/wiki/Web_API>