# **Szekszárdi SZC Apáczai Csere János Szakgimnáziuma,**

# **Szakközépiskolája és Kollégiuma**

# **Az 54 213 05 számú Szoftverfejlesztő szakképesítés**

# **záródolgozata**

# **Gyakori Kérdések Weboldal**

# **Készítette:**

# **Kiss Ottó Kristóf**

# **Dombóvár**

# **2020**

# Tartalomjegyzék

[Bevezetés 5](#_Toc38315179)

[1. Témaválasztás indoklása 5](#_Toc38315180)

[1.1. Kifejtés 5](#_Toc38315181)

[1.2. Hasonló alkalmazások, mint az én projktmunkám 5](#_Toc38315182)

[1.3. Célközönség 6](#_Toc38315183)

[2. Fejlesztői dokumentáció 6](#_Toc38315184)

[2.1. Igényfelmérés 6](#_Toc38315185)

[2.1.1. Összefüggések és célok 6](#_Toc38315186)

[2.1.2. A rendszer jelenlegi felhasználói és feladatuk 6](#_Toc38315187)

[2.2. Adatbázis tervezés és készítés 7](#_Toc38315188)

[2.2.1. E-K diagram 7](#_Toc38315189)

[2.2.2. Adatbázis diagram 8](#_Toc38315190)

[2.2.3. Normalizálás 9](#_Toc38315191)

[2.2.4. Osztálydiagram 9](#_Toc38315192)

[2.3. Programozási nyelvek 9](#_Toc38315193)

[2.3.1. HTML 9](#_Toc38315194)

[2.3.2. CSS 9](#_Toc38315195)

[2.3.3. PHP 10](#_Toc38315196)

[2.3.4. JAVASCRIPT/JQUERY 11](#_Toc38315197)

[2.3.5. AJAX 11](#_Toc38315198)

[2.3.6. SQL 11](#_Toc38315199)

[2.4. Fejlesztői környezet 11](#_Toc38315200)

[2.4.1. Brackets 11](#_Toc38315201)

[2.4.2. Visual Studio Code 11](#_Toc38315202)

[2.4.3. Laragon 12](#_Toc38315203)

[2.4.4. HeidiSQL 12](#_Toc38315204)

[2.4.5. PhpMyAdmin 12](#_Toc38315205)

[2.4.6. Chrome böngésző 12](#_Toc38315206)

[2.5. Diagramok 12](#_Toc38315207)

[2.6. Ábrák 13](#_Toc38315208)

[3. Felhasználói dokumentáció 15](#_Toc38315209)

[3.1. Rendszerkövetelmények 16](#_Toc38315210)

[3.1.1. Hardver 16](#_Toc38315211)

[3.1.2. Szoftver 16](#_Toc38315212)

[3.2. Program telepítése 16](#_Toc38315213)

[3.3. A program használatának részletes leírása 16](#_Toc38315214)

[3.3.1. Főoldal 17](#_Toc38315215)

[3.3.2. Bejelentkezés 18](#_Toc38315216)

[3.3.3. Regisztráció 18](#_Toc38315217)

[3.3.4. Egy véletlen kérdés/kattintás egy kérdésre 19](#_Toc38315218)

[3.3.5. Kérdések 19](#_Toc38315219)

[3.3.6. Tagok 20](#_Toc38315220)

[3.3.7. Címkék 20](#_Toc38315221)

[3.3.8. Keresés 20](#_Toc38315222)

[3.3.9. Információk 20](#_Toc38315223)

[3.3.10. Profil 20](#_Toc38315224)

[3.3.11. Értesítések 20](#_Toc38315225)

[3.3.12. Kérdéseim 20](#_Toc38315226)

[3.3.13. Válaszaim 20](#_Toc38315227)

[3.3.14. Új kérdés 21](#_Toc38315228)

[3.3.15. Jobb oldali sáv 21](#_Toc38315229)

[4. Tesztelés 22](#_Toc38315230)

[4.1. A tesztelés menete 22](#_Toc38315231)

[4.2. Adatbázis tesztelése 23](#_Toc38315232)

[4.3. Kezelőfelület tesztelése 23](#_Toc38315233)

[5. Összefoglaló 24](#_Toc38315234)

[5.1. Tapasztalatok 24](#_Toc38315235)

[5.2. Továbbfejlesztési lehetőségek 24](#_Toc38315236)

[6. Ábrajegyzék 25](#_Toc38315237)

[7. Irodalomjegyzék, hivatkozás 25](#_Toc38315238)

[Köszönetnyilvánítás 27](#_Toc38315239)

# 

# Bevezetés

Napjainkban nagyon népszerűek a webes felületen elérhető, fórumok, kérdésekkel és adott témákkal kapcsolatos weboldalak. Számtalan oldalon található mindenféle kategóriában és stílusban hasonló oldal. Nagy vonzereje a kis közösségépítő hangulat vagy a válaszok nem egyet értéséből született kisebb viták. Egyszerűen és gyorsan kezelhető és kereshető. A felhasználók leginkább többször kevesebb időt töltenek el a weboldalon válaszok keresése céljából, mintsem, hogy egy vagy pár alkalommal böngésszenek az adott oldalon hosszabb ideig. E szakdolgozat keretében szeretnék fejleszteni egy ingyenes, de jól átlátható és felhasználóbarát webes online, egyének által feltett, és a közösség által kérdésekre válaszolt oldalt. Igyekszem úgy tervezni a rendszert, hogy azok a szabályok, amelyek bevezetése meghaladja e szakdolgozat keretét, könnyen beilleszthetők legyenek egy esetleges továbbfejlesztés alkalmával.

1. Témaválasztás indoklása
   1. Kifejtés

A vizsgamunkám egy online böngészős a felhasználók által bővített kérdések tárhelye. A vizsgamunkám egy online böngészős, regisztrált tagok kérdéseiből és a rá kapott válaszokból álló weboldal. A felhasználó hozzá tud szólni a kérdéshez, a válaszokat pedig értékelni tudja. Az emberek, mindig megoldást és választ szeretnének a kérdéseikre. Legtöbb esetben ezt egy oldalon nem biztos, hogy meg fogják találni. Jó esetben perceket, de rosszabb esetben órákat lehet eltölteni, mire a kérdésre választ kap valaki.

* 1. Hasonló alkalmazások, mint az én projektmunkám

Mostanában nehéz egyedi ötletet kitalálni, és a legtöbb ember nem is akar teljesen új dologba belekezdeni, inkább a meglévő dolgokat fejleszteni és javítani. Így történt az én vizsgamunkámban is. Az én, és szerintem sok mások kedvenc oldala, a gyakorikerdesek.hu weboldal. Rengetegen látogatják, kérdeznek, és kapnak választ a kérdéseikre. Úgy gondoltam, hogy teljesen tökéletes ez az oldal a célpiac számára, vagyis majdnem, hisz ennek az oldalnak a kinézete, már nagyon régóta nem változott.

* 1. Célközönség

A kiválasztott célcsoport igényei szerint kell kialakítani a terméket. Figyelmen kívül kell hagyni az egyéni érdekeket, a szegmensek igényeire kell figyelmet összpontosítani. A fogyasztók elégedettségét is fontos vizsgálni, hogy a célcsoportokat megfelelően lettek-e kialakítva.

A célközönség gyakorlatilag minden magyar ember, aki interneteléréssel rendelkezik. Vannak olyan emberek, akik adott témában igen jártasak, azok fognak leginkább válaszolni a kérdésekre, és a kérdésekre kapott válaszokat magabiztosabban értékelni. Ugyan ezeknek a személyeknek is lehet rengeteg kérdése, amire akár az oldalon is meg tudja találni a választ, de ha nem, akkor pedig fel tudja tenni viszonylag rövid időn belül.

1. Fejlesztői dokumentáció

A fejlesztés egyik legfontosabb lépése az igényfelmérés, majd a tervezés. Ezután, kiderül, hogy az ügyfél vagy a megcélzott csoport(ok) mit is szeretnének látni az oldalon, és mit szeretne csinálni rajta. Ennek alapján a tervezés elkezdődhet, az adatbázist fel lehet építeni, a diagramokat el lehet készíteni, és ez alapján a fejlesztés meg tud valósulni

* 1. Igényfelmérés
     1. Összefüggések és célok

Rengeteg ember szeretne választ kapni a kérdéseire, és ehhez egy felhasználóbarát, több eszközre is optimizált weboldalra van szükség, hogy ezután az emberek, akiknek hasonló kérdéseik vannak, szintén választ kapjanak a kérdéseikre. Továbbá az adminok naprakész statisztikával rendelkezzenek a továbbfejlesztés érdekében.

* + 1. A rendszer jelenlegi felhasználói és feladatuk
* Látogató: Ez a csoport, ami nincs feljegyezve az adatbázis alapján, de a célközönség nagy részét ők teszik ki. Ők csupán csak böngészgetnek az oldalon.
* Felhasználók: A látogatóból vált személyek, akik már elvégezték a regisztrációt, és bejelentkeztek a profiljukba. Kérdéseket tesznek fel, Válaszolnak más kérdéseire, és a válaszokat tudják értékelni.
* Admin: Ők felügyelik az oldal zökkenőmentesebb menetét. A felhasználói kérdésekre válaszolnak, próbálnak még észre nem vett hibákat felkutatni és ezek szerint fejlesztik az oldalt.
  1. Adatbázis tervezés és készítés

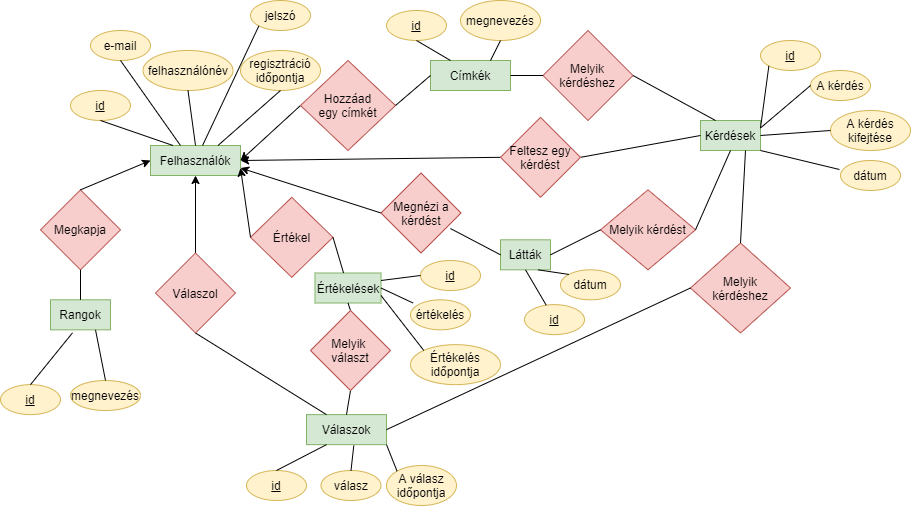
Az adatbázis fontos meghatározója egy webes alkalmazásnak. A felhasználói, vagy akár a rendszer oldaláról is nézve hasznos, sőt elengedhetetlen egy adatbázis. A tervezés során, fontos elsősorban az adat sémákat egy diagramon keresztül kialakítani. Eközben fontosak már a tervezési alapelvek. A valósághű modellezés, a redundancia elkerülése, egyszerűség.

* + 1. E-K diagram

Egyed-kapcsolat diagram elemei:

**Egyed (entitás)**: valódi vagy fiktív dolog, amely tulajdonságokkal, jellemzőkkel bír és önálló elemként tartjuk számon (pl. autó, személy, könyv, stb.) Az egyedeket téglalappal jelöljük és beleírjuk az egyed nevét.

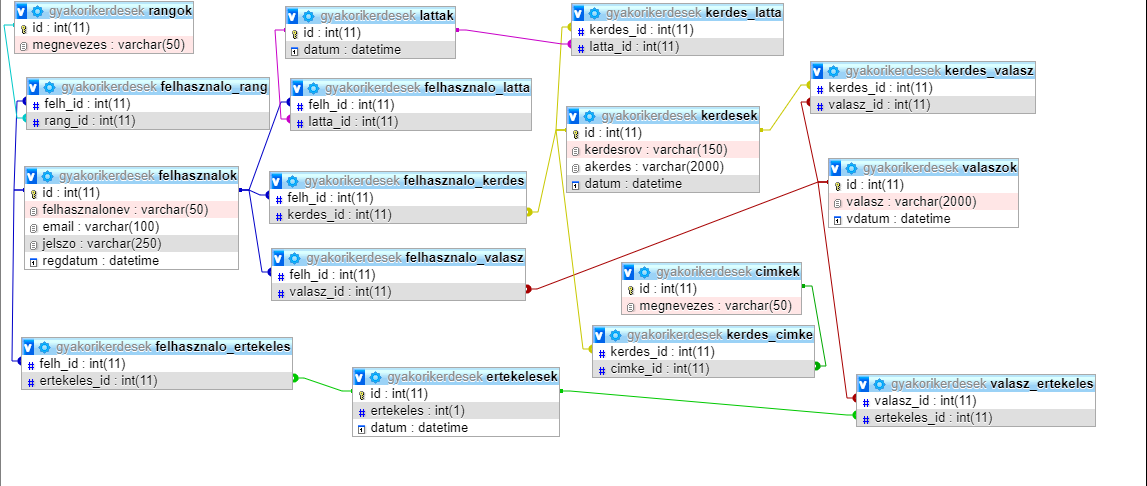
**Kapcsolat**: két vagy több egyed között határoz meg relációt (pl. könyvtári adatbázisban ha van egy személy egyed és egy könyv egyed, akkor a köztük lévő kapcsolat a kölcsönzés). Rekurzív kapcsolat esetén az egyed önmagával van relációban (pl. ha van egy személy egyedünk, akkor az anyja vagy apja kapcsolat rekurzív kapcsolat lesz, hiszen mindenkinek az apja vagy anyja szintén személy). A kapcsolatokat rombusszal jelöljük, és beleírjuk a kapcsolat nevét. A példában az egyik helyen nyíllal jelöltük a kapcsolatot, ennek külön jelentése van, amelyet később tárgyalunk.[[1]](#footnote-1)



1. ábra: E-K Diagram

* + 1. Adatbázis diagram

Ez a diagram az webalkalmazás adatbázis felépítését mutatja. Jól szemlélteti a táblák közötti kapcsolatokat is. A kapcsolatok külső kulcsként vannak jelen.Egy külső kulcs megszorítás, azt határozza meg, hogy egy tábla adott oszlopa, egy másik tábla bizonyos oszlopának az értékeit veheti csak fel. Mindegyik külső kulcsnál CASCADE-t állítottam be, hogy ha egy érték változik egy adott táblában, akár törlődik vagy frissül, akkor vele együtt változzon a vele azonos, másik táblában található érték.



2. ábra: Adatbázis diagram

* + 1. Normalizálás

A normalizálás fontos lépés, az adatbázis fejlesztésében. Az adatok rendszerezésének folyamatára szükség van, az anomáliák kiküszöbölése érdekében. A normalizálás nélkül, az adatok rugalmatlanabbak, a redundáns adatok lemezterületet pazarolnak, és karbantartási problémát okozhatnak. A normalizálást, öt lépésre tagoljuk, minden lépést az adot sorszámú normálformának nevezik. Ha egy táblázat eleget tesz egy szintnek, de a következőnek nem, akkor abban a normálformában van. Az adatbázisomat nem lépésről lépésre, hanem a normálformák ismeretében egyszerre alakítottam ki.

* + 1. Osztálydiagram

Az osztályok és összefüggéseik ábrázolására szolgál. A webalkalmazásom négy osztályt különböztet meg egymástól.

* 1. Programozási nyelvek

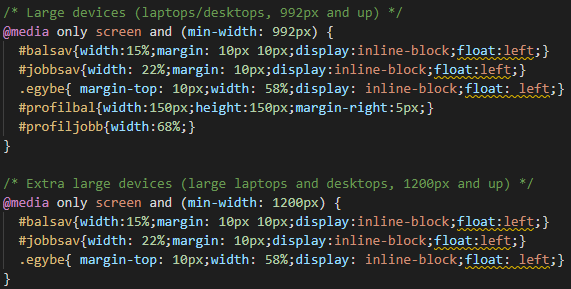
A programozási nyelvek elengedhetetlenek egy program elkészítéséhez. Fontos azoknak a kiválasztása, és magasabb szintű ismerete, hogy zökkenőmentesebb legyen a fejlesztés.

* + 1. HTML

Minden weboldal mögött ott található a HTML. Nem programozási nyelvnek számít, hanem jelölőnyelvnek, ahogy a fordításban is észre tudjuk venni.Segítségével kialakítottam a weboldalam struktúráját, az elképzelésem szerint.

* + 1. CSS

A CSS egy stílusleíró nyelv, amivel tetszetőssé és más eszközökre optimalizálva létrehoztam a projektmunkát. A weboldal stílusát nem maga a HTML tagekben szabályzom, hanem egy külön fájlban, id-k és class-ok segítségével.

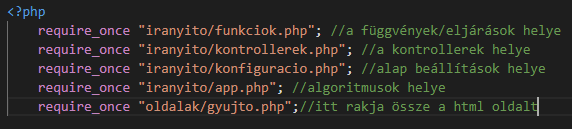


3. ábra: A több eszközre való optimizálás css kódja

* + 1. PHP

Ez a programnyelv, ami leginkább meghatározza a weboldalamat. A projektmunka nagy részében ezt a szerveroldali szkriptnyelvet alkalmaztam, főleg az adatbázissal való kommunikáció érdekében. Segítségével létrehoztam, egy dinamikus weboldalt.

A webalkalmazásom egy MVC alkalmazás. Magyarosítva: Model-Nézet-Vezérlő. Az alkalmazás felépítésének főbb célja, a több rétegre való felbontás. Ez a szoftvertervezésben egy gyakori szerkezeti minta célja, hogy a felhasználói felület ne befolyásolja az adatkezelést, és könnyen átszervezhetőek legyenek az adatok, a felület átszervezése nélkül.



4. ábra: Az mvc alkalmazás főbb felépítésének kódja

* + 1. JAVASCRIPT/JQUERY

Javascript egy script nyelv, amely a böngészőn fut.

A JQuery egy JavaScript könyvtár, mely a HTML kód és a kliens oldali JavaScript közötti kapcsolatot hangsúlyozza.

* + 1. AJAX

Ez egy aszinkronizált Javascript és XML, ami segítségével az oldal betöltése nélkül tudtam adatokat küldeni és fogadni. A webalkalmazásomban, ezt az űrlapoknál használtam fel, hogy amíg a nem megfelelő szabályok szerint van érték egy beviteli mezőben, addig ne engedje az oldalt elküldeni.

* + 1. SQL

Egy strukturált lekérdezőnyelv, amivel kialakítottam az adatbázist.

* 1. Fejlesztői környezet

A fejlesztői környezet fontos meghatározója és hozzájárulója, a program elkészítésének módjához, stílusához, időigényéhez és egyéb más tényezőkhöz.

* + 1. Brackets

A weboldalam kialakításának legelső szerkesztési fázisában a Brackets nevezetű szövegszerkesztőt alkalmaztam. Az idő haladtával, és a feladat megoldásához szükséges segítő eszközök hiánya okán, szükség volt egy másik program használatára

* + 1. Visual Studio Code

A Visual Studio Code program segítségével készítettem el a webes alkalmazás legtöbb részét.

Kiegészítő segítségének igénybevételével lett fejlesztve a fájlok kód sorai.

* + 1. Laragon

Egy szabad és nyílt forrású  webszer-szoftvercsomag, amelynek legfőbb alkotóelemei az Apache webszerver, a HeidiSQL adatbázis kezelő, valamint a PHP programozási nyelvek végrehajtó rendszerei. Egy integrált rendszert alkot, mely alkalmazások készítését, tesztelését és futtatását irányítja.

* + 1. HeidiSQL

Egy nyílt forráskódú asztali adatbázis kezelő rendszer. Az átláthatóság és a gyorsaság

* + 1. PhpMyAdmin

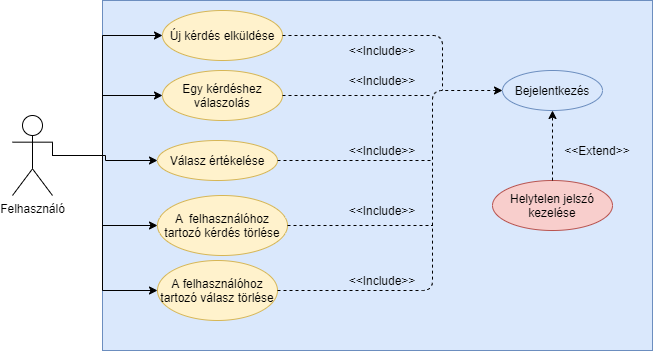
Egy nyílt forráskódú webes adatbázis kezelő rendszer. A felhasználónév és a jelszó begépelése után,

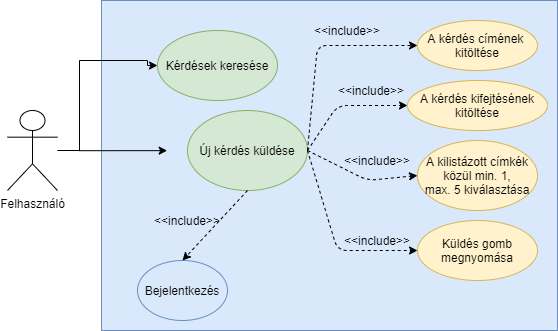
* + 1. Chrome böngésző

A webalkalmazás böngészőn keresztül tekinthető meg. A létrehozását és a tesztelését Google Chrome böngésző segítségével oldottam meg.

* 1. Diagramok

Mindent ezeken az ábrákon keresztül szemléltetem a folyamatok lépéseit.Nagyon egyszerű volt a dolgom az ábrák elkészítése során. A program amit használtam hozzá az a Visual Paradigm és egy online diagram készítő alkalmazás volt. A diagramok rendkívül hasznosak az adatok bemutatásában és egy adott folyamat leírásában. Ilyen például a regisztráció, a belépés, új kérdés elküldése. Ezeken az ábrákon keresztül szemléltetem a folyamatok lépéseit. Az ábrák elkészítéséhez, Visual Paradigm asztali alkalmazást, és egy online diagramkészítőt használtam.



5. ábra: A felhasználók számára elérhető funkciók

6. ábra: Új kérdés elküldése use case diagram

* 1. Ábrák

Use case diagram, néhány elérhető funkcióról.

|  |  |  |  |
| --- | --- | --- | --- |
| Ki? | Milyen gyakran? | Mit? | Megjegyzés |
| F,L,A | 1 | Regisztáció | e-mail, felhasználónév, jelszó, regisztráció dátuma |
| F,L,A | 1..n | Keresés az oldalon |  |
| F,L,A | 1..n | Egy kérdés megtekintése |  |
| F,A | 1..n | Belépés | felhasználónév, jelszó |
| F,A | 1..n | Új kérdés küldése |  |
| F,A | 1..n | Kérdéshez válaszolás |  |
| F,A | 1..n | Válasz értékelése |  |
| F,A | 1..n | Saját értesítések megtekintése |  |
| F,A | 1..n | Kijelentkezés | COOKIE-k törlése |

Itt lépésről-lépésre látható a regisztráció folyamata.

|  |  |
| --- | --- |
| **Azonosító**  001 | **Név**  Regisztráció |
| **Rövid leírás**  A felhasználó regisztál, majd böngész az oldalon. Keres, talál, válaszol, értékel |  |
| **Érintett aktorok**  Felhasználó, Admin |  |
| **Kiváltó esemény**  Amikor a valaki igénybe szeretné venni, a felhasználóknak elérhető funkciókat |  |
| **Paraméterek**  e-mail, felhasználónév, jelszó, regisztráció dátuma |  |
| **Standard Lefutás**  1.Regisztráció  2.Adatok megadása  3.Végélegesítés | **Alternatívák**  a, Jelszó mégegyszer  b, a Felhasználónév már foglalt |
| **Utófeltétel**  Adatok ellenőrzése  Véglegesítés |  |
| **Eredmény**  Regisztráció sikeressége |  |
| **Átlagos időtartam**  2 perc |  |
| **Utalások**  002 |  |
| **Megjegyzések**  Kérdezni csak így lehet |  |

Ezután szemügyre vesszük az „Új kérdés” folyamatot.

|  |  |
| --- | --- |
| **Azonosító**  002 | **Név**  Új kérdés küldése |
| **Rövid leírás**  A felhasználó a kérdésére nem talál megfelelőt |  |
| **Érintett aktorok**  Felhasználó, Admin |  |
| **Kiváltó esemény**  Amikot a felhasználó kérdezni szeretne |  |
| **Paraméterek**  A kérdés címe, a kérdés kifejtése, a hozzákapcsolódó címkék kiválasztása |  |
| **Standard Lefutás**   1. A mezők kitölése 2. Címkék kiválasztása 3. Küldés gomb megnyomása | **Alternatívák**  a, Létezik már ilyen kérdés |
| **Utófeltétel**  Adatok ellenőrzése  Véglegesítés |  |
| **Eredmény**  A kérdés sikeres elküldése |  |
| **Átlagos időtartam**  5 perc |  |
| **Utalások**  002-1 |  |
| **Megjegyzések**  Mindegyik mező kitölése kötelező, legalább egy, maximum 5 címke kiválasztása lehetséges |  |

1. Felhasználói dokumentáció

A felhasználói dokumentáció fontos a leendő felhasználóknak, az üzembe helyezőnek és a betanítónak.

* 1. Rendszerkövetelmények
     1. Hardver

Processzor: Intel(R) Core(TM) i3-4030U CPU @ 1.90GHz

RAM: 4GB

HDD: 1TB

MONITOR MÉRET: 15,6”

* + 1. Szoftver

Operációs rendszer: Windows 10 Education

Böngésző: Google Chrome (80.0.3987.163 64-bit)

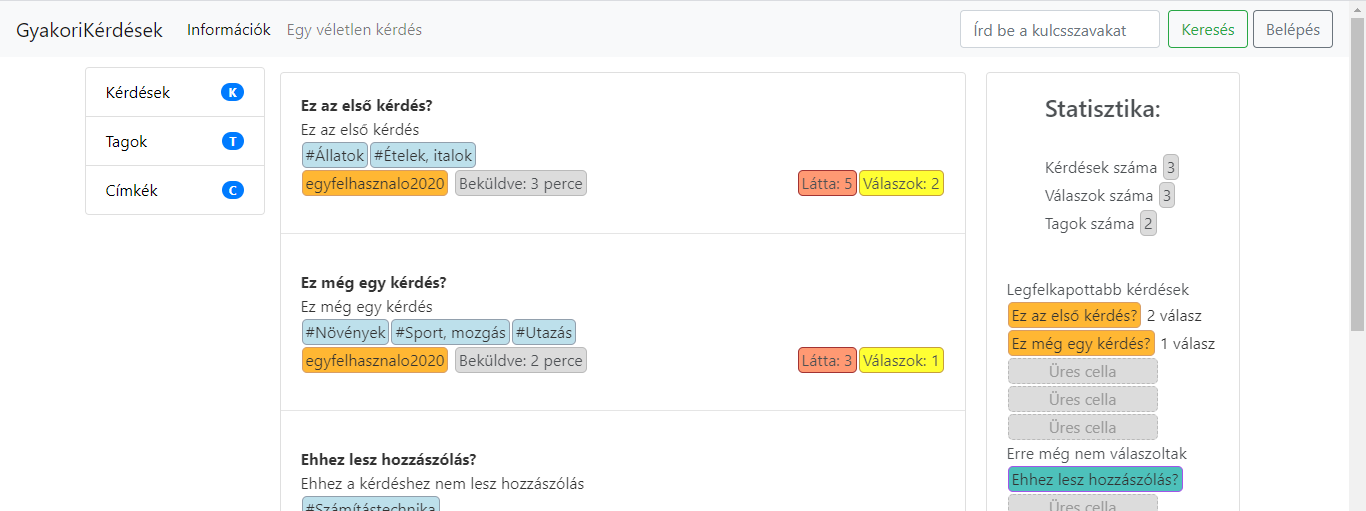
* 1. Program telepítése

A programhoz valójában nincs szükséges telepítéshez, mert a böngésző segítségével tudja elérni a felhasználó. A továbbfejlesztések során nem lesz szükség az alkalmazás frissítésére.

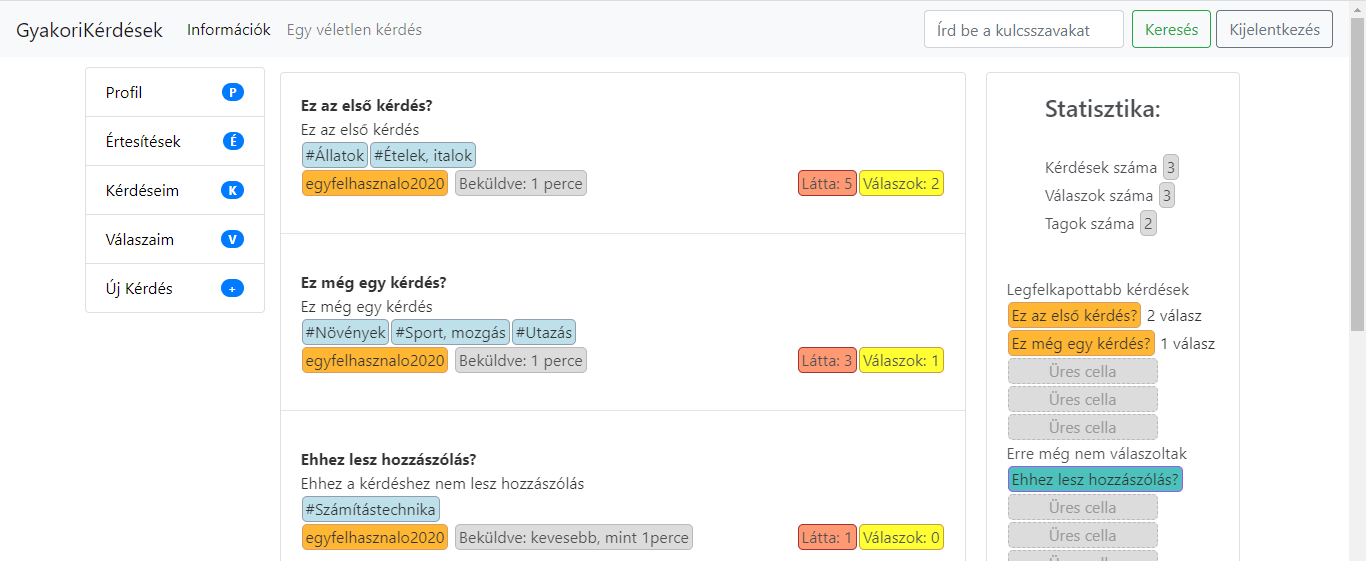
* 1. A program használatának részletes leírása

A program jelenleg csak webes felületen elérhető. Telepítéshez nincs szükség a felhasználó számára.

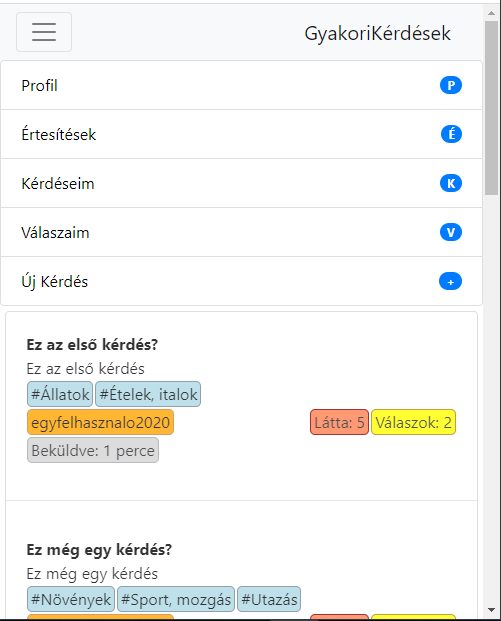
* + 1. Főoldal



7. ábra: A főoldal szerkezete kijelentkezve



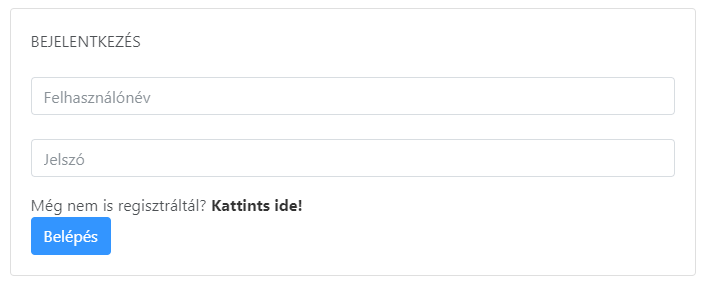
8. ábra: A főoldal szerkezete bejelentkezve



9. ábra: A főoldal telefonos kinézete bejelentkezve

* + 1. Bejelentkezés

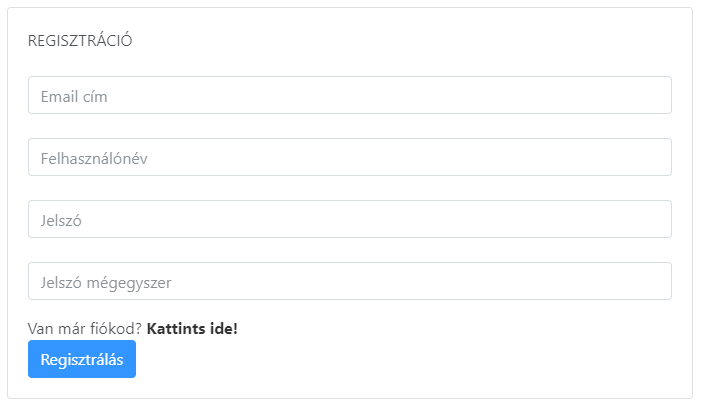
Itt tud belépni a felhasználó, helyes felhasználónév és helyes jelszó páros megadása után.



10. ábra: Bejelentkezés űrlap szerkezete

* + 1. Regisztráció

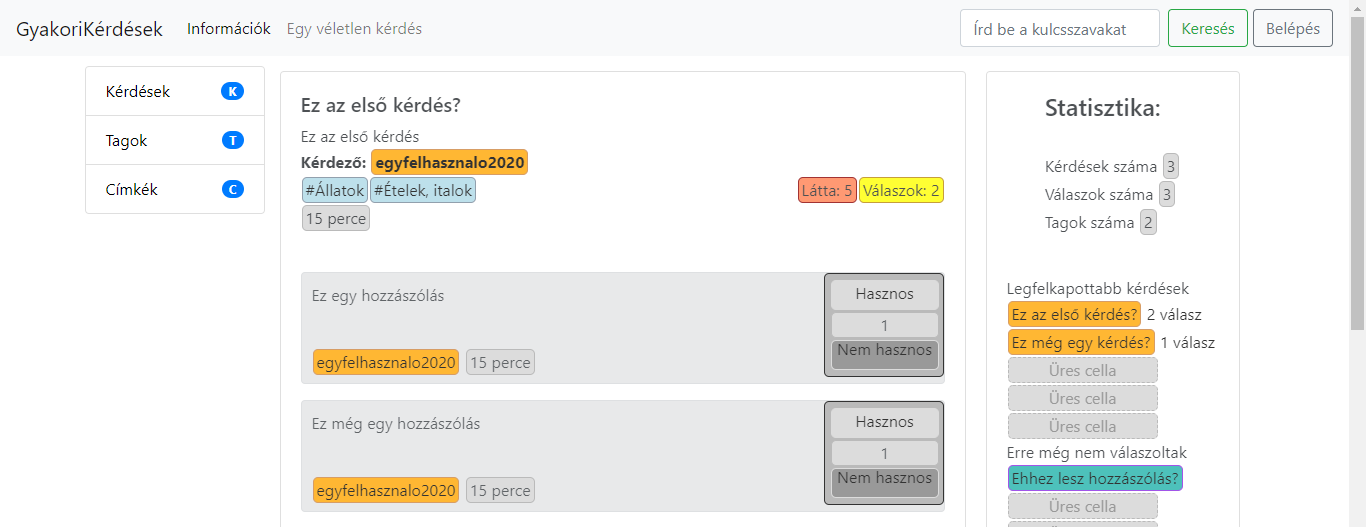
A regisztrációt itt lehet elvégezni, ahol szükséges megadni a jövőbeli felhasználó e mail címét, felhasználónevét és a jelszavát.



11. ábra: A regisztrációs űrlap szerkezete

* + 1. Egy véletlen kérdés/kattintás egy kérdésre

Egy véletlen kérdés nevezetű menüpontra vagy egy kérdésre kattintva egy adott kérdéshez tartozó oldalt kapunk. Látható lesz rajta rengeteg adat. A kérdés címe, a kérdés, a kérdező, a kérdés beküldésének dátuma, a válaszok száma, és az is, hogy mennyi megtekintésnél jár.Fontos hozzá sorolni, a kapott válaszokat, amit bejelentkezés után, minden felhasználó tud értékelni hasznos vagy nem hasznosként.



12. ábra: Egy kérdé oldal felépítése

* + 1. Kérdések

Erre a menüpontra kapcsolva, az összes kérdést kilistázza az oldal, amit lehet rendezni és szűrni,

* + 1. Tagok

Ez az oldal kilistázza a tagokat. Az adott felhasználóra kattintva, megkapjuk az adott ember profilját.

* + 1. Címkék

A címkék azok, amivel csoportosítani lehet a kérdéseket. Erre az oldalra kattintva, megkapjuk azt az oldalt, ami az adott címkéhez rendezve kapjuk meg a kérdéseket. Itt az összes többi címkét is kilistázza az oldal.

* + 1. Keresés

A keresés funkció a leghasználatosabb minden oldalon. Ha valaki valamit nem talál, legtöbbször ezeknek a segítségével keresi meg.

* + 1. Információk

Itt kisebb segítségre talál, aki erre az oldalra tekint. Az alapvető funkciókról tudhat meg többet az olvasója.

* + 1. Profil

Ez az oldal jelenleg, még nem sok hasznos funkcióval tud büszkélkedni. Megjeleníti a felhasználónevet, az adott felhasználónak az e-mail címét, ha be van jelentkezve.Megkapjuk a kérdések számát, a válaszok számát, az értékelések számát, és a leggyakrabban használt címkét.

* + 1. Értesítések

Ennél az oldalnál, a felhasználó a jelenlegi nap és az előző két nap kérdéseire kapott válaszokat tudja megtekinteni, napokra felbontva.

* + 1. Kérdéseim

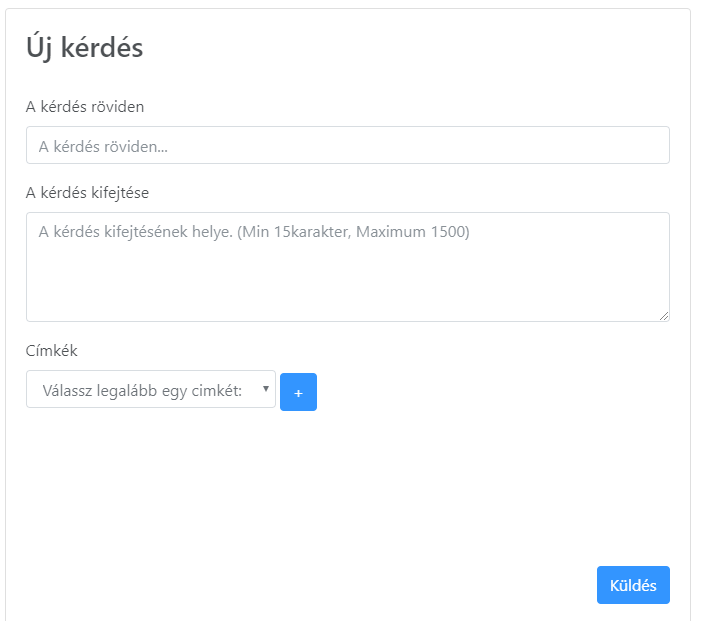
Lehetőség nyílik megtekinteni a saját feltett kérdéseket. Itt lehet azokat törölni is.

* + 1. Válaszaim

Az az oldal, ahol az adott felhasználó, az adott kérdés(ek)hez fűzött válaszait láthatja rendezve. Lehetőség van azok törlésére.

* + 1. Új kérdés

Egy felhasználó az “Új kérdés” menüpontra kattintva, behoz egy oldalt, ahol a szabályoknak megfelelően tud kérdést feltenni. A kérdés címe, a kérdés kifejtése és legalább egy, de maximum öt címke megadásával, a “Küldés” gomb megnyomása után lehetsége elküldeni az adott űrlapot.

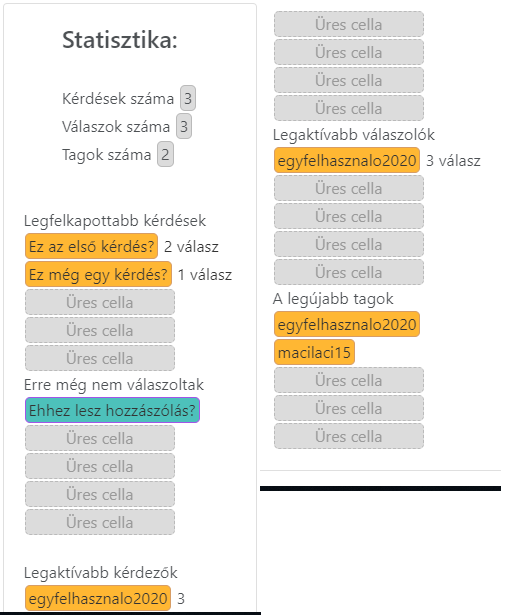


13. ábra: Új kérdés oldal szerkezete és kinézete

* + 1. Jobb oldali sáv

A jobb oldali sáv mindegyik oldalon megjelenik. Rajta láthatóak az oldal statisztikái.

Kérdések száma, válaszok száma, tagok száma. Alattuk több rész is van. A legfelkapottabb kérdéseknél, az öt legtöbb válasszal rendelkező kérdés van kilistázva.A következő azok a kérdések ötös listája, amelyekre még nem válaszoltak. Fontos tudni, hogy a látható “üres cella”-k az oldal kitöltését szolgálják. Egy kérdés vagy felhasználó, ha megfelel az adott kritériumnak, akkor eltűnik egy ilyen rész, és helyette az foglalja el a helyét. A legaktívabb kérdezők részen, a legtöbb kérdéssel rendelkező felhasználók foglalnak helyet. A legaktívabb válaszolóknak a helyén, a legtöbb válasszal rendelkezők állnak. A legutolsó helyen az új tagok állnak, ahol a legújabban regisztrált felhasználók vannak.



14. ábra: A jobb oldali sáv

1. Tesztelés

A tesztelés feladata a program helyességének, megbízhatóságának, tudásának, vizsgálata. Két alapvető célja a rendszerben található hibák felderítése és annak ellenőrzése, hogy a rendszer a felhasználó céljai szerint használható.

* 1. A tesztelés menete

„Egy tesztelési terv tartalmazza, hogy a rendszer milyen tulajdonságait akarjuk tesztelni, melyeket nem. Le kell írni benne, hogy a szoftver képességeinek milyen kombinációit akarjuk ellenőrizni. Bele kell venni, hogy milyen megközelítéssel akarjuk tesztelni, azaz minden funkciócsoportra kell egy megközelítés, amellyel megfelelően tudjuk azt tesztelni. Ez magában foglalja a fő tesztfeladatokat, illetve az azok elvégzéséhez szükséges időt. Specifikáljuk benne, hogy mi az az átfogási szint, amit el szeretnénk érni, megadjuk a technikákat, amellyel ellenőrizzük, hogy mennyire széleskörűek a teszt erőfeszítések, illetve milyen technikákkal ellenőrizzük a követelmények követhetőségét. A teszt tervben rögzítjük továbbá, hogy hogyan döntjük el egy teszt futtatásról, hogy az sikeres, avagy sikertelen. Előre meg kell határozni, hogy milyen környezeti igényei vannak a tesztelésnek, alapvető, illetve elvárt szinten. Ez tartalmazza a szükséges hardver, illetve szoftver eszközök leírását, illetve egyéb különleges igényeket. Fontos rögzíteni azt is, hogy milyen felelősségekkel jár, kinek milyen jogköre van (tervezés, futtatás, ellenőrzés stb). Készíteni kell egy ütemtervet, amellyel ellenőrizhető a folyamat előrehaladásának mértéke, meg kell adni a részfeladatokhoz szükséges időráfordítás mértékét, illetve mérföldköveket. Meg kell említeni a rizikófaktorokat, eshetőségeket is, amelyek kockázatokkal járhatnak.”[[2]](#footnote-2)

* 1. Adatbázis tesztelése

„Mint látható, az adatbázis rendszerek több modult foglalnak magukba, hiszen tartalmaznak adattároló funkciókat, erőforrás menedzselő funkciókat és programkód végrehajtó funkciókat is egy egységben. Emiatt szokás az adatbázisok tesztelését öszetettebb feladatnak tekinteni, mely a már megismert szoftver tesztelési mechanizmusok integrálására épül.

Az adatbázis tesztelése azonban legalább ugyanolyan fontos lépés a szoftverrendszerek fejlesztése során, mint az alkalmazások kódjának tesztelése, hiszen az adatbázis jelenti az alkalmazások információ forrását. Ezen függőség miatt a rossz adatbázis működés kihat az egész alkalmazásra. Mivel a fejlesztés során az alkalmazási kód az adatbázis sémára épül, ezért a séma módosítás költsége nagyságrenddel nagyobb értékű, mint a kód javításának költsége. „[[3]](#footnote-3)

* 1. Kezelőfelület tesztelése

A felhasználói felület (angolul user interface) a site azon eleme, amelyen keresztül a látogató interakcióba lép a weboldallal. Ha a felhasználói felület nem konzekvens, zavaros, nehezen áttekinthető, akkor a látogatók elkattintanak az oldalról. Tesztelése éppen ezért kritikus a siker szempontjából.

* A felhasználói felületek tesztelése során az alábbi területeket vizsgáljuk:
* A felhasználói felületekre vonatkozó sztenderdeknek történő megfelelés ellenőrzése
* A design elemek ellenőrzése: elrendezés, színek, betűtípusok, feliratok, szöveg blokkok, szöveg formázás, nyomógombok, CTA-k, ikonok, linkek
* A felület tesztelése különböző felbontásokon
* Nyelvi változatok tesztelése
* Különböző eszközökön (mobil, tablet, desktop) történő tesztelés[[4]](#footnote-4)

1. Összefoglaló
   1. Tapasztalatok

A webalkalmazás készítése során, rengeteg számos tényezővel találkoztam, amire nem számítottam. A régebbi ismereteim során ritkán készítettem hasonló bonyolultságú alkalmazásokat. A feladat első fázisaiban nem volt meg minden tudásom a jelenlegi projektmunkámhoz szükséges elvárásokhoz képest. Rengeteg fórumot olvastam, videót néztem, annak érdekében, hogy az elért eredmény minél előbb, határidőre elkészülhessen. Úgy gondolom, hogy elvárt volt ez a feladat ahhoz, hogy megtapasztaljuk azt, hogy miként áll össze egy alapvető alkalmazáshoz képest, egy ilyen szintű komolyabb projekt. Rengeteg hiba merült fel a fejlesztés során, amikre sokszor képes voltam magamtól is megoldani, akár néhány percen belül, de voltak olyan problémák, amiknél szükséges volt más segítséget is igénybe vennem, ha az tartok akár napokig is. A szerzet tapasztalataimat igyekszek a következő, ennél magasabb szintű projekteknél is felhasználni.

* 1. Továbbfejlesztési lehetőségek

Az alkalmazás rengeteg szinten még továbbfejleszthető.A szakdolgozat leadásának idejére eddig a szintig sikerült kidolgozni.

A főbb lehetőségek a következők lennének:

* Az űrlapoknál további szabályok alkalmazása szükséges annak érdekében, hogy kevesebb esély legyen a hibák megjelenésére.
* Üzenet privátban. Amikor a weboldal online elérhetővé válik, és a felhasználók száma gyarapodni fog, szükség lesz mindenképp üzenetküldési lehetőségre is.
* A továbbfejlesztési lehetőségek egyik legfontosabb tagja, a pontrendszer. A felhasználók, az adott tevékenységük során pontot szerezhetnek, vagy esetleg veszíthetnek (Az oldalon értelmetlenül sok, haszontalan kérdezés, válaszadás, a többi értékeléssel nagyrészt ellenkező értékelés). Ezek alapján, napi, heti és havi statisztikák készítése.
* Különböző statisztikák készítésének támogatása az admin felületen. Például: Időszakokra felosztani a feltett kérdésekett, a válaszokat, a láttamokat és a hasznos/nemhasznos funkciókat .
* A bővítési lehetőségek között szerepel, hogy egy felhasználó profilját megtekintve, látható lenne a feltett kérdései és a válaszai.
* Későbbiekben a egy megadott aktivitást követően bővíteni lehetne úgy, hogy telefonos applikációkat tudják használni az érdeklődök.

1. Ábrajegyzék

[1. ábra: E-K Diagram 7](#_Toc38312811)

[2. ábra: Adatbázis diagram 7](#_Toc38312812)

[3. ábra: A több eszközre való optimizálás css kódja 9](#_Toc38312813)

[4. ábra: Az mvc alkalmazás főbb felépítésének kódja 10](#_Toc38312814)

[5. ábra: A felhasználók számára elérhető funkciók 12](#_Toc38312815)

[6. ábra: Új kérdés elküldése use case diagram 12](#_Toc38312816)

[7. ábra: A főoldal szerkezete kijelentkezve 14](#_Toc38312817)

[8. ábra: A főoldal szerkezete bejelentkezve 14](#_Toc38312818)

[9. ábra: A főoldal telefonos kinézete bejelentkezve 15](#_Toc38312819)

[10. ábra: Bejelentkezés űrlap szerkezete 15](#_Toc38312820)

[11. ábra: A regisztrációs űrlap szerkezete 16](#_Toc38312821)

[12. ábra: Egy kérdé oldal felépítése 16](#_Toc38312822)

[13. ábra: Új kérdés oldal szerkezete és kinézete 18](#_Toc38312823)

[14. ábra: A jobb oldali sáv 19](#_Toc38312824)

1. Irodalomjegyzék, hivatkozás

<https://www.inf.u-szeged.hu/~gnemeth/adatbgyak/exe/EK_diagram/az_egyedkapcsolat_diagram_elemei.html>

<http://mit.bme.hu/~micskeiz/education/onlab/kovacs_adam/J2EE-alkalmazasok-tesztelese-Dolgozat-5.pdf>

<https://regi.tankonyvtar.hu/hu/tartalom/tamop425/0046_szoftverteszteles/ch10.html>

<https://www.tutorial.hu/weboldal-keszites/weboldal-tesztelesi-eljarasok-teljes-utmutato/>

Köszönetnyilvánítás

Szeretném megköszönni Fodorné Csepeli Diána osztályfőnökömnek folyamatos segítséget és motivációt , hogy elérjem azt a szintet amivel el tudtam kezdeni a projektmunkámat, és el tudjam kezdeni a szakmai vizsgák megszerzéséhez szükséges felkészülést.

Závoda Ferenc tanár úrnak, aki próbált minél több információt átadni ahhoz, hogy a lehető kevesebb gondunk akadjon a vizsgák folyamán.

Noé Zoltán tanár úrnak, aki folyamatosan segített azokban az esetekben is, amikor elakadtam valamelyik feladatban.

Kiss Ottó Kristóf

1. <https://www.inf.u-szeged.hu/~gnemeth/adatbgyak/exe/EK_diagram/az_egyedkapcsolat_diagram_elemei.html> [↑](#footnote-ref-1)
2. <http://mit.bme.hu/~micskeiz/education/onlab/kovacs_adam/J2EE-alkalmazasok-tesztelese-Dolgozat-5.pdf> [↑](#footnote-ref-2)
3. <https://regi.tankonyvtar.hu/hu/tartalom/tamop425/0046_szoftverteszteles/ch10.html> [↑](#footnote-ref-3)
4. <https://www.tutorial.hu/weboldal-keszites/weboldal-tesztelesi-eljarasok-teljes-utmutato/> [↑](#footnote-ref-4)