**Szakképesítés neve:** **Szoftverfejlesztő**

**OKJ száma: 54 213 05**

**ZÁRÓDOLGOZAT**

**Dokumentum megosztó és kezelő webalkalmazás**

**Balogh Krisztián**

14.Inf

**Dombóvár, 2020.**

Tartalom

[1. Bevezetés 4](#_Toc33999163)

[2. Célközönség meghatározása 5](#_Toc33999164)

[3. A feladat megoldásának ismeretei 6](#_Toc33999165)

[3.1 Használt eszközök: 6](#_Toc33999166)

[Visual Studio Code 6](#_Toc33999167)

[Laragon 6](#_Toc33999168)

[GitHub 6](#_Toc33999169)

[Visual Paradigm 7](#_Toc33999170)

[Gravit Designer 7](#_Toc33999171)

[3.2 Alkalmazást felépítő elemek 7](#_Toc33999172)

[PHP 7](#_Toc33999173)

[MySQL 7](#_Toc33999174)

[HTML 8](#_Toc33999175)

[CSS 8](#_Toc33999176)

[AJAX 8](#_Toc33999177)

[JavaScript, jQuery 8](#_Toc33999178)

[Bootstrap 8](#_Toc33999179)

[4. Hasonló alkalmazások 9](#_Toc33999180)

[Doksi.hu 9](#_Toc33999181)

[Érettségi.com 9](#_Toc33999182)

[5.Megvalósítandó funkciók 10](#_Toc33999183)

[6. Futtatási környezet 11](#_Toc33999184)

[7. Rendszertervezés 12](#_Toc33999185)

[7.1 Követelmények specifikációi 12](#_Toc33999186)

[7.1.1 Use Case Diagram(Használati esete diagram) 15](#_Toc33999187)

[7.1.2 Use Case Folyamat leltár 15](#_Toc33999188)

[7.1.3 Use Case Dokumentum feltöltése, Kérés kezdeményezése 16](#_Toc33999189)

[7.1.4 Use Case Kérés teljesítése, Keresés, Tantárgy hozzáadása 17](#_Toc33999190)

[7.1.5 Use Case Dokumentum értékelése, Hozzászólás bejelentése 18](#_Toc33999191)

[7.1.6 Use Case Rendszer karbantartása 19](#_Toc33999192)

[7.1.7 Az alkalmazás használatbavételének üzleti folyamat táblázata 20](#_Toc33999193)

[7.1.8 A rendszer karbantartásának üzleti folyamat táblázata 21](#_Toc33999194)

[7.2 A rendszer folyamatainak modellezése 22](#_Toc33999195)

[7.2.1 Alkalmazás használatbavétele 22](#_Toc33999196)

[7.2.2 Swimlane diagram: dokumentum feltöltése, kezelése 23](#_Toc33999197)

[7.3 Logikai adatmodell 24](#_Toc33999198)

[7.4 Osztálydiagram 25](#_Toc33999199)

[Irodalomjegyzék 26](#_Toc33999200)

# 1. Bevezetés

Napjainkban a tanulás egy elengedhetetlen dolog, melyre minden téren szükség van. A tanulás folyamatának megkezdéséhez alkalomadtán szükség van a megfelelő tartalmú dokumentumra, melynek megtalálása olykor nem is olyan egyszerű. Vagy azért mert rossz helyen keresünk, vagy azért mert az adott dokumentum számunkra nem elérhető.

Záró dolgozatom célja tehát egy olyan alkalmazás fejlesztése mely az előző két problémát megpróbálja kiküszöbölni. Egy olyan alkalmazás, amely kifejezetten a tanuláshoz szükséges dokumentumokra épül és szorosan összekapcsolja a közösséget az információáramlással. Eszerint szakdolgozatom témája egy iskolai képzéshez kapcsolódó dokumentum megosztó web alkalmazás melynek feladata a felhasználói interakciókra történő dokumentumkezelés, statisztikák készítése, felhasználók kezelése valamint a zavartalan munkavégzés élményének biztosítása. Rendelkezésre állnak hasonló alkalmazások, de az általam kitűzött céloknak egyikük sem felel meg. Mindegyik csak alapvető szolgáltatásokat nyújt és szinte egyikük sem tér el a másiktól. Természetesen alkalmazásom alapjául ezen alkalmazások működése segítség lehet, de az alkalmazás jelentős részét kizárólag saját gondolkodás alapján alakítom ki. Az alkalmazás webes céllal készül, ezáltal meg kell felelnie bizonyos mai web fejlesztési szempontoknak. Ezek a felhasználóbarát és reszponzív szempontok. Az alkalmazás fejlesztéséhez a PHP7 programozási nyelvet választottam a szerveroldali feladatok elvégzéséhez MySQL-t és AJAX -ot az adatbázisos részekhez. HTML5-öt, Bootstrap4-et, CSS3-at, JavaScript-et a kliensoldali dolgokhoz pl. megjelenítés. Az alkalmazás funkcióinak használatbavétele csak bejelentkezés után lehetséges. Az alkalmazást használatba vevőket két csoportra oszthatjuk, az egyik csoportba tartoznak az átlag felhasználók, akik az alkalmazás közösségéért, a dokumentumok feltöltéséért valamint az esetleges felhasználók bejelentéséért felelnek. A másik csoportba pedig azok, akik az alkalmazás megfelelő működéséért a felhasználók valamint a dokumentumok karbantartásáért kezeskednek.

Az alkalmazást illetően az egyik fő elvárt szempontom a platform függetlenség, amely lehetővé teszi az alkalmazás használatát eszköz függetlenül. Csupán olyan eszközre van szükség melyen böngésző futtatható pl. okostelefon, tablet, laptop stb. Fontos figyelembe venni, hogy a minimális képernyőhatár méretet ne lépjük túl, mert az alkalmazás felhasználói élményét ronthatja.

# 2. Célközönség meghatározása

Az alkalmazás célközönsége azok az emberek, akik iskolai képzésben vesznek részt valamint azok, akik iskolai képzésen kívüli önfejlesztést végeznek. Akik számára a tanuláshoz szükséges dokumentumok nehezen fellelhetők és jobban szeretik az egyszerűbb dolgokat. Emellett jobban kedvelik a felhasználóbarát platformokat. Azok, akik a közösséget esetleg össze akarják kapcsolni a tanulással. Mindezt úgy, hogy a mindennapi használatot ne konkrét eszköz használata határozza meg.

# 3. A feladat megoldásának ismeretei

## 3.1 Használt eszközök:

### **Visual Studio Code**

A Visual Studio Code egy általános kódszerkesztő, mint pl: Brackets, Notepad++ stb. Hatékonyan és egyszerűen használható felhőfejlesztéshez. Ingyenes és nyílt forráskódú. Futtatható macOS, Windows és Linux rendszereken. A rengeteg rendelkezésre álló bővítmények rendkívül felgyorsíthatják a munkavégzést. Például a PHP Debug  egy hibakeresésre szolgáló bővítmény, amely segítségével hibákat leplezhetünk le az adott kódban, és nyomon követhetjük, a kód működését pl. megvizsgáljuk, hogy mikor kap értéket egy adott változó. Emellett a program beépített Git támogatással is rendelkezik mely lehetővé teszi az azonnali kódbeli változások commit-álását.

### **Laragon**

A Laragon egy alkalmazás kiszolgáló programcsomag, amely lehetővé teszi a helyi fejlesztést, melynek főbb komponensei a PHP az Apache és a MySQL adatbázis kezelő. Társaihoz képest pl. XAMMP, WampServer telepítése és konfigurálása egyszerűbb valamint nyilvánvalóbb. Az Apache nyílt forráskódú webkiszolgáló alkalmazás, mely egy lokálisan futtatható web szerver. Az Apache lehetővé teszi, hogy megismerjük mind a statikus mind a dinamikus weboldalak, alkalmazások működését. A MySQL az egyik legelterjedtebb adatbázis-kezelő, és egyszerűen beállítható megoldást ad dinamikus webhelyek szolgáltatására. A Laragon az adatbázis kezelésre különféle eszközöket biztosít pl. HeidiSQL, phpMyAdmin stb. Konkrét eszköz használatbavétele olykor konfigurációt eredményez. A MySQL által adatokat olvashatunk ki és vihetünk be az adatbázisba. A dinamikus weblapoknál, alkalmazásoknál elengedhetetlen az adatbázis kezelés.

### GitHub

A GitHub egy verziókezelő rendszer mely lehetővé teszi a fejlesztés nyomon követését. Ezáltal ha fejlesztés során végzetes hibába ütközünk, akkor lehetőség van a könyvtár bármelyik korábbi állapotának visszaállítására. Szinkronizálni tud egy másik gépen levő hasonló könyvtárral, közben intelligensen átvezeti a változásokat, illetve jelez, ha nem megy neki. Használata esetén lehetőség van a mellékágak kialakítására, ha esetleg ügy döntenénk, hogy más irányba folytatjuk tovább a fejlesztést. Természetesen ekkor a főágon történő fejlesztések ettől függetlenül változatlanok megmaradnak. Elérhető belőle asztali alkalmazás a Github Desktop mely futtatható macOS, Windows operációs rendszereken. Ha Visual Studio Code mellett használjuk a programot, akkor folyamatos szinkronizáció zajlik le a két program között mely lehetővé teszi a fájl tartalmának folyamatos változásának szemléltetését mely különböző módon mutatkozhat meg.

### Visual Paradigm

UML diagrammodellező program mely lehetővé teszi, hogy a fejlesztendő programok funkciói áttekinthető formában jelenjenek meg. Többféle diagram típust támogat pl. osztály diagram, használati eset diagram, tevékenység diagram stb. Használatával a fejlesztendő funkciók megértésére fordítandó idő jelentősen lecsökken. Általában a szoftverfejlesztés kezdeti, tervezési fázisában van rá a legnagyobb igény. A program elérhető Windows, macOS és Linuxon.

### Gravit Designer

Online szerkesztő alkalmazás melynek van platform független asztali változata is. Lehetővé teszi a képek szerkesztését, vektoros illusztrációt valamint web dizájn létrehozását webes felületen. Mivel támogatja a vektografikát így a Visual Paradigm programmal készített .svg kiterjesztésű fájlok utólagos szerkesztése sem okoz akadályt.

## 3.2 Alkalmazást felépítő elemek

### PHP

Szerveroldali szkriptnyelv az az elemi utasítások sorozata, mely lehetővé teszi az oldalak dinamikussá tételét. Szerveroldali tehát futásidőben semmi nem látható belőle minden a háttérben történik. PHP kódot általában külön .php kiterjesztésű fájlban valamint a HTML kódba szokás elhelyezni. A PHP kód feldolgozására a web szerver interpretert vagyis értelmezőt alkalmaz. A fejlesztést érdemes mindig a legfrissebb verzióval kezdeni, mert a korábbi verziók használatával olykor hibákba ütközhetünk.

### MySQL

Ingyenesen elérhető kiszolgálón futó SQL adatbázis kezelő rendszer, amely lehetővé teszi a különböző felhasználói interakciókra történő adat tárolást, módosítását, törlését valamint keresését az adatbázisban. Az adatbázisok kezelhetők különféle programozási nyelven megírt programokkal pl. PHP, C# stb.

### HTML

Jelölőnyelv mely meghatározza egy weboldal kinézetét. Egyszerű szöveges fájl telis tele jelölésekkel, mely jelöléseket értelmezve a böngésző a sima szövegből formázott weboldalt állít elő. A HTML elemeket gyakran használatba veszik a HTML-el kompatibilis eszközök pl. CSS, Java Script stb. mely használatbavétel célja az oldal tartalmának formázása, esetleg dinamikussá tétele stb.

### CSS

Stílusleíró nyelv melynek feladata a HTML fájl tartalmának testreszabása. Használatával szabadon formázhatjuk a HTML elemek tulajdonságait.

### AJAX

Használata lehetővé teszi a weboldal folyamatos újratöltésének kiküszöbölését. Újratöltés akkor fordulhat elő, ha pl. adatot szeretnénk felvinni illetve lekérni az adatbázisból. A lekért adatok megjelenítéséhez az AJAX használata nélkül az újratöltés elhagyhatatlan. Az AJAX tehát lehetővé teszi az oldal betöltése után az adatok küldését, fogadását, kérését a szerver felé és mind ezt a háttérben. Valamint az oldal tartalmának változását az oldal újratöltése nélkül. Az AJAX használatbavételéhez elengedhetetlen a JavaScript, jQuery használata.

### JavaScript, jQuery

Kliensoldali szkriptnyelv amely első sorban a weboldalak interaktív működéséért felel. Feladata beviteli mezők tartalmának ellenőrzése, valamint hibásan beírt adatok esetén figyelmeztető üzenet megjelenítése stb. A jQuery pedig egy JavaScript könyvtár mely előre megírt függvényeket tartalmaz. Feladata a JavaScript bonyolultabb dolgainak megoldása. Használatával akár a fejlesztési idő is jelentősen lecsökkenhet.

### Bootstrap

Előre megírt keretrendszer mely CSS, HTML, JS alkotóelemekből, sablonokból épül fel. Lehetővé teszi a reszponzív weboldalak egyszerűbb fejlesztését, ezáltal a fejlesztési időt is jelentősen lecsökkenti.

# 4. Hasonló alkalmazások

### Doksi.hu

Online dokumentum megosztó webalkalmazás mely lehetővé teszi a felhasználóknak a dokumentum feltöltést, keresést valamint értékelést. Emellett számos fogalom, műelemzés, életrajz érhető el az oldalon. Az oldal teljesen ingyenes, ezáltal bárki használatba veheti.

### Érettségi.com

Érettségi tételek, feladatsorok, könyvek online gyűjtőhelye. Az oldalon lehetőség van tételek beküldésére illetve letöltésére. Valamint az oldal felvételi pontszámítóval és a legfrissebb hírekkel szolgál az érettségivel illetve továbbtanulással kapcsolatban. Hasonló alkalmazások állnak rendelkezésre pl. erettsegid.hu mely szinte semmi funkcióban nem tér el az előzőtől, egyedül a felületben mely felhasználóbarátabb.

# 5.Megvalósítandó funkciók

Felhasználó:

* Dokumentumok feltöltése, kérvényezése, teljesítése, értékelése és keresése.
* Felhasználók bejelentése.
* Profil szerkesztése.
* Tantárgyak bővítése.
* Saját feltöltések, kedvencek megtekintése.

Admin:

* Dokumentumok ellenőrzése, kezelése.
* Felhasználók karbantartása.
* Kérések ellenőrzése.
* Az oldal bővülése esetén a felhasználók jogosultságainak módosítása, mely módosítás esetén megkapja az admin alapvető funkcióit.
* Tantárgyak bővítésének ellenőrzése.

Rendszer:

* Statisztikák készítése.

# 6. Futtatási környezet

Az alkalmazás használatbavételéhez csupán megfelelő internet kapcsolatra, és olyan eszközre van szükség, melyen böngésző található. Minimális hardver konfigurációnak a fejlesztési környezetemet tekintem melynek alkotó komponensei a következők. Számítógép: Processzor: Intel Core 2 Duo E8400 3GHz. RAM: 4 x 2 GB 800MHz DDR2. VGA: NVIDIA GeForce 9800 GT 1GB 256bit DDR3. HDD: Toshiba 500 GB. OS: Windows 7 Ultimate 64bit. Okostelefon: Processzor: 2,0 GHz Octa-Core. RAM: 3 GB. ROM: 32 GB. OS: MIUI11, Android 7.0. Ajánlott hardver konfiguráció az általam használt fejlesztési környezettől valamennyivel erősebb alkotó komponensekkel rendelkező eszköz használata. Régebbi, gyengébb alkotó komponensekkel rendelkező eszköz használata esetleg a felhasználói élmények romlásához vezethet. Böngésző hiánya esetén az alkalmazás nem használatba vehető. A használni kívánt eszköznek meg kell felelni egy további szempontnak is mely a minimális képernyőméret. Átlépése szintén a felhasználói élmények romlásához vezethet.

# 7. Rendszertervezés

## 7.1 Követelmények specifikációi

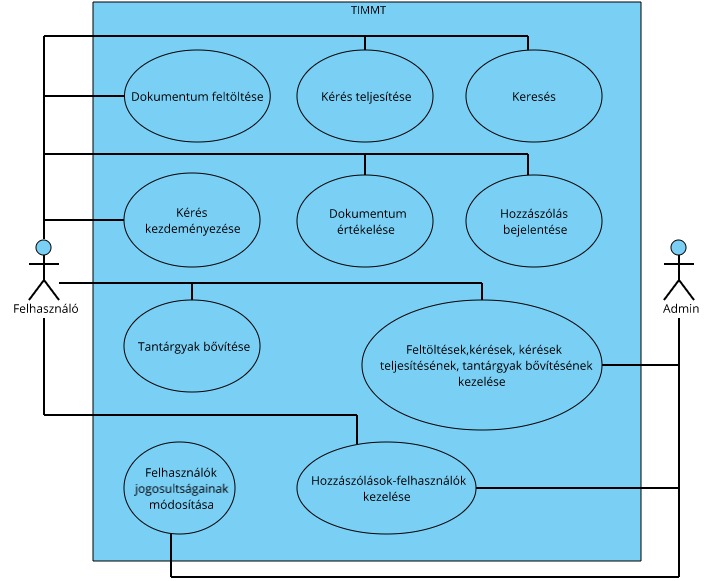
**f: Felhasználó** **a: Admin**

| **Ki** | **Milyen gyakran** | **Mit** | **Megjegyzés** |
| --- | --- | --- | --- |
| - | 1 | **Alkalmazás**  **használatbavétele** |  |
| f | 1 | Felhasználó  bejelentkezése | E-mail, Jelszó páros használata. Kizárólag regisztráció után. |
| f | 1 | Regisztráció | Email, jelszó, név stb. megadása. |
| f | 1..n | Dokumentum feltöltése | Az oldal tartalmához hasonló dokumentumok. |
| f | 1 | Tantárgy  kiválasztása | Az alapján hogy melyik tantárgyhoz köthető a dokumentum teljes tartalma. |
| f | 1 | Kategória  kiválasztása | Dokumentumnak megfelelően pl. Kidolgozott érettségi tétel, könyv, dolgozatminta stb. |
| f | 1 | Év megadása | Melyik évben készült a dokumentum. |
| f | 1 | Oldalszám megadása | Hány oldalból áll az adott dokumentum. |
| f | 1 | Forrás megnevezése | Ha saját, akkor a készítő nevét, ha pl. letöltött, akkor a link megadása. |
| f | 1 | Dokumentum kiválasztása | Helyi gépről. |
| f | 1 | Feltöltés | Műveletek érvényesítése. |
| f | 1..n | Kérés kezdeményezése | Felhasználó által, ha nem talált magának megfelelő tartalmú dokumentumot. |
| f | 1 | Tantárgy kiválasztása | Pontosan, mint a feltöltés esetében. |
| f | 1 | Kategória kiválasztása | -”- |
| f | 1 | Dokumentum megnevezése | Az kérvényező által kért dokumentum megnevezése. |
| f | 1 | Kérés indítása | Műveletek érvényesítése. |
| f | 1..n | Kérés teljesítése | A felhasználó dokumentumot szeretne feltölteni a kérő számára. |
| f | 1 | Teljesítendő kiválasztása | Kérés kiválasztása. |
| f | 1 | Fájl kiválasztása | Feltöltendő dokumentum megkeresése, kiválasztása. |
| f | 1 | Teljesítés | Műveletek érvényesítése. |
| f | 1..n | Keresés | Keresés az elérhető dokumentumok között. |
| f | 1 | Keresendő meghatározása | Dokumentum témájának  meghatározása. |
| f | 1 | Szűrés használata | Keresési eredmények személyre  szabása. |
| f | 1 | Tantárgy kiválasztása |  |
| f | 1 | Kategória kiválasztása |  |
| f | 1 | Év kiválasztása | Dokumentum előállításának éve. |
| f | 1 | Keresés indítása | Műveletek érvényesítése. |
| f | 1..n | Tantárgyak bővítése | Ha nincs az általunk kiválasztandó  tárgy a listában. |
| f | 1 | Tárgy megnevezése | Hozzáadni kívánt tárgy nevének megadása. |
| f | 1 | Hozzáadás | Műveletek érvényesítése. |
| f | 1..n | Dokumentum értékelése |  |
| f | 1 | Dokumentum kiválasztása | Melyik dokumentumot szeretné  értékelni. |
| f | 1 | Pontszám megadása | Hány pontra értékelem az 5-ből. |
| f | 1 | Új értékelés | Szöveg megfogalmazása(comment). |
| f | 1 | Küldés | Műveletek érvényesítése. |
| f | 1..n | Hozzászólás bejelentése | pl. vulgáris megjegyzés esetén |
| f | 1 | Hozzászólás kiválasztása |  |
| f | 1 | Bejelentés okának megadása | Bejelentés indoklása. |
| f | 1 | Bejelentés | Műveletek érvényesítése. |
| a | 1..n | **Rendszer karbantartása** |  |
| a | 1..n | Feltöltések,kérések, kérések teljesítésének, tantárgyak bővítésének kezelése | Az adott elem ellenőrzése. |
| a | 1 | Mentés | A dokumentum, kérés, kérés teljesítés, tantárgy elérhetővé válik a felhasználók számára |
| a | 1 | Törlés | Az adott elem törlődik az összes tulajdonságával együtt az adatbázisból. |
| a | 1..n | Hozzászólások-felhasználók kezelése |  |
| a | 1 | Törlés | A hozzászólás törlődik az adatbázisból. |
| a | 1 | Tiltás | Adott hozzászólást létrehozó személy eltiltása pl. 15 napig az oldaltól. |
| a | 1..n | Felhasználók jogosultságainak módosítása | Az oldal bővülése esetén. |
| a | 1..n | Felhasználó megkeresése | Kinek szeretném módosítani a  jogkörét. |
| a | 1..n | Felhasználó kiválasztása |  |
| a | 1..n | Jogkör megváltoztatása | Pl. közreműködőből szerkesztő. |
| a | 1..n | Mentés | Műveletek érvényesítése. |

## 7.1.1 Use Case Diagram(Használati esete diagram)

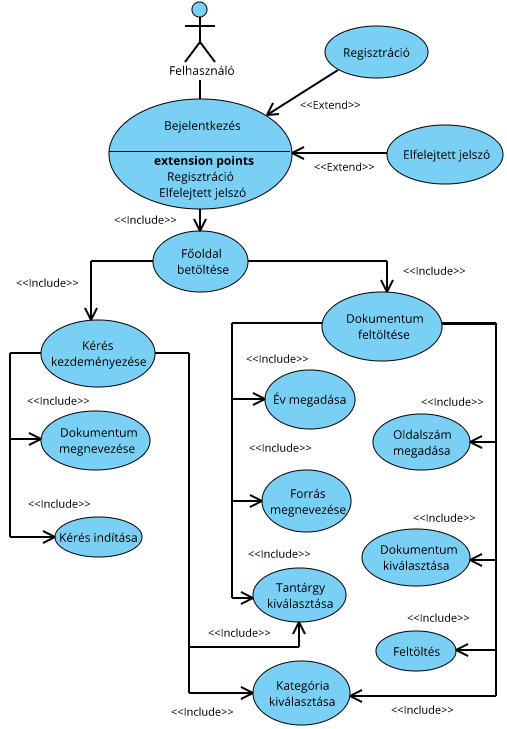
Az Use Case Diagram célja a követelményrögzítés a szoftverfejlesztés kezdeti (tervezési) fázisában. Megmondja, hogy mit kell tudnia a rendszernek, milyen funkciói legyenek a megtervezett rendszernek. Mindezt szemléletesen könnyen áttekinthetően.

## 7.1.2 Use Case Folyamat leltár



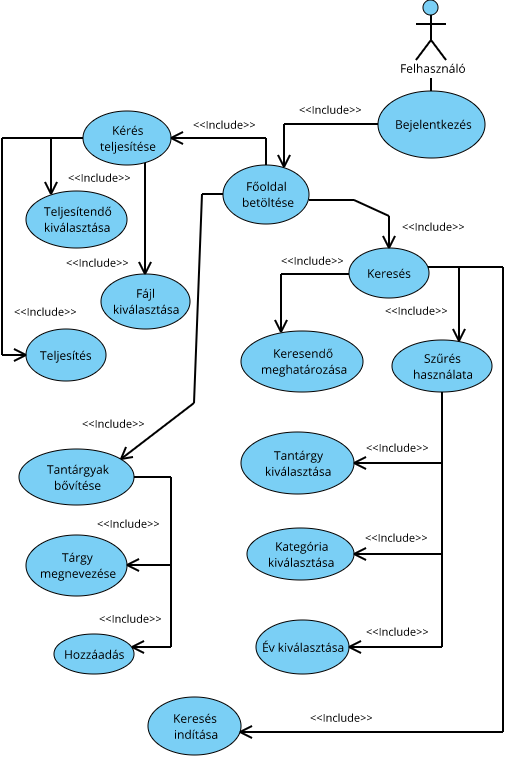
A diagramon a rendszer főbb komponensei láthatóak melyek különböző aktorokhoz (A rendszerrel kapcsolatban álló személyek vagy dolgok melyek az üzleti folyamatra hatással vannak.) köthetőek. A komponensek külön felhasználói fiókokhoz köthetőek, aktoroknak megfelelően. A feltöltések, keresések, keresések teljesítésének, tárgyak bővítésének kezelése valamint a hozzászólások-felhasználók kezelése komponensek csak abban az esetben köthető a felhasználó aktorhoz ha az admin megváltoztatja az adott felhasználó jogkörét. Különben nem férhet hozzá az adott komponensekhez.

## 7.1.3 Use Case Dokumentum feltöltése, Kérés kezdeményezése



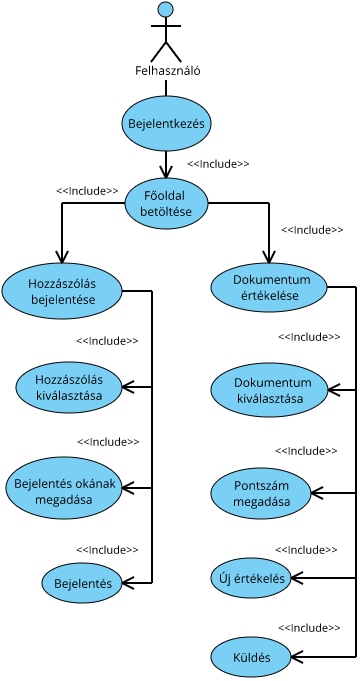
A diagramon a dokumentum feltöltése és a kérés kezdeményezése látható melyet a felhasználó végez el. A komponensek csak bejelentkezés után érhetőek el a felhasználó számára melyek a főoldalon menüpont formájában jelennek meg és egyszerre csak egy művelet hajtható végre. Ha a bejelentkezés nem lehetséges rossz jelszó vagy fiók hiánya esetén lehetőség van a jelszó megváltoztatására illetve felhasználói fiók létrehozására (regisztráció).

## 7.1.4 Use Case Kérés teljesítése, Keresés, Tantárgy hozzáadása



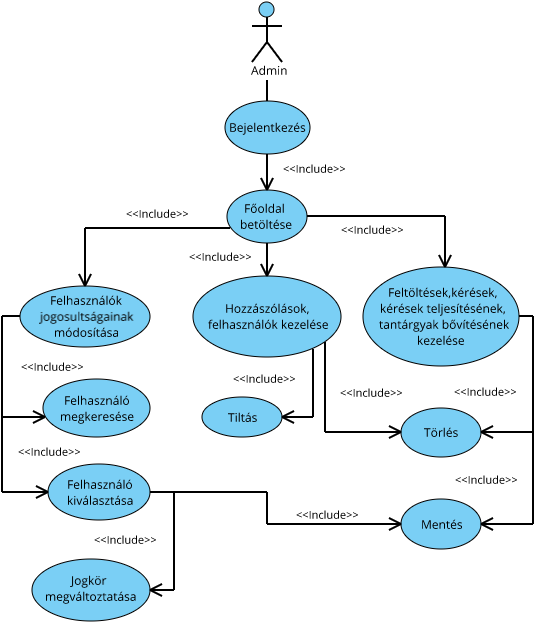
A diagramon a keresés, kérés teljesítése és a tantárgy bővítése látható melyet ismét a felhasználó végez el. Természetesen mind ezt külön időben.

## 7.1.5 Use Case Dokumentum értékelése, Hozzászólás bejelentése



A diagramon a hozzászólás bejelentése valamin a dokumentum értékelése komponensek láthatóak melyet ismét a felhasználó végez el. Bejelentésre akkor kerülhet sor, ha egy felhasználó értékelés során olyan szöveget fogalmazott meg amely nem oda illő.

## 7.1.6 Use Case Rendszer karbantartása



A diagramon az admin számára elérhető komponensek láthatóak. Az admin feladatköre eltér a sima felhasználóétól, ezáltal a főoldalon is más menüpontok kapnak helyet. Előfordulhat azonban, hogy az oldal növekedése hatására az admin egyes felhasználók jogkörét megváltoztatja, ezáltal hozzáférhetnek bizonyos további komponensekhez, ami eddig csak az admin számára volt elérhető.

## 7.1.7 Az alkalmazás használatbavételének üzleti folyamat táblázata

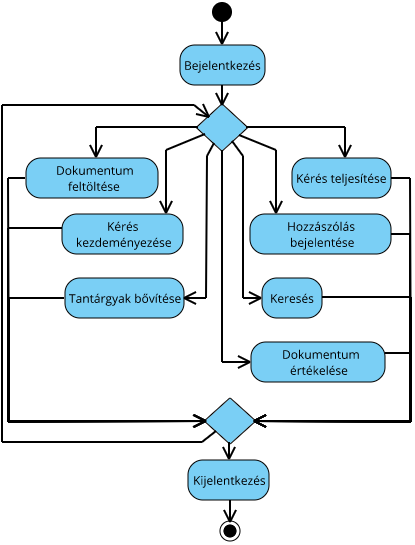
|  |  |
| --- | --- |
| **Azonosító** TIMMT-1 | **Név** Alkalmazás használatbavétele |
| **Rövid leírás** Az alkalmazás használatbavételének teljes folyamata a bejelentkezéstől egészen a kijelentkezésig. | |
| **Érintett aktorok** 1.Felhasználó 2. TIMMT | |
| **Kiváltó esemény** Az adott felhasználó tananyagot keres a tanuláshoz. Illetve az adott felhasználó tananyagot szeretne megosztani mások számára. | |
| **Előfeltétel** Ismeretszerzés, tartalommegosztás célja. | |
| **Standard lefutás** 1.Felhasználó bejelentkezése  2.Dokumentum feltöltése  2.Keresés  2.Kérés kezdeményezése  2.Kérés teljesítése  2.Dokumentum értékelése  2.Hozzászólás bejelentése  2.Tantárgy bővítése  3.Kijelentkezés | **Kivételek és alternatívák** a, A felhasználónak nem sikerül bejelentkezni vagy azért mert hibás felhasználó-jelszó párossal próbálkozik vagy azért mert még nincs is fiókja. b. A kérés teljesítése nem lehetséges, ha nincs teljesítendő kérés.  c. A keresésnél előfordulhat, hogy nincs a keresésnek megfelelő dokumentum.  d. A dokumentum értékelése csak fellelhető dokumentumok esetében lehetséges. |
| **Utófeltétel** Nincs. | |
| **Eredmény** Az alkalmazás használatbavétele sikeresen lezajlik. | |
| **Gyakoriság** Időszakoktól függ. | |
| **Utalások**  TIMMT-2 | |
| **Megjegyzések, nyitott kérdések** nincs | |

## 7.1.8 A rendszer karbantartásának üzleti folyamat táblázata

|  |  |
| --- | --- |
| **Azonosító** TIMMT-2 | **Név** Rendszer karbantartása |
| **Rövid leírás** Az alkalmazás megfelelő működésének vezérlése. | |
| **Érintett aktorok** 1. Admin 2. TIMMT | |
| **Kiváltó esemény** A felhasználók dokumentumokat töltenek fel, kéréseket nyújtanak be, teljesítenek, valamint felhasználókat jelentenek be. | |
| **Előfeltétel** Felhasználói tevékenység az oldalon. | |
| **Standard lefutás** 1.Admin bejelentkezése  2. Feltöltések,kérések, kérések teljesítésének, tantárgyak bővítésének kezelése.  2. Hozzászólások-felhasználók kezelése  2. Felhasználók jogosultságainak módosítása  3.Kijelentkezés | **Kivételek és alternatívák** a, Nem történt felhasználói tevékenység ezáltal nincs is mit ellenőrizni, kezelni.  b. Az oldal közössége nem növekszik, ezáltal nincs szükség a jogosultságok módosítására sem. |
| **Utófeltétel** Nincs. | |
| **Eredmény** Az alkalmazás megfelelő működése. | |
| **Gyakoriság** Naponta. | |
| **Utalások**  TIMMT-1 | |
| **Megjegyzések, nyitott kérdések** nincs | |

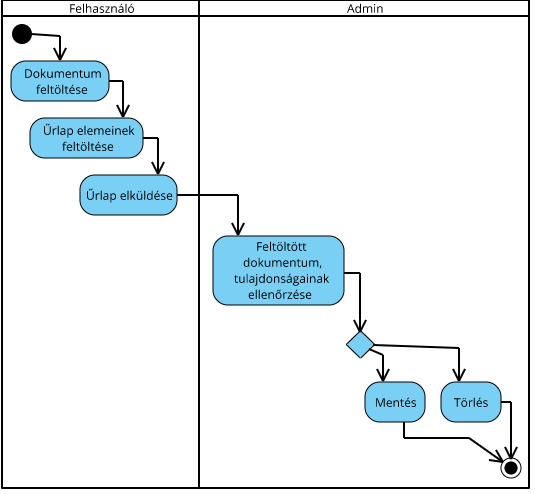
## 7.2 A rendszer folyamatainak modellezése

### 7.2.1 Alkalmazás használatbavétele



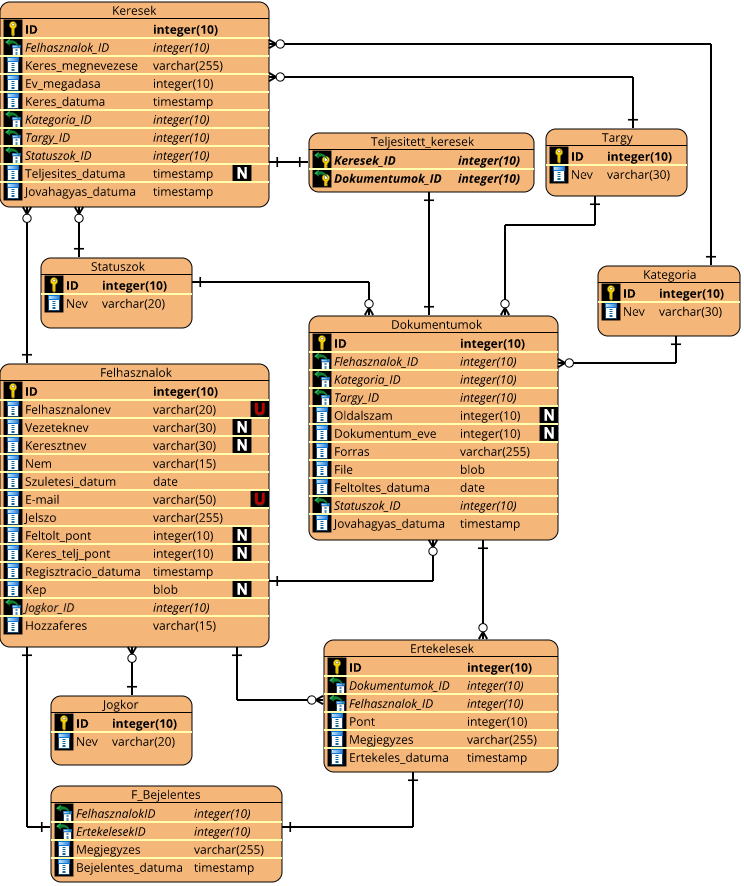
A diagramon az alkalmazás használatbavételének tevékenysége látható melyet a felhasználók végeznek el, és ami bejelentkezéssel indul és kijelentkezéssel zárul. A két szakasz között számtalan tevékenység végezhető el. Az oldal véletlenszerű bezárása esetén az azonnali újbóli megnyitáskor a bejelentkezés elhagyható.

### 7.2.2 Swimlane diagram: dokumentum feltöltése, kezelése



A swimlane diagram a dokumentum feltöltésének teljes folyamatát ábrázolja időrendi sorrendben. Dokumentum feltöltés után az admin veszi át a szerepet és ellenőrzés alá veszi a feltöltött dokumentumot. Ha a dokumentum minden elvárásnak megfelelt, akkor véglegesen eltárolásra kerül az adatbázisban és elérhetővé válik keresés alapján a felhasználók számára. Ellenkező esetben törlésre kerül az adatbázisból minden tulajdonságával együtt. A feltöltés és az ellenőrzés közötti idő az ellenőrzendő dokumentumok számától függ.

## 7.3 Logikai adatmodell



## 7.4 Osztálydiagram

# Irodalomjegyzék