Pannon Egyetem

Műszaki Informatikai Kar

Villamosmérnöki és Információs Rendszerek Tanszék

Programtervező informatikus BSc

**SZAK-/DIPLOMADOLGOZAT**

**Edzőterem menedzsment webalkalmazás fejlesztése**

**Selyem Péter Ferenc**

Témavezető: Nagy Zsuzsanna

Külső/belső konzulens: [Konzulens neve]

2025

Témakiírás

A szkennelt formában megkapott témakiírás beillesztése a dolgozatba.



Hallgatói nyilatkozat

Alulírott Selyem Péter Ferenc hallgató (Neptun kód: WOKEUE ) kijelentem, és a dolgozat feltöltésével egyidejűleg nyilatkozom, hogy a Edzőterem menedzsment webalkalmazás fejlesztése című szakdolgozatot (a továbbiakban: dolgozat) a Pannon Egyetem Műszaki Informatikai Kar Villamosmérnöki és Információs rendszerek tanszék készítettem a <<végzettség>> oklevél megszerzése érdekében.

Kijelentem, hogy a dolgozatban csak a megadott és hivatkozott forrásokat használtam fel, és ezekre a vonatkozó idézési szabályok szerint hivatkoztam.

Nyilatkozom, hogy a dolgozat érdemi része saját szellemi alkotásom eredménye, és azt más intézményben, szakon, vagy felsőfokú képesítés megszerzésére nem nyújtottam be. Tudomásul veszem, hogy a plágium vagy szerzői jogsértés esetén a dolgozatom elutasításra kerülhet, és ellenem fegyelmi eljárás indulhat. Tudomásul veszem továbbá, hogy szerzői jogsértés esetén az Egyetem jogosult a dolgozat elérhetőségét korlátozni, valamint eltávolítani a dokumentumot a dolgozatok tárolására szolgáló, a témát vezető szervezeti egység által meghatározott elektronikus zárt rendszerből.

Tudomásul veszem továbbá, hogy a Pannon Egyetem a dolgozat eredményeit saját céljaira eltérő írásbeli megállapodás hiányában a Pannon Egyetem Szellemi Tulajdon Kezelési Szabályzatában foglaltaknak megfelelően szabadon felhasználhatja.

Nyilatkozom, hogy a dolgozat elkészítése során mesterséges intelligencia eszközöket *használtam /nem használtam[[1]](#footnote-1).*

Nyilatkozom, hogy a dolgozat elkészítése során az alábbi táblázatban feltüntetett mesterséges intelligencia eszközöket kizárólag a kutatási, illetve fejlesztési feladat támogatására használtam fel, az érdemi munka, elemzés és következtetések teljes mértékben saját szellemi alkotásomat képezik.

Példa a táblázat kitöltésére:

|  |  |  |  |
| --- | --- | --- | --- |
| **Alkalmazott technológia** | **Alkalmazás módja** | **Előállított tartalom** | **MI használat aránya** |
| GPT-4o (OpenAI) | szöveges összefoglaló generálása | 2.2 fejezet | 80% |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Dátum: Veszprém, 20.. <<hónap neve>> <<nap>>.

*Selyem Péter Ferenc*

Témavezetői nyilatkozat

Alulírott Nagy Zuszsanna témavezető kijelentem, hogy a Edzőterem menedzment webalkalmazás fejlesztése című dolgozatot *Selyem Péter Ferenc* a Pannon Egyetem Villamosmérnöki és Információs Rendszerek Tanszékén készítette <<végzettség>> végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozat védésre bocsátását engedélyezem.

Dátum: Veszprém, 20.. <<hónap neve>> <<nap>>.

*Nagy Zsuzsanna*

Köszönetnyilvánítás

A hallgató köszönetet nyilvánít mindazoknak, akiktől (elméleti, gyakorlati, erkölcsi stb.) segítséget kapott.

Tartalmi összefoglaló

Az szakdolgozatom egy Edzőterem menedzsment webalkalmazás létrehozásáról,

megtervezéséről és megvalósításáról szól. Az alkalmazás ötlete 2024 nyarán jutott eszembe, amikor kezdtem komolyan edzőterembe járni és önmagamat fejleszteni. A terembe járva láttam, hogy még mindig a régimódi papírbérletet használják. A videókban, amiket megnéztem, hogy informálódjak afelől, hogy mit hogyan is kéne csinálni az edzés alatt, láttam sokféle edzőtermet. Voltak olyanok, ahol egy olyan beléptető kapu van, ahol be kell szkennelni a bérletet, hogy beengedjen. Evvel kapcsolatban beszéltem másokkal, hogy mennyivel egyszerűbb és jobb lenne egy olyan terem, ahol online lehetne rendezni a jegyeket és bérleteket. Emellett, a terem a videóban látotthoz hasonló beléptető rendszerrel rendelkezne.

A program megvalósításához a Visual Studio-t és a Visual Studio Code-ot használom, mivel ezeket már volt szerencsém használni, valamint mindkettő az egyik legjobb és legelterjedtebb a maga területén. Az alkalmazást C# nyelven írom a .Net keretrendszert használva, mivel ezek modernek és elterjedtek a hasonló alkalmazások megvalósításához. Emellett a .Net keretrendszerben sok segítséget és bővítményt lehet találni, így ezt találtam a legalkalmasabbnak a feladat backend részéhez. Adatbázis kezeléshez SQLite alkalmazást választottam, mert ingyen elérhető és tökéletes a feladatra. A frontend részhez pedig a jól megszokott és bevált HTML, CSS és JavaScript nyelveket használom, és mellette a Bootstrap-et az egyszerűbb, esztétikusabb megvalósításért és a jobb felhasználói élményért.

Tartalmi összefoglaló magyarul. Az összefoglalónak tartalmaznia kell (rövid, velős és összefüggő megfogalmazásban) a következőket:

* téma megnevezése,
* megoldott feladat megfogalmazása,
* megoldási mód,
* elért eredmények,
* kulcsszavak (4-6 darab)
* terjedelme nem lehet több 1 A4-es oldalnál.

Az összefoglalót magyar és angol nyelven kell készíteni. Sorrendben a dolgozat nyelvével megegyező kerül előrébb. A cím Title stílusú, formázása: Times New Roman, nagybetű, 14 pt, félkövér, középre igazított; az összefoglaló Normál stílusú, formázása: Times New Roman, 12 pt, sorkizárt, 1.5-ös sortávolság.

**Kulcsszavak:** [4-6 kulcsszó felsorolása, vesszővel elválasztva]

Abstract

Abstract in English

**Keywords:** [list 4-6 keywords]

Tartalomjegyzék

[Jelölésjegyzék 8](#_Toc99107193)

[1. Fejezet 9](#_Toc99107194)

[1.1. Alfejezet 9](#_Toc99107195)

[1.1.1. Alfejezet 9](#_Toc99107196)

[2. Új fejezet 10](#_Toc99107197)

[2.1. Új alfejezet 10](#_Toc99107198)

[3. Instrukciók 11](#_Toc99107199)

[Irodalomjegyzék 12](#_Toc99107200)

[Mellékletek 13](#_Toc99107201)

[Ábrajegyzék 14](#_Toc99107202)

[Táblázatjegyzék 15](#_Toc99107203)

Jelölésjegyzék

|  |  |
| --- | --- |
| AI: | Artificial Intelligence (Mesterséges Intelligencia) |
| GPU: | Graphical Processing Unit (Grafikus Processzor / Grafikus Feldolgozó Egység) |
| API: | Application Programming Interface (Alkalmazásprogramozási Felület) |
| CPU: | Central Processing Unit (Központi Feldolgozó Egység / Processzor) |
| GUI: | Graphical User Interface (Grafikus Felhasználói Felület) |
| HCI: | Human Computer Interaction (Ember-gép kapcsolat) |
| CIS: | Cognitive Information System (Kognitív információs rendszer) |

# Felhasznált Technológiák

A **szöveg** normál stílusú: Times New Roman, 12 pt, 1.5-ös sortávolságú, sorkizárt. A változók szövegben dőlt betűvel szerepeljenek. Az új bekezdés első sora behúzással új sorban, nem előzi meg üres sorköz (Normál stílusban beállítva).

**Címek** értelemszerűen számozva, Heading 1: 14 pt, Times New Roman, félkövér, további Heading: 12 pt, félkövér, Times New Roman, minden cím előtt és után a cím stílusában vannak beállítva a sorközök, cím utáni első bekezdés stílusa First paragraph.

Általános szabályok:

* minden műveleti jelet (számtani, halmazelméleti stb.) megelőz és követ egy-egy szóköz
* minden írásjelet (pont, vessző, kérdőjel, stb,) követ egy szóköz
* a zárójelek: normál (nem dőlt)

**Nyelvi ajánlás**: magyar ill. angol nyelv szempontjából a Magyar Helyesírási Szabályzat, ill. a megfelelő – brit, amerikai stb. – angol nyelvi szabályzat. **Terjedelem**: a tartalmi rész legalább 40 oldal, de legfeljebb 60 oldal. **Margók**: normál (felső, alsó, bal és jobb oldali margók is egyaránt 2,54 cm-esek, a kötésmargó 1 cm.) **Oldalszámozás**: középre alulra. Fejléc tartalma fejezetenként a fejezetcímek középre rendezve.

## Visual Studio

A backend részhez Visual Studio-t használok, ami a Microsoft által fejlesztett integrált

fejlesztőkörnyezet (IDE). Ezt széles körben használják különféle szoftverek fejlesztésére,

beleértve az asztali, mobil és webes alkalmazásokat is. Ez az eszköz különösen népszerű a .NET fejlesztések során, például C#, ASP.NET és más Microsoft technológiák használatakor, hiszen átfogó eszközkészletet kínál a fejlesztés minden szakaszára; a kódírástól kezdve a hibakeresésen át egészen az alkalmazások telepítéséig. A Visual Studio támogatja a különböző nyelveket és platformokat és megkönnyíti a fejlesztést, ezáltal ideális választás kezdő és tapasztalt fejlesztők számára is.A tartalmi részt a témavezető és a hallgató közösen határozzák meg, mely a jelölt idézetek nélkül legalább a dolgozat 2/3-a, legalább 40 oldal.

Néhány mondatnál hosszabb szövegszerű idézeteket az érdemi részbe berakni NEM szabad. Hosszabb idézetet a mellékletben kell elhelyezni.

### Alfejezet

Ábrák, képletek középre rendezve, feliratozva és számozva kerüljenek a dolgozatba. A felirat Caption stílusú és az ábra alatt foglal helyet.

Diagram

Description automatically generated

1. ábra: ábrafelirat (ha szükséges, akkor a forrás megjelölésével) [1]

#### Alfejezet

A táblázatok középre rendezve, feliratozva és számozva kerüljenek a dolgozatba. A táblázat felirata a táblázat felett helyezkedik el.

Táblázaton belül a szöveg függőlegesen középre igazítva. Az adatok vízszintes igazítását az adattartalom határozza meg (decimális értékek esetén javasolt a decimális igazítás). [2] [3]

1. táblázat megnevezése, ha szükséges hivatkozás a forrásra

|  |  |  |
| --- | --- | --- |
|  | **Mért adatok** |  |
| **Adatok 1.** | **Adatok 2.** | **Adatok 3.** |
| 1,5 | 2,1 | 1,8 |

## C# programozási nyelv

Az általam választott programozási nyelv a C#, amivel van egy kevés tapasztalatom és azt mondják, hogy ez az egyik legjobb nyelv webalkalmazás fejlesztésre. Emellett sok

segédanyag található az interneten hozzá, ezért erre a nyelvre esett a választásom. A C# egy általános célú, modern és objektumorientált programozási nyelv, amelyet a Microsoft fejlesztett ki 2000-ben, és része a .NET ökoszisztémának. A célja, hogy egyszerre legyen könnyen használható a kezdő programozók számára, és hatékony eszköz a professzionális fejlesztők kezében, akik komplex alkalmazásokat készítenek. A C# szintaxisa tiszta és intuitív. Ez lehetővé teszi a fejlesztők számára, hogy gyorsan elsajátítsák a nyelvet, különösen, ha már rendelkeznek előzetes tapasztalattal más programozási nyelvekkel. A C# teljes mértékben támogatja az objektumorientált programozást (OOP), amely lehetővé teszi a fejlesztők számára, hogy moduláris, újra felhasználható és karbantartható kódot hozzanak létre. A C#-ot eredetileg a Windows ökoszisztéma számára tervezték, de a .NET Core (és az újabb .NET verziók) megjelenésével a nyelv platformfüggetlenné vált, így támogatja a Windows, macOS és Linux rendszereket is. Az ASP.NET Core segítségével a C# ideális webalkalmazások és API-k fejlesztésére csak úgy mint asztali alkalmazások

mobilalkalmazások vagy játékok fejlesztésére.

## ASP.NET Core

Az előbb is emlegetett ASP.NET a Microsoft által fejlesztett nyílt forráskódú, nagy teljesítményű keretrendszer webes alkalmazások fejlesztésére. Az ASP.NET segítségével

dinamikus, modern, skálázható és biztonságos webalkalmazások, API-k, valamint valós idejű szolgáltatások fejleszthetők különféle platformokon (pl. Windows, macOS és Linux

rendszereken).

Az ASP.NET Core az egyik leggyorsabb webes keretrendszer, köszönhetően a fejlett optimalizálási technikáknak, például az aszinkron feldolgozásnak, a Kestrel webszervernek és a hatékony memóriahasználatnak. ASP.NET-ben lehetőség van különböző csomagokat becsatolni a programunkhoz, amik megkönnyítik a fejlesztést.

## Visual Studio Code

A frontend megvalósításához a Visual Studio Code nevű programot használom, ami egy nyílt forráskódú, könnyűsúlyú, de rendkívül erőteljes szövegszerkesztő és fejlesztőkörnyezet (IDE), amelyet a Microsoft fejlesztett. 2015-ben jelent meg, és gyorsan az egyik legnépszerűbb eszközzé vált a fejlesztők körében, köszönhetően a sokoldalúságának, bővíthetőségének és gyorsaságának.

A Visual Studio Code támogatja a különféle programozási nyelveket és fejlesztési környezeteket, mint például JavaScript, Python, C#, Java, HTML/CSS, és még sok más. Az Extension Marketplace segítségével rengeteg bővítményt lehet hozzáadni, amik segítik a fejlesztést akár csak annyival, hogy átláthatóbbá vagy színesebbé teszik a kódot. Vannak olyan bővítmények is, amelyek a kód írása közben próbálják kitalálni, hogy mit akarunk és javaslatokat ad, amiket ha elfogadunk, akkor ki sem kell írni az akár több soros parancsokat.

## Bootsrap

A Bootstrap egy népszerű, nyílt forráskódú front-end keretrendszer, ami lehetővé teszi, hogy gyorsan és hatékonyan építsünk fel vizuálisan tetszetős és mobilbarát felületeket HTML, CSS és JavaScript kombinációjával. Az előre definiált stílusok és komponensek segítségével időt takaríthatunk meg, miközben a webes szabályokat is betartjuk. Ezeket egyszerűen csak be kell csatolni a HTML fájlba és már használhatjuk is a weboldalunk fejlesztésére.

# Hasonló oldalak vizsgálata

## Flex Gym

Flex Gym oldala egy magyar példa hasonló oldalra. Nekem nem tetszik, hogy sok a kép a főoldalon, de a viszont vannak jó megoldásaik. Tetszik, hogy a nyitva tartás megtalálható az oldal alján egyszerűen, de szerintem mehetne egy menüpont alá, az elérhetőségekkel együtt, amik viszont alig látszódnak a menüben.

A jegyárak egy menüpont alatt vannak, amit hasonlóan oldanék meg én is, de a főoldalra

kitenném 1-2 gyakoribb jegy árát. Online vett jegyeket, ha minden igaz, e-mailben küldik el. Ezt én inkább az oldalon oldanám meg, a profilban, ahol egy „jegyeim” menüpont lenne megtalálható. Az „edzőink” (2. ábra) menüpontban tetszik, hogy felsorolja az edzőket és a bemutatkozásaikat. Szerintem, ha lenne pár vélemény emberektől, akiket edzettek, akkor jobb lenne.

A képen szöveg, Emberi arc, ember, személy látható

Automatikusan generált leírás1. ábra: A Flex Gym főoldala

A képen szöveg, Emberi arc, nő, ruházat látható

Automatikusan generált leírás

2. ábra: A Flex Gym „Edzőink” oldala

## Planet Fitness

Planet Fitness egy külföldi példa, aminek a főoldalán (3.ábra) kint vannak az árak. Ezáltal

A képen szöveg, ruházat, képernyőkép, lábbelik látható

Automatikusan generált leírásegyből látni mennyivel kell számolni, ha oda akar menni az ember. Szerintem itt is sok képet tettek ki az oldalra. Legalábbis sok helyet foglalnak el. Viszont a nyitvatartás eléréséhez először ki kell keresni, hogy melyik terembe tervezünk menni és ott lesz leírva, ami érthető. Szerintem egy átlagos nyitvatartást írhattak volna egy könnyebben elérhető helyre.

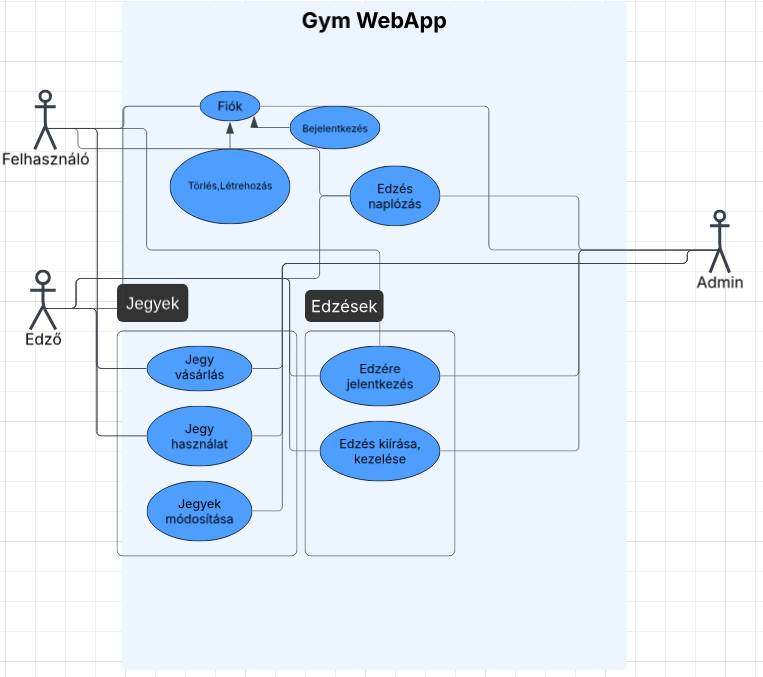
3. ábra: A Planet Fitness főoldala

# Rendszerterv

A webalkalmazásban három féle felhasználói jogosultság és szerepkör található.

* felhasználó, aki tud fiókot létrehozni, ebbe be és kijelentkezni és tudja törölni a saját fiókját. tud még edzést naplózni, jegyeket venni és felhasználni továbbá kiírt edzésekre jelentkezni.
* Edzői jogosultság az alap felhasználói képességek mellett tud edzéseket kiírni és neki meg kell adni a telefonszámát, milyen edzés típusra szakosodott és képet is feltehet magáról.
* Admin jogosultsággal egy személy az edzői jogokon kívül tud még másik személy fiókját törölni, jegyeket módosítani, új jegy típust hozzáadni, edzéseket kiírni és kezelni és mások jogosultságát kezelni

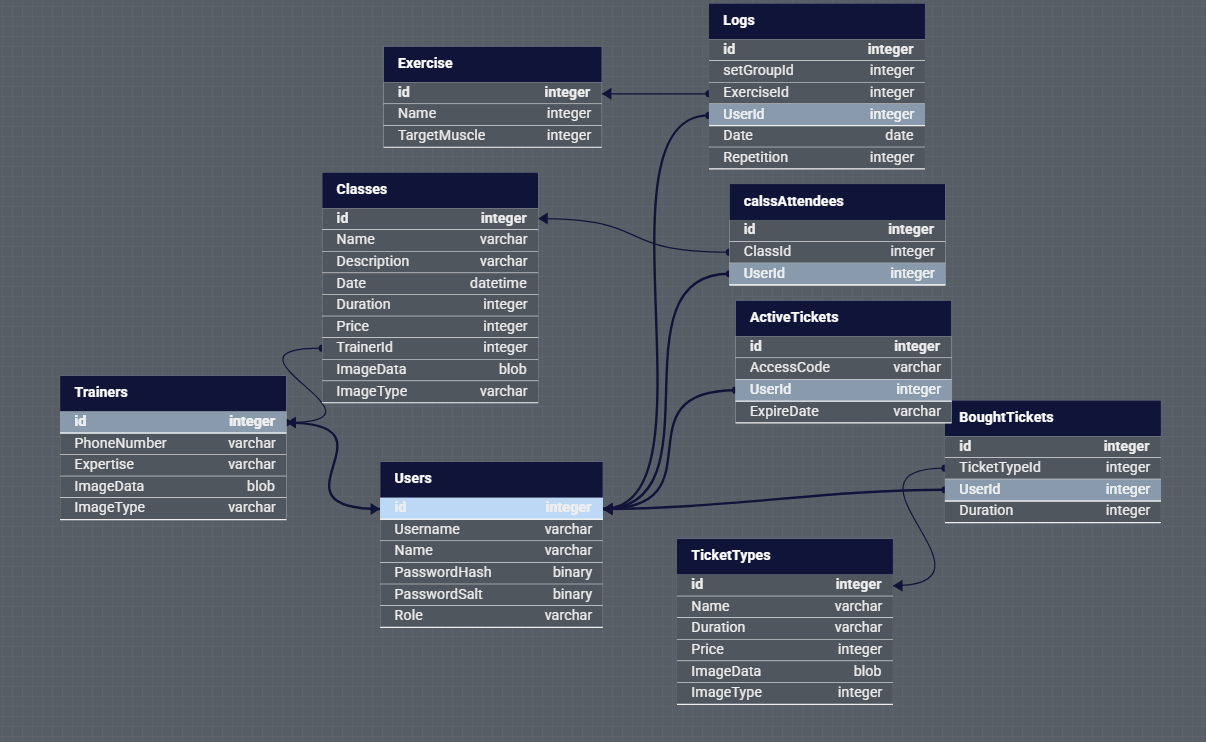
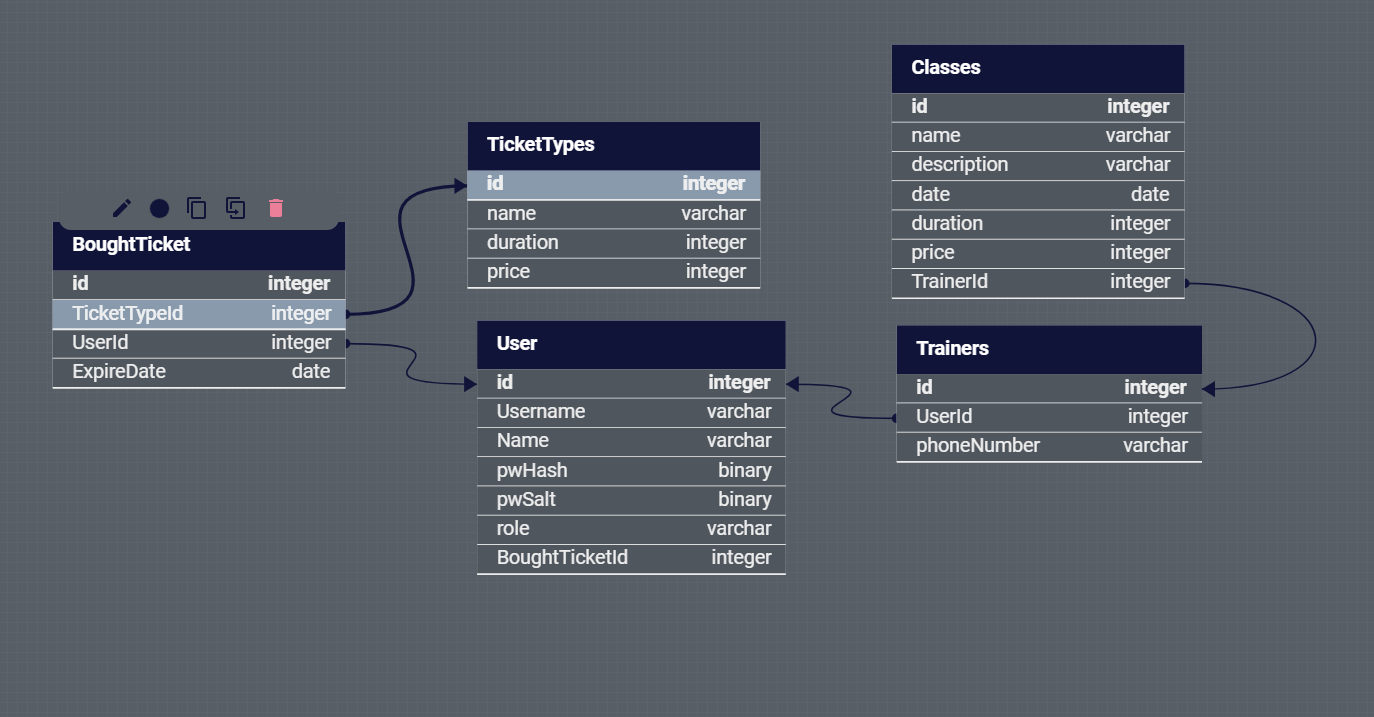
4. ábra: Az alkalmazás Use Case diagramja



# Adatbázis

Az adatokat adatbázisban tárolom, ami kezdetileg 5 táblából állt, de a mostani változat már 9 táblát foglal magába.

5. ábra: Az adatbázis kezdetleges



6.ábra: Az adatbázis jelenlegi felépítése

## Users tábla

* id, egyedi azonosító minden felhasználónak
* Username, a felhasználónév
* Name, felhasználó neve
* passwordHash/-Salt , a jelszó titkosítva
* Role, a felhasználó jogosultsága

## Trainers tábla

* id, egyedi azonosító és idegen kulcs a User táblával összekötve
* PhoneNumber, az edző telefonszáma
* Expertise, az edző szakosodása
* ImageData, edző képe
* ImageType, a kép fájl formátuma

## Classes tábla

* id, az edzés egyedi azonosítója
* Name, edzés megnevezése
* Description, leírás az edzésről
* Date, az edzés időpontja
* Duration, az edzés időtartama
* Price, az edzés ára
* TrainerId, az edzéshez rendelt edző azonoítója
* ImageData, kép az edzéshez
* ImageType, kép fájlformátuma

## ClassAttendees

* id, egyedi azonosító
* ClassId, adott edzés azonosítója, amihez személyt rendelünk
* UserId, az edzéshez rendelt személy azonosítója

## TicketTypes

* id, az adott jegy egyedi azonosítója
* Name, jegy megnevezése
* Duration, jegy élettartame
* Price, jegy ára
* ImageData, kép a jegyhez
* ImageType, a kép fáj lformátuma

## BoughtTickets

* id, megvásárolt jegy azonosítója
* TicketTypeId, jegy típus azonosító
* UserId, a felhasználó azonosítója, aki megvette a jegyet
* Duration, visszamaradt idő/használat

## ActiveTickets

* id, egyedi azonosytó
* AccessCode, a felhasznált jeggyel kapott belépési kód
* UserId, a jegy tulajdonos azonosítója
* ExpireDate, a jegy érvényességének ideje

## Logs

* id, egyedi azonosító
* setGroupId, egy gyakorlatbol összecsoportosítja a setteket
* ExerciseId, adott gyakorlat azonosítója az Exercise táblából
* UserId, a felhasználó azonosítója aki felvitte a naplózást
* Data, a gyakorlat dátuma
* Repetition, gyakorlat ismétlészáma

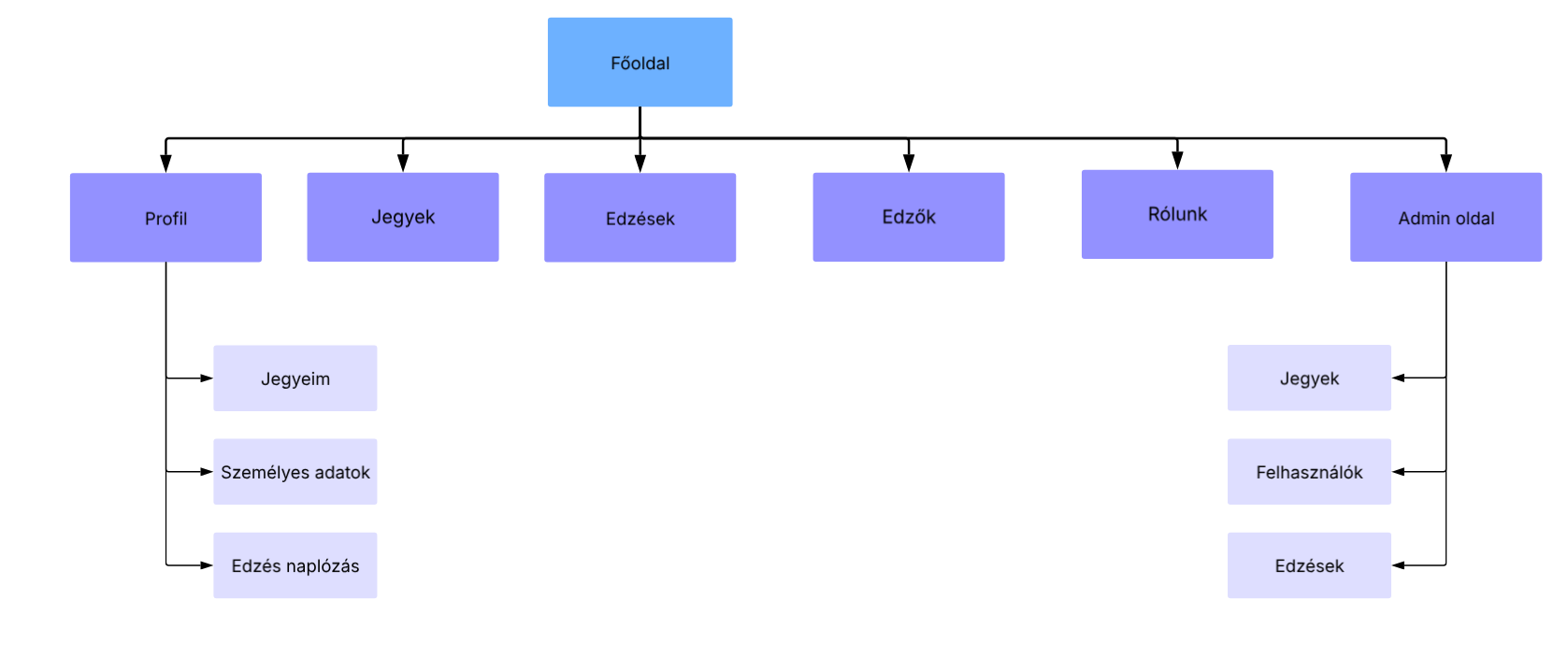
## Exercise

* id, egyedi azonosító
* Name, a gyakorlat neve
* TargetMuscle, melyik izmot edzi az adott gyakorlat

# Megvalósítás

## Oldal Térkép

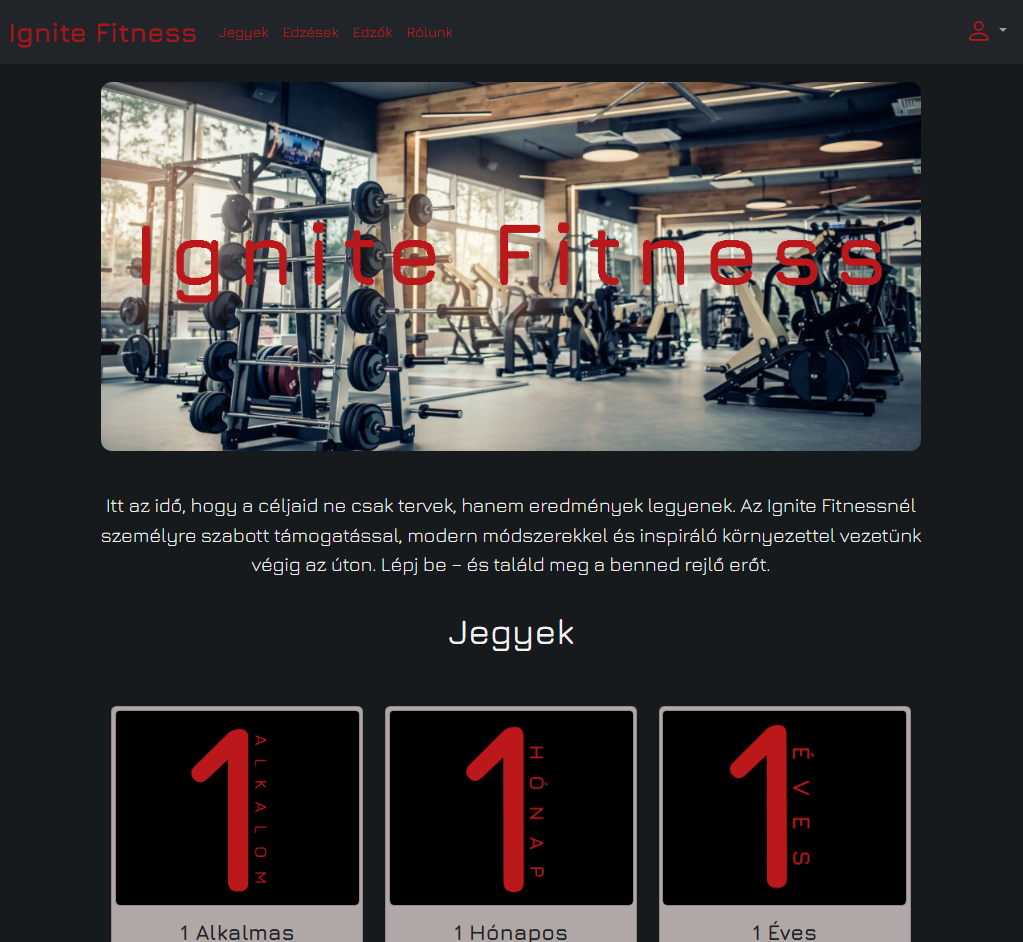
Az oldal térképe az alábbi ábrán (7. ábra) látható.



7.ábra: az oldal térképe

A Főoldalról (8. ábra) minden külön oldal elérhető egyszerűen.

8.ábra: Főoldal



A jobb feni emberkére kattintva előjön egy kis lista aho eleinte bejelentkezni tudunk vagy elérni a regisztrációs oldalt, bejelentkezést követően meg el tudjuk érni a profilunkat.

|  |  |
| --- | --- |
|  | C:\Users\selyemp\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logged in.PNG |
| 9.ábra: Bejelentkező mező | 10.ábra lenyíló menü bejelentkezés után |



11.ábra: Regisztrációs oldal

# Új fejezet

## Új alfejezet

Az irodalomjegyzék szó Title stílusú. A jegyzék automatikusan generált. Az irodalomjegyzék hivatkozások a szövegben szögletes zárójelben számokkal megadva kerülnek elhelyezésre. Az irodalomjegyzék sorrendjét a szövegbeli említés határozza meg.

Az internetcímeket aláhúzás nélkül kell. Internetcímek esetén a weblap utolsó felkeresésének dátumát az url-t követően zárójelben meg kell adni.

Az egyenletek sorszámozása az egyenlet alatt, jobbra igazítva, a minta szerint (1).

(

(1)

# Instrukciók

Tanszékek:

* Alkalmazott Informatikai Tanszék (Nagykanizsa)
* Informatikai Rendszerek és Alkalmazásai Tanszék (Zalaegerszeg)
* Matematika Tanszék
* Rendszer- és Számítástudományi Tanszék
* Villamosmérnöki és Információs Rendszerek Tanszék

Szakok nevei:/ végzettség megnevezése

* Gazdaságinformatikus BSc - gazdaságinformatikus
* Mérnökinformatikus BSc - mérnökinformatikus
* Programtervező informatikus BSc – programtervező informatikus
* Villamosmérnök BSc - villamosmérnök
* Üzemmérnök informatikus BProf – üzemmérnök-informatikus
* Mérnökinformatikus MSc – okleveles mérnökinformatikus
* Programtervező informatikus MSc – okleveles programtervező informatikus

Irodalomjegyzék

|  |  |
| --- | --- |
| [1] | T. De Smedt and W. Daelemans, “Pattern for python.,” *The Journal of Machine Learning Research,* vol. 13, no. 1, pp. 2063-2067, 2012. |
| [2] | „Záróvizsga információk,” [Online]. Available: https://mik.uni-pannon.hu/index.php/hu/oktatas/zarovizsga.html. [Hozzáférés dátuma: 04 03 2022]. |
| [3] | D. J. Wetherall és A. S. Tanenbaum, Computer networks, Pearson Education, 2013. |

Mellékletek

Mappaszerkezet

+chatbot

| backen.bat

| backend.py

| files.doc

| fixedlinks.json

| ipcheck.py

| linkek.json

| linkfix.py

| log.txt

| sqlwriter.py

| textprocessor.py

|

+---backend

| pyvenv.cfg

+---lara

| | entities.py

| | nlp.py

| | parser.py

| | stemmer.py

+---static

| +---css

| | chat.css

| |

| \---js

| chat.js

|

+---templates

| index.html

|

[PÉLDA!!! Megjegyzés: A Python csomagkezelője által telepített fájlok, illetve a különböző cache fájlok a fenti listából kimaradtak, mivel ezekkel indokolatlanul és aránytalanul hosszú lenne a fenti felsorolás. A beadott fájlok között azonban a teljesség kedvéért szerepelnek ezek a fájlok is.]

Ábrajegyzék

[1. ábra: ábrafelirat (ha szükséges, akkor a forrás megjelölésével) [1] 1](#_Toc97890941)

Táblázatjegyzék

[1. táblázat (forrás megjelölésével) [2] 1](#_Toc97890942)

1. A megfelelő rész aláhúzandó. [↑](#footnote-ref-1)