**Szegedi Tudományegyetem**

**Informatikai Intézet**

**szakdolgozat**

**Szabó Krisztián Dávid**

**2025**

**Szegedi Tudományegyetem**

**Informatikai Intézet**

**„TaskManager” feladatkezelő webalkalmazás**

Szakdolgozat

*Készítette:* *Témavezető:*

**Szabó Krisztián Dávid** **Pénzes János**

Gazdaságinformatika szakos hallgató üzletági igazgató

SZEGED

2025

# Feladatkiírás

A szakdolgozat keretein belül a hallgató egy feladatkezelő webalkalmazást fejleszt, amely lehetővé teszi nagy volumenű projektek részfeladatokra bontását és azok menedzselését.

A hallgató szakdolgozatát a GriffSoft Informatikai Zrt.-nél készíti. A cég fő terméke a Forrás ügyviteli rendszer, amelyet főként Önkormányzatok, Költségvetési intézmények és középvállalatok használnak.

Az ügyviteli rendszer fejlesztése során számos esetben felmerül az igény, hogy legyen egy alkalmazás, amelyben a cég dolgozói átláthatóan és könnyen tudják kezelni a projekteket, illetve a projektekhez tartozó részfeladatokat.

Ennek megoldására készít a hallgató egy feladatkezelő webalkalmazást, amelyben a projekteket részfeladatokra bonthatjuk és a részfeladatok haladását nyomon követhetjük.

# Tartalmi összefoglaló

* ***A téma megnevezése:***

„TaskManager” feladatkezelő webalkalmazás

* ***A megadott feladat megfogalmazása:***

A feladat célja egy olyan felhasználóbarát webalkalmazás fejlesztése, amely lehetővé teszi a felhasználó számára, hogy hatékonyan és szervezetten kezelje a projekteket és a hozzájuk tartozó feladatokat. Lehetőséget kínál emellett a különféle feladatokhoz prioritást, munkaszámot és határidőt rendelni.

* ***A megoldási mód:***

A keretrendszer által nyújtott technikák segítségével a program mindig napra készen tudja tartani a feladatokat. A feladatokhoz rendelt munkaszámokat a cég által használt Azure DevOps felhőalapú platformról szinkronizálja a webalkalmazás.

* ***Alkalmazott eszközök, módszerek:***

A fejlesztés Visual Studio 2022 nevű fejlesztő környezetben, C# nyelven készült. A webalkalmazás elkészítésében a Blazor keretrendszer és a hozzá tartozó nyílt forráskódú MudBlazor komponens könyvtár nyújtott segítséget. A feladatok eltárolásában az SQL Server Management Studio 19 adatbázist kezelő rendszer segített.

* ***Elért eredmények:***

A webalkalmazást a GriffSoft Informatikai Zrt. használja céges környezetben.

* ***Kulcsszavak:***

webalkalmazás, feladatkezelés, prioritások, határidők, felhasználóbarát

# Tartalomjegyzék

[Feladatkiírás 1](#_Toc163562086)

[Tartalmi összefoglaló 2](#_Toc163562087)

[Tartalomjegyzék 3](#_Toc163562088)

[Bevezetés 4](#_Toc163562089)

[1. Felhasznált módszerek, eszközök bemutatása 5](#_Toc163562090)

[1.1. Keretrendszer 5](#_Toc163562091)

[1.1.1. Blazor 5](#_Toc163562092)

[1.1.2. .NET 5](#_Toc163562093)

[1.2. Komponens könyvtár 5](#_Toc163562094)

[1.2.1. MudBlazor 6](#_Toc163562095)

[1.3. Programozási nyelv 6](#_Toc163562096)

[1.3.1. C# 6](#_Toc163562097)

[1.3.2. Egyéb programozási nyelvek 7](#_Toc163562098)

[2. A fejlesztés menetének bemutatása 7](#_Toc163562099)

[2.1. Backend 7](#_Toc163562100)

[2.2. Frontend 7](#_Toc163562101)

[3. Elért eredmények bemutatása 8](#_Toc163562102)

# Bevezetés

A GriffSoft Informatikai Zrt. fő terméke a Forrás ügyviteli rendszer, amelyet főként Önkormányzatok, Költségvetési intézmények és középvállalatok használnak. A fejlesztés magában foglal több, nagy volumenű projektet, amelyek menedzselése gyakran összetett és nehezen átlátható.

A növekvő feladatmennyiség következtében a projektvezetőknek könnyebbséget jelentene egy olyan applikáció, amely strukturált és átlátható módon kezeli az egyes projekteket és a hozzájuk tartozó feladatokat. Ezen igény felmerülése miatt esett a választás a „TaskManager” feladatkezelő webalkalmazás elkészítésére, amely fejlesztése 2022-ben vette kezdetét.

A „TaskManager” webalkalmazás célja egy olyan átfogó, hatékony és felhasználóbarát feladatkezelő rendszer fejlesztése, amely lehetővé teszi a projektek és feladatok strukturált kezelését, a csapatmunka optimalizálását, illetve a hatékony időmenedzsmentet. A programnak lehetővé kell tennie a határidők nyomon követhetőségét, illetve a kommunikációt a projektben részvevő emberek számára.

Eredetileg a fejlesztés egy gyakorlófeladatként indult számomra, hogy minél mélyebben belemerülhessek a C# programozási nyelvbe, a Blazor keretrendszerbe, valamint a MudBlazor komponens könyvárba. A fejlesztés előrehaladtával a program 2023-ban már használható volt.

A webalkalmazás első verziója 2024-ben készült el, amikor egy beépített szűrőrendszer is implementálásra került, amiben a feladatok státuszai szerint lehet szűrést végezni. Emellett külön funkcióként került bele, hogy egy kereső mező segítségével minden feladathoz felelőst lehet beállítani a cég dolgozói közül.

# Felhasznált módszerek, eszközök bemutatása

## Keretrendszer

A keretrendszer egy előre elkészített struktúra. Lényege, hogy a különböző alkalmazásokban leggyakrabban használt elemeket egy helyen gyűjtik össze és így kínálják fel a fejlesztők számára. A keretrendszerek sok olyan funkcionalitást és eszközt kínálnak, amelyek újra felhasználhatók, így a fejlesztőknek ezeket nem kell minden egyes programnál újra implementálniuk. Ezáltal a keretrendszerek növelik a fejlesztési folyamat hatékonyságát és csökkentik a fejlesztési időt.

### Blazor

A Blazor egy Microsoft által fejlesztett, nyílt forráskódú keretrendszer, amely lehetővé teszi a C# és a .NET alkalmazások kliens és szerver oldali fejlesztését.

Két fő változata létezik, a Blazor Server és a Blazor WebAssembly. A különbség a kettő között, hogy a Blazor Server szerver-oldali, a Blazor WebAssembly pedig kliens-oldali alkalmazások felépítéséhez nyújt segítséget. A TaskManager alkalmazás Blazor Server architektúrát használ, hiszen ebben a megoldásban az üzleti logika és az adatok a szerveroldalon tárolódnak el, így a kliens oldalon kevesebb érzékeny információ található.

### .NET

A .NET szintén a Microsoft fejlesztése alatt álló, nyílt forráskódú, platform független keretrendszer.

Legfrissebb verziója a .NET 8, amely képes futtatni a C# kódot, illetve számos NuGet csomag használatára ad lehetőséget. A NuGet csomagok harmadik féltől származó külső könyvtárak és eszközök. Ilyen például a szakdolgozatban is használt „EntityFrameworkCore”, amely az adatbázisok kezelését egyszerűsíti meg.

## Komponens könyvtár

A komponens könyvtárak olyan csomagok, amelyek előre elkészített UI komponenseket tartalmaznak, amiket a fejlesztő testre tud szabni, illetve be tudja illeszteni az alkalmazásába. Céljuk megkönnyíteni és felgyorsítani a fejlesztés menetét.

### MudBlazor

A MudBlazor egy ingyenes, nyílt forráskódú Blazor komponens könyvtár, amely testre szabható UI komponenseket biztosít a Blazor alkalmazások számára.

A TaskManager felhasználói felülete az ebben a könyvtárban található komponensekből épül fel. Ezek a komponensek reszponzív megjelenítést biztosítanak, így különböző eszközökön és képernyőméreteken is jól működnek. Nem elhanyagolható, hogy a MudBlazor könyvtárhoz tartozik egy mindent átfogó dokumentáció, amely rendkívül hasznos példákon keresztül mutatja be a különböző komponensek használatát.

## Programozási nyelv

Általánosságban beszélve a programozási nyelvek lehetőséget adnak kommunikációra a számítógéppel.

Hasonló módon, mint az emberi nyelv, a programozási nyelv is rendelkezik egy saját szabályrendszerrel, aminek a betartásával lehetősége van a fejlesztőnek megfogalmazni egy utasítást vagy utasítások sorozatát, egy programot. A szabályrendszer felhasználásával a megírt kódot a számítógép lefordítja a saját nyelvére, majd elvégzi a kiadott utasításokat.

### C#

A C# egy modern, objektumorientált programozási nyelv, amelyet a Microsoft fejlesztett ki a .NET keretrendszer részeként.

A nyelv platformfüggetlen, így Windows, Linux, MacOS, IOS, és Android operációs rendszerekben is alkalmazható. A C# tiszta, jól strukturált szintaxissal rendelkezik, így könnyen tanulható a kezdő programozók számára is.

Az objektumorientáltság miatt a C# nyelven megírt alkalmazások komponensekre bonthatók, ezzel növelve a kód újra felhasználhatóságát és karbantarthatóságát.

A C# programozási nyelvet folyamatosan fejlesztik és támogatja a modern fejlesztési szabványokat és tervezési mintákat is. Lehetővé teszi az aszinkron programozást, így az alkalmazásunk több szálon futhat.

Szintén beépített szolgáltatás a LINQ, ami az SQL-hez hasonló lekérdező nyelvet valósít meg. Használatával könnyen tudunk nagy mennyiségű adaton műveleteket végrehajtani.

Összegezve, ezek az előnyök teszik a C# nyelvet az egyik legnépszerűbb programozási nyelvvé a világon, különösen a vállalati és az üzleti alkalmazások fejlesztése terén.

### Egyéb programozási nyelvek

A CSS egy stílusleíró nyelv, amelyet a weboldalak és webalkalmazások megjelenésének formázására használnak. A CSS lehetővé teszi a webfejlesztők számára, hogy megadják a HTML elemek megjelenését, például a színeket, a betűtípusokat, a betűméreteket, a térközöket és az elrendezést.

Az SQL egy speciális nyelv, amelyet relációs adatbázisok kezelésére és manipulálására használnak. Az SQL segítségével adatokat lehet lekérdezni, módosítani, beszúrni és törölni az adatbázisokból. A relációs adatbázisokban az adatok táblákban vannak tárolva, és az SQL segítségével lehet kapcsolatokat létrehozni ezek között a táblák között.

A JSON egy könnyen olvasható és írható adatátviteli formátum, amelyet gyakran használnak az adatok strukturált átvitelére a webes alkalmazások és szolgáltatások között. A JSON szöveges alapú és emberi olvasási és írási szempontból is egyszerű, de mégis hatékonyan reprezentálja a strukturált adatokat.

# A fejlesztés menetének bemutatása

## Backend

A backend fejlesztés azon tevékenységek összessége, amelyek a webalkalmazások és szolgáltatások háttérkomponenseinek fejlesztésére összpontosítanak. A backend a felhasználók elől rejtett része a webalkalmazásnak, amely az adatokat tárolja, feldolgozza és szolgáltatja a felhasználóknak. Ide tartozik az adatbázis kezelés, az üzleti logika és minden más olyan folyamat vagy szolgáltatás, amit a felhasználó elől el szeretnénk rejteni.

### Adatbázis kezelés

A „TaskManager” feladatkezelő webalkalmazás a már korábban is említett „EntityFrameworkCore” keretrendszert használja, amely lehetővé teszi az adatbázissal történő kommunikációt a C# nyelven belül.

## Frontend

## Elért eredmények bemutatása