Program Specifikáció

Smart Manager

Készítette:

Magyar Márk, Nagy Huba



### Tartalomjegyzék

[Tartalomjegyzék 1](#_Toc208531755)

[1. Bevezetés 3](#_Toc208531756)

[1.1 Rövid projekt bemutatás 3](#_Toc208531757)

[1.2 Témaválasztási magyarázat 3](#_Toc208531758)

[1.3 Mi célt szolgál 3](#_Toc208531759)

[1.4 Funkciók és extrák 3](#_Toc208531760)

[1.5 Célközönség 3](#_Toc208531761)

[2. Fejlesztőkörnyezet és hardver - Fejlesztői Dokumentáció 5](#_Toc208531762)

[2.1 A fejlesztéshez használt főbb eszközök 5](#_Toc208531763)

[2.2 Fejlesztésre használt számítógép 5](#_Toc208531764)

[2.3 UI/UX tervezés – Figma 5](#_Toc208531765)

[2.4 Technikai konzultáció 5](#_Toc208531766)

[2.5 Tárolt adatok 5](#_Toc208531767)

[3. Elvárások a feladattal kapcsolatban 7](#_Toc208531768)

[3.1 Operációs rendszer, környezet 7](#_Toc208531769)

[3.2 Felhasználandó programozási nyelv 7](#_Toc208531770)

[3.3 Megoldás formátuma 7](#_Toc208531771)

[3.4 Szoftverfejlesztés 7](#_Toc208531772)

[3.5 Modulok 7](#_Toc208531773)

[4. Szoftver specifikáció 8](#_Toc208531774)

[4.1 Megjelenés 8](#_Toc208531775)

[5. Dokumentáció 11](#_Toc208531776)

[5.1 Erőforrás-terv, munkaidő nyilvántartás 11](#_Toc208531777)

[5.2 Technikai dokumentáció 11](#_Toc208531778)

[5.3 Forráskód dokumentáció 11](#_Toc208531779)

[6. Projekt adatlap 12](#_Toc208531780)

[6.1 Projekt neve 12](#_Toc208531781)

[6.2 Feladat rövid ismertetése 12](#_Toc208531782)

[6.3 Programozási nyelvek 12](#_Toc208531783)

[6.4 Specifikációt összeállította 12](#_Toc208531784)

# Bevezetés

## Rövid projekt bemutatás

A Smart Manager egy projektmenedzsment rendszert képvisel. Szoftver Megoldást nyújt, amely a tanárok és diákok közötti hatékony együttműködést segíti elő, illetve támogatja az általános felhasználást is. A rendszer célja, hogy biztosítson egy jó megoldást a projektek egy helyen tárolására.

## Témaválasztási magyarázat

Ez a munka elsősorban azért jött létre, mert sok tanár számolt be nehézségekről, illetve a diákok nagy száma és a különféle projektmunka-lehetőségek, valamint a közös munkát támogató platformok sokfélesége is kihívásokat jelent. Hiszen vegyük alapul van 4 fő 2-2 fős csapatot alkotnak az egyik csapat csak githubot használ, ahol követhető ki mit csinált és mikor, a másik csapat meg használt messengert, discordot és drive-ot, ahol nem mindig lehet vissza követni, hogy ki mikor és mit változtatott.

## Mi célt szolgál

A Smart Manager célja egy olyan projektmenedzsment rendszer létrehozása, amely egyszerre segíti a diákok tanulását és a tanárok adminisztrációs munkáját, átláthatóvá teszi a feladatokat, a statisztikákat, valamint támogatja a digitális tanulási környezetet. Továbbá remek lehetőség általános felhasználóknak is mivel, ha van egy cég és minket választ simán tudja kezelni dokumentumait, illetve egyéb munkát és alkalmazottait követni, hogy ki mikor mit csinált a projektben.

## Funkciók és extrák

Három fő funkciót egyesít: verziókezelés, feladatmenedzsment és valós idejű kommunikáció. A verziókezelő rész lehetővé teszi a projekthez tartozó fájlok rendszerezett tárolását és verziókövetését. A feladatmenedzsment egy Kanban tábla segítségével támogatja a csapatok munkaszervezését és nyomon követését. A beépített chat modul biztosítja a gyors és egyszerű kommunikációt a csapattagok között, így minden fontos eszköz egyetlen felületen érhető el.

## 

## Célközönség

A platform elsődleges célközönsége a diákok és tanárok, akik egyszerű, átlátható eszközt keresnek közös projektek, beadandók és csapatfeladatok megszervezésére. Segíti a feladatok kiosztását, a közös dokumentum- és kódrendszerezést, valamint a gyors kommunikációt. Emellett a rendszer hasznos lehet kisebb fejlesztői csapatoknak, hobbi projektekhez vagy nonprofit szervezeteknek is, ahol fontos a könnyen kezelhető, integrált munkakörnyezet.

# Fejlesztőkörnyezet és hardver - Fejlesztői Dokumentáció

## A fejlesztéshez használt főbb eszközök

IDE: Visual Studio 2022   
Célplatform: Windows, Android, IOS   
Verziókezelés: Git (GitHub)

## Fejlesztésre használt számítógép

|  |  |  |
| --- | --- | --- |
| Specifikáció | Minimum | Maximum |
| RAM: | 16GB DDR4 | 16GB DDR4 |
| OS: | Windows 10 64bit | Windows 11 64bit |
| GPU: | AMD Radeon RX 570 8g | Geforce rtx 2070 super 6gb |
| CPU: | AMD Ryzen 5 2600 | Intel Core i7-10970H |

## UI/UX tervezés – Figma

A játék felhasználói felületének és képernyőelrendezésének megtervezésére a Figma nevű online UI/UX eszközt használtuk. Ez lehetővé tette, hogy már a fejlesztés megkezdése előtt vizuálisan átgondolt terveket készítsünk a menürendszerről, szövegkörnyezetről, valamint az interaktív elemek elhelyezkedéséről. Figma miatt tudtuk elkerülni a lehetőségre okot adó hibákat hiszen mivel meg volt tervezve egy alap így tudtuk csökkenteni a felesleges újra tervezések számát és a felesleges gondolkodást.

## Technikai konzultáció

A közös munka során megbeszélésekre és egyeztetésekre a Discordot használtuk hiszen rendkívül segítőkész alkalmazás olyan téren, hogy meg tudjuk egymással osztani hívás közben a képernyőnket és élőben tudjuk követni mit csinál a másik miközben közösen agyaltunk a teendőkön.

## 

## Tárolt adatok

A játékhoz kapcsolódó összes változó adat, ide tartoznak többek között:

Felhasználói adatok (felhasználónév, jelszó, email cím)  
Projektmunka (ha feltöltésre kerül, ha nem akkor nincs eltárolva semmi)  
A feltöltött munka adatai. Pl: css, js, html, cs, xml, txt…stb.  
Valós idejű kommunikáció (Admin - User, User - User)  
Statisztika   
Aktivitás (kiírja az admin számára ki mikor volt utoljára elérhető)  
Admin, User  
To-Do táblázat adatai (Határidő, feladat státusz: Befejezett, folyamatban, Késés, ha későn adja le a User akkor számolja mennyi ideje nem adta le)  
Jogosultságok és szerepkörök  
pl: Admin, tanár, diák, projektvezető  
jogosultság szint: be lehet állítani, hogy ki mit láthat és mit módosíthat  
létrejött csapat(ok) adatai

# Elvárások a feladattal kapcsolatban

## Operációs rendszer, környezet

A rendszer többplatformos működésre lett tervezve.   
Backend: egyedileg készített API, SQL adatbázis.  
Frontend: Reszponzív web alkalmazás, amely modern böngészőkben (Chrome, Edge, Safari) és mobil eszközökön is működik.   
Asztali kliens: Windows környezetre készített WinForms alkalmazás, amely közvetlenül a backend API-hoz csatlakozik.

## Felhasználandó programozási nyelv

Backend: C#  
Frontend: Vue.js  
Asztali kliens: C# (Windows Forms / Avalonia)  
Adatbázis: SQL (MySQL)  
Valós idejű kommunikáció: Websocket (Socket.IO)

## Megoldás formátuma

A végső megoldás három komponensből áll:  
1.Webes frontend – diákok számára (projektek kezelése).  
2.Backend REST API – a központi logika és adattárolás.  
3.Asztali kliens – tanári felület (statisztikák kezelése).  
A teljes forráskód GitHub repóban kerül tárolásra, verzió követéssel és dokumentációval együtt.

## Szoftverfejlesztés

A fejlesztés során Visual Studio-t és Figma-t használunk, a kommunikáció Discordon zajlik. A tiszta kód elveit követjük, és folyamatosan teszteljük a rendszer stabilitását.

## Modulok

Felhasználókezelés (bejelentkezés, regisztráció, jogosultságok).  
Projektfeladat-kezelés (Kanban board, státuszok).  
Statisztikák és riportok (átlagpontszám, legjobb diák, export).  
Adminisztrációs felület (tanári desktop kliens).

# Szoftver specifikáció

## Megjelenés

Diákoknak windows forms-os exe állomány készült.  
Amit tartalmazni fog:  
1. Belépés  
Diák bejelentkezés / regisztráció -> e-mail + jelszóval.

2. Főoldal  
Közelgő határidők (beadandók).  
Saját státusz (hány feladat folyamatban, hány kész).  
Legutóbbi tanári közlemények.

3. Feladatkezelés  
Tanártól kapott feladatok listája.  
Feladat beadása (fájl feltöltés → pl. .zip, .docx, .cs).  
Saját beadások állapotának követése (pl. „Folyamatban”, „Elküldve”, „Értékelve”, „Elkésve”).  
Határidő figyelmeztetés (pl. piros kiemelés, ha lejárt és időzítő indul, hogy mióta nincs leadva).

4. Csapatmunka  
Kanban tábla: „Teendő – Folyamatban – Hiányzik – Kész”.  
Feladatkártyák: név, felelős, határidő.  
Közös fájlok feltöltése.  
Változásnapló: ki mikor adott hozzá/módosított valamit.

5. Chat modul

Egyéni chat diák ↔ diák.  
Csapatchat (projekt szinten).  
Tanár-diák chat (pl. kérdés beadandóval kapcsolatban).

6. Statisztika / Visszajelzések  
Beadott feladatok száma és státusza.  
Értékelések, pontszámok megtekintése.  
Grafikon: teljesítési arány, késésekkel együtt.

7. Beállítások  
Dark/Light mód.  
Profil Adatok (e-mail, jelszó).

Tanároknak Weboldal készült.  
Amit tudni kell róla:  
1. Belépés  
Tanár/Admin bejelentkezés (magasabb jogosultságokkal).

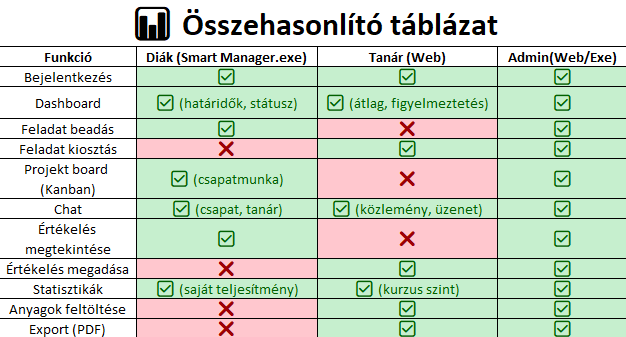
2. Főoldal  
Legutóbbi beadott feladatok.  
Figyelmeztetések (pl. „5 diák nem adta le a feladatot időre”).  
Statisztikai összegzés (átlag pontszám, teljesítési arány).

4. Feladatkiosztás  
Új feladat létrehozása (név, leírás, határidő, csatolmány).  
Feladat kiadása.  
Határidők követése → listázás, ki adta le időben.

5. Értékelés  
Beadott feladatok megtekintése (ki mikor adta le, milyen fájlt adott le).  
Pontszám / jegy megadása.  
Szöveges visszajelzés írása.  
Automatikus státusz frissítés (pl. „Értékelve”).

6. Statisztikák és riport, osztályátlag megjelenítése.  
Legjobb és legrosszabb teljesítők listája.  
Hiányzó beadások, késések kimutatása.  
Exportálás PDF-be.

7. Kommunikáció  
Üzenet küldés.  
Egyéni üzenet diákoknak.  
Visszajelzés beadott munkákhoz közvetlenül.

Összefoglalva Excel táblázatban:

# Dokumentáció

## Erőforrás-terv, munkaidő nyilvántartás

Csapatlétszám: 2 fő fejlesztő.  
Munkamegosztás: egyik fő a backend + desktop fejlesztésért felel, a másik fő a frontend + UI megvalósításáért.  
Munkaidő: heti 8–10 óra fejenként, összesen kb. 150–160 munkaóra.  
Ütemezés: 8 hét, heti mérföldkövekkel (lásd Gantt-diagram).

## Technikai dokumentáció

Adatbázismodell: ERD diagram, SQL dump.  
Rendszerarchitektúra: backend, frontend és desktop komponensek kapcsolatának leírása.  
API dokumentáció: Swagger/OpenAPI specifikáció a végpontokról.

## Forráskód dokumentáció

Kommentek: tiszta kód elvek, metóduson kenti rövid leírás.  
README fájl: telepítési és futtatási útmutató.  
Kódfelépítés: modulok és mappastruktúra bemutatása. Tesztdokumentáció: futtatott tesztek és eredmények.

# Projekt adatlap

## Projekt neve

Smart Manager

## Feladat rövid ismertetése

Projekt menedzser ,mely kiszolgál egy teljes projekt folyamat lebonyolításához szükséges minden igényt.

## Programozási nyelvek

C#...

## Specifikációt összeállította

Magyar Márk József, Nagy Huba Kende