

**Budapesti Gépészeti SZC
Eötvös Loránd Technikum**

Műszaki informatikus
54 481 05

Szűcs Erik Dániel

2024

Záródolgozat

Füstérezékelő

Szűcs Erik Dániel

Budapest

2024



Budapesti Gépészeti SZC

**Eötvös Loránd
Technikum**

**Műszaki informatikus
54 481 05**

Készítette

Szűcs Erik Dániel

Konzulens

Molnár József

KONZULENSI NYILATKOZAT

TANULÓI NYILATKOZAT

Alulírott Szűcs Erik Dániel műszaki informatikus tanuló kijelentem, hogy ezt a záró dolgozatot meg nem engedett segítség nélkül, saját magam készítettem csak a megadott forrásokat (szakirodalom, eszközök stb.) használtam fel. Minden olyan részt, melyet szó szerint vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Hozzájárulok, hogy a jelen munkám alapadatait (szerző (k), cím, magyar nyelvű tartalmi kivonat, készítés éve, konzulens (ek) neve) a BGéSZC Eötvös Loránd Technikum nyilvános hozzáférésű elektronikus formában, a munka teljes szövegét pedig az iskola belső hálóján keresztül (vagy hitelesített felhasználó számára) közzétegye. Kijelentem, hogy a benyújtott munka és annak elektronikus verziója megegyezik.

Kelt: Budapest, 2024 április 17.

.....

Szűcs Erik Dániel

TARTALOMJEGYZÉK

Konzulensi Nyilatkozat.....	5
TANULÓI NYILATKOZAT.....	6
<i>Bevezetés.....</i>	<i>8</i>
Köszönetnyilvánítás.....	8
<i>Választott téma indoklása</i>	<i>9</i>
<i>Téma kifejtése</i>	<i>10</i>
Fejlesztői környezett alkalmazásai	10
A záró dolgozatban használt eszközök	10
Segéd oldalak.....	Hiba! A könyvjelző nem létezik.
<i>rendszer bemutatása</i>	<i>12</i>
Rendszertervrajz	12
A program felépítése.....	13
A weboldal	16
Az Arduino program szerkezete, és működése.....	18
A modell szerkezet.....	18
<i>Saját vélemény</i>	<i>23</i>
Továbbfejlesztés lehetőségei	23
<i>Összefoglalás</i>	<i>25</i>
<i>Irodalomjegyzék.....</i>	<i>26</i>
<i>ÁBRAJEGYZÉK.....</i>	<i>27</i>

BEVEZETÉS

Az IoT, vagyis a dolgok internete terminus olyan elektronikai eszközöket, esetleg magas intelligenciájú eszközöket takar, amik képesek érzékelni, továbbá lényeges információkat továbbítani az internetes hálózaton keresztül más eszközök részére, hogy kommunikáljanak, esetleg feldolgozzák ezeket az adatokat adott célokra. Ezek a magas intelligenciájú eszközök bírnak beépített szenzorokkal, esetleg érzékelőkkel, amik segítségével adatokat gyűjtenek. Az IoT egy olyan technológia, mely folyamatosan terjed a modern világban, ezen kívül újabb lehetőségeket kínál az életünk megkönnyítésére, valamint az online tér kiterjesztésére. Ilyen eszközök közé tartoznak példának okáért az okosórák, okosriasztók, okosfüggönyök, illetve okosfűtésvezérlők ami az általam vizsgált témát képezi.

Az IoT jelentőségét tovább növeli az általa kínált mindenre kiterjedő alkalmazások palettája, amik megkönnyítik, valamint hatékonyabbá teszik mindennapi tevékenységeinket. Az okos otthonokban például az IoT támogatásával az eszközök egymással való kommunikációja lehetővé teszi, hogy távolról is vezérelhessük otthonunkat, például távolról bekapcsolhatjuk a fűtést, esetleg megfigyelhetjük a ház biztonságát. Ezzel együtt a gazdaságban is nagy szerepet kap, javítja a termelés és az üzemeltetés hatékonyságát.

Köszönetnyilvánítás

Szeretném megköszönni osztálytársamnak és egyben csapatársamnak **Vadász Dávidnak** munkásságát és a projektben való aktív részvételét. A sok segítségét, és a kreativitását.

Szeretném külön kiemelni **Molnár József** tanárurat mint a záródolgozat konzulensét a projektben való segítségnyújtásért, véleménynyilvánításért és ellenőrzését.

Illetve Szénásy Zsolt tanár úrnak a 3d modell kivitelezése segítségét.

VÁLASZTOTT TÉMA INDOKLÁSA

A téma kiválasztása leginkább az motivált, hogy szeretek az informatika számos területével foglalkozni. Leginkább a programozási területen és a modellezés terén. És ezzel a projecttel szereztem sok új tapasztalatot, hogy hogyan készülnek ezek az okos eszközök. Meg mindig is érdekelt, hogyan tudom a környezetemet fejleszteni, és összekötni őket, esetleg egy nagy weboldalra, és ha valami információra szükségem van, akkor azt könnyen elő tudom keresni.

Az informatikában szeretek tevékenykedni, van egy pár másik projektem is, ami a közösségi médiában meg található. Ezek játék kiegészítők, amik a játékhoz adnak pályákat, amiket én modelleztem le, és a pályát is nagyrészt én állítottam össze. E projekt által kezdtem el érdeklődni még általános iskolában a 3d modellezés és az informatika világában és mind két területen próbálom továbbfejleszteni magam továbbra is.

Ezek mellett az informatikában még az is megfogott, hogy mennyi problémát meg lehet egy-egy programmal oldani, és én szeretek foglalkozni ezekkel a probléma megoldási körökkel, illetve örömmel tölt el az is amikor látom a végeredményt amit sikerül alkotnom. A kiválasztott projektemhez is így állok, sok ötletem van még hozzá, hogy hogyan is tudom tovább fejleszteni, és stabilabb rendszert alkotni. De az idő szűkültével azokra már nincs kapacitásunk.

Érdeklődési körömbe még beletartozik a webmunkák fejlesztése és kezelése is, ami plusz egy indok volt hogy a végén sok projekt ötletünk közül ennek álljunk neki, mivel ehhez egy elég komoly weboldal felület kell, alkalmazzunk, ami adatokat, funkciókat is megjelenít, és az eszközöket is tudja kezelni. A weboldalas részhez leginkább az adatbázis műveletek, adat kezelése és űrlap készítése érdekelt, hogy hogyan is lehet kivitelezni. Hasonló weboldallakkal foglalkoztam már, egy ismerősömnek a weboldalát karbantartom és frissítem az aktuális adatokat, de a laravel keretrendszerrel is öröm volt meg ismerkedni, hogy miket lehet létrehozni benne. Hogy hogyan tudok benne adatokat megjeleníteni, és a fejlesztői környezete hogyan is működik.

Ezek mellet még az is érdekelt, hogy az iskolai eszközökből mit tudok összerakni, ami egy fontos problémakört fedhet le. Leginkább az épület biztonsága érdekeben lenne ez a rendszer ki találva, hogy a veszélyes helyzeteket el lehessen kerülni, itt arra gondolok, ha tűz üt ki, vagy a levegőszennyezettség már káros az egészségre a teremben. Továbbá a közös projektünk végeredményeként megalkotott füstérzékelővel azon diákokat is ki lehetne szűrni, aki az iskola épületében dohányzik.

TÉMA KIFEJTÉSE

Fejlesztői környezett alkalmazásai

A záródolgozat készítéséhez több alkalmazást, program, keretrendszereket és könyvtár struktúrát is alkalmaztunk a projektünk meg valósításához.

A webes megjelenéshez használtuk a **Visual Studio Code** programot. Az arduinos eszközök fejlesztéséhez az **Arduino IDE** programot használtuk. A weboldalhoz még egy adatbázist is kötöttünk, ami az **XAMPP** programban tároltuk. Illetve a külső fizikai modellhez a **SolidWorks** programot alkalmaztuk.

A záró dolgozatban használt eszközök

A webes felület kialakításához az ingyenes **Laravel** keretrendszerre támaszkodtunk, mely lehetővé teszi PHP alapú webszerverek fejlesztését. Az alkalmazás elkészítéséhez szükség volt a **Composer** nevű PHP függőség kezelő telepítésére, mely szintén ingyenesen elérhető. A webfejlesztés során a Microsoft által kifejlesztett **Visual Studio Code** 1.87.0-ás verzióját használtam, ahol a Composer, továbbá a Laravel telepítése egyszerűen elvégezhető volt a terminálban kiadott `composer global require laravel/installer` parancs segítségével. Ezzel együtt a programban telepítésre kerültek a **Laravel Blade Snippets**, ezen kívül a **Laravel Snippets** kiegészítők is, amik könnyebbé tették a kódolást a szintaxisok, valamint prefixek kiemelésével. A **Thunder Client** kiegészítő segítségével alternatíva nyílt az API-on keresztüli adatfeltöltés tesztelésére.

Az okoseszköz fejlesztése során az **Arduino IDE** 2.3.2-es verzióját használtuk, melyben az **esp8266**-os board kezelőt is telepíteni kellett, ami lehetővé tette a WiFi modulra való kódírást. Emellett telepítve volt az **Adafruit Unified Sensor** is.

DHT sensor library nevezetű könyvtár, melyek szükségesek voltak a szenzorokhoz való kód készítéséhez. Ezáltal pedig adatokat kaphatunk vissza.

A füstérzékelő modelljének megtervezéséhez a **SolidWorks 2023** volt alkalmazva, a modell az iskolában lett kinyomtatva egy 3D nyomtatóval. Az eszközök pontos

méretével lett el látva, hogy minél kisebb legyen, de viszont annál több védettséget adjon a külső burkolat.

A webszerver, ezen kívül adatbázis kezeléshez az **XAMPP Control Panel v3.3.0**-as változata volt használva. Ezzel tudtuk az adatbázisunkat kezelni

A rendszerbe beépített eszközök:

- 1db ESP8266 Wemos - NodeMCU 1.0 (ESP-12E modul)

- USB-C kábel

- 1db MQ-135 Gas Sensor

- 1db DHT11 Temperature & Humidity Sensor

- 2db Arduino LED

- 1db 3k Ω , és 1db 2k Ω ellenállás

- Vezetékek

- Egyedi, a technikum által nyomtatott 3D modell

RENDSZER BEMUTATÁSA

A rendszerünket az egyszerűségekre terveztük. Minél felhasználó barátába tesszük, és a megfelelő információkat jelenítjük csak meg.

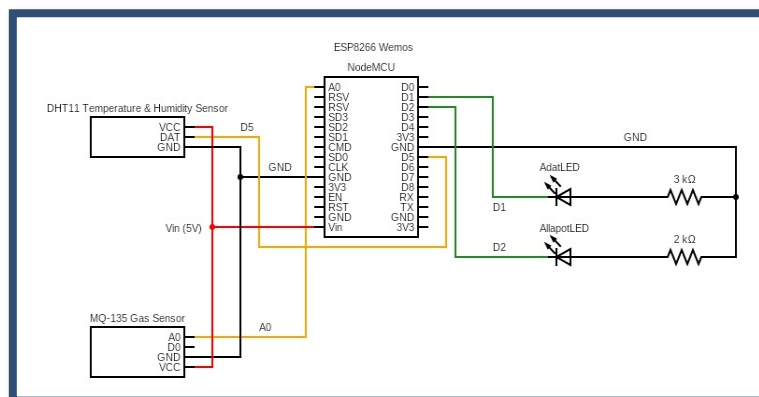
A rendszerünk lényegét tekintve egy olyan eszköz, ami képes a weboldalunkkal Kommunikálni, illetve képes a mért adatokat elküldeni rá. A weboldalhoz tartozó adatbázisban tároljuk el az összes olyan adatot, ami az eszköz állapotát is mutatja, és az esetleges hibásan mért adatokat is.

Rendszertervjajz

A fizikai tervezésben, létrehoztunk egy áramköri rajzot. A következő képen látható az elektronikai összeszerelése. A rajz egy fontos szerepet játszik hogy az eszközök milyen csatlakozási portokba kell rakni. A LED-eket, továbbá ellenállásokat az adott feladatokra ellátott vezetékekkel, kiemelten fontos figyelembe venni a vezetékekhez kapcsolódó csatlakozások neveit, hisz ha nem megfelelőképpen vannak, bedugva zárlatossá tehetik a rendszert, ami tönkre is teheti az alkatrészeket, esetleg elégetheti az ellenállásokat.

Az ábrázolásban felhasznált csatlakozások jelentései:

- **D1, D2, D5**: Digitális csatlakozás
- **A0**: Analóg csatlakozás
- **GND**: Földelés
- **Vin**: 5 Voltos áramellátás
- **VCC**: Áramfogadás
- **DAT**: Adat csatlakozás, digitális csatlakozáshoz kell kötni



1. ábra füstérzékelő áramköri megjelenése (A diagramot a <https://www.circuit-diagram.org> weboldalon készítettük el)

A program felépítése

A programok minden egyes eszközön külön meg található, és egyedi felfelprogramozással is beállítható. A weboldalas meg jelenítéshez is külön IP címezést alkalmazunk hogy meg különböztessük a termék adatát. A részleteket lemondtam fejezetekre.

A weboldal szerkezete

A laravel egy keretrendszer, ami al-mappákat köt össze. Ez a keretrendszer már működő képes, és amikor létrehozzuk a projektet, egy alap weboldallal és migrációkkal hoz létre.

- **Views mappa:** a views mappában olyan fájlokat találhatunk, amik a weboldalon megjelenő oldalak. Ezekért az oldalakért egy **master.blade.php** nevű fájl felel az összekapcsolásért. Ebben a fájlban hívjuk meg azt a contentet, amit szeretnénk meg jeleníteni a weboldalon és master.blade-et egy **layouts** mappában tároljuk.
 - A **master** fájlban olyan elemek találhatóak, mint például a baloldali menüsor, mely lehetővé teszi a webfelület különböző lapjaira való navigálást, ezen kívül az érvényben lévő oldal címe is megjelenik.
 - A **welcome** fájlban a főoldal további tartalmai találhatóak, amik az emeletekre vannak tagolva, így a termék a szintjük alapján találhatóak meg. A termék sorba rendezését a terem szám alapján zajlik, így szintenként könnyebb meg keresni az adott termelt.

- A **fusterzekelo2** fájl tartalmazza azt a programkódot, amely abban az esetben látható a weboldalon, amikor egy terem megnyitunk. Ezen a felületen állíthatjuk le az adatok küldését az adatbázisban, vagy A ledtest gombbal elérhetjük, hogy a fizikai eszközön ki kapcsoljuk a ledeket, de teszt üzemhez vissza kapcsolhatóak.
- A **terepfelvétel** fájlban kódolva vannak a teremfelvételhez szükséges űrlap elemek. Ezen a nézeten újabb termeket vehetünk fel a weboldalra, ahol megadhatjuk a terem számát, szintjét, az elhelyezendő eszköz IP címét, a terem jellegét (például tanári) és csatolhatunk egy képet a teremhez, amely a későbbiekben mindenhol meg fog jelenni.
- A **teremlista** fájl tartalmazza az összes felvett terem nézetének. Ezen a nézeten táblázatos listát láthatunk az adatbázisban lévő termekről, amelyeket módosíthatunk, esetleg törölhetünk.
- A **hibakódok** fájl tartalmazza az összes eszköz hibáit, amit lista szerűen ír ki. A listában időrendben van rendezve, mindig a legfrissebb hibák jelennek meg és a hozzá tartozó hiba jelentéssel, eszköz ip-vel, teremszámmal jelenik meg.
- **Routes mappa:** Ebben a mappában találhatóak azok a fájlok, amik a weboldal url-it kezelik. Fontos szerepet kapnak mert ezek alapján tud a weboldalunk át irányítani.
 - A **web. Php** fájl tartalma, itt találhatóak azok az elkészített útvonalak amelyben azt a blade fájl tartalmat hívhatjuk meg, és a felhasználó számára elérhetővé tesszük. A megoldásunkban mi egy **controller** meg hívásával kötjük össze az oldalt, mert az adatok lebonyolításához szükséges.
 - Az **api.php** fájl tartalma, hasonló, mint a web.php, de itt adatok továbbítása zajlik le. Az api segítségével létrehozzuk azt az elérést, amivel csak adatokat fogadhatunk az adatbázisunkba.
- **Controllers mappa:** a pontos elhelyezkedése pedig az'. App\Http\Controllers' helyen található.
 - A feladatai közé tartozik, hogy a web.php-ból meg hívót függvényeket esetleges megjelenítéseket is kezeljen.
 - A projektben használt controllerek is itt találhatóak, ezek is a megjelenésért felelnek, de ezekben zajlik az eszközökkel való kommunikáció is. Amikor lenyomunk a weboldalon egy gombot a kontrolleren keresztül küldjük ki arra az elérési útvonalra a jelet, amelyik eszköz ip címét tartalmazza az aktuális oldal.
- **Models mappa:** pontos helye 'App\models'. A modellek felelősek a weboldalon megjelenő adatokért. Ezek, mint egy hidat képeznek az adatbázis, a

- webfeldolhozás között. Funkcionalitásuk a weben, mint egy objektum működnek az oldalon, és lekérdezésekkel tudunk adatokat kizsedni.
 - A projektünkhöz alkalmaznunk kellet két tábla össze kapcsolást, ami egy a többhöz kapcsolatot létesít.
- **Migrations mappa:** pontos helye ' \database\migrations'. Az adatbázis szerkezetét itt határozzuk meg. Ebben a mappában találhatóak azokat a segéd fájlok, amik a **php artiasn migrate** paranccsal hozhatóak létre.
 - Az adatbázisunkban egy olyan szerkezetett kellet ki alakítani, ahol két tábla kialakításra van szükségünk.
 - A feladat megoldásához alkalmaztunk egy idegen kulcs össze kapcsolást. E a két tábla a **helyszinek** és **meresek**, a meréseket kötitek össze hastomany kapcsolás van alkalmazva, a két oldalt pedig mindig frissítjük, hogy ha törlik a hozzá tartozó adatok is törlődjenek.
 - Egy példa a meresek migrációból:


```

○ Schema::create('meresek', function (Blueprint $table) {
○ $table->id("m_id");
○ $table->foreignId('h_id')->references('h_id')->on('helyszinek')->onDelete('cascade')->onUpdate('cascade');
○ $table->integer('ppm');
○ $table->float('homerseklet',4,2);
○ $table->integer('paratartalom');
○ $table->integer('hibakod');
○ $table->dateTime('meres_ideje');
○ });
          
```
- **Seeders mappa:** pontos helye ' \databas\seeders'. A projekt során használtuk, hogy tesztelhessük a weben meg jelenő adatokat. A feladata leginkább az adatbázisba való teszt adatok beszúrása, ha esetleg töröljük az adatbázis van kíváncsiak vagyunk, hogy több teremmel, hogy képes üzemelni az oldalunk. De tényleges adatokat csak az eszközökről kapott információkat hagyjuk az adatbázisban.
- **Public mappa:** A weboldalt ezen itt meg nyitni az **index.php**-val. De itt tároljuk a projectekhez felhasznált jegyzeteket, arduino kódot és a 3d modellt.
- **Egyéb fájl(ok) és mappák:** A php sajátos adatbázisa is ide tartozik aminek nincs neve, csak kiterjesztése van, ez pedig az .env fájl. Ez az adatbázishoz szükséges konfigurációs fájl. amit a projektünkhöz módosítanunk kellet benne.
 - **DB_HOST=localhost**
 - **DB_DATABASE=project2024**

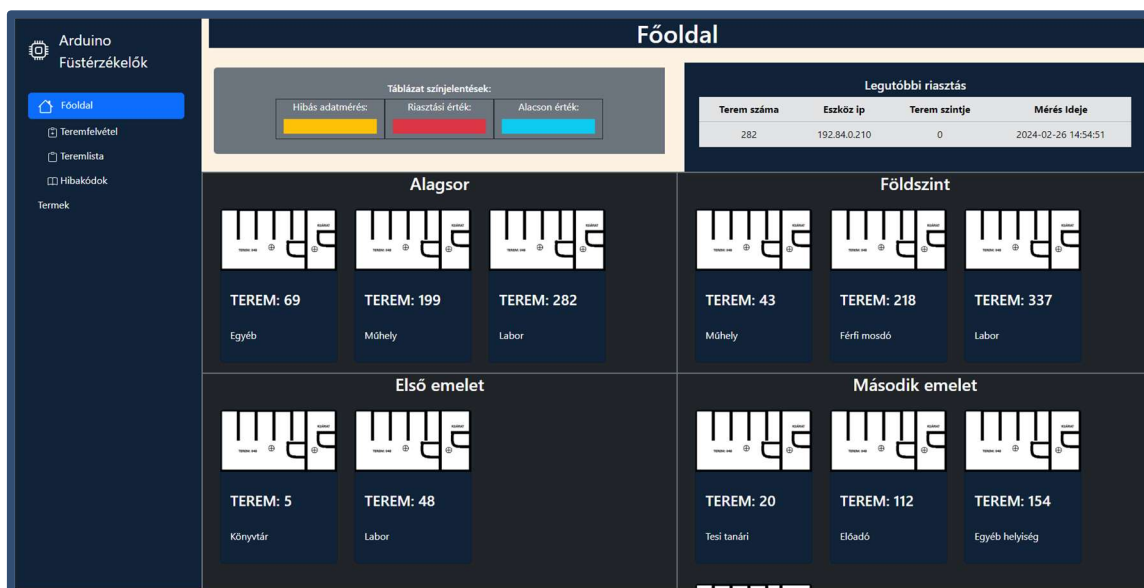
A host változóval az adatbázisnak azt mondjuk meg, hogy milyen IP címen találja az adatbázist, localhost ez esetben, mely a lokális számítógépet jelenti, a database változóval a nevét változtatjuk az adatbázisnak.

Továbbá, ha szeretnénk képeket tárolni a weboldalhoz, ahhoz a **storage** mappát használjuk. Ez a mappa akkor jön csak létre ha a projektet ott azon a gépen hozzák létre. A képek elérése pedig 'storage\app\public\teremkepek'. Fontos hogy itt tároljuk, mert a felhasználók számára nem egyszerűen módosítható, és másolható. Kifejezetten a biztonsági szempontból kerülnek ide a képek.

A weboldal

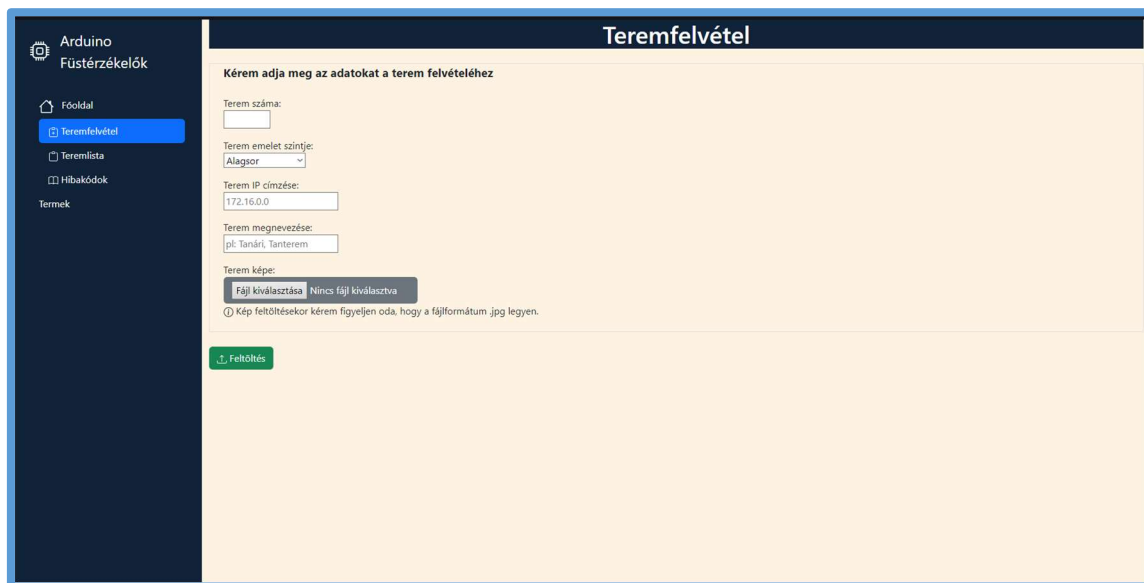
A weboldalunk 4 oldalból áll össze, mindegyik oldalnak különböző funkciói vannak.

A **fő oldal**: amikor meg nyitjuk a weboldalt, először a fő oldalra irányít. Itt láthatjuk az összes termet, ami az adatbázisban meg található. Ezeket a termeket pedig ki tudjuk szűrni, hogy az esetleges riasztás hol található



2. ábra A weboldal fő oldala

A **terem felvétel**: ezen az oldalon lehet új termeket meg adni, ahogy említettem már, a mezők meg felelő kitöltésével lehet az adatbázisba szűrni. Fontos, hogy a kép formátuma jpg kiterjesztésű legyen.



3. ábra A weboldal teremfelvétele

A teremlista: ezen az oldalon találhatóak csak a termek, itt lehet őket módosítani, vagy törölni. A módosításnál nem lehet a Terme #ID mert az mindegyik teremnek az egyedi azonosítója. A törlést viszont nem lehet vissza hozni, ha már rá nyomtunk.



Terem #ID	Teremszám	Teremszint	Eszköz IP	Módosítás	Törölés
1	112	3	192.112.0.182	Módosítás	Törölés
2	5	2	172.53.0.159	Módosítás	Törölés
3	43	1	168.78.0.25	Módosítás	Törölés
4	20	3	168.27.0.36	Módosítás	Törölés
5	337	1	172.36.0.179	Módosítás	Törölés
6	199	0	127.5.0.169	Módosítás	Törölés
7	48	2	172.60.0.153	Módosítás	Törölés
8	154	3	168.10.0.112	Módosítás	Törölés
9	282	0	192.84.0.210	Módosítás	Törölés
10	244	3	172.202.0.187	Módosítás	Törölés
12	218	1	192.55.0.67	Módosítás	Törölés
13	69	0	192.168.0.18	Módosítás	Törölés

4. ábra A weboldal Teremlistája

A Hibakódok: ezen az oldalon találhatóak az eszközök hibái. Ezeket az adatokat úgy jelenítjük meg hogy le írjuk hogy melyik terem, melyik szinten és hogy milyen hiba kóddal lépet fel ez a probléma.

A táblázatban időrendben vannak állítva, mindig a legfrissebb adatok jelennek meg, mellette pedig a hibának az okát is láthatjuk. A hibák csak akkor tűnnek el ha az hozzá tartozó adatbázist kiüritjük.

Az Arduino program szerkezete, és működése

Az Arduino kódunk egy nagy fájlból áll össze. A szükséges könyvtárak meghívása, változók deklarálása, ezen kívül a webszerver kapcsolat beállítása. A kód további részeiben találhatóak az eszköz kapcsolóinak, valamint LED-jeniek tesztelését biztosító kódrészletek. Ezt követően következik a setup részlete, mely egyszeri futást biztosít az eszköz felállításakor, például a hálózatra kapcsolódást, ezen kívül a csatlakozások bekapcsolását. A setupot követően található az adatküldés kódrészlete, mely a mért adatokat továbbítja a webszervernek a jó teremhez. A különféle hibatípusok jelzése külön részt kapott, az összes típusnak saját jelzése van.

A programunkhoz házzá tartozik egy hibakód tábla is. Ezzel a hiba táblázattal lehet meg nézni hogy az eszköznek milyen gondok adódtak.

Hibakód száma	Hibakód jelentése	Hibakód jelzése
0	Minden megfelel	ÁllapotLED: Folyamatos AdatLED: 1 rövid villogás
1	Sikertelen páratartalom és hőmérsékletbeolvasás	ÁllapotLED: 1 rövid, 1 hosszú villanás
2	Sikertelen levegőminőségbeolvasás	ÁllapotLED: 1 rövid, 2 hosszú villanás
3	Sikertelen IP címzés	ÁllapotLED: 1 rövid, 3 hosszú villanás
4	Sikertelen adatküldés az adatbázisba	ÁllapotLED: 2 rövid, 3 hosszú villanás

A modell szerkezet

A 3d modell szerkesztését én vállaltam el, és szeretném egy kicsit részletesebben bemutatni a modelletemet.

Az eszköznek egy olyan külsővédő modellt szerkesztettem, ami védi a belső eszközöket. A modell tartalmazza a tok darabjait, ezek alátámasztások és csavar furatok, illetve a Wifi modul és mérő eszközöknek a helyét. A projektünkben meg találhatóak a 3d fájlok a **Project2024\public\3dmodel** mappájában

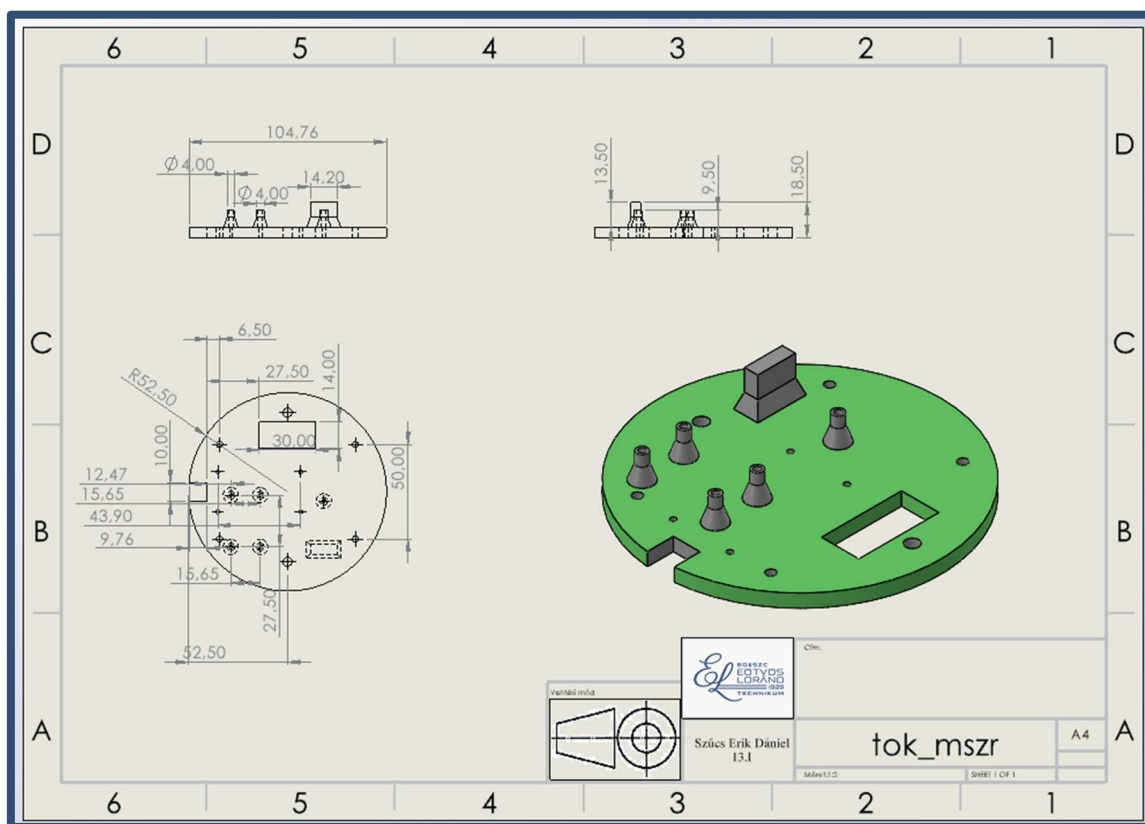
A modell tervezés folyamata alatt figyelembe vettem a következőket:

- **a modell egyszerű összerakhatósága:** a modell darabokból és csavarokból áll, melyeket könnyen össze lehet rakni.
- **az eszközök elhelyezhetősége:** minden felhasznált szenzor és eszköz mérete le lett mérve milliméter pontosan, hogy a modellben is precízen el lehessen helyezni őket. És egyszerűen meg különböztetni, hogy melyik szenzor hova tartozik.
- **a LED-ek kivezetése:** a LED-ek ki vannak vezetve a modell oldalára, hogy bárki láthassa az általuk küldött jelzéseket.
- **az esetleges plafonra szerelés:** a modell oldalán található szárnyaknak köszönhetően az eszköz akár plafonra is helyezhető.
- **szűrő, ami eléri a szenzorokat:** a modell tetején lévő szűrőn keresztül bejut a levegő a szenzorokhoz, így a szenzorok sikeresen végre tudják hajtani a méréseket. Mivel a modell több emeletről áll így, ha esetleg valamilyen folyadék éri az eszközt legfeljebb a szűrőn tud átjutni, megvédve az alkatrészeket sérüléstől.
- **a vezetékek törés és szakadásvédelme:** a modell belülről tágas és több szintből áll, így a vezetékek nem sérülhetnek.
- **tápegység bevezetése:** a modell alja úgy lett tervezve, hogy a Wi-Fi modulra lehessen helyezni valamilyen tápegységet. Alapesetben a Wi-Fi modul USB-C kábellel kap áramellátást, viszont biztos, hogy ez nem lenne kivitelezhető mindenhol, így más módszerek alkalmazásához lett hagyva hely a modellben.

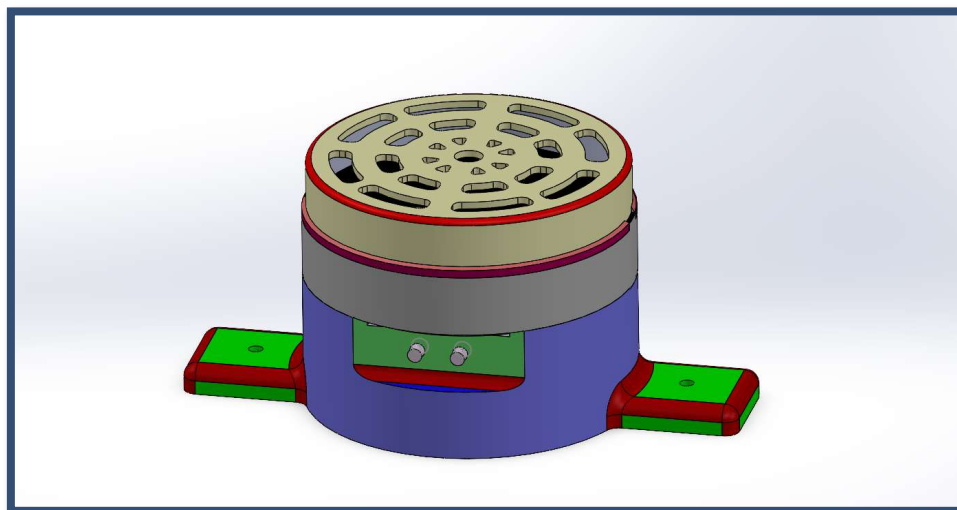
A modellbe csavar készlete, tudni valók és csavar típusok:

- A modellben a csavarok elhelyezkedése úgy lettek elkészítve, hogy a csavar vége fele tudja csak a modellben meg fogni, mert ugyan is a lábak, amik például tartják a weomost nagyon vékony falúak és a csavar könnyen szét repesztené. Ezért alkalmaztam azt a megoldást, hogy csak abba a részbe foghatbele amire rá lenne fogatva az alaplap is.
 - **Eszközök:**
 - **mq sensor:** D-fejű, kereszthornyos csavar DIN 7985 Rozsdamentes acél A2 (fej átmérő normál: 4mm), átmérője: M2, hosszúság: 16 mm, (4db)
 - **dht11:** D-fejű, kereszthornyos csavar DIN 7985 Rozsdamentes acél A2 (fej átmérő normál: 4mm), átmérője: M2. hosszúság: 16 mm, (1db)
 - **wemos:** D-fejű, kereszthornyos csavar DIN 7985 Rozsdamentes acél A2 (fej átmérő mini: 4,7mm), átmérője: M2,5 hosszúság: 12 mm, (4db)
 - **Modelben:**
 - **bottom_to_alaplap:** D-fejű, kereszthornyos csavar DIN 7985 Rozsdamentes acél A2 (fej átmérő normál: 8mm), átmérője: M4 hosszúság: 50 mm, (4db)
 - **-alapla_to_top:** D-fejű, kereszthornyos csavar DIN 7985 Rozsdamentes acél A2 (fej átmérő normál: 8mm), átmérője: M4 hosszúság: 16mm, (2db)

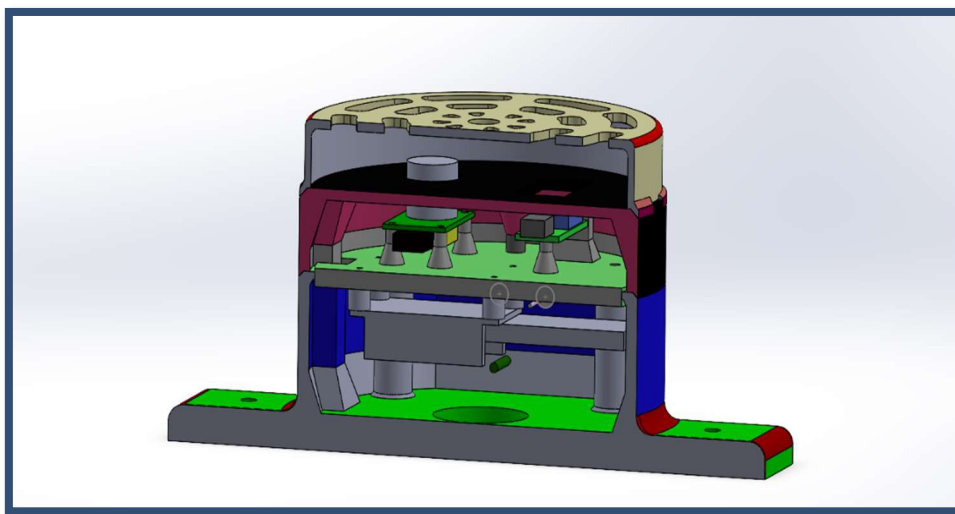
A 3d modellek mappában találhatóak még a műszaki rajzok között, a többi modell pontos méretével kapcsolatosan. Illetve a 3d modellek mappában találhatóak még a modelltől régebbi képek, és portó típusokról is. A jelenlegi eszköz készleteink is meg találhatóak le modellezve amiket a pontos méretezés miatt hoztunk létre, hogy minden egyes komponenst a megfelelő helyére elhelyezhessünk.



5. ábra A füstérzékelő tok_mszt műszaki rajza



6. ábra A füst érzékelő 3D modellje



7. ábra a Füst érzékelő félbevágott 3D modellje

SAJÁT VÉLEMÉNY

A projektre leginkább az motivál, hogy a technikumban eddig itt töltött időm alatt nehezen viseltem el azokat a körülményeket mikor a mosdót kellett használnom. Számtalanszor ahogy beléptem egy ilyen helységekre, nem kellemes szagok fogadtak és keveredve a cigifüsttel. A legtöbbször már nem is mertem el menni mert néha már olyan mértékben folytó volt a levegő, hogy már nem is lehetett miatta bemenni. Legtöbbször ezért is kellett azt a módszert alkalmazzam, hogy órán meg próbálok ki kérekedni, de ezeket nem lehetett sokszor eljátszani. Amíg még fontosabb, hogy ezek a diákok ellepik olyankor a mosdót, és egész szünetben befoglalják, ezzel el ellehetetlenítve, hogy ne óráról kelljen ki menni. Ami leginkább a bosszantott, hogy ezzel az egészségemet is elég erősen kockáztatják, és ha felmerek szólalni, hogy hagyják abba, akkor egy olyan fallal nézek szembe, ami lehet, hogy meg is támad. Nem beszélve a stílusokról, amit azok a diákok képviselnek, ha valami nem tetszik nekik.

Ezzel a projektünkkel megoldható lenne, hogy ezeket az eseteket el tudjuk kerülni. Mert a kialakítást tekintve úgy terveztük meg hogy legyen egy olyan rendszerünk, ami néma riasztást tud küldeni a tanárok számára, ami képes meg mondani, hogy melyik teremben, melyik szinten és hogy milyen mértékben áll fent a riasztás. Ezáltal hatékonyabbá tesszük nekik az ehhez hasonló probléma körök megoldását. És rajta kapni a diákok nagyobb százalékát, akik ehhez a cselekményekhez folyamodnak, és elnyerjek méltó büntetéseiket. Ezáltal sokkal jobbra tudnánk tenni a diákok napját, és egészségét, illetve a tanárok munkásságát is, hogy nem kell az egész épületben rohángálniuk, hanem egy vagy több adott pontra is ki tudnak menni intézkedni.

Továbbá a rendszerünk növelné az épület biztonsága érdekeit, időben jelezné, ha valamilyen gond adódna. A rendszerünk 6 másodpercenként tudja küldeni az adatokat, de az adat küldési időt még le lehet csökkenteni a kívánt adatt frissítési gyorsaságát.

Továbbifejlesztés lehetőségei

A szerkesztések között felmerült egy pár olyan tovább fejleszthetőségi lehetőség, amit jó ötletnek tartok, de mivel már nem lenne elegendő időnk, hogy meg tegyünk a hozzá tartozó lépéskét.

Amit a legfontosabbnak tartok, azok a jobb szenzorok alkalmazása lenne, ami képes, nagyobb körben és rövidebb idő alatt érzékelni a füstöt vagy a hőmérsékletváltozást. A jelenlegi szenzorjaink körülbelül 15 másodperc telik el, mire az eszközünk érzékelési zónájába esik. A modernebb eszközök képesek lennének távolabbi pontról is érzékelni.

A modell át alakításával is lehetne fejleszteni, leginkább arra gondolok, hogy a modellben több fajta érzékelőt és alaplapot is lehet elhelyezni. A modellünket egyfajta moduláris kialakítást kéne ki alakítani, ezek lehetnek mozgó alkatrészek, és más fajta csavar típusok elfogadása.

A mi még igazan fontos lenne a modellünkkel kapcsolatban, hogy erősebb anyagból legyenek a falak, lehet fémből is vagy erősebb műanyagból, hogy ne lehessen könnyen megrongálni, és tartósabb csavar készlettel kellene ellátni. Tovább, olyan szűrővel ellátni, ami a folyadékot nem engedi át, de a levegőt nem változtatja nagy mértékben.

Egy infravörös távirányítóval lehetne fel konfigurálni az eszköz manuális IP címzését, vagy DNS kiosztást lehet választani. A távirányítón be lehetne gépelni az ip címet is, és vissza jelzésként egy kis kijelzőn mutatná, hogy sikeres a konfiguráció, és esetleges hibákat is ki írná.

Az adatbázis optimális működése. Az adatbázisunkban a merések táblába óriási adathalmaz kerülhet, függően attól hogy mennyi eszközt helyezünk fel az épületben. Egy olyan automata beépítése a rendszerünkben, ami meg gátolja, hogy az adatbázisunk meg teljen és a weboldalon meg jelenő adat mennyiség ne okozza a rendszer kiakadását.

A web felületre egy globális kapcsoló létre hozása. A fő oldalon ki lehetne választani, hogy esetleg melyik csoportot vagy az összes eszközt le lehet állítani, vagy a teszt ledjüket bekapcsolni.

Az eszközök szellőztetése, vagy hűtése. A jelenlegi eszközein nem igényelnek nagy mértékű hűtést, de a nagyobb teljesítményű eszközök között már el kell gondolkozni, hogy milyen hűtést vagy szellőztetést igényelnek. A hűtés abból a szempontból is fontos a szerepe, mert ez is határozza meg az eszköz élettartalmát. Egy egyszerű pc ventilátor is elegendő lehet a megfelelő hűtéshez.

Az arduinos kód egyszerűsítése, és funkcionalitás javítása. A kódunk, amit eddig sikeresen meg futtatunk, és jó eredményekkel büszkélkedik, lehetne tovább fejleszteni, hogy kevesebb idő alatt le tudjon futni. Esetleg több hiba vissza jelzést be integrálni, és pontosabb hibaköri leírás megadása.

A weboldal responsive-vá váltása. A jelenlegi weboldalunkon használjuk a bootstrap 5 adta lehetőségeket, de sajnos nem mindent tudunk a telefonos meg jelenítéshez is alkalmazni. Az elemek egymás alárendezése, és az oszlopok tömörítése.

Több képformátum felvétele a termék felvételénél. Illetve nagyobb képek elhelyezése, és a fő oldalon való meg jelenítése.

A weboldalon lévő színtéma kiválasztás lehetősége. Ha esetleg máshol is szeretnék a rendszert alkalmazni, akkor az intézmény is meg választhatja a weboldalon megjelenő színeket.

Lehetőség a weboldal nyelv kiválasztásra, hogy esteleg nemzetközi szinten is lehessen alkalmazni.

DOKUMENTÁCIÓ – ÖSSZEFOGLALÁS

ÖSSZEFOGLALÁS

Egy olyan rendszert hoztunk létre, ami képet több termemnek az adatai megjeleníteni. A termék ezzel kell az adatokkal rendelkeznek: légminőség, páratartalom és hőmérséklet. Dinamikusan bővíthető az oldalunk. De mellette törölhető az elavult adatokat. Továbbá a weboldalunk tartalmazza még az eszközök hibakódját is. amit az eszközöknél is meg található, illetve egy nagyobb táblázatban is meg találhatóak a ki gyűjtött hibák. Ez eszközök egy teremhez vannak rendelve, ami, egyedi azonosítóval rendelkezik, és ezáltal a fő oldalon is meg találhatóak a termék, amiknek az egyedi adataik találhatóak meg.

Egy olyan eszközt hoztunk létre, ami képes kommunikálni a weboldalunkkal, és a környezetében a páratartalmat, hőmérsékletet és a légminőséget képes mérni. Továbbá képes fizikai ledéken jelezni

IRODALOMJEGYZÉK

A záródolgozatomban használt források:

- <https://laravel.com/>
- <https://getcomposer.org/>
- <https://www.arduino.cc/>
- <https://www.circuit-diagram.org/>
- <https://www.apachefriends.org/hu/index.html>
- <https://www.infojegyzet.hu/webszerkesztes/zarodolgozatmintak/>
- <https://www.w3schools.com/>
- <https://getbootstrap.com/>
- <https://stackoverflow.com/>
- <https://github.com/>
- <https://www.solidworks.com/>
- <https://color.adobe.com/create/image>
- <https://hu.wikipedia.org/wiki/Kezd%C5%91lap>

ÁBRAJEGYZÉK

1. ábra füstérzékelő áramköri megjelenése (A diagramot a https://www.circuit-diagram.org weboldalon készítettük el)	13
2. ábra A weboldal fő oldala	16
3. ábra A weboldal teremfelvétele	17
4. ábra A weboldal Teremlistája	17
5. ábra A füstérzékelő tok_mszt műszaki rajza	21
6. ábra A füst érzékelő 3D modellje	21
7. ábra a Füst érzékelő félbevágott 3D modellje	22