

Kodolányi János Egyetem

SZAKDOLGOZAT

MAJOR LEVENTE

ÜZEMMÉRNÖK-INFORMATIKUS

Budapest

2025

Kodolányi János Egyetem

Informatika Tanszék

Nyílt területi időjárás állomás építése és adatok publikálása mikrovezérlő segítségével

Konzulens: Dr. Pitlik László

KÉSZÍTETTE: MAJOR LEVENTE

ÜZEMMÉRNÖK-INFORMATIKUS

Budapest

2025

Tartalomjegyzék

1. Bevezetés és célkitűzések	7
1.1. Motiváció és témaválasztás indoklása	7
1.2. Célkitűzés	9
1.3. További fejlesztési lehetőségek	9
1.4. A dolgozat szerkezete	10
1.5. Célcsoport	12
1.6. Hasznosság	12
2. Elméleti alapok, szakirodalmi áttekintés	16
2.1. Gazdasági és humán ismeretek	16
2.1.1. Európai civilizáció és identitás	16
2.1.2. Komplex társadalomtudományi ismeretek	17
2.1.3. Emberi viselkedés és kommunikáció	18
2.1.4. Vállalati gazdaságtan	18
2.1.5. Vezetési és vállalkozási ismeretek	19
2.2. Jogi ismeretek	19
2.2.1. Szakterületi jogi ismeretek	19
2.2.2. Projekt kapcsolódása a jogi szabályzásokhoz	20
2.3. Természettudományos alapismeret	20
2.3.1. Matematikai alapok	20
2.3.2. Az elektronika fizikai alapjai	21
2.3.3. Elektronikus áramkörök	21
2.4. Rendszerfejlesztés és szoftverarchitektúrák	24
2.4.1. Rendszertervezés	24
2.4.2. Rendszermodellezés	25
2.4.3. Szoftverarchitektúrák	26
2.4.4. Felhasználói interfészek és vizualizáció	27
2.5. Adatbázis kezelés	27
2.5.1. Adatbázisok I	27
2.5.2. Adatbázisok II	28
2.6. Hálózatok és operációs rendszerek	28
2.6.1. Hálózatok és számítógép architektúrák	28
2.6.2. Operációs rendszerek	29
2.7. IT-biztonság és fejlesztés	30

2.7.1. Innovatív információs és kommunikációs technológiák az IT-biztonság kapcsán	30
2.7.2. IT-biztonsági fejlesztések minőség- és projektmenedzsmentje.....	31
2.7.3. Mesterséges intelligenciák az IT biztonság területén.....	32
2.7.4. Tudásmenedzsment az IT biztonság területén.....	33
2.8. Programozás	33
2.8.1. Programozási alapelvek és módszertanok	34
2.8.2. Adatszerkezetek és algoritmusok	34
2.8.3. Programozás I.....	35
2.8.4. Programozás II	35
2.8.5. Programozás III	36
2.9. Szoftverüzemeltetés és tesztelés.....	36
2.9.1. Szoftverüzemeltetés.....	36
2.9.2. Szoftvertesztelés	37
2.10. A horgászat biológiai, ökológiai törvényszerűségei.....	37
3. A projekt bemutatása	39
3.1. Szerkezeti modell és fő részei	39
3.1.1. Első rész: Slave	40
3.1.2. Második rész: Master	40
3.1.3. Harmadik rész: Adattovábbítás	41
3.2. Alkalmazott hardvereszközök szerkezeti és működési áttekintése	41
3.2.1. ESP32-WROOM-32U mikrovezérlő.....	41
3.2.2. DS18B20 hőmérséklet szenzor	46
3.2.3. 18650 Lítium akkumulátor V3 bővítőmodul.....	47
3.2.4. Antenna	48
3.2.5. BME280 szenzor	48
3.2.6. RS-FXJT-N01 szélirány mérő	49
3.2.7. RS-FSJT-N01 szélsébség mérő	50
3.3. Hardverkomponensek kapcsolódása	51
3.3.1. A Slave eszköz.....	51
3.3.2. A Master eszköz	52
3.3.3. Csatlakozás a webszerverhez	53
3.3.4. A fejlesztői környezet konfigurálása és a programok működési elve	54
3.3.5. A mikrovezérlő fejlesztői környezetének telepítése	54
3.3.6. Webhelyen elhelyezett fájlok fejlesztői környezete	57
3.4. Forráskódok és működésük	58

3.4.1. MAC-cím kiolvasás.....	58
3.4.2. Master eszköz kódjának bemutatás	59
3.4.3. Slave eszköz kódjának bemutatás	63
3.4.4. Webes adatbázis kezelése	66
3.4.5. Adatok GUI-n keresztüli megjelenítése	69
3.5. Speciális aspektusok.....	70
3.5.1. Adatbiztonság és titkosítás	70
3.5.2. Hálózati biztonság	71
3.5.3. Hozzáférés-kezelés és autentikáció	71
3.5.4. Adatintegritás és megbízhatóság	72
3.5.5. Webszerver és adatbázis biztonsága.....	73
3.5.6. Fizikai biztonság.....	75
3.5.7. DDoS és más szolgáltatásmegtagadási támadások elleni védelem	75
3.5.8. Tesztelés	76
3.5.9. Mesterséges Intelligencia	78
3.6. A fejlesztési perspektívák rövid elemzése	83
3.6.1. PH szenzor.....	83
3.6.2. A víz oldott oxigénszintjének mérése.....	83
3.6.3. Vízhőmérséklet mérésének a kiterjesztése különböző helyeken, különböző vízmélységben.....	84
3.6.4. LoRa kommunikáció kialakítása nagyobb távolságok elérése érdekében;	85
3.6.5. Tápellátás fejlesztése napelemes rendszer kialakításával.....	85
3.6.6. Szoftverfejlesztés az energiatakarékosság elérése érdekében	85
3.6.7. Hardveres fejlesztés élőkép stream-elése a felhasználók felé	85
3.6.8. Esőszenzor kiépítése.....	86
3.6.9. Online lehetőség biztosítása a felhasználóknak online fogási eredmények regisztrálása, annak érdekében, hogy a MI pontosabb predikciót készítsen	86
4. Kérdőív	87
5. Vita.....	96
5.1. Az adatok megbízhatósága és pontossága	96
5.2. Az adattovábbítás és a rendszer biztonság	96
5.3. Hozzájárul-e a horgászat sikeréhez és az élményhez?	96
5.4. Van-e értékelhető piaci igény a rendszerre?	96
6. Következtetés.....	98
6.1. Az adatok megbízhatósága és pontossága	98
6.2. Az adattovábbítás és a rendszerbiztonság	98

6.3. Hozzájárul-e a horgászat sikeréhez és élményéhez?	98
6.4. Van-e értékelhető piaci igény a rendszerre?	99
7. Jövőkép.....	100
8. Összefoglalás.....	101
8.1. Rezümé – HU	101
8.2. Rezümé – EN.....	101
9. Felhasznált irodalom.....	103
10. Mellékletek.....	114
10.1. A projektben felhasznált programok letöltési helye és verziószáma:	114
10.2. Ábrajegyzék	115
10.3. Definíció jegyzék	116
10.4. Rövidítések jegyzéke.....	117
10.5. ChatGPT – konverzációk	119
10.5.1. IEEE 802.11b/g/n/e/i szabvány leírása.....	119
10.5.2. IT Biztonsági aspektusok	122
10.5.3. Számítógép hálózatok	123
10.5.4. Többlet-érték becslés.....	124
10.5.5. Licence díj kalkuláció	126
10.6. Licence	129
10.7. Műszaki adatlapok.....	130
10.7.1. esp32-wroom-32d_esp32-wroom-32u_datasheet_en.....	130
10.7.2. DS18B20	130
10.7.3. Battery Shield V3	130
10.7.4. bst-bme280-fl000	130
10.7.5. RS-FXJT-N01.....	130
10.7.6. RS-FSJT-No1	130

1. Bevezetés és célkitűzések

Az első fejezetben bemutatásra kerül a szakdolgozat témaválasztásának indoklása (vö. 1.1 alfejezet), ami a horgászat és a modern technológia összekapcsolásán alapszik. Ismertetem a projektem célkitűzéseit (vö. 1.2 alfejezet stb.), a szenzorjaim által mért adatokat, hogy azok milyen funkciókat töltenek be és miért jelentősek a horgászatban, a dolgozatom szerkezetének felépítését (vö. 1.4 alfejezet), a célcsoportomat (vö. 1.5 alfejezet), mely jóformán a technológiai újításokra nyitott horgászokról szól. Ismertetem a projekt gyakorlati hasznosságát (vö. 1.6 alfejezet), amely a környezeti tényezők pontosabb elemzésével hozzájárulhat a sikeresebb horgászathoz, továbbá röviden bemutatom a rendszerem továbbfejlesztésének lehetőségeit (vö. 3.6 alfejezet).

1.1. Motiváció és témaválasztás indoklása

Véleményem szerint ebben a felgyorsult világban, a kikapcsolódás és a béke esszenciális lételeme a mentális egészség fenntartásának. Számomra a nyugalom a horgászatban mutatta meg magát. Semmihez sem fogható élmény az, amikor az ember távol kerül a modern eszközöktől, a tömegtől és a vízparton teljes nyugalomban tölthet el pár órát, napot, esetleg barátok vagy család közelében. Jómagam is a sporthorgászok közösségébe tartozom már 2012 óta. Bár már maga a természetben töltött idő is élvezetes, de a kapás, a halak fárasztása, megmondhatatlan pluszt képvisel számomra. Ez a sikerélmény ugyanakkor tévhitekkel szembe menve nem csak a szerencsén múlik. Sok információ van, mely számottevően növeli a horgászati sikereket érintő kilátásainkat például: a helyismeret és a tapasztalat az adott vízterületről, a halak étkezési szokásairól, táplálkozásuk formájáról. Mindezek kiismerésére elköteleződés és rendszeres látogatás szükséges egy adott területhez, ami időigényes és hosszadalmas folyamat. Ez egyben megérteti a laikusokkal is a KNUTH-i elvet: „*tudás az, ami forráskódba átírható, minden más emberi aktivitás művészet*” (vö. https://miau.my-x.hu/miau2009/index_tki.php3?filter-Text0=*knuth). Azonban, sajnos rendkívül kevés ember engedheti meg magának, hogy ennyi időt foglalkozzon hobbijával. Bátran kijelenthető, hogy a horgászok többsége alkalmazza a „*rapid peca*” módszerét mely, mint a nevében is megjelenő gyors horgászatot jelenti, mikor az embernek csak pár órája van a hobbjára kiélvezésére. (Halcatraz – <https://www.halcatraz.hu/blog/2022/08/01/elmeletbol-gyakorlatba-method-gyorstalpalo> – Elméletből gyakorlatba - Method gyorstalpaló – Letöltés: – 2025.03.16.)

A kutatásom, célja főképp ezen rétegbe tartozók sikerélményének és élvezeti faktorjának növelését célozza meg, bár ezen információk, még a legtapasztaltabb pontyhorgászoknak is hasznára válnak. Ugyanis nem csak saját tapasztalatból (vagyis intuitív úton módon) lehet eredményeket

elérni! Az általam készített műszaki megoldás olyan lényegesnek vélt adatokat szolgáltat a horgászoknak, melynek tudatában az „*esélyeik a nagyobb fogásdarabszám/fogás-össztömeg kapcsán határozottan növekednek*”. (Nahs Magyarország YouTube-csatorna – https://www.youtube.com/@nashtackle_hungary/videos – Letöltés: 2025.03.16.) Statisztikai kimutatás jelenleg nem áll rendelkezésre, ugyanakkor a tapasztalt horgászok körében ez az empirikus tény.

A halak táplálkozási szokásait befolyásolják többek között a „*léghőmérséklet*”, „*víz hőmérséklet*” (Haldorado – Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-1-a-homerseklet-a-legnyomas-es-a-frontok-a2722?utm> – Letöltés: 2025.02.20.), „*szélsebesség*”, „*szélirány*” (Halcatraz – Az időjárás és a halak táplálkozása https://www.halcatraz.hu/blog/2012/06/26/az-idojaras-es-a-halak-taplalkozasa?srsId=AfmBOornd0EaaE-emwAVWS9MaO0dEMIJWI9_bMyUOWFDfZi2IO48oL54f&utm – Letöltés: 2025.03.20.), „*páratartalom*” (Xfish – A halak étvágyára ható időjárási tényezők – https://www.xfish.hu/horgaszmagazin/a-halak-etvagyara-hato-idojarasi-tenyezok/980?srsId=AfmBOopQ6oGeY_KjKFwV9qe2SaVCEbj39FNwYrYVXH8RUP_108X5sRZ_&utm – Letöltés: 2025.03.20.) és a „*légnyomás*” (Haldorado – Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-1-a-homerseklet-a-legnyomas-es-a-frontok-a2722?utm> – Letöltés: 2025.02.20.). Minden hal szervezete és szokásai különböznek, jelen esetben a célhal a ponty, mert a projekt alapján mért adatok erre a halfajtára vannak hatással.

Az alacsony vízxigén szint esetében a szervezetük nem képes feldolgozni a fehérjét és annak formáit, ebben esetben egy szénhidrátos csali az ideális választás, „*alacsony víz hőmérséklet esetén az emésztésük lelassul, mely azt eredményezi, hogy kevés táplálékot vesznek föl*” (Haldorado – Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-1-a-homerseklet-a-legnyomas-es-a-frontok-a2722?utm> – Letöltés: 2025.02.20.) Ilyenkor a koncentrált helyre való etetés nélküli, „*pva bages módszer a legeredményesebb*”, a kicsi, kirívó színnel és aromával rendelkező csali a jó döntés. (Halcatraz – Pontyozás hideg vízben <https://www.halcatraz.hu/blog/2023/01/11/pontyozas-hideg-vizben?srsId=AfmBOooPs88751zWe3aUS-lxEZIo7BHGRArFTIh91ZH-XMbIZWxZDm8Tw> – Letöltés: 2025.03.16.)

Vegyük például a szélirányt és sebességet, mely a helyválasztás szempontjából a legfontosabb tényező. A pontyok hidegvérű állatok, tehát mindig is a melegebb vízrétegeket érdemes megcélozni. Ez szélirány tekintetében azt jelenti, hogy meleg időben a szél mozgatja a felsőbb vízréteget melyet a halak követnek, ellenben hideg időben a szél hűlő vizet szállít, mely taszítja a halakat. Van egy régi mondás is ezzel kapcsolatban mely így szól: „meleg időben köszönj a szélnek, hideg időben pedig tudd a szelet a hátad mögött”. Egy fizetős real-time adatszolgáltatás lehetősége, időt takarít meg és elősegíti a gyors döntéshozatalt az eredmények elérése érdekében amellet, hogy pénzt spórol a horgászoknak (vö. 1.6 alfejezet).

1.2. Célkitűzés

A cél olyan hardver eszköz elkészítése (vö. 3.2 – 3.3 alfejezet), illetve olyan adatelemzőrendszer alapjainak a kialakítása (vö. 3.5.9 alfejezet), mely széles körű adatgyűjtésre alkalmas a horgásztavaknál és ami internetes kapcsolaton keresztül ezeket webszerverre tölti. Az ott lévő adatokat nyílt internethasználók megtekinthetik és kiértékelhetik (vö. 3.4.5 alfejezet) a horgász élményeinek növelése érdekében. A szakdolgozat elsődleges célja tehát, ezen hardvereszköz megépítésének (vö. 3. fejezet), valamint a részegységek működésének a megismertetése (vö. 3.2 alfejezet). Továbbá valamennyi eszköz működési összhangjának a megteremtése (vö. 3. fejezet) és annak bemutatása a műszaki paraméterektől, a kommunikációs lehetőségeken (vö. 3.2 alfejezet) és a programozáson keresztül (vö. 3.4 alfejezet), az általános GUI megjelenítéséig (vö. 3.4.5 alfejezet).

1.3. További fejlesztési lehetőségek

A projekt megvalósítása során több fejlesztési lehetőség fogalmazódott meg, melyek a jelenlegi terjedelmi és időbeli korlátok miatt nem lehetnek témái a dolgozatnak, azonban a konkrét elképzelések röviden ismertetésre kerülnek a 3.6. alfejezetben. Ezek a következők:

- PH szenzor alkalmazása
- Víz oxigénszintjének a mérése
- Vízhőmérséklet mérésének a kiterjesztése: különböző helyeken, különböző vízmélységben
- LoRa kommunikáció kialakítása nagyobb távolságok elérése érdekében
- Tápellátás fejlesztése napelemes rendszer kialakításával
- Szoftverfejlesztés az energiatakarékosság elérése érdekében (Deep Sleep)
- Hardveres fejlesztés élőkép stream-elése a felhasználók felé
- Esőszenzor kiépítése

- Online lehetőség biztosítása a felhasználóknak: online fogási eredmények regisztrálása, annak érdekében, hogy a MI pontosabb predikciót (darab, tömeg) készítsen.

A felmerült ötletek egyik része műszaki jellegű, mely segít költséghatékonyabbá tenni az üzemeletetést (pl.: napelemes tápellátás, Deep Sleep kialakítás). A fejlesztések másik része externális adatokra számít (pl.: adott időpontban kifogott halak mennyisége, mérete, alkalmazott csali jellege). Aktív szerepet szán a motivált közreműködő horgászoknak, felhasználóknak azáltal, hogy GUI felületen az aktuális eredményekről (pl.: milyen csalitípussal, mikor milyen halat fogtak milyen tömegben) beszámolnak. Továbbá segítené az adatvagyon építését az elérhető online halfogási statisztikák (pl.: HORINFO, a Magyar Országos Horgász Szövetség információs rendszere) bekapcsolása a rendszerbe.

A harmadik rész pedig mindezeket ötvözve, a mesterséges intelligencia támogatásával segítené az élmények gyarapítását, ezzel együtt emlékezetesebbé téve a kikapcsolódással töltött értékes pillanatok. Az érzékelt adatok predikciót segítő optimalizálása (vö. 3.5.9 alfejezet), finomítása szintén a jövő kihívásait gyarapítja, mivel ezek szinte önálló matematikai (vö. 2.3.1 alfejezet), informatikai (vö. 2.8 alfejezet) és ökológiai (vö. 2.10 alfejezet) problémák. Például a szélirány kihasználása, a nagy vízfelületű tavak víz hőmérséklet mérése különböző mélységekben stb.

1.4. A dolgozat szerkezete

A dolgozat bemutatja a téma és a tanult tantárgyak összefüggéseit, az ezzel kapcsolatos kompetencia változás eredményét (vö. 2.* fejezet), azonos logika szerint felépítve, ezzel is segítve az értelmezést.

Az Olvasót lépésről lépésre eljuttatja a projekt minden egyes elemének és működésének a megértéséhez, igyekszik láttatni a globális módon a teljes projektet. (vö. 3.* fejezet).

A dolgozat szerkezeti, formai felépítésének az alapját „Kodolányi János Egyetem Szakdolgozati Szabályzata”:

(Kodolányi János Egyetem – https://www.kodolanyi.hu/ugyfelszolgalat/images/tartalom/File/informatika_tsz/kje_egyseges_szakdolgozati_szabalyzat_2021marc17_hatalyos-1.pdf – Letöltés: 2025.02.12.),

a „KJE BPROF-képzésének szakdolgozati specialitásai”:

(Kodolányi János Egyetem – https://www.kodolanyi.hu/ugyfelszolgalat/images/tartalom/File/informatika_tsz/kje_bprof_szakdolgozat_specialitasok.docx – Letöltés: 2025.02.12.)

és az egyetemi könyvtár végzős hallgatói számára publikált „Minta Szakdolgozat”:

(Kodolányi János Egyetem – https://www.kodolanyi.hu/konyvtar/images/tartalom/File/Honlapra/Feltoltes/szakdoli_minta.pdf – Letöltés: 2025.02.12.) képezi.

A dolgozat első fejezetében került megfogalmazásra a témaválasztás motivációja, valamint az, hogy milyen rövidtávú célt szeretnék megvalósítani, amellet, hogy a körvonalazódott távoli elképzelések is kirajzolódtak.

A második fejezet mutatja be a mintatanterv tantárgyainak és a dolgozat témájának közvetlen kapcsolatát, valamint a tanultak alkalmazását a projekt megvalósítása közben. Ez a struktúra biztosítja a dolgozat teljes szakirodalmi előzményének feldolgozását. Ezen tantárgy-orientált alfejezetek mellett a horgászat ökológiai összefüggésrendszere is megjelenik a klasszikus szakirodalmi fejezetben (vö. 2.10 alfejezet).

A harmadik a fejezet részletezi az eszközök elektronikai, fizikai kialakításának és működésének részleteit. Választ ad azokra a kérdésekre, hogy milyen hardvereket tartalmaz, azok milyen műszaki paraméterekkel rendelkeznek, hogyan kapcsolódnak egymáshoz, hogyan kommunikálnak, milyen adatbázis épül, hogyan és milyen módon, milyen eszközökkel történt a programozásuk.

A negyedik fejezet egy online kérdőíves felmérést és annak értékelését tartalmaz.

Az ötödik fejezetben kerültek megfogalmazásra a projekt megvalósítása közben felmerülő kérdések, melyek érintik az adatok megbízhatóságát, rendszerbiztonságot, és a projekt tényleges hozzájárulását a horgászat sikeréhez.

A hatodik fejezetben igyekszem választ adni a negyedik fejezetben felmerült vitatható témákra. A hetedik fejezetben, azaz a jövőképben, mind azon fejlesztési perspektívákról van szó, amelyek a projekt elkészítése közben előkerültek és életszerű igényként jelentkeztek, vagy jelentkezhetnek a későbbiek folyamán.

A nyolcadik fejezetben, röviden összefoglalom a projekttel kapcsolatos tudnivalókat és célokat, magyar, illetve angol nyelven is.

A kilencedik fejezet a projekttel kapcsolatban áttekintett és felhasznált források részletei tartalmazza

A tizedik fejezet tartalmaz minden olyan mellékletet melyek segítették a projekt összeállítását és terjedelmi okokból indokolatlan volt a szakdolgozat szövegében szerepeltetni.

A dolgozat a terjedelmi korlátok okán nem foglalkozik részletesen az alábbi gondolkörökkel:

- Nem cél a COCO (<https://miau.my-x.hu/my-x-factor-y>) működésének bemutatása, csak annak felhasználása a bizonyításához.
- Nem volt cél az alkalmazott protokollok részletes ismertetése, csak azok használata.
- Továbbá nem volt cél a forráskódok programsor szerinti részletezése.

- Nem készült pontos kalkuláció egy valós üzleti tervhez, hanem csak egy becslés a többletérték megfogalmazásához.

A dolgozat formátum-szabályai követik a Kodolányi János Egyetem Szakdolgozati Szabályzatában és a KJE BPROF-képzés szakdolgozati specialitásainak az előírásait, azonban a forráskódok a szemléletesség érzékeltetése céljából – eltérő Consolas betűtípussal, 10,5-ös betűmérettel és sötétszürke háttérrel (hex: #1F1F1F), az egyéb magyarázó programsorok Consolas betűtípussal, 10,5-ös betűmérettel, fehér színű háttérrel kerültek kiemelésre. A beépítésre kerülő eszközök műszaki specifikációi terjedelmi okok (78 oldal) miatt nem kerültek tételesen a mellékletek közé, hanem az elérhetőségeik linkjei vannak felsorolva (vö. 10.7 alfejezet).

1.5. Célcsoport

A fejlesztésem elsődleges célcsoportját azok a horgászok alkotják, akik nyitottak a modern technológia horgászatban való implementálására vagy szeretnek kísérletezni a tapasztalataik mellett a műszaki adatok összevetésével is. Szerencsére ez a célközösség exponenciálisan gyarapszik. Egyre több horgász ismeri föl a környezeti adatok szükségességét a horgászatban, valamint a különböző rendszerek és szenzorok ötvözéséből álló eszközök pozitív hatását. Kiemelt figyelmet kapnak azok a horgászok, akik már rendelkeznek modern eszközökkel például GPS-es etetőhajóval, szonárral, halradarral, fogáskísérő applikációval vagy egyéb digitális horgászati eszközökkel. Ők nagyobb valószínűséggel érdeklődnének egy olyan rendszer iránt, amely valós idejű adatokat szolgáltat a vízterületről, ezzel is megkönnyítve a horgászat tervezési fázisát is. Másodlagos célcsoportnak tartom azokat a turistákat, akik évente csak néhány alkalommal, kikapcsolódásként kívánnak horgászni.

1.6. Hasznosság

A horgászok körében köztudott tény, hogy a környezeti tényezők befolyásolják a halak táplálkozását és helyezkedését is. A rendszerem valós idejű adatszolgáltatást nyújt az érdeklődő horgászok számára, aminek segítségével pontosabb képet kaphatnak az általuk szeretett vagy egy új és ismeretlen vízterületről egyaránt. Nemcsak a horgászatot napokkal megelőző tervezést, hanem közvetlen a horgászat megkezdése előtti időszakot is egyszerűsíti számunkra. Például a víz hőmérséklet, szélesség és a szélirány a helyválasztásban nyújt támpontot, a „*légnyomás és a víz hőmérséklet*” (Haldorado – Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – [12](https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-</p>
</div>
<div data-bbox=)

2025.03.18.) az etetés mennyiségét és minőségét határozza meg.

A rendszer nemcsak a tapasztalt profi horgászok számára kedvez, hanem a kezdő és kíváncsi horgászoknak is egyaránt.

Többszörösérték-becslés (vö. 1. ábra) a ChatGPT információi alapján:

Tényezők	Profi horgász	Amatőr horgász
Horgászatok száma évente	50 db	20 db
Egy horgásznap költsége	15 000 Ft	6 000 Ft
Elkerült "rossz napok" az eszköz információ alapján	10 db	4 db
Fogási siker (jobb helyszín/ideje miatt)	5 hal	2 hal
Egy extra hal értéke (élmény + költség)	3 000 Ft	1 500 Ft
Elkerült felesleges napok értéke:	10 nap x 15 000 Ft = 150 000 Ft	4 nap x 6 000 Ft = 24 000 Ft
Többszörösérték értéke:	5 hal x 3 000 Ft = 15 000 Ft	2 hal x 1 500 Ft = 3 000 Ft
Éves információs többszörösértékek:	165 000 Ft	27 000 Ft

1. ábra: Többszörösérték becslés

Forrás: ChatGPT konverzáció (vö. 10.5.5. alfejezet)

„Ha egy rendszer megbízható időjárási és vízhőmérsékleti adatokkal látja el a horgászokat, az információs többszörös több tízezer forintnyi értéket is jelenthet évente egy-egy felhasználó számára. Ez a becsült érték szegmentálható célcsoportonként és akár üzleti modell alapjául is szolgálhat (pl.: előfizetési alkalmazás).” (ChatGPT – vö. 10.6.4 – Letöltés: 2025.04.05.). A becslés a nyereséget, úgy tárgyasítja, hogy a sikertelen alkalmakra el nem költött összeget meghatározza, amihez hozzáadja a kedvezőbb helyszínek hozzávetőleges eredményét. Profi horgászok esetében ez évi kb. 165.000 Ft, míg amatőrök esetén kb. 27.000 Ft. (vö. 1. ábra) A tényleges hasznosság megközelítését kiegészíti a kérdőívem (vö. 4. fejezet). A kérdőív 13. kérdése, mely szerint: *Használna-e egy olyan alkalmazást, amiből napra készen látná egy hazai víz időjárási viszonyait (szélirány, szélereősség, levegő és vízhőmérséklet, légnyomás), a válaszadók 69,5 %-a „igen” -nel válaszolt, amennyiben ez ingyenes. 13.3 % mondta, hogy fizetne is érte (vö. 56. ábra). Üzletpolitikai szempontból átgondolandó, hogy ne a horgász, hanem a tavak üzemeltetői fizessenek az eszközért, így növelve a saját horgászai sikerét, közvetve pedig a bevételeiket.*

Szintén a ChatGPT segítségével készítettem egy rövid szoftverfejlesztési/licence díj kalkulációt (vö. 2-3. ábra).

A számításnál figyelembe vett alapadatok:

- Szoftverfejlesztés 250.000 Ft
- Hardvereszközök értéke 150.000 Ft/db
- Hardvereszközök átlagos havi fix költsége (internet előfizetés, tárhelybérlet, domain bérlet, áram) 10.000 Ft/eszköz
- Marketing 200.000 Ft/összes eszköz
- Havi terméktámogatás 10.000 Ft/eszköz
- Felhasználó száma körülbelül 50 db horgásztónál, 5.000 fő.

Fix költségek			
Költségtípus	Mennyiség	Egységköltség	Összesen
<i>Szoftverfejlesztés</i>	<i>1 db</i>	<i>250.000 Ft</i>	<i>250.000 Ft</i>
<i>Eszközbeszerezés</i>	<i>50 eszköz</i>	<i>150.000 Ft</i>	<i>7.500.000 Ft</i>
<i>Marketing</i>	<i>1 kampány</i>	<i>200.000 Ft</i>	<i>200.000 Ft</i>
Összesen (egyszeri)			7.950.000 Ft

2. ábra: Fix költségek

Forrás: Saját képernyőkép – Jelmagyarázat: felmerülő fixköltségeket tartalmazza forintban kifejezve.

Havi költségek		
Költségtípus	Egységköltség / eszköz	Összesen / hó
<i>Fix költség</i>	<i>10.000 Ft</i>	<i>500.000 Ft</i>
<i>Terméktámogatás</i>	<i>10.000 Ft</i>	<i>500.000 Ft</i>
Havi összesen		1.000.000 Ft

3. ábra: Havi költségek

Forrás: Saját képernyőkép – Jelmagyarázat: felmerülő havi költségeket tartalmazza forintban kifejezve.

„Kalkulált fajlagos költségek

Egyszeri költség / felhasználó:

$$7.950.000 \text{ Ft} / 5.000 \text{ fő} = 1.590 \text{ Ft} / \text{fő}$$

Havi működési költség / felhasználó:

$$1.000.000 \text{ Ft} / 5.000 \text{ fő} = 200 \text{ Ft} / \text{fő} / \text{hó}$$

Következtetések és javaslatok

- *A rendszer költséghatékony, ha eléri az 5.000 fős felhasználóbázist, hiszen havi 200 Ft/fő fenntartási díj nem magas.*
- *Egyszeri belépési díj (pl.: éves előfizetésbe építve) érdemes lehet, pl.: 2.000–2.500 Ft/fő, amivel a kezdeti beruházás 1 év alatt megtérülhet.*
- *Célszerű lehet freemium modellt alkalmazni: alapszolgáltatás ingyenes, prémiumért (pl.: részletes elemzések, múltbeli trendek, fogási előrejelzések) fizetnek a felhasználók.*
- *A profi horgászok célcsoportja hajlandó lehet akár 500–1000 Ft/hó díjat is fizetni a plusz adatokért.*
- *Az amatőrök esetében inkább az egyszerűség, élményfaktor a döntő – itt lehet alacsonyabb havi díjjal számolni.” (Költségkalkuláció – www.chatgpt.com, Letöltés: 2025.04.11.)*

A fentiek figyelembevételével és az alapadatok pontosabb meghatározásával már készíthető egy üzleti terv, ami megalapozhatja egy vállalkozás beindítását.

2. Elméleti alapok, szakirodalmi áttekintés

A fizikailag megvalósult projekt, illetve és a megépített hardver eszköz egy hosszú felkészülési folyamat eredménye. Nem nélkülözhetette sem az elméleti, sem pedig a gyakorlati tudást, melyet a képzés során kaptam. E fejezetben mutatom be tudásmodulonként, tantárgyanként, a tananyag és a projekt elméleti kapcsolatait megvilágítva, a tanultak gyakorlati alkalmazását.

A mesterséges intelligencia ma már megkerülhetetlen része a fejlődésnek. Az eddig felhalmozott tudás megfelelő szegmenseihez való gyors hozzájutás a jövőbeli sikerek egyik alapja. Tudni kell méríteni ebből a globális ismeretből, tudni kell hitelességéről meggyőződni és tudni kell a gyakorlatba átültetni azokat. A dolgozat részletezi a GPT-4-turbo támogatását a projekt megvalósítása során (vö. 10.5 alfejezet), ahol teljes konverzációk megtalálhatók.

2.1. Gazdasági és humán ismeretek

Az informatika tudományának nemcsak műszaki kötődései vannak, hanem jelentős gazdasági és társadalmi vonatkozásai is. Egyre inkább együtt kell működnie a pénzügyi, jogi szakterületekkel, valamint a humán erőforrás és projektmenedzsmenttel. Ezek az ismeretek segítenek egy komplex szemléletet kialakítani.

A halászat a gazdaság része, a horgászat pedig a sporthorgászat kivételével az egészséges táplálkozást is szolgálja. Emellett a sporthorgászattal együtt jelentős sport, versenysport és hobbi, ami a humán tevékenységek és az egészséges életmód fontos területe.

„A WHO egyik alapidokumentuma szerint: „az egészség nem pusztán a betegségek hiányát, hanem az egyén teljes testi, lelki, szellemi és szociális jól – létét jelenti.” (Butor Klára: Intézményi sport stratégia és hallgatói jól - lét program – Kodolányi János Főiskola PIQ&LEAD képzési modelljében – <https://www.kodolanyi.hu/kutatas-fejlesztes/kiadvanyaink>– Letöltés: 2025. 04. 05. – In: Schumann Júlia: A közösségfejlesztés útjai, 2016 – Kodolányi János Főiskola)

Mivel a szakdolgozatom legfőképpen a horgászatot, de emellett a turizmust és vízi sportokat is segíti, pozitív hatást gyakorol magára az emberre, de a humán ismeretekre és az akvakultúrára is.

2.1.1. Európai civilizáció és identitás

A természethez való viszony az európai identitás nagyon fontos és meghatározó része. A horgászat nemcsak pénzkeresési lehetőség és nem is csak hobbi, hanem igen komoly történelmi és kulturális megjelenése van az európai életben. Az ősember már a kezdetektől halászott és vadászott. Valójában az európai és magyar civilizáció részeként tekintünk rá.

„Adj halat az éhezőnek és aznap megmented a haláltól. Tanítsd meg őt halászni-örök életére jóllakott lesz, sem ő sem családja nem szenved majd hiányt.” tartja egy régi kínai mondás. „Az őskori leletek jól mutatják, hogy a korabeli (ős)ember kézzel, nyárssal, majd fa, később, kő és fém horgokkal és szigonyokkal próbált a vizekből zsákmányt szerezni. Az ókorban a különböző kultúrákban (kínai és római birodalom), térben és időben egymástól távol alakultak ki napjaink halgazdálkodásának-haltenyésztésének előfutárai...” (Dr. Mézes Miklós, Dr. Bokor Zoltán, Csorbai Balázs, Dr. Ferincz Árpád, Dr. Staszny Ádám, Dr. Hegyi Árpád, Dr. Eszterbauer Edit, Dr. Urbányi Béla 2020- – Halászat, haltenyésztés – Hermann Ottó Intézet Nonprofit KFT., Budapest.) A saját történelmünkben is nagy jelentőséggel bír a halászat, ugyanis a magyar vándorlás során, mindenhol *„folyók, tengerek mellett telepedtünk le”* (V. Árpád népe – https://nat2012.nkp.hu/tankonyv/tortenelem_5/lecke_05_040 – Letöltés: 2025.03.18). A tavaink, folyóink az ivóvíz, az állatok itatása mellett, a halászat adta lehetőségek kihasználását is jelentette.

„Európában a kereszténység is sokat lendített a halászatban: a böjti időszakban is fogyasztani lehetett a halételeket...” *„A kolostorok környékén alakultak ki az első kezdetleges halgazdaságok...”* (Dr. Mézes Miklós, Dr. Bokor Zoltán, Csorbai Balázs, Dr. Ferincz Árpád, Dr. Staszny Ádám, Dr. Hegyi Árpád, Dr. Eszterbauer Edit, Dr. Urbányi Béla 2020- – Halászat, haltenyésztés – Hermann Ottó Intézet Nonprofit KFT., Budapest.)

A kutatásom, az évszázadok alatt összegyűjtött horgászati tapasztalatot és a modern technológiát ötvözi, a hatékonyság garantált növelésének érdekében.

2.1.2. Komplex társadalomtudományi ismeretek

Az órán megtanultam, hogy a technológia és társadalmi hatások szoros összhangban vannak egymással. Az általam készített érzékelőkkel ellátott szerkezet nem csupán egy technológiai eszköz, társadalmi hatással is bírhat a horgászok közösségében. Napjainkban már rengeteg új szerkezet és fejlesztés van, köztük az én projektem is. Átalakítja a horgászatot tudományosabbá és hatékonyabbá téve azt, ami egy nagyszerű példa a társadalmi változásokat vizsgálóknak. Ugyanis az emberek szokásait, életmódját, szinte kivétel nélkül befolyásolja az innováció.

A horgászok között is több, valójában az összes generáció megtalálható.

Az X és az idősebb generációk nem tudják és nem is szeretik használni az, az internetes alkalmazásokat. Őket a kifejlesztett mérőeszközök érdeklik majd valószínűleg. Az Y és fiatalabb generációk a kifejlesztendő applikációt is használni fogják, hiszen az élet minden területén, például az álláskeresés során is alkalmazzák.

„Az X generáció még digitálisan szinte tudatlan, ritkán keres az interneten munkahelyet...”

„Az Y generációk tagjainak már meg kellett tanulni az internetes munkalehetőséget ajánlókat, az online platformokat, a felhő alapú adattárolást és a videóhívásokat ... megtanulták, kikényszerítette az élet a digitális tartalmak feldolgozását, létrehozását” (Malatyinszki Szilárd: Az új generáció, mint erőforrás – In: Multidiszciplináris kihívások, sokszínű válaszok, ISSN 2630-886X , 2020. (9. évf.), 1. sz. – :http://epa.oszk.hu/03400/03448/00019/pdf/EPA03448_multi-diszciplinaris_2020_1_090-101.pdf – Letöltés: 2025. 04. 05.).

2.1.3. Emberi viselkedés és kommunikáció

A projektben a hangsúly bár az adatgyűjtésen van, de közel sem csak ennyiről szól. Jelentős fontossággal bír ugyanis az adatok megfelelő feldolgozása és a felhasználók felé való kommunikálás is. Az információkat és adatbázisokat egy weboldalon keresztül teszem elérhetővé az oldalam látogatói, felhasználó számára. Ez a folyamat szorosan kapcsolódik a kommunikációhoz, hogy az adatot könnyen értelmezhető állapotban jelenítsük meg.

„A mai fiatalok viselkedési szokása, reagálása, nézőpontja több területen megváltozott. Újak a hozzájuk vezető kommunikációs eszközök, eltérő a motivációs rendszerük, más az eszközigényük és teljesen megváltoztak a nézőpontjaik is.” (Malatyinszki Szilárd: Az új generáció, mint erőforrás - In: Multidiszciplináris kihívások, sokszínű válaszok, ISSN 2630-886X, 2020. (9. évf.), 1. sz.,http://epa.oszk.hu/03400/03448/00019/pdf/EPA03448_multidiszciplinaris_2020_1_090-101.pdf – Letöltés: 2025. 04. 05.)

2.1.4. Vállalati gazdaságtan

A vállalati gazdaságtan egyik alapvető eleme a fejlesztés, annak az esetleges piaci értéke. Az általam fejlesztett rendszer akár piaci értékkel is bírhat, hiszen a felhasználása nemcsak a horgászok körében lehetséges, hanem akár a strandokon, nyílt vizeken, öblökben és különböző turisztikailag jelentős területeken is. Megfelelő ambíciókkal és üzleti modellel vállalkozás is épülhet rá, horgászok, illetve turisztikai irodák számára, előfizetési alapon.

„A vállalkozásalapítás ösztönzőleg hat a gazdaság fejlődésre, a politikai döntéshozók a vállalkozói tevékenységet a foglalkoztatás, a gazdasági, valamint regionális fejlődés motorjának tekintik.” (Timár Gigi, Szennay Áron, Csákné Filep Judit – A vállalkozásoktatás hatása a vállalkozói aktivitásra a közép-kelet-európai tranzíciós országokban – In: Közgazdaság - Gazdaságelméleti és -politikai szemle 2025. évf. 1. szám)

2.1.5. Vezetési és vállalkozási ismeretek

A piaci részek kiaknázása a sikeres vállalkozás alapja. Az én rendszerem egy ilyen piaci résre is kínálhat megoldást. A környezeti adatok meghatározása és az esetleges statisztikai elemzések kimutatása a kezdő és a profi horgászoknak is egyaránt hasznos és keresett szolgáltatás lehet. A halastavainkat sokszor magánszemélyek, családi vállalkozások üzemeltetik, néha nagyobb cégek. Mindegyikhez másképpen vezet az út és másképpen kell megismertetni velük az alkalmazásomat. Minden generációnak az alkalmazás öt érdeklő tulajdonságait kell kiemelni.

„Magyarországon igen nagyok az eltérések a cégek tőkeerejében, rugalmasságában, innovációjában és természetesen a fiatalokkal való kapcsolataikban is. Munkám során többször tapasztaltam, hogy egy-egy vezető ugyanúgy próbál a különböző generációkhoz utat találni, ugyanazokkal az eszközökkel, motivációs eszközökkel, stílussal, támogatói háttérrel.” (Malatyinszki Szilárd: Az új generáció, mint erőforrás - In: Multidiszciplináris kihívások, sokszínű válaszok,

ISSN 2630-886X, 2020. (9.évf.), 1.sz.,
http://epa.oszk.hu/03400/03448/00019/pdf/EPA03448_multidiszciplinaris_2020_1_090-101.pdf – Letöltés: 2025. 04. 05.)

2.2. Jogi ismeretek

A környezeti mérésekből származó adatok online megjelenítésekor a jogi követelmények és előírások figyelembevétele egy kötelező aspektus. A rendszer kialakítása során nemcsak a műszaki megvalósításra, hanem az információk jogszerű és biztonságos kezelésére is kiemelt figyelmet kell fordítani. A jogi ismeretek fejezet célja, hogy bemutassa a tantárgyakon tanult jogi szabályok, projektemhez kapcsolódó részeit.

2.2.1. Szakterületi jogi ismeretek

A jog teljesen áthatja a szakdolgozatom minden területét, mint például a horgászat és a különböző platformokon történő megjelenés jogi szabályozása, a GDPR és a szellemi munka szabaddalmi kérdései által, hogy csak néhányat említsek.

Ez a munka nem jött volna létre, ha nem lennék hosszú évek óta horgász. Ehhez azonban először is egy horgászvizsgát tettem le, aminek a részeként elsajátítottam a halak ismeretén túl, a betartandó szabályokat. *„Az 1997. évi XLI. törvény a halászsátról és a horgászsátról, valamint a 78/1997. sz. rendelet az, ami szabályozza ezt a tevékenységet. Megtalálhatjuk benne a tilalmi időket, a méret- és fajlagos tilalmat, a tiltott helyeket, módszereket és eszközöket és a védett halfajainkat, amelyek a hetvenes évektől folyamatosan bővülnek.”* (Dr. Juhász Lajos, 2011 –

Magyar horgászvizeken – Tóth Könyvkereskedés és Kiadó KFT., Debrecen – Letöltés: 2025.03.22.)

A szakdolgozat készítése során is be kell tartani a szabályzatokat, törvényeket.

2.2.2. Projekt kapcsolódása a jogi szabályzásokhoz

A Kodolányi János Egyetem szakdolgozatkészítési előírásait, valamint a szerzői jog védelmét, melyet az 1999. évi LXXVI. törvény garantál. A „*plagizálás*”. (Nemzeti Jogszabálytár – <https://njt.hu/jogszabaly/1999-76-00-00.24> – A szerzői jogokról – Letöltés: 2025.03.22.). elkerülése érdekében például elengedhetetlen a felhasznált források pontos megjelölése.

Ez a törvény „*már külön műfajként, a VI. fejezetben foglalkozik a számítógépi programalkotásokkal.*” (Horváth László, Pirkó József, 2001 – Informatikai tudástár, Kiskapu Kiadó, Budapest.)

A szakdolgozat szellemi terméke egy számítógépes program, alkalmazás, amit a „*szoftver irányelv*” is véd, „*az Európai Tanács 2009/24/EK*” irányelve. (Mezei Péter, Harkai István, 2024 – A platformgazdaság szerzői jogi kihívásai – Orac Kiadó, Budapest).

A megvalósuló projekt során a GDPR alapján, „*személyes adatot (pl.: név, e-mail), csak az illető beleegyezésével kezelhetünk, különleges adatot (pl.: szexuális irányultság, vallási hovatartozás) egyáltalán nem is kezelhetünk.*” (MIAU – <https://miau.my-x.hu/miau/322/itb/> – Szemelvények az információbiztonság teréről – Letöltés: 2025.03.22.)

2.3. Természettudományos alapismeret

Természettudományos alapismeretek nélkül, nem lehet az informatika területén jól teljesíteni. Ezek a tudományok megalapozzák az algoritmikus, programozói szemlélet kialakulását, továbbá elengedhetetlenek a műszaki eszközök világában történő tájékozódáshoz

2.3.1. Matematikai alapok

„*Olyan ismeretek gyűjteménye, melyek minden későbbi és párhuzamos (nem csak az IT-tárgyak esetén, hanem az összes) tantárgy kapcsán segítik a knuth-i/pirsig-orientált értelmezést, a tanulást (vö. GPS = general problem solving vs. kompetencia).*” (MIAU – https://miau.my-x.hu/mediawiki/index.php/BROF-tant%C3%A1lrgyak#Matematikai_alapok_.28ea.29 – Matematikai alapok (ea) – Letöltés: 2025.03.19.)

A matematika nélkül, nincs számítástechnika. Számelméleti, halmazelméleti, kombinatorikai logikai ismeretek nélkül, nem lehetséges informatikai tervező munkát végezni és leprogramozni egy-egy probléma megoldását.

Esetleges statisztika készítésére a matematikai alapok elengedhetetlenek. A halak viselkedése és a környezeti tényezők közötti párhuzamot, összefüggéseket matematika modellek segítségével könnyű ábrázolni. A statisztikai pontosság meghatározására korreláció számolása van szükség. Ez egy nagyon hasznos művelet, mellyel akár azt is megérthetjük, hogy milyen feltételek vannak leginkább hatással a halak táplálkozási szokásaira, ami egy eredményesebb horgászathoz vezethet.

2.3.2. Az elektronika fizikai alapjai

„A mikroelektronikai eszközök alkalmazása mára igazán tömegessé és meghatározóvá vált mind a termelés, mint a szórakozás területén. A mikroelektronika beépült minden napjaink eszközeibe és nehezen tudjuk elképzelni életünket nélküle.” (Dr. Mojzes Imre, 2005 – Mikroelektronika és technológia; Műegyetemi kiadó). Tapasztalatom szerint ez a gondolat napjainkban hatványozottan állja meg a helyét. A szakdolgozat sokat épít a mikroelektronika eredményeire, nem érintve az alkalmazott eszköz teljes belső működését. Fő hardverelemei a mikrovezérlők, vagy más néven mikrokontrollerek. *„A mikrokontrollerek többnyire 8 bites redukált utasítás készlettel és nagy számú párhuzamos adatkimenettel ill. bemenettel rendelkező processzorok, amelyeknek fő alkalmazási területe a szabályozástechnika.”* (Dr. Mojzes Imre, 2005 – Mikroelektronika és technológia; Műegyetemi kiadó).

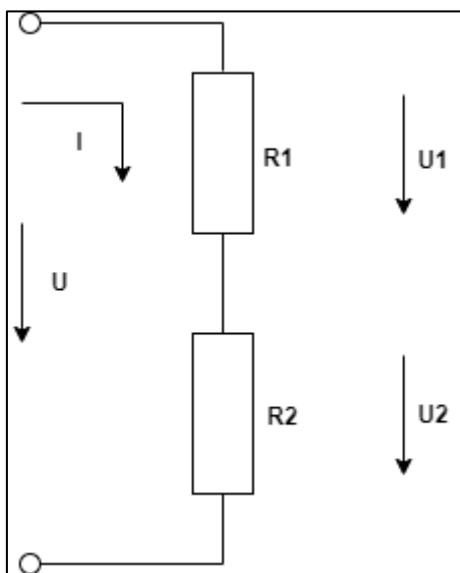
A projektben az ESP32-es mikrovezérlő és a hozzá kapcsolt érzékelők közvetlen kapcsolatban állnak Az elektronika fizikai alapja tantárggyal. Az elektronika törvényei mind jelentős részei voltak az eszköz bekötési rajzának és tervének a felépítésében. Elengedhetetlenek ezek az ismeretek egy stabil, megbízható eszköz felépítéshez. A beépített áramkörök és az érzékelők dolgozzák föl és továbbítják az adatot, melyeket ideális áramköri elvek és megfelelő áramellátás mellett tesznek biztonságossá.

2.3.3. Elektronikus áramkörök

A fizikai számítástechnika *„Szűkebb értelemben olyan interaktív rendszerek létrehozását jelenti hardverek és szoftverek segítségével, melyek képesek érzékelni a világban létrejövő jeleket és reagálni is tudnak rá.”* (Harsányi Réka, Juhász Márton András 2014 – Fizikai számítástechnika: elektronikai alapok és Arduino programozás – Typotex Kiadó, Budapest.) A projektet kifejezetten támogatta ez az könyv, mivel az alapoktól a megvalósításig hasznos tudással látott el. *„A fizikai számítástechnika fontos részét képezi még a DIY (do it yourself = csináld magad) vonal.”* (Harsányi Réka, Juhász Márton András 2014 – Fizikai számítástechnika: elektronikai alapok és Arduino programozás. – Typotex Kiadó. Budapest.)

Maga a mikrovezérlő egy komplex eszköz, melynek belső kialakítása maximálisan kiszolgálja annak használatját. (vö. 3.2.1 alfejezet). Néhány alkatrész beépítésére azonban szükség volt a helyes működés, illetve az alkatrészek védelme szempontjából. Ilyen volt a szélesség és a szélirány érzékelő szenzor. A mikrovezérlő adatlapja alapján, annak lába legfeljebb 3,3V feszültséggel terhelhető, azonban mindkét eszközök – gyári kialakítás szerint – 0-5V a kimeneti feszültséget ad. Mindkét esetben feszültségosztó kapcsolásra volt szükség ellenállások segítségével.

A feszültségosztás alapelve (vö. 4. ábra), hogy „soros kapcsolásban az egyes ellenállásokon fellépő feszültségek úgy aránylanak egymáshoz, mint az ellenállásaik értékei.” (Centroszet – https://centroszet.hu/tananyag/elektro_bovitett/343_feszltsgoszt.HTML – Feszültség osztás – Letöltés: 2025.03.18.)



4. ábra Feszültségosztás alapelve
Forrás: Saját szerkesztés – Draw.io

Az Ohm törvény „A fogyasztóra kapcsolt feszültség egyenesen arányos a fogyasztón átfolyó áram erősségével. Ez Ohm törvénye. Nevét felfedezőjéről, Georg Simon Ohm (1787–1854; ejtsd: georg szimon om) német fizikusról kapta. Róla nevezték el az elektromos ellenállás mértékegységét.” (NKT – https://nat2012.nkp.hu/tankonyv/fizika_8/lecke_01_006 – Ellenállás, Ohm törvénye – Letöltés: 2025.03.19.) alapján:

$$\text{ellenállás}(R) = \frac{\text{feszültség}(U)}{\text{áramerősség}(I)}$$

betűjelekkel:

$$ohm [\Omega] = \frac{volt [V]}{amper[A]}$$

ez alapján:

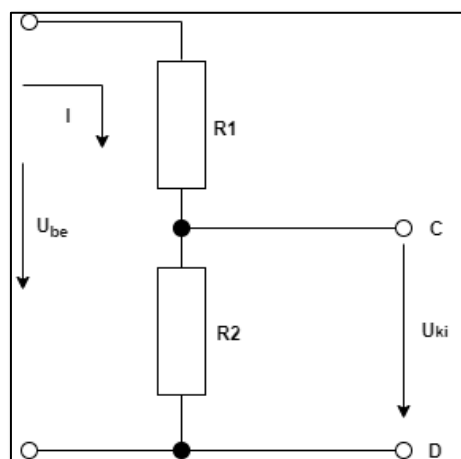
$$I = \frac{U_1}{R_1} \quad I = \frac{U_2}{R_2}$$

$$\frac{U_1}{R_1} = \frac{U_2}{R_2} \quad \rightarrow \quad \frac{U_1}{U_2} = \frac{R_1}{R_2}$$

„A feszültségosztó egy négyfázis. A legegyszerűbb esetben két sorba kötött ellenállásból áll A rendszer A és B pontja közé feszültséget kapcsolva (ez a négyfázis bemenete) áram alakul ki, és az ellenállásokon feszültségesés lép fel. Kirchhoff II. törvénye értelmében a két feszültség összege mindig megegyezik a tápláló generátor feszültségével.” (Centroszet –https://centroszet.hu/tananyag/elektro_bovitett/343_feszltsgosztas.HTML – Feszültségosztás – Letöltés: 2025.03.10.)

A 5. ábra alapján, ha $U_{be}=U_1+U_2$ és $U_2=U_{ki}$, akkor felírható a feszültségosztás törvénye:

$$\frac{U_{ki}}{U_{be}} = \frac{R_2}{R_1 + R_2} \quad \rightarrow \quad U_{ki} = U_{be} \frac{R_2}{R_1 + R_2}$$



5. ábra Feszültségosztás törvénye
Forrás: Saját szerkesztés – Draw.io

Mivel az $U_{ki}=3,3V$, a $U_{be}=5V$, valamint, ha a $R_1=1k\Omega$, így a fenti képletbe behelyettesítve, majd rendezve az egyenletet:

$$3,3V = 5V \frac{R_2}{1k\Omega + R_2}$$

$$R_2 = \frac{3,3V}{1,7V} = 1,9k\Omega \sim 2k\Omega$$

Tehát egy $1k\Omega$ -os és egy $2k\Omega$ -os ellenállást választva, megfelelő jelfeszültséget kapunk, ami már közvetlen beköthető a mikrovezérlőbe.

Az alapelvek, melyeket többek között az Elektronikus áramkörök tantárgyon tanultak nagyban hozzásegítettek a rendszer kialakításában és megfelelő tesztelésében. Az ilyen típusú áramkörök tervezése és optimalizálása a tantárgy ismereteire épít.

A projektben felhasznált szenzorok (DS18B20, BME280, RS-FXJT-N01, RS-FSJT-N01) és a mikrokontroller (ESP32) közti kapcsolatot próbapanelen történő lágyforrasztás biztosítja.

2.4. Rendszerfejlesztés és szoftverarchitektúrák

Egy projektben kulcsfontosságú szerepet tölt be a rendszer gondos megtervezése, hiszen ez határozza meg a működés alapjait. A fejezet bemutatja a fejlesztéshez szükséges lépéseket, a rendszertervezés és rendszermodellezéstől kezdve a szoftveralkotás és a vizualizáció megalkotásáig.

2.4.1. Rendszertervezés

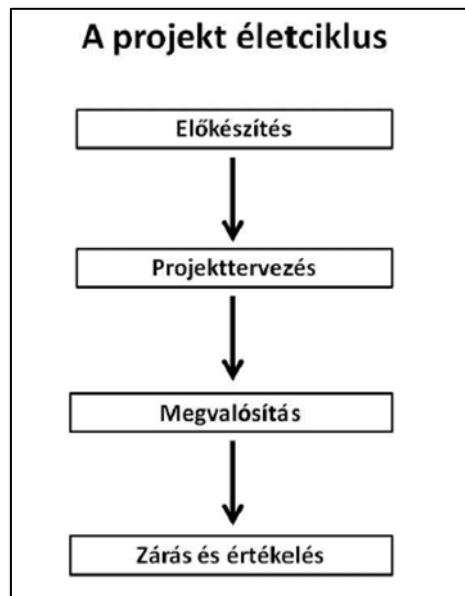
„A rendszertervezés egy komplex folyamat, amelynek a célja egy adott cél elérése érdekében működő információs rendszer tervezése, létrehozása, telepítése és a működési feltételeinek a biztosítása.” (Komló Csaba 2013 – Információs rendszerek tervezésének módszertana). A szerző szerint az 1990-es években ezek a szakmai ténykedések sok időt és egyedi megoldásokat igényeltek, azonban a kiadvány megjelenésekor már (2013), előtérbe kerültek a kevésbé időigényes és aprólékos eljárások. Az informatikai fejlődés exponenciális növekedése, a gazdasági kérdések és a folyamatos időhiány újabb és újabb rendszertervezési eljárásokat igényeltek. A tantárgy alapot adott a projekt gondolati megtervezésére, logikájának kialakítására, a megvalósítás sorrendjének meghatározására. Így meg kellett határozni:

- az alapproblémát,
- az alapprobléma megoldásának lehetőségeit,

- a hardvereszközök szükségességét,
- az adatbázis szerkezetét,
- a forráskódok felépítés,
- a GUI megtervezését.

A rendszertervezés tantárgy segített megérteni, hogy hogyan lehet a szoftvereket és a hardverkomponenseket összehangoltan működtetni. Segített még átlátni, hogy hogyan épül fel egy rendszer és hogy hogyan kapcsolódnak egymáshoz különböző komponensek. Az én projektem esetén a hardver komponensek különböző szenzorokból (vízhőmérséklet, léghőmérséklet, páratartalom, légnyomás szélesség, szélirány) és egy ESP32-es mikrovezérlőből, amíg a szoftver a háttérben futó operációs rendszer és a szenzorok működtetéséhez szükséges telepített programkódokból áll.

A projektmegvalósítás klasszikus szakaszai ebben a munkában is fellelhetők (vö. 6. ábra)



6. ábra: A projekt megvalósítás szakaszai

Forrás: Információs rendszerek tervezésének módszertana Komló Csaba 2013

2.4.2. Rendszermodellezés

„Hogyan garantálhatjuk, hogy a végső termék nem lesz hibás, megfelelő lesz a teljesítménye, és a rajta dolgozó mérnökökön kívül később más is képes lesz megérteni és módosítani a programot? Ezekre a kérdésre ad választ a Rendszermodellezés tárgy a komplex rendszerek modellezésének alapvető aspektusainak bemutatásával. A modellezés egy jól bevált technika a bonyolult rendszerek komplexitásának kezelésében.” (Bergmann Gábor, Darvas Dániel, Molnár Vince, Szárnyas Gábor, Tóth Tamás 2024 – Rendszermodellezés, Kritikus Rendszerek

Kutatócsoport) A modellezési technológiák régóta ismertek és folyamatos fejlődésen mennek keresztül, mivel a körülmények is folyamatosan változtak.

„A vízesés modell az első publikált rendszerfejlesztési modell, a hetvenes években Winston W. Royce publikálta „Managing the Development of Large Software Systems” címen.” (Komló Csaba 2013 – Információs rendszerek tervezésének módszertana) Eszerint tervezési folyamat egymás utáni lépések sorozata. Előnye a könnyű átláthatósága hátránya, hogy a menet közbeni eltéréseket nem, vagy rosszul kezeli.

A rendszerfejlesztési modellek következő meghatározó csoportja az UML. *„Az UML (Unified Modeling Language) szabványos, általános célú modellező nyelv, üzleti elemzők, rendszertervezők, szoftvermérnökök számára.”*

(Wikipédia – Unified Modeling Language – https://hu.wikipedia.org/wiki/Unified_Modeling_Language – Letöltés: 2025.03.21.) Ez a módszer különböző grafikus ábrákat használ a projektek leírására.

A legnagyobb szabadságot az agilis modell jelenti. *„Az agilis fejlesztési módszerek (nevezik adaptívnak is) egyik fontos jellemzője, hogy a résztvevők, amennyire lehetséges megpróbálnak alkalmazkodni a projekthez. Ezért fontos például, hogy a fejlesztők folyamatosan tanuljanak.”*

(Dr. Kusper Gábor és Dr. Radványi Tibor 2012 – Jegyzet a projekt labor című tárgyhöz). Napjainkban is meghatározó ez a modell, mely jellemző a dolgozatomban szereplő projektekre is. A működőképesség megoldása a fő szempont.

A rendszermodellezés alapvető eleme volt a projektem létrehozásának. A mikrovezérlőbe kötött különböző szenzorok érzékelők és kapcsolók egy rendszert alkotnak. A rendszer modellezése segít átlátni a komponensek közti kapcsolódást ezáltal segíti a hibakeresésben és az optimalizálásban is. A különböző komponensek közötti interakciók és kapcsolatok tisztázása alapvető a projekt sikeres megvalósításában.

2.4.3. Szoftverarchitektúrák

„Szoftver architektúrának nevezzük a szoftver fejlesztése során meghozott elsődleges tervezési döntések halmazát, meghatározzák a rendszer magas szintű felépítését és működését, az egyes alkotóelemek csatlakozási pontjait, megváltoztatásuk később jelentős újratervezését igényelné a szoftvernek” (Eötvös Loránd Tudományegyetem Informatikai Kar – Giachetta Roberto, 2017 – Komponens alapú szoftverfejlesztés, Szoftver architektúrák alapvetései)

A tantárgyon elsajátított ismeretek segítettek megérteni, hogy milyen fontos is egy szoftverarchitektúra gondos megtervezése. Esetemben az ESP32-es mikrovezérlő értékeket küld egy weboldalnak, mely az adatok pillanatnyi értékét kiírja, majd ezt bizonyos időközönként

továbbítja egy adatbázisnak. Például a tárgynak köszönhetően az is figyelembe lett véve a rendszer architektúrájának kialakításakor, hogy a későbbiekben a rendszeremet tovább szeretném fejleszteni, legyen lehetőségem plusz szenzorokat, érzékelőket kényelmesen elhelyezni a mikrovezérlőn.

2.4.4. Felhasználói interfészek és vizualizáció

A felhasználói interfészek és vizualizáció tanóra témája volt a megfelelő megjelenítés alapelveinek megismertetése, vagyis a felhasználó hogyan látja az adatokat. *„Maga az ergonómia kifejezés két görög szónak (az ergos=munka és a nomos=törvények) az ötvözete. ... Figyelembe véve, hogy a szoftver is termék, belátható, hogy az ergonómiának ez az aspektusa ugyancsak nagy jelentőségű az információs technológiák fejlesztésében, bevezetésében és alkalmazásában”* (Dr. Izsó Lajos, Dr. Antalovits Miklós, 2000 – Bevezetés az információ-ergonómiába, Emberi tényezők az információs technológiák bevezetésében és alkalmazásában)

Nem csak a funkcionalitásról szól egy program, applikáció vagy weblap, hanem esztétikai megjelenésről, felhasználóbarát vizualizációról, melynek fényében alakítottam ki a saját projektben használt weboldal kinézetét. Figyeltem arra is, hogy a weboldal képes legyen a valós idejű adat fogadásra és vizualizációjára is.

2.5. Adatbázis kezelés

„Az adatbázisok és az adatbázis-technológia nagy hatással vannak a számítógépek elterjedésére. Azt mondhatjuk, hogy az adatbázisok kritikus szerepet játszanak szinte minden olyan területen, ahol számítógépeket használunk, beleértve az üzletet, az elektronikus kereskedelmet, a mérnöki tudományokat, az orvostudományt, a jogot, az oktatást és a könyvtártudományt. Az adatbázis szó használata annyira elterjedt, hogy annak a definiálásával kell kezdenünk, mi is az az adatbázis. Az adatbázis egymással kapcsolatban álló adatok gyűjteménye” (A válogatott fejezetek Ramez Elmasri és Shamkant B. Navathe 2011 – Fejezetek az adatbázisrendszerek elméletéből – Fundamentals of Database Systems című könyve alapján készültek.)

2.5.1. Adatbázisok I

A tárgy célja az adatbázisok elméleti alapjainak bemutatása. Az első lépés megismerni az *„adatbázisokkal kapcsolatos alapfogalmakat, az adatok különböző strukturáltsági szintjeit és a fontosabb adatmodelleket”* (Dr. Balázs Péter, egyetemi docens Dr. Németh Gábor, adjunktus; 2019 – Adatbázisok a tervezéstől az alkalmazásfejlesztésig – Szegedi Tudományegyetem Természettudományi és Informatikai Kar Informatikai Intézet)

Adatbázisok I órán az adatbázis kezelés és létrehozásával ismerkedtünk meg elsőnek, majd későbbiek folyamán megtanultuk, a helyes adatszerkezet kialakítását. Ezt eleinte főleg Access-ben hajtottuk végre, de később adaptáltuk ezt az alaptudást MySQL és PHP környezetben is. A tantárgy lényege az adattábla átláthatósággá tételének a logikája és az alap működések és az egyszerűbb függvények megismertetése.

2.5.2. Adatbázisok II

Itt már részleteiben foglalkoztunk az adatbáziskezelési módszerekkel webes környezetben, kihasználva a MySQL *„A MySQL az egyik legelterjedtebb adatbázis-kezelő, aminek egyik oka lehet, hogy a teljesen nyílt forráskódú LAMP (Linux–Apache–MySQL–PHP) összeállítás részeként költséghatékony és egyszerűen beállítható megoldást ad dinamikus webhelyek szolgáltatására.”* (Wikipédia – MySQL – <https://hu.wikipedia.org/wiki/MySQL> – Letöltés: 2025.04.05.) és a PHP *„A PHP egy általános szerveroldali szkriptnyelv dinamikus weblapok készítésére....A kódok végezhetnek adatbázis-lekérdezéseket, létrehozhatnak képeket, fájlokat olvashatnak és írhatnak, kapcsolatot létesíthetnek távoli kiszolgálókkal.”* (Wikipédia – PHP – <https://hu.wikipedia.org/wiki/PHP> – Letöltés: 2025.03.12.) nyújtotta lehetőségeket. A fő cél volt egy webes adatbázis kialakítása, űrlapon történő adatbejegyzés és lekérdezések készítése.

Összességében az adatbázis rendszerek órákon elsajátított tudás nagy hatással volt az eszközökben lévő potenciálok kiaknázásában. Azzal, hogy tisztában voltam az adatbázisrendszerek alapvető létrehozásának feltételeivel, így sikeresen ki tudtam alakítani az adatbázist, mely képes adatokat fogadni. Mindez jelentős nehézséggel jöhetett volna csak létre, ha az alapokat és a gondolkodásmódot nem tudom a birtokomban venni az adatbázisokkal foglalkozó órákon.

2.6. Hálózatok és operációs rendszerek

Már két számítógép is hálózatot alkot és képes szükség esetén kommunikálni egymással. Azonban a megfelelő kapcsolat kialakítása bonyolult folyamat. Számtalan lehetőség kínálkozik a megoldásra, melyet már a tervezés folyamán át kell gondolni. Nem elég a hálózati adatközlést megoldani, a hálózatba kapcsolt számítógépek egyedi működését, operációs rendszereit is figyelembe kell venni, illetve ennek a célnak kell alávetni.

2.6.1. Hálózatok és számítógép architektúrák

Mi is a számítógépes hálózat: *„A számítógép-hálózat olyan speciális rendszer, amely a számítógépek egymás közötti kommunikációját biztosítja. A számítógépek az egymással való információcseréhez digitális összeköttetéseken keresztül közös kommunikációs protokollokat*

használnak. Ezek a kapcsolódások különböző távközlési technológiákból épülnek fel, amelyek fizikailag lehetnek vezetékes, azon belül réz vagy optikai kábeles, illetve vezetékek nélküli, különféle rádiófrekvenciás megoldások.”

(Wikipédia – Számítógép-hálózat – <http://hu.wikipedia.org/wiki/Sz%C3%A1m%C3%ADt%C3%B3g%C3%A9p-h%C3%A1ll%C3%B3zat> – Letöltés: 2025.03.22.)

Chat GPT szerint a hálózatok előnye:

„A számítógép-hálózatok számos előnyt kínálnak mind a magánfelhasználók, mind a vállalatok számára. Néhány fő előny:

- *Erőforrás-megosztás*
- *Hatékony adatmegosztás és kommunikáció*
- *Központi adatkezelés és biztonság*
- *Távoli elérés és mobilitás*
- *Költségcsökkentés és hatékonyság*
- *Könnyű skálázhatóság”* (OpenAI – Számítógép hálózatok– www.chatgpt.com –

Letöltés: 2025.02.22.)

A hálózat kiemelt fontossággal bír a projektben, ugyanis a kész rendszerem és eszközeim saját hálózatot alkotnak, valamint csatlakoznak az internethez. A műszaki kialakítások miatt többféle hálózati protokoll szerint csatlakoznak egymáshoz. Más a két mikrovezérlő között, és más a nyílt internet felé. Ismereteim hasznomra váltak a kábel nélküli hálózati csatlakoztatás és a folyamat közben hibák keresése és megoldása során.

2.6.2. Operációs rendszerek

„Az ISO nemzetközi szabványosítási szervezet definíciója szerint az operációs rendszer olyan programrendszer, amely a számítógépes rendszerben a programok végrehajtását vezérli: így például ütemezi a programok végrehajtását, elosztja az erőforrásokat, biztosítja a felhasználó és a számítógépes rendszer közötti kommunikációt.”

(Wikipédia – Operációs rendszer – https://hu.wikipedia.org/wiki/Oper%C3%A1ci%C3%B3s_rendszer#cite_note-1, – Letöltés: 2025.03.19.)

A tantárgy az Operációs rendszerek felépítését, részeit, folyamatait mutatta be. PowerShell és a Bash programozás világossá tette, hogy a számítógépműködésnek léteznek mélyebb dimenziói is. Ezek ismeretével a könnyebben és hatékonyabban tudunk folyamatokat kezelni. Ezek az ismeretek segítik egy-egy hardvereszköz működésének a megértését.

„Használható számítógép működtető rendszer nélkül? Ezt a használati módot nevezzük a hardveren való direkt futtatásnak. A válasz tulajdonképpen igen, de csak a kis bitszélességű mikrokontrollereknél szokásos ma már.” (Dr. Vadász Dénes, 2002 – Operációs rendszerek.)

Az ESP32-es mikrovezérlő, amit a projektem során felhasználtam, ugyan rendelkezik egy beágyazott freeRTOS nevű operációs rendszerrel, de nem képes teljes értékű asztali operációs rendszer alatt futni. Valós idejű operációs rendszerrel rendelkezik, amely képes kezelni a különböző alapfeladatokat mely szükséges az eszköz működéséhez.

Operációs rendszer feladatai közé tartozik a(z):

- Memória kezelés
- Feladatkezelés és időzítés
- Eszköz és driver kezelés

A projekt során az operációs rendszer közvetlen elérésére nem volt szükség.

2.7. IT-biztonság és fejlesztés

Az informatikai rendszerek fejlődésével párhuzamosan az IT-biztonság szerepe is egyre jelentősebbé válik. A biztonságos fejlesztés nem csupán technikai kérdés, hanem komplex feladat, amely kiterjed a korszerű információs és kommunikációs technológiák alkalmazására, a projektek megfelelő menedzselésére és a mesterséges intelligencia lehetőségeinek kihasználására is. Ugyanakkor a tudás hatékony kezelése és megosztása szintén elengedhetetlen a biztonságos informatikai környezet kialakításához. Ebben a fejezetben ezeknek a területeknek a szerepét és jelentőségét mutatjuk be a fejlesztési folyamatok tükrében.

2.7.1. Innovatív információs és kommunikációs technológiák az IT-biztonság kapcsán

A digitális kommunikációs technikák robbanásszerűen felgyorsították az információcserét. Hatalmas digitális adatmennyiség keletkezik percenként a világban, melyek biztonságos közvetítését a végpontok között meg kell oldani az adatvédelmi incidensek nélkül.

Az egy évtizede, a Sony Corporation elleni nagyszabású hackertámadás *„rávilágított arra, hogy a vállalatok mennyire kiszolgáltatottak a kiberbűnözéssel szemben.”* (Yossi Sheffi, 2019 – A rugalmasság ereje, Pallas Athéné Könyvkiadó, Budapest)

„Az internet több, mint világkönyvtár: a világ legnagyobb archívuma és adatbázisa...” (Szabó Katalin, 2002 – Kommunikáció felsőfokon, Kossuth Kiadó, Budapest) Ami egyszer ide felkerül, az már nem törölhető ki, tehát fontos, hogy az adatainkat itt is biztonságban tudhassuk. Ezt például a blokklánc technológia is biztosítja.

A szakdolgozatom elkészítéséhez is használtam az innovatív információs és kommunikációs technológiák egy részét. Például a felhő alapú tárolást, ami felveti a távoli elérés lehetőségét a világhálón keresztül, a Google-drive, valamint egy szerver felhőalapú tárolását is. A jövő lehetőségeinek feltérképezésére pedig kérdőívet állítottam össze, hogy jobban megismerjem a célközönség ismereteit a témámmal kapcsolatban, véleményét, szokásait és a kutatásomhoz kapcsolódó témákhoz való hozzáállását. Ennek során kiemelt figyelmet szántam a GDPR és az adatvédelmi törvény betartására. Csak 18 éven felülieket kérdeztem, a személyes adatokat pedig mellőztem a kérdéseim során, viszont használtam a Chat-GPT-t is.

Az IT-biztonság az egyik legfontosabb területe a digitális kor időszakában. A mesterséges intelligencia is megkerülhetetlen az információs és kommunikációs technológiában. Ahogy azt a mentorom is hangsúlyozta:

„A mesterséges intelligencia fejlesztések kapcsán tehát etikai szinten elvárható és így praktikus is elvárt, hogy a jog és/vagy a tudomány területén induljon meg végre a rendteremtés”.

(Pitlik László: Stratégiai és operatív irracionalitások a (pályázat)értékelésben – https://matarka.hu/cikk_list.php?fusz=191502 – Letöltés: 2025. 03. 28.)

2.7.2. IT-biztonsági fejlesztések minőség- és projektmenedzsmentje

A szakdolgozat is egy IT fejlesztés, ami valójában egy projekt is. Ezért a szakdolgozati célok kitűzése után időtervet készítettem és a rendszeres napi *„projektidőt be is kellett tartani”* (Peter Hobbs, 2000 – Projektmenedzsment – Scholar Kiadó). Ez azért is fontos, mert az esetleges nehézségek ellenére is el kell készülnünk vele időben. Tehát mi vagyunk a projektmenedzserei a szakdolgozatunknak. Szükségünk van egy szigorúan ellenőrző személyre, aki nem más, mint a mentorunk. A minőségi előírásokat a KJE és a tanszék határozta meg.

Ez tulajdonképpen a minőségmenedzsmentben is használt PDCA ciklus *„Plan-Do-Check-Act”*, Tervezés-Készítés-Ellenőrzés-Cselekvés. (Kristály Mátyás, 2001 – Minőségirányítási alapismeretek Okker Kiadó, Budapest)

Az első feladat a tervezés után a szükséges adatok megszerzése. Többféle adatgyűjtési és kutatási módszert használtam. Az így nyert adatok feldolgozása során a *„tudáspiramis lépcsőfokait magam mögött hagyva alakult át az elvont adat információvá, majd az információ elemzése, rendszerezése során megszereztem azt a tudást, amit alkalmazva a piramis tetején találtam magam, ahol a döntés meghozatala történt”.* (Amaël Cattaruzza, 2019 – A digitális adatok geopolitikája, Pallas Athéné Könyvkiadó KFT., Budapest)

„Erőforrásaink mozgósításával” tehát, időt, energiát és költséget tudunk megtakarítani a projektünk, jelen esetben szakdolgozatunk számára úgy, hogy közben etikus magatartást tanúsítunk

és figyelembe vesszük az adat- és információbiztonság alapelveit. (Zahary Wong, 2019 – Nyolc nélkülözhetetlen humán készség a projektmenedzsmentben, Pallas Athéné Könyvkiadó KFT., Budapest)

2.7.3. Mesterséges intelligenciák az IT biztonság területén

„The recent advances in machine learning are breath-taking and important. However, it is critical to understand that we are very far from artificial general intelligence that is, the kind of AI than spans the full range of human intelligence.” A gépi tanulás terén elért közelmúltbeli fejlemények lélegzetelállítóak. Fontos azonban megérteni, hogy nagyon messze vagyunk az általános mesterséges intelligenciától, vagyis attól a fajta mesterséges intelligenciától, amely az emberi intelligencia teljes skáláját lefedi. mondta Erik Brynjolfsson, a Stanford Digital Economy Lab Igazgatója. (Erik Brynjolfsson – Artificial Intelligence and the Future of Work Congressional Testimony – <https://www.congress.gov/116/meeting/house/109981/witnesses/HHRG-116-SY15-Wstate-BrynjolfssonE-20190924.pdf> – Letöltés: 2025.03.28.)

Maga a mesterséges intelligencia nem része a dolgozatnak, de használata kikerülhetetlen. Az oktatáson kívül a mindennapi élet során is egyre gyakrabban használjuk. Sokan úgy gondolják, hogy ez valami nagyon új, nagyon modern dolog, pedig már 1950-es évektől létezik.

Mi is ez valójában? Byron Reese kétféle mesterséges intelligenciát különböztet meg, a „*keskeny MI*” -t, (Byron Reese, 2020 – A negyedik kor – Typotex Kiadó, Budapest) ami ma is létezik és bizonyos problémákat meg tud oldani és az „*általános MI (AGI)*”-t (Byron Reese, 2020 – A negyedik kor – Typotex Kiadó, Budapest), ami elképesztő dolgokra képes.

Mindkettőben hatalmas potenciál található mind az IT biztonság, mind a kiberfenyegetettség területén. Nem véletlen, hogy az európai adatvédelmi biztos szigorú törvényi szabályozás mellett érvel, míg az „*európai MI-ipar*” (Zódi Zsolt 2024 – A mérnöki és a jogi gondolkodásmód ellentéte a mesterséges intelligenciáról szóló jogszabályban, In: Rab Árpád, Majó-Petri Zoltán, Koltay András, 2024: Felelős AI, Gondolat Kiadó, Budapest) ennek felülvizsgálatát, enyhítését kéri.

Martin Ford könyvében (Martin Ford: Jövünk a robotok korában, HVG Könyvek, Budapest, 2022) a szuperintelligencia embert túlszárnyaló veszélyeiről ír. Például az „*eltűnő állások, amikor is a „mesterséges intelligencia és a robotika végső soron a kiszámítható rutinmunkák jelentős részét fel fogja váltani”*. Említi a „*Kínában elterjedt intenzív megfigyelés*” rendszerét, ami emberi jogi kérdéseket is felvet. (Martin Ford, 2022 – Jövünk a robotok korában – HVG Könyvek, Budapest)

A szakdolgozat során többször használtam én is, amikor elakadtam egy-egy témánál, ötletek nyerésére. Azonban más forrásokból is utánanéztem a témának.

A jövőbeni elképzeléseknek is a része, hiszen a projekt beérkezett adatainak feldolgozásához jelentős segítséget tud nyújtani, hiszen a gyűjtött adatok nyilvánosak, nem személyes jellegűek.

2.7.4. Tudásmenedzsment az IT biztonság területén

Az adat, minden tudományos mű alapja. A 2001-ben megjelent Informatikai Tudástár szerint, nem más, mint a „*kódolt információ*” (Horváth László, Pirkó József, 2001 – Informatikai Tudástár Kiskapu Kiadó, Budapest). Napjainkban azonban, amikor a digitális kor rohamos fejlődésének vagyunk tanúi, abban egyet érthetünk, hogy „*legtöbbször digitális adatot*” értünk rajta. (Amaël Cattaruzza, 2019 – A digitális adatok geopolitikája, Pallas Athéné Könyvkiadó KFT., Budapest)

A digitális információ növekedésével felértékelődött az IT biztonság szerepe is, valamint az ehhez kapcsolódó tudásmenedzsment. A szakdolgozat tudásmenedzsmentje kapcsán felmerül a titkosítás kérdése is.

A „*titkosítás és a kriptográfiai kulcsok kezelése*” mind a szakdolgozat alkotásakor, mind a hozzá kapcsolódó kérdőív adatainak megőrzése miatt fontos. Vigyázni kell, „*hogy illetéktelenek ne férhessenek hozzá a titkos adatokhoz*”. (Magyar Gábor, Nemeslaki András, Syi, 2021 – A digitális transzformáció kérdései, Gondolat Kiadó, Budapest).

Az adatvédelmi biztos és a „*Nemzeti Adatvédelmi és Információszabadság Hatóság (NAIH), mint azt tudjuk, egy helytelen adatkezelés esetén korlátozhatja, vagy akár meg is tilthatja az adatkezelést*”. (Szabó Endre Győző, 2022 – A védelmi lépcső elmélete, Ludovika Egyetemi Kiadó, Budapest.) Szakdolgozatom készítése során többször is használtam a ChatGPT szolgáltatásait, melyeket kritikusan alkalmaztam több területen. Ennek jelentőségét megerősítette mentorom egyik írása: „*a ChatGPT is egy fajta újszerű kiegészítésként hat a tudásmenedzsment eddigi szakterületére.*” (Pitlik László – ChatGPT szerepe a tudásmenedzsmentben, MTA / Gazdálkodástudományi Bizottság / Tudásmenedzsment Munkabizottsága, „A tudásmenedzsment új utakon - a MI és az Upgrading összefüggésrendszereiben” c. XXI.)

2.8. Programozás

A programozás a modern technológia egyik alapvető eleme. Ez azonban nemcsak program sorok írásából áll, hanem lényegesen komplexebb összetettebb feladat. Magába foglalja a feladatok megértését, azok szelektálását, kódolható részekre történő szétbontását. Fontos megismerni az alapvető elemi részeit, eszközeit, az alapvető programozási tételeket, vezérlési szerkezeteket,

melyekkel minden feladatmegoldásnál találkozhatunk. Nem szabad megfeledkezni az objektorientáltságról, mely napjainkban alapvető ismeretté vált. Egy magasszintű programnyelven keresztül mindezek megismerhetők és elsajátíthatók.

„Valamilyen programot megírni nem nehéz, de jó programot írni bizony nem egyszerű feladat. Nehéz lenne megmondani, hogy melyik az aranyút, amely a jó programkészítés iskolájának tekinthető, de az bizonyosan állítható, hogy jó és hatékony programot csakis megfelelő fejlesztési környezetben tudunk készíteni.” (Illés Zoltán, 2005 – Programozás C# nyelven; Jedlik Oktatási Stúdió Budapest)

2.8.1. Programozási alapelvek és módszertanok

A tantárgy platformfüggetlen módon mutatta be a programozás alapjait, *„az alapvető programozói háttérismereteket és alapfogalmakat mind a hardver, mind a szoftver oldaláról. Bemutatja az algoritmusok és programok alapvető építőelemeit, valamint a strukturált programok készítésének alapvető szabályait”* (Simon Gyula, 2011 – A programozás alapjai). Az elméleti programozás alapja a kód olvasása és megfelelő rendezés, hiszen egy programozónál fontos készségének kell lennie a más programjának olvasása, feljavítása, átírása és értelmezése. Ezeket az alap készségeket többek között programozási alapelvek órán sajátítottam el. Ezen alapok nélkül a szenzorjaim, Master/Slave eszközeim kapcsolatának felépítése és programozása, a kábel nélküli kapcsolat kialakítása, a megfelelő helyre való továbbítása nem valósulhatott volna meg. A vezérlési szerkezetek, a változók ismerete jelen projekt kevésbé összetett kódjainak megírásánál is elengedhetetlen

2.8.2. Adatszerkezetek és algoritmusok

„Az adatstruktúrák és algoritmusok elengedhetetlen alapjai a szoftvertervezés, szoftverfejlesztés folyamatának, komplex informatikai programoknak, szoftvereknek.” (Adonyi Róbert, 2011 – Adatstruktúrák és algoritmusok – Typotex kiadó)

Az algoritmusok és a programok megtervezéséhez ismerni kell az alapvető eljárásokat, tételeket (vezérlési szerkezetek, programozási tételek), melyek szinte bármely programozási probléma megoldásában előfordulhatnak és jelentősen leegyszerűsíthetik azt. Ezek alapos ismeretével gyorsan átgondolható és adaptálható egy-egy eljárás.

„A megfelelő programozási ismeretekhez elengedhetetlen, hogy tudjuk és ismerjük

- *hogyan lehet az információt a számítógép memóriájában tárolni,*
- *milyen adatszerkezetekbe lehet az adatot szervezni,*
- *mik az előnyei és hátrányai a kiválasztott adatszerkezetnek,*

- *hogyan lehet algoritmusokat felépíteni és az algoritmusokkal a tárolt adatot módosítani,*
- *mik a kiválasztott adatszerkezeteknek és a hozzájuk kapcsolódó algoritmusoknak a számítási igénye.*” (Adonyi Róbert, 2011 – Adatstruktúrák és algoritmusok – Typotex kiadó)

2.8.3. Programozás I

A Programozás I órán már a gyakorlati programozás volt hangsúlyban, ahol egy adott nyelv tanulására koncentráltunk, ami az én esetemben a C# volt. *„A C# tisztán objektumorientált, típusbiztos, általános felhasználású nyelv. A tervezésénél a lehető legnagyobb produktivitás elérését tartották szem előtt. A nyelv elméletileg platform független (létezik Linux és Mac fordító is), de napjainkban a legnagyobb hatékonyságot a Microsoft implementációja biztosítja.*” (Reiter István, 2012 – C# programozás lépésről lépésre – Jedlik Oktatási Stúdió) A tantárgy témája a nyelvspecifikus módon a változók és deklarációk, valamint az alapvető elágazások, ciklusok, operátorok, változók, utasításblokkok, (if, else, else if, while, do while stb...), voltak. Kezdetben alacsonyabb szintű matematikai egyenletek és logikai feladatokkal foglalkoztunk melyek segítségével sajátítottuk el a tömbök, listák és a mátrix alapelveit. A szakdolgozati projektben a forráskódok elkészítésénél, módosításánál jelen vannak ezek az elemek (vö. 3.4 alfejezet).

2.8.4. Programozás II

A Programozás II tantárgy előtérbe helyezte az objektum orientáltság ismereteit. *„Az objektumorientált vagy objektumelvű programozás (angolul object-oriented programming, röviden OOP) az objektumok fogalmán alapuló programozási paradigma. Az objektumok egységbe foglalják az adatokat és a hozzájuk tartozó műveleteket. Az adatokat ismerik „mezők”, „attribútumok” vagy „tulajdonságok” néven, a műveleteket „metódusokként” szokták emlegetni. Az objektum által tartalmazott adatokon általában az objektum metódusai végeznek műveletet.*” (Wikipédia –Objektumorientált programozás – http://hu.wikipedia.org/wiki/Objektumorient%C3%A1lt_programoz%C3%A1s – Letöltés: 2025.03.08.) Az osztályokkal, példányosítással, mezőkkel védelmi szintekkel, konstruktorokkal, destruktorkkal foglalkozott. Az eseményvezérelt programozás komplex architektúrája számos hasznos információt tanított meg a számomra. A szakdolgozati projekt nem igényelt kiterjed objektumkezelési programozást.

2.8.5. Programozás III

A Programozás II tantárgy alapján elsajátított tudást egészítettük ki a grafikus felületek programozásával. Ez a „*Windows Presentation Foundation (WPF)*” (Kovácsnai Gergely, Biró Csaba 2013, – .NET-es programozási technológiák – Eszterházy Károly Főiskola Matematikai és Informatikai Intézet) Ezen tudásmenetet követően bepillantást nyertünk más, így a Python programozási nyelvbe. „*A Python általános célú, nagyon magas szintű programozási nyelv, melyet Guido van Rossum holland programozó kezdett el fejleszteni 1989 végén, majd hozott nyilvánosságra 1991-ben. A nyelv tervezési filozófiája az olvashatóságot és a programozói munka megkönnyítését helyezi előtérbe a futási sebességgel szemben*”. (Wikipédia – Python (programozási nyelv) – [https://hu.wikipedia.org/wiki/Python_\(programoz%C3%A1si_nyelv\)](https://hu.wikipedia.org/wiki/Python_(programoz%C3%A1si_nyelv)) Letöltés: 2025.03.08.) A Python nyelv a projekt jelen állapotáig nem került alkalmazásra.

A Programozás I-III tanult ismereteimet külön jellemezni véleményem szerint nem lehet. A fő nyelvem a C# volt, ezáltal a C nyelvek értelmezése egyszerűbbé vált számomra, ami egy jelentős segítség volt a rendszerem programkódjának a készítése során. Az Arduino IDE-ben alapjaiba véve C és C++ alapú programnyelvet lehet használni, ugyanakkor rendelkezik néhány egyszerűsített kiegészítéssel a mikrokontroller programozásának megkönnyítése ellenére nem minősül külön programnyelvnek. A programozások órán elsajátított tudás minden szintjét felhasználtam az algoritmikus gondolkodásom fejlesztésében, ami talán a projektem sikerességéhez a legfontosabb és egy elengedhetetlen szükséglet volt.

2.9. Szoftverüzemeltetés és tesztelés

A verziókezelés és a karbantartás fontos perspektívák, nem csak hardveresen, hanem szoftver szinten is, hibajavítás, rendszer frissítés és hibakeresés és még sok más. Ezen lépéseket a Szoftverüzemeltetés óra mutatta meg. Esetemben az eszköz és weboldal folyamatos figyelmet igényel, hogy biztosítani tudjam a hatékony működést és adatszolgáltatást.

2.9.1. Szoftverüzemeltetés

Ennek a rendszere az ITIL. „*Az ITIL (A ITIL® az Axelos Limited bejegyzett védjegye. Minden jog fenntartva.) az Information Technology Infrastructure Library (ITIL) rövidítése. Az IT-szolgáltatások irányítására és támogatására vonatkozó legjobb gyakorlatok keretrendszerét határozza meg. Az ITIL biztosítja, hogy a szolgáltatások és a folyamatok folyamatosan javuljanak, ezáltal növelve az értékkeremtést az egész vállalatban belül*” (OTRS – Mit jelent az ITIL? – <https://otrs.com/hu/blog/itsm/mi-az-til/> – Letöltés: 2025.03.08.) Ezen lépéseket a Szoftverüzemeltetés óra mutatta meg. Két fő területe van: „*A rövidtávú feladatokat a szolgáltatás*

támogatása, a hosszabb távú, taktikai, stratégiai feladatokon keresztül a szolgáltatások tervezését és nyújtását pedig a szolgáltatás biztosítása csoport tartalmazza” (Sarkadi Nagy István, 2002 – ITIL - az informatikaszolgáltatás módszertana – http://www.itsmf.hu/documents/itil2modszertan_osszefoglalo_v3.1.pdf – Letöltés: 2025.04.05.).

Esetemben az eszköz és weboldal folyamatos figyelmet igényel, hogy biztosítani tudjam a hatékony működést és adatszolgáltatást. Elsősorban az üzemeltetési feladatok megszervezése, akkor lesz releváns, ha a projekt üzletszerű megvalósulása biztosított lesz.

2.9.2. Szoftvertesztelés

A szoftvertesztelés különböző módszertanokkal és technikákkal ellenőrzi a rendszerek megbízhatóságát. „A tesztelés nemcsak a hibák detektálására korlátozódik, hanem nagyban hozzájárul a megfelelő működéshez és segíti a funkcionális és nem funkcionális jellemzők változásainak követését is.” (Horváth László, 2007 – Szoftvertesztelés a gyakorlatban). Az órán a programteszteléssel kapcsolatos logikai feladatokat oldottunk meg amely hozzájárult ahhoz, hogy különböző nézőpontokból közelítsünk meg és elemezzünk egy adott problémát. Nézőpontom szerint az óra legfőbb célja a logikai és racionális programozói gondolkodás fejlesztése volt. A szakdolgozatom írása során erre komoly szükség volt.

2.10. A horgászat biológiai, ökológiai törvényszerűségei

Az ökológia napjaink igen fontos, kiemelt jelentőségű témája, a bolygónk megőrzésének alapja. „Az ökológia a környezetvédelem igen fontos, és a természetvédelem kétségkívül legfontosabb alaptudománya.” (Horváth Balázs, Pestiné Rácz Éva Veronika – Ökológia. https://dtk.tankonyvtar.hu/xmlui/bitstream/handle/123456789/8106/0021_Okologia.pdf?sequence=1&isAllowed=y – Letöltés: 2025. 04.04.)

„Az ökológiai gazdálkodás az IFOAM (Ökogazdálkodók Nemzetközi Szövetsége) meghatározása szerint a talajok és az élővilág épségének, valamint az emberek egészségének megőrzését célzó élelmiszertermelési rendszer. Az ökológiai gazdálkodás a helyi erőforrások használatán és a természetes körfolyamatokon alapul, így kerül a kémiai inputanyagok (szintetikus növényvédő szerek, műtrágyák) és a géntechnológiával módosított szervezetek és származékaik alkalmazását” (Drexler Dóra, Varga Korinna, Allacherné Szépkuthy Katalin – Az ökológiai gazdálkodás 2027-ig tartó cselekvési terve, agrárium7, 2022/07/03 – <https://agrarium7.hu/cikkek/2190-az-okologiai-gazdalkodas-2027-ig-tarto-cselekvesi-terve> – Letöltés: 2025.04.05.)

Tehát a halak etetéséhez a helyi ökológiai tápanyagokat kell felhasználni.

Az ökológiai haltenyésztés abból a szempontból is hangsúlyos, mert az egészséges táplálkozás részeként a világban fokozatosan növekszik a halfogyasztás mértéke. Ezért is folyamatos a törvényi szabályozás változása is.

Az Európa Tanács ökológiai termelésre vonatkozó alapelvei között szerepel többek között:

„Az ökológiai állattenyésztésnek tiszteletben kell tartania a magas szintű állatjólléti szabványokat és ki kell elégítenie az állatok fajspecifikus viselkedési szükségleteit, ... Ezenfelül a fajták kiválasztásakor figyelembe kell venni a helyi viszonyokhoz való alkalmazkodási képességüket.”

(Az Európa Tanács 834/2007/EK rendelete – <https://eur-lex.europa.eu/legal-content/HU/TXT/HTML/?uri=CELEX:32007R0834> – Letöltés: 2025. 04. 05.)

Eszerint főként őshonos halakat, hazánkban többek között a pontyféléket ajánlott tenyészteni, mert azok széles tűrésűek, jól alkalmazkodnak a helyi éghajlati viszonyokhoz.

„Magyarország halfaunája folyamatosan változott, illetve változik ma is.” 2018-ban 89 faj alkotta. A vizeink *„halközösségei természetvédelmi szempontból nemcsak az emberi beavatkozások miatt abiotikus oldalról, hanem az invazív fajok okán biotikus oldalról is kiemelten sérülékenyek.”* (Dr Halasi-Kovács Béla, 2019 – A magyarországi vízfolyások halközösségeinek ökológiai szempontú elemzése – Fazekas Nyomda, Szarvas)

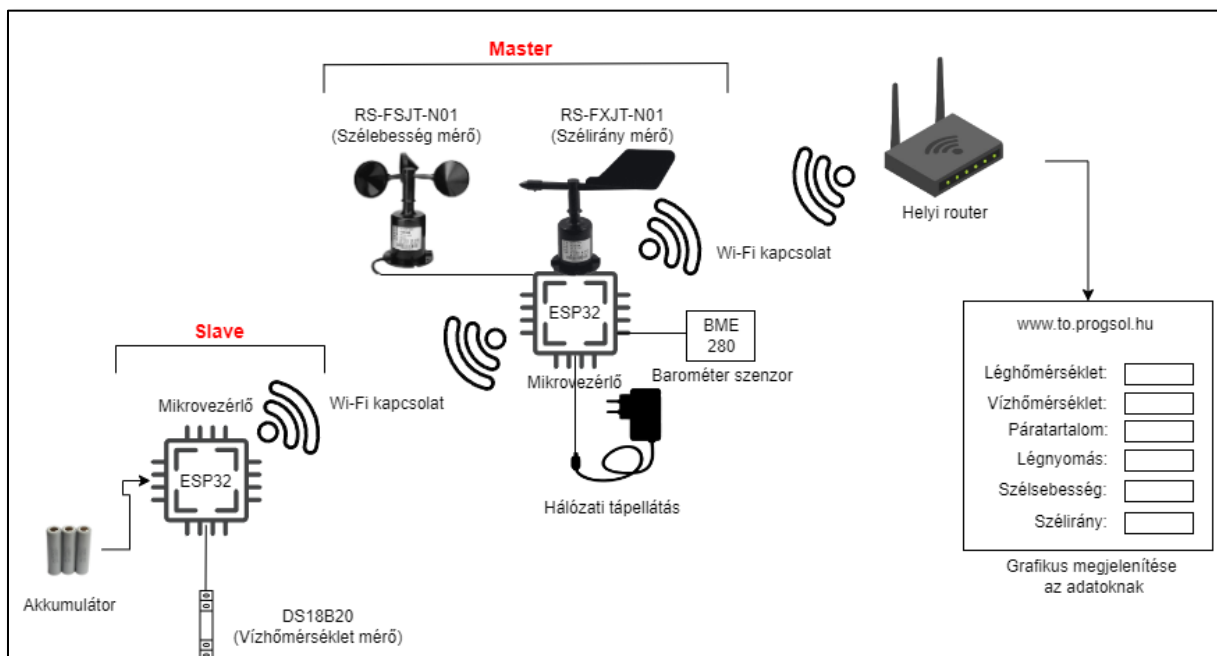
Ezért az idegenhonos halak, például a gyorsan szaporodó törpeharcsa visszaengedése a horgászat során tilos.

3. A projekt bemutatása

Ebben a fejezetben részletesen ismertetem a felhasznált szenzorokat és eszközöket. Az eszköz és szoftver komponensek kapcsolatát és szerkezetét, valamint a rendszer működési elveit. Továbbá szó lesz az adattovábbításról, webserveren történő tárolásról, webes felületen, Graphical User Interfacen keresztüli megjelenítésről is. Bemutásra kerülnek a fejlesztői környezetek és a forráskódok, majd végül a fejlesztési ötletek és tervek kerülnek részletesebb kiértékelésre.

3.1. Szerkezeti modell és fő részei

Az alábbi 7. ábra mutatja a hardvereszközök szerkezeti felépítését és az egymáshoz való kapcsolódását, a 8. ábra pedig a megvalósult projekt asztali változatát.



7. ábra: Szerkezeti modell

Forrás: Saját szerkesztés - Adobe Photoshop CS6 Extended

Slave: A képen ez jelképezi az alárendelt eszközt, szenzorát és tápellátását. A Slave feladata a mérés és adatküldés a Master eszköznek. Az eszköz utasításokat hajt végre, de kommunikációt nem kezdeményezhet, innen ered az elnevezése is.

Master: A képen ez jelképezi a főlérendelt mikrovezérlőt és annak perifériáit. A Master eszköz feladata az adatfogadás a Slave eszköztől a saját szenzorainak a kiértékelése és adattovábbítás a webserverver felé. Az eszköz begyűjti a Slave adatait, kezdeményezi a kommunikációt, utasításokat oszt a Slave eszköznek, innen ered a Master elnevezése is.



8. ábra: Összeállított rendszer működő asztali verziója
Forrás: Saját kép – Telefon fotó

3.1.1. Első rész: Slave

Vízhőmérséklet mérés, szenzor érzékelés mikrovezérlő segítségével („Slave” eszköz). Ebben a modulban ESP32-WROOM-32U mikrovezérlőhöz kapcsoltam egy vízálló DS18B20 típusú hőmérsékletérzékelő szenzort, mely Wi-fi kapcsolaton keresztül küldi az adatokat a Master eszköznek. Tápellátását egy 3.7 voltos 18650-ös tölthető akkumulátor biztosítja, így lehetőség van az eszköz hálózati áramforrástól távol, akár egy bójában való elhelyezésére is, a kommunikációs távolság figyelembevételével.

3.1.2. Második rész: Master

Léghőmérséklet, páratartalom, légnyomás, szélesség és szélirány mérés szenzorok és mikrovezérlő segítségével („Master” eszköz). Ebben a modulban szintén egy ESP32-WROOM-32U mikrovezérlőhöz kapcsolódik egy BME280 típusú, léghőmérséklet, légnyomás, páratartalom mérésre alkalmas szenzor (barométer szenzor), egy RS-FXJT-N01 szélirány mérő, valamint egy RS-FSJT-N01 szélesség mérő szenzor. A mikrovezérlő táplását egy 5 Volt kimeneti feszültséggel rendelkező tápegység biztosítja míg a szélirány és sebesség mérésre alkalmas szenzorok tápellátása pedig egy 12V-os tápegység.

3.1.3. Harmadik rész: Adattovábbítás

A mért adatok webszerverre és adatbázisba történő eljuttatása, valamint publikálása külső felhasználók részére. Ez a modul helyi routeren keresztül valósul meg, mely ezen keresztül kapcsolódik a nyílt internet hálózatra, illetve megfelelő a webszerverre. A webszerverről az adatok a napjainkban a szokásos módszer szerint HTML és PHP nyelv használatával kerül megjelenítésre a felhasználók számára.

3.2. Alkalmazott hardvereszközök szerkezeti és működési áttekintése

A fejezetben a használt hardvereszközök, azok szerkezeti felépítése és működési elvei kerülnek bemutatásra. Részletesen megismerhetjük a ESP32-WROOM-32U mikrovezérlőt, DS18B20 hőmérséklet szenzort, BME280 légnyomás- és páratartalom szenzort, szélirány- és szélesség mérőket, valamint az eszközre felszerelt összes kiegészítőt.

3.2.1. ESP32-WROOM-32U mikrovezérlő



9. ábra: ESP32-WROOM-32U

Forrás: <https://modulshop.hu/esp32-wroom-cp2102-nodemcu-38-pinantenna-csatlakozo-1626?keyword=antenna>

A mikrovezérlő (vö. 9. ábra) egy apró számítógép egyetlen chipen, amely tartalmaz egy központi feldolgozó egységet, azaz processzort (CPU), memóriát (RAM, ROM/Flash) és perifériákat. Ezeket különböző rendszerekbe beágyazva használjuk. Ugyanakkor nem csak tervezésre és egyéni felhasználásra alkalmas eszköz, sokkal többek annál. Napjaink nagy vállalatai előszeretettel alkalmazzák ezen eszközöket elektromos rendszerek irányítására, mint például autókban, háztartási gépekben és még rengeteg más okoseszközben. A legfőbb céljuk a vezérlés. Az első mikrovezérlőt 1971-ben hozta létre az Intel a „4004 processzorral” (Juhász Róbert – Mikrovezérlők története – <https://jrobi.hu/mikrovezerlok/mikrovezerlok-toertenete> – Letöltés: 2025.03.04.) melyet az első, egyetlen chipre integrált mikroprocesszor volt, amit érdekes módon eredetileg számológépekhez fejlesztettek. Bár ez még nem mondható teljes értékű

mikrovezérlőnek, de ez a fejlesztés volt a katalizátor a rapid fejlődéshez. Az első valódinak mondható mikrovezérlőt (TMS1000) a Texas Instruments fejlesztette ki 1974-ben, ami már egyetlen chipben tartalmazott RAM-ot, ROM-ot és I/O perifériákat is mely lehetővé tette a vezérlést önálló számítógépes rendszerrel is. Az idő teltével egyre olcsóbbak és elérhetőbbé váltak a nyilvánosság számára, ezáltal elterjedt az ipari és fogyasztói környezetben is egyaránt. Az Atmel, Microchip, Motorola és az STMicroelectronics kulcsszerepet játszott ezen eszközök fejlesztésében és elérhetővé tételében.

A 2000-es évek elején már WiFi-val és Bluetooth-tal rendelkező vezérlők is megjelentek, amely beindította az IoT(Internet of Things) forradalmát. Az „Espressif Systems” 2016-ban mutatta be az ESP23-es mikrovezérlőt, mely a mai napig a leghasználtabb mikrovezérlő IoT fejlesztésekhez. Legismertebb termékük az ESP8266 és az ESP32-es, ami már képes az analóg és digitális jelek fogadására.

Összességében a mikrovezérlők kulcsfontosságú szerepet töltenek be a modern elektronikai rendszerekben. Alacsony költségük és hatékonyságuk miatt mellőzhetetlenek bizonyos eszközök adatkezelésében és vezérlésében. A fejlődésük folyamatos, bővülő funkciókkal és platformokkal a jövő elektronikai innovációjában is meghatározó szerepet játszhat.

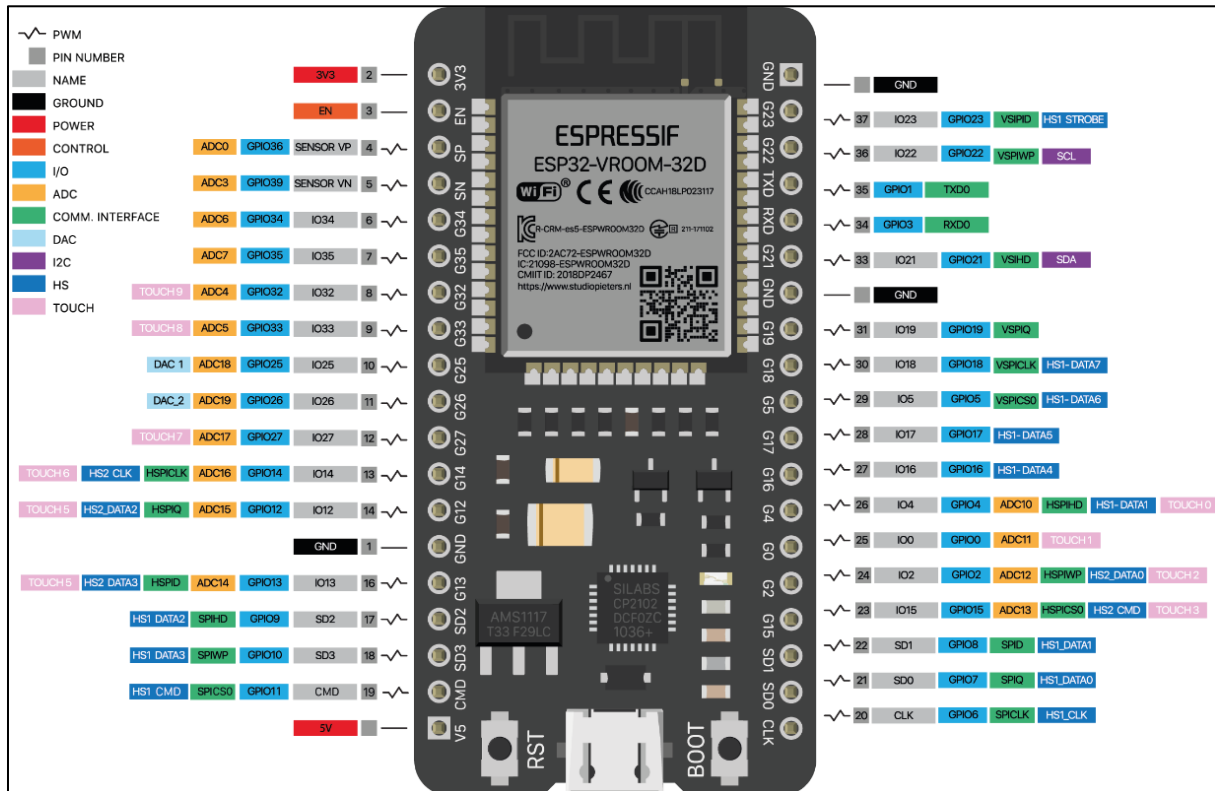
Ebben a projektben alkalmazott mikrovezérlő alkalmas a műszaki paramétereinek alapján (vö. 10-11. ábra) – valamennyi igény szerinti adat feldolgozására és továbbítására a megfelelő módon. Ismer bizonyos mennyiségű kommunikációs protokollt, amely a szenzorokból érkező jelek feldolgozását lehetővé teszi. A mikrovezérlő külső felhasználói felület segítségével programozható a program fordítása után feltölthető az eszközre a számítógép soros portján keresztül. A jól megírt program későbbiek folyamán megbízható működést biztosít.

A projektben alkalmazott eszköz 38 tűs, antenna csatlakozóval rendelkező típus. Jellemzői:

Az eszköz főbb műszaki paramétereinek:

- WiFi:150.0 Mbps, IEEE 802.11b/g/n/e/i
- Bluetooth:
 - Bluetooth classic (állandó rendszerrel sugároz)
 - BLE (kis mennyiségű adatátvitel, alvó állapot)
- Processor(CPU): Tensilica Xtensa Dual-Core 32-bit LX6 microprocessor, 240 MHz
- Memória:
 - ROM: 448 KB (szükséges programok betöltése), SRAM: 520 KB (adatok és parancsokhoz), 16KB az egyéb funkciókhoz (pl.: CPU alvó állapotból való felébresztéséhez)

- Low Power: biztosítja, hogy alacsony energiafelhasználású üzemmódban is használható legyen az ADC.
- Titkosítás: AES, WPA2-PSK, WPA, WPS
- USB Illesztés: CP2102 chip
- Üzemi hőmérséklet: -40...85°C



10. ábra Mikrovezérlő és annak lábkiosztása

Forrás: https://www.studiopieters.nl/esp32-pinout/#google_vignette

	lábak száma (1-38)
	lábak elnevezése (pl.: GPIO 22)
	GND (Ground) – negatív feszültség
	POWER – pozitív feszültség
	CONTROL – irányítói/felügyeleti láb
	I/O – ki- és bemenetre alkalmas láb
	ADC – analóg-digitális átalakító
	COMM. INTERFACE (Communication Interface) – kommunikációs interfész
	DAC – digitális-analóg átalakító
	I2C (Inter-Integrated Circuit) – kétvezetékes soros kommunikációs interfész
	HS (High Speed) – nagysebességű mód bemenete
	TOUCH – érintésérzékelő bemenet

11. ábra: Jelmagyarázat (vö. 10. ábra.)

Forrás: <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/>

Kapcsolások kialakítása során a barométer szenzor jeleinek a lekezeléséhez, annak tápellátásán túl igénybevételre került a GPIO21 és a GPIO22 kivezetések, melyek alapértelmezetten alkalmasak az I²C protokollt használó eszközök jelfeldolgozására.

A szélesség és szélirány szenzorok mivel analóg jelet közvetítenek így azok a GPO32 és GPO34-es kivezetésre kerültek bekötésre.

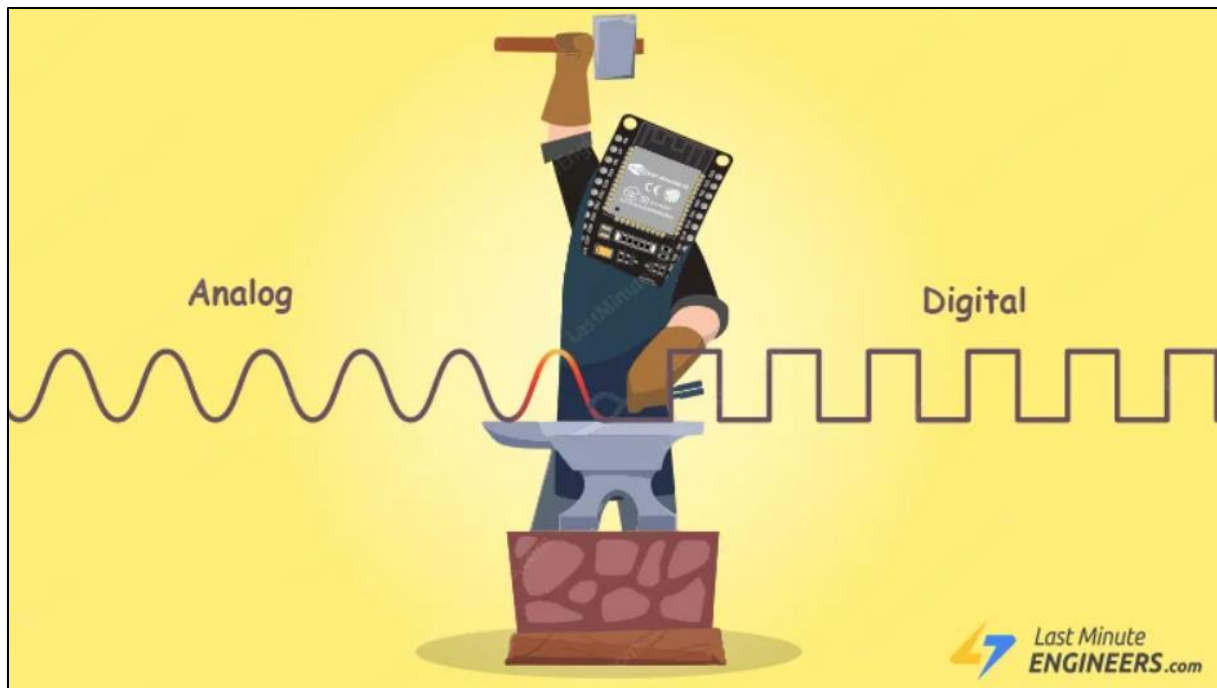
A lábak nagyrésze szoftveresen átdefiniálható, de mivel összesen 3 szenzor került alkalmazásra így ennek a lehetőségnek a kiaknázására nem volt szükség, így elegendő volt a célnak megfelelő alapértelmezett láb használatba vétele.

Az eszköz sokoldalúságát bizonyítja, hogy rendelkezik beépített WiFi antennával, mely szükség esetén kiegészíthető külső antennával is, mely hatékonyabbá teszi kommunikációt. Jelen esetben mivel kültéri eszközről van szó mind a Master mind a Slave eszköz ellátásra került külső antennával is.

Az eszköz tápellátása biztosítható USB C csatlakozón keresztül adapterrel vagy a specifikációk szerinti meghatározott feszültségű tűkön keresztül külső önálló tápegységgel. Az USB C csatlakozó kiemelt fontosságú jelentőséggel bír, ugyanis azon keresztül történik az eszközre írt, kész program feltöltése. Továbbá amennyiben az eszköz számítógépre van csatlakoztatva és fut a programozói felhasználói felület (Arduino IDE), akkor a program futása (jól megírt program esetén) nyomon követhető a soros monitoron (vö. 28. ábra).

„Az analóg-digitális átalakító más néven ADC”, azaz analóg jel digitális jellé való átalakítása. Egy olyan eszköz, amely az analóg jeleket, például feszültséget vagy áramerősséget digitális értékke alakítja át (vö. 12. ábra). Ez azt jelenti, hogy adott időközönként a folyamatosan változó jeleket leolvassa és numerikus értékke raktározza el. A digitális jel nem lesz teljes egészében folyamatos, mivel az átalakítás csak bizonyos időközönként történik. „Elektronikai berendezésekben széles körűen alkalmazzák, mivel a digitális jel stabilabb és pontosabban feldolgozható mint az analóg jel, ezzel egy biztonságosabb működést hozva létre.”

(Wikipedia – Analóg-digitális átalakító – https://hu.wikipedia.org/wiki/Anal%C3%B3g-digit%C3%A1lis_%C3%A1talak%C3%ADt%C3%B3 – Letöltés: 2025.03.10.)



12. ábra: Szemléltetés analóg és digitális jelekről
forrás: <https://lastminuteengineers.com/esp32-basics-adc/>

A ESP 32-es mikrokontroller egyik leglényegesebb műszaki tulajdonsága a „ADC” (Espressif – ADC – <https://espressif-docs.readthedocs-hosted.com/projects/arduino-esp32/en/latest/api/adc.HTML> - 2025.03.13). Az érzékelő 12 bites felbontással rendelkezik, ami azt jelenti, hogy a $2^{12} = 4096$ analóg jelszintet képes észlelni és egész számmá alakítani, vagyis 0-4095 állapot különböztethető meg. Valamennyi lábra legfeljebb 3,3 V feszültséget képes mérni. Ez a későbbiekben azért válik fontossá, mert mind a szélesség, mind pedig a szélirány mérő kimeneti feszültsége 0-5V között van. Erre nyújt megoldást a „feszültségosztó” (vö. 2.3.3 pont) (Last Minute Engineers – Analog-to-digital Converter; – <https://lastminuteengineers.com/esp32-basics-adc/> – Letöltés: 2025.03.13.)

Az Arduino a jelek feldolgozását a `map()` függvénnyel teszi lehetővé, ami leképez egy számot egyik tartományból a másikba.

Alkalmazott függvény:

`„map (value, fromLow, fromHigh, toLow, toHigh)”`

(Arduino – Language Reference – <https://docs.arduino.cc/language-reference/en/functions/math/map/> – Letöltés: 2025.03.13.)

A programozás során:

```
map (sensorValue, 0, 4095, 0, 360);
```

A szélirány érzékelésnél a kiolvasott feszültség alapján 0-4095 közötti érték helyett, visszaad egy 0°-360° közötti érték

A Master és a Slave mikrovezérlő közötti adattovábbítás az ESP-NOW protokoll segítségével történik. Ez egy vezeték nélküli kis fogyasztású kommunikációs lehetőség.

„Az ESP-NOW egyfajta vezeték nélküli Wi-Fi kommunikációs protokoll, amelyet az Espressif fejlesztett ki. Az ESP-NOW-ban az alkalmazás adatokat egy gyártó-specifikus műveleti keretbe foglalják, majd vezeték nélkül továbbítják egyik Wi-Fi-eszköztől a másikra.” (Espressif – https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/net-work/esp_now.HTML – Letöltés: 2025.03.28.)

3.2.2. DS18B20 hőmérséklet szenzor



13. ábra: DS18B20 szenzor

Forrás: <https://makeradvisor.com/tools/ds18b20-digital-temperature-sensor/>

A Slave eszköz mikrovezérlőjéhez csatlakozik a DS18B20 típusú vízálló szenzor (vö. 13. ábra), mely a víz hőmérséklet mérésére alkalmas. Az eszköz egyszerű felépítésű, megfelelő tápellátás mellett egy vezetéken közvetíti a digitális adatokat, mely a programozás során jól és egyszerűen feldolgozható. Ebben a projektben az eszköz a GPIO4-es lábra került bekötésre a mikrovezérlőn. Leglényegesebb műszaki paraméterei:

- kábelén keresztül kommunikál,
- tápfeszültség: 3.0V és 5.5V között,
- működési hőmérséklet tartomány: -55°C és 125°C között,
- pontosság: +/-0.5°C (-10 °C és 85 °C között).

Az eszköz a 1-Wire protokoll szerint adja át az adatokat.

„Az 1-Wire a Dallas Semiconductor által tervezett vezetékes félduplex soros busz, amely alacsony sebességű (16,3 kbit/s [1]) adatkommunikációt és tápfeszültséget biztosít egyetlen vezetőn keresztül. Az 1-Wire koncepciója hasonló az I2C-hez, de alacsonyabb adatsebességgel és nagyobb hatótávolsággal. Általában kis, olcsó eszközökkel, például digitális hőmérőkkel és időjárásmérő műszerekkel való kommunikációra használják. Az egyvezetékes eszközökből álló

hálózatot egy hozzárendelt főeszközzel” (Wikipédia – 1-Wire – <https://en.wikipedia.org/wiki/1-Wire> – Letöltés: 2025.03.13.)

A programozás során a megfelelő osztály könyvtárak (<OneWire.h>, <DallasTemperature.h>) beillesztésével, a `sensors.requestTemperatures()` függvénnyel az adatok közvetlenül kiolvashatók programozás során. (Last Minute Engineers – Interfacing DS18B20 1-Wire Digital Temperature Sensor with Arduino – <https://lastminuteengineers.com/ds18b20-arduino-tutorial/> – Letöltés: 2025.03.13.)

3.2.3. 18650 Lítium akkumulátor V3 bővítőmodul



14. ábra: Akkumulátor burok

Forrás: <https://modulshop.hu/akku-shield-5-v-es-3-v-kimenet-1-db-18650-cella-mikro-usb?keyword=Akku>

A Slave eszköz áramforrását ez a mobil tápegység bővítő modul (vö. 14. ábra) biztosítja, mely egy „kettős feszültségkimenetű 18650 Lithium Battery Shield V3 egy mobil tápegység bővítő modul, amelyet kifejezetten Arduino és ESP mikrovezérlők számára terveztek, hogy biztosítsa a stabil és megbízható áramellátást hordozható projektekhez. A modul egy darab 18650-es lítium akkumulátorral működik, és 5V/2A vagy 3V/1A kimenetet biztosít, így ideális választás sokféle alkalmazáshoz, beleértve az IoT eszközöket, érzékelő hálózatokat és egyéb mobil projektekhez szükséges áramellátási megoldásokat.” (<https://modulshop.hu/akku-shield-5-v-es-3-v-kimenet-1-db-18650-cella-mikro-usb?keyword=Akku> – 2025.03.06. – modulshop.hu webáruház – Leírás, paraméterek.)

3.2.4. Antenna



15. ábra: Antenna

Forrás: <https://modulshop.hu/antenna-24-ghz-wifi-bluetooth-zigbee-sma-to-ipex-1779?keyword=antenna>

A mikrovezérlő rendelkezik antenna csatlakozóval, így lehetőség van Wi-Fi jel minőségének a javítására. Az eszköz (vö. 15. ábra) „*kiváló minőségű 2.4GHz-es 3dBi WiFi antenna ideális választás vezeték nélküli routerekhez, ESP8266 és ESP32 modulokhoz. Az RP-SMA Male csatlakozóval könnyen csatlakoztatható a különböző eszközökhöz, míg a 17 cm-es PCI U.FL IPX to RP-SMA Male Pigtail kábel rugalmas és megbízható kapcsolatot biztosít.*” (modulshop.hu webáruház – Leírás, paraméterek – <https://modulshop.hu/antenna-24-ghz-wifi-bluetooth-zigbee-sma-to-ipex-1779?keyword=antenna> – Letöltés: 2025.03.06.)

3.2.5. BME280 szenzor



16. ábra: BME280 szenzor

Forrás: <https://www.amazon.es/dp/B093C7JK4X?tag=makeraadviso02-21&linkCode=ogi&th=1&psc=1>

A BME280 (vö. 16. ábra) egy érzékeny eszköz, melyen több kis szenzor is van. Ezek léghőmérséklet, légnyomás és páratartalom mérésére alkalmas. A szenzor által közvetített adatok, az I²C protokoll segítségével könnyen és jól feldolgozható szoftveres szinten. Alapvető műszaki

paraméterei jól megfelelnek a projektnek és jól illeszthetők a mikrovezérlőkhöz. Szempon-
tunkból releváns műszaki adottságok:

- tápfeszültség: 1.2V - 3.6V,
- érzékelők működési tartományai:
 - légnyomás: 300 ... 1100 hPa,
 - hőmérséklet: -40 ... +85 °C,
 - páratartalom eltérés: $\pm 3\%$,
 - páratartalom frissítési idő: 1s,
- Működési hőmérséklet tartomány: -40°C és 85°C között.

„Az I²C (Inter-Integrated Circuit) egy multi-Master, multi-Slave, csomagkapcsolt soros busz, melyet a Philips Semiconductor (ma NXP Semiconductors) fejlesztett ki. Az I²C jellemzően kis távolságú, viszonylag alacsony sebességű IC és fedélzeti rendszerek közötti kommunikációra szokás alkalmazni”

(Wikipédia – I²C – <https://hu.wikipedia.org/wiki/I%C2%B2C> – Letöltés: 2025.03.13.)

Az adatok közvetítéséhez a tápfeszültségen túl, két vezeték szükséges, Az egyik órajelhez (SCL), a másik pedig az adatok küldésére és fogadásához (SDA)

3.2.6. RS-FXJT-N01 szélirány mérő



17. ábra: RS-FXJT-N01 szélirány mérő

Forrás: <https://www.amazon.es/dp/B0CQXBCP6F?tag=makeradviso02-21&linkCode=ogi&th=1>

Az alkalmazott széliránymérő (vö. 17. ábra) jelfeldolgozása feszültség alapú (analóg). Lénye-
gében 0°-tól 360° és a 0V-tól 5V feszültség egymáshoz viszonyított megfeleltetésén alapul,
vagyis, ha 0 fokot tekintjük északi iránynak, akkor a kimeneti jelfeszültség 0V, így a déli irány-
nak a 180° 2,5V-nek felel meg. Pontos iránymeghatározás ezek alapján szoftveresen és függ-
vények segítségével történhet, melyek részleteit a 3.4.2 pontban ismertetett forráskódok tartal-
mazzák. Az eszköz lényeges műszaki adatai:

- tápfeszültség: 10~30V DC
- energiafelhasználás: $\leq 0.3W$
- működési feltételek: $-20^{\circ}C \sim +60^{\circ}C$
- válaszidő: 0,5s
- kimeneti feszültség: 0~5V

Ennél az eszköznél az eddigiektől eltérően külön tápellátás biztosítása szükséges az eredményes működés érdekében. A kimeneti jelfeszültséget nem befolyásolja az alkalmazott tápfeszültség nagysága. Ugyanazok a kimenő jelfeszültség mérhető 10V tápfeszültség esetén is és ugyanaz mérhető 30V tápfeszültségnél is. Jelen esetben az eszköz táplálása 12V-os adapterrel történt.

3.2.7. RS-FSJT-N01 szélesség mérő



18. ábra: RS-FSJT-N01 szélesség mérő
Forrás: <https://www.ebay.co.uk/itm/264909611629>

Ennek az eszköznek (vö. 18. ábra) szintén feszültség alapú a jelfeldolgozása (analóg) azzal a különbséggel, hogy minél nagyobb a forgási sebesség, annál nagyobb a kimenő jelfeszültség. Ennek a feszültségnek feldolgozása szintén szoftveresen, függvényekkel történt, figyelembe véve a gyártó által meghatározott műszaki paramétereket. Hasonlóan a szélirányméréshez az 0V-tól 5V-ig tartó kimeneti feszültséget kell feldolgozni. A jelfeszültség arányos, a gyártó által meghatározott 0-70 méter/secundum szélességnek. Ezek alapján a figyelembe vett lényeges gyártói adatok az alábbiak:

- tápfeszültség: 10~30V DC
- energiafelhasználás: $\leq 0.3W$
- működési feltételek: $-20^{\circ}C \sim +60^{\circ}C$
- eltérés $\pm 0.3m/s$
- mérési tartomány: 0~70m/s

- kimeneti feszültség: 0~5V

Ez az eszköz szintén külön tápellátást igényel, mely hasonlóan a szélirány mérőhöz 12V-os hálózati adapterrel történt.

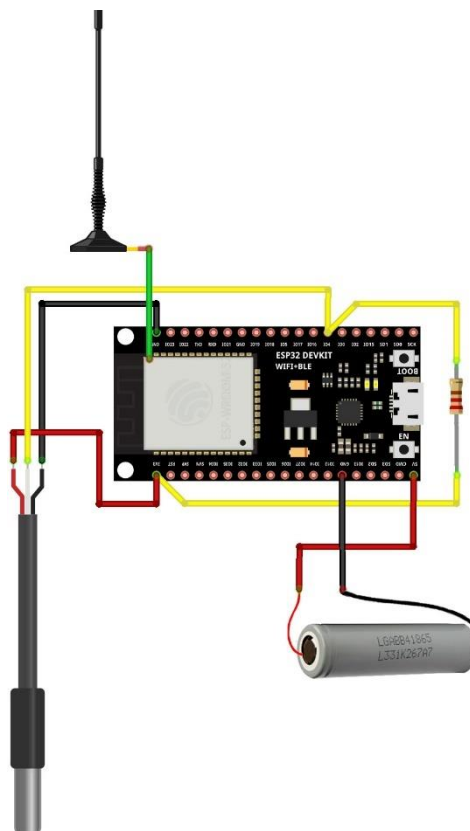
A projektben használt valamennyi eszköz összes műszaki paraméterét a dolgozat mellékletét képező műszaki adatlapok tartalmazzák.

3.3. Hardverkomponensek kapcsolódása

A fejezetben bemutatásra kerülnek a hardverkomponensek mikrovezérlőhöz való csatlakozásai, melyek grafikus ábrázolással is megjelennek. Szó van még a webszerverre való csatlakozásról, az adatok küldéséről és fogadásáról is egyaránt.

3.3.1. A Slave eszköz

Ez az eszköz 3 elemből áll, egyik maga a mikrovezérlő, a másik a hozzákapcsolt szenzor, a harmadik pedig az áramforrás (vö. 19. ábra). Ennek eszköznek a feladata a víz hőmérséklet mérése és az adatok vezeték nélküli kapcsolat használatával továbbítása a Master eszköz részére.



19. ábra. Slave eszköz felépítése
Forrás: Saját szerkesztés - Fritzing

A DS18B20 szenzor 3 kivezetéssel rendelkezik pozitív és negatív tápcsatlakozással, valamint a DQ elnevezésű adatkivezetéssel. Az eszközt tápfeszültséggel maga a mikrokontroller látja el

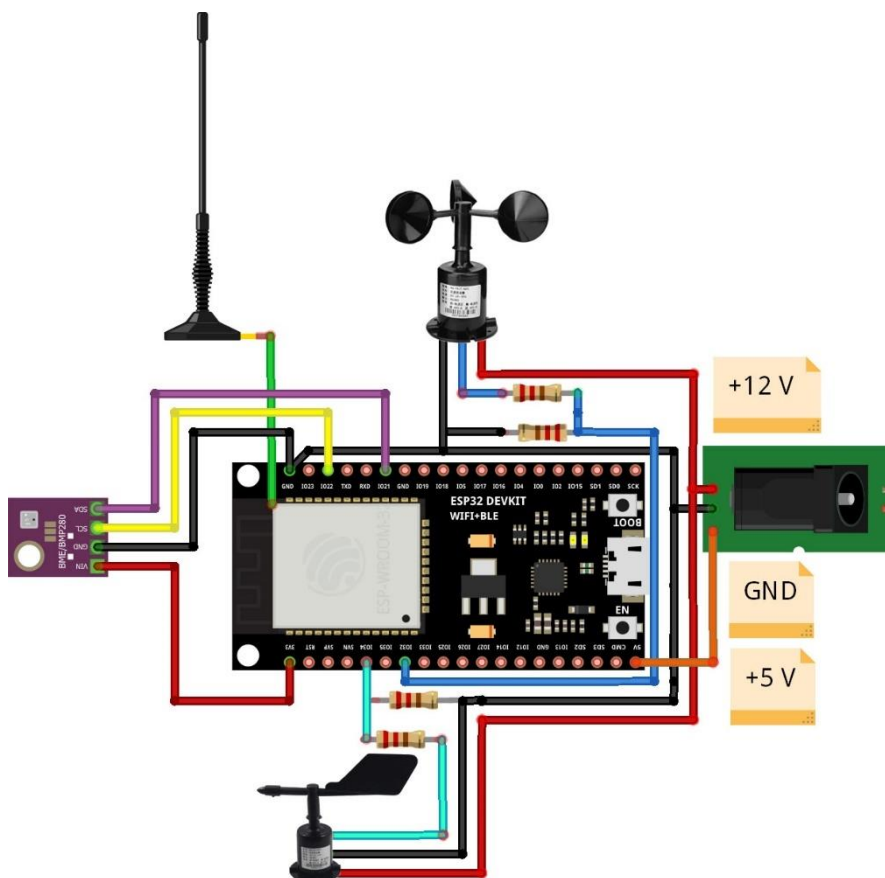
egy 3,3V-os, valamint a GND lábon keresztül. A DQ kivezetés a mikrovezérlő 26-os lábára, azaz a GPO4-es kivezetésére van kötve. A láb alapértelmezetten alkalmas a szenzorról érkező jelek közvetlen kiolvasására és feldolgozására.

Mivel maga az eszköz rendelkezik antenna csatlakoztatási lehetőséggel így ennek a bekötése is megtörtént, egy 2.4 Ghz-es antenna segítségével.

A tápellátást a 3.2.3 Litium battery. pontban részletezett mobil tápegység bővítő modul biztosítja, mely egy 3,7V-os tölthető akkumulátor befogadására alkalmas. Ez közvetlen rákötésre került az 5V-os tű és a GND láb segítségével.

3.3.2. A Master eszköz

A projekt során a legtöbb funkciója ennek az eszközcsoportnak (vö. 20. ábra) van. Önállóan fel kell dolgoznia a barométer szenzor adatait, a szélesség és széliránymérő adatait, kommunikálni kell egyrészt a Slave eszközzel, másrészt biztonságos kapcsolaton keresztül az adatokat továbbítani kell a vezeték nélküli router felé. Megfelelő programozással ez mind biztosítható.



20. ábra Master eszköz felépítése

Forrás: Saját szerkesztés - Fritzing

Az alkalmazott barométer szenzor négy kivezetéssel rendelkezik 3,3V-os pozitív, valamint negatív tápcsatlakozással és az alkalmazott kommunikáció szerinti SDA, SCL elnevezésű adatbusszal. Ezeknek megfelelően az eszköz bekötésre került a 2-es számú, 3,3V-os, a 33-as számú GPIO 21-es SDA, a 36-os számú GPIO 22-es, valamint a közös GND lábra. A GPIO 21 és 22-es lábak szintén alapértelmezetten képesek feldolgozni a beérkezett jeleket az I²C kommunikáció szerint.

A használt szélirány és szélssebesség mérőhöz három kivezetés került csatlakoztatásra. A pozitív 12V-os, valamint negatív tápcsatlakozással, valamint a 0-5V kimenő feszültséget adó adatkivezetéssel. Jelen esetben problémát jelentett, hogy a mikrovezérlő ADC lábai maximum 3,3V feszültség feldolgozására alkalmasak. A probléma megoldását ellenállásokból kialakított feszültségosztó kapcsolat jelenti, ahol arányosan elérhető a 3,3V maximális jelfeszültség.

A széliránymérő a feszültségosztó után bekötésre került a mikrovezérlő 6-os számú GPIO 34-es a szélssebességmérő szintén feszültségosztó után a 8-as számú GPIO 32-es lábra. Az antenna csatlakoztatása itt is megtörtént.

A mikrovezérlő tápellátása egy 5V-os, a szélirány és szélssebességmérőé pedig egy 12V-os hálózati adapter segítségével történik.

3.3.3. Csatlakozás a webszerverhez



21. ábra Hálózati csatlakozás

Forrás: Saját szerkesztés - Adobe Photoshop CS6 Extended

A Master eszköz a megírt program szerint csatlakozik a helyi routeren keresztül a nyílt internet hálózathoz (vö. 21. ábra). Az internet hálózat végpontját, a helyi routert, valamint a nyílt

internethez való hozzáférést a szolgáltatási szerződés alapján távközlési szolgáltató (Magyar Telekom Távközlési Nyilvánosan Működő Részvénytársaság) biztosítja. A projekt megvalósításához domain név és tárhely szolgáltatás került igénybevételre, szintén szerződés alapján. A tárhelyszolgáltató cég Nethely Kft.

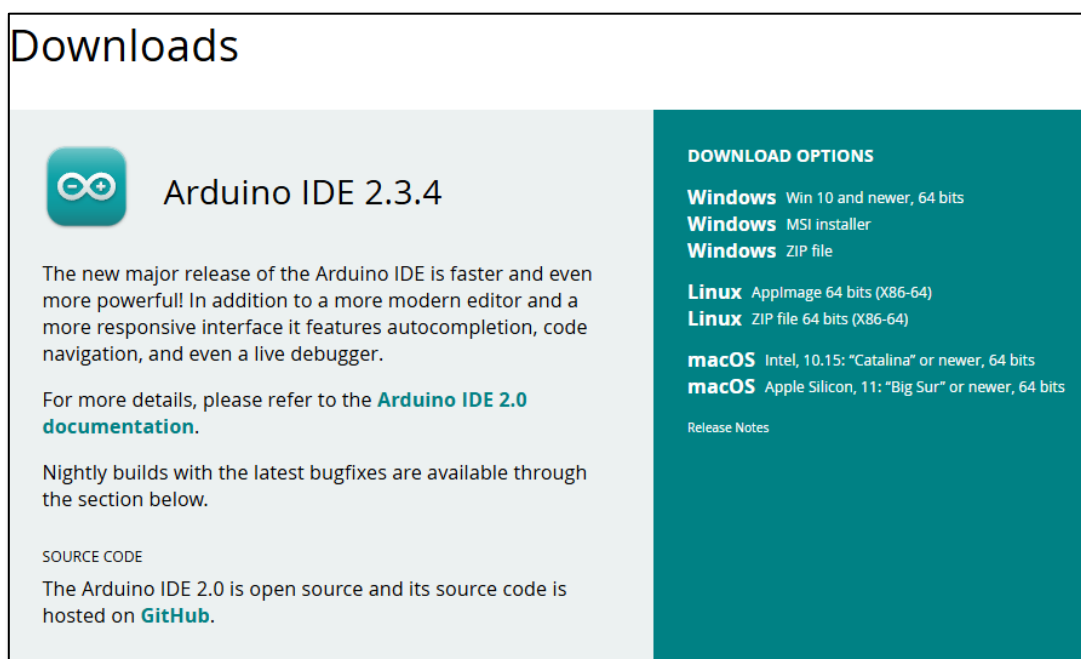
A webserververhez, illetve az adatbázishoz való csatlakozás az általános eljárás szerint PHP szkriptek meghívásával történik. (vö. 3.5.2. alfejezet)

3.3.4. A fejlesztői környezet konfigurálása és a programok működési elve

A fejezetben bemutatásra kerül a fejlesztői környezet telepítése és a különböző szoftverkönyvtárak (package) hozzáadása, amiket képek illusztrálnak, valamint az eszköz számítógéphez való csatlakoztatása és a feltelepítése is.

3.3.5. A mikrovezérlő fejlesztői környezetének telepítése

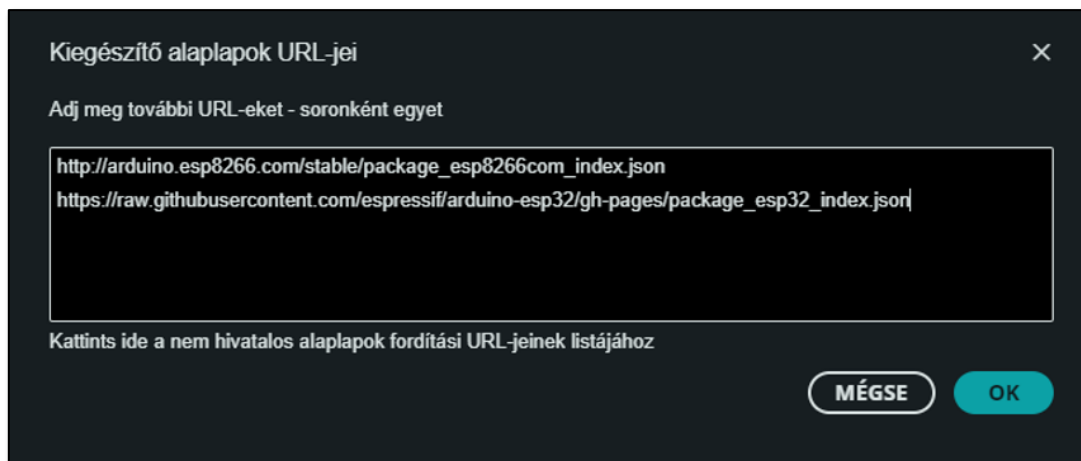
A mikrovezérlők programozásához, futtatásához és teszteléséhez a legalkalmasabb fejlesztői környezet az Arduino IDE (vö. 22. ábra). A letöltése és telepítése a megszokott módon történik, melyet az Arduino hivatalos weboldalon (<https://www.arduino.cc/en/software#experimental-software>) el is érünk.



22. ábra: Fejlesztői környezet letöltése
Forrás: Saját képernyőkép

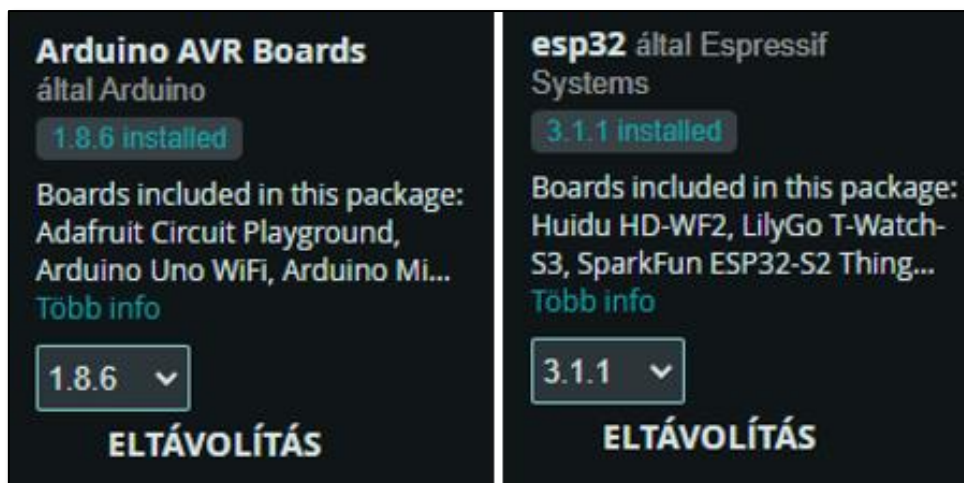
A használt verzió esetében a „Win 10 and newer, 64 bits”. A program használatát segíti a rendelkezésre álló angol nyelvű dokumentáció, mely tartalmazza a szoftver használatára vonatkozó lehetőségeket, mintakódokat, könyvtárak használatát az alkalmazott C programnyelv

referenciáit, függvényeit. További támogatást jelent a fórumrendszer, mely segít megoldani a programozás közben felmerülő problémákat. A munkalap megnyitása után, még nem kezdhető el a program írása, ugyanis telepíteni kell a kiválasztott mikrovezérlő programozását lehetővé tevő fájlokat, könyvtárakat. Esetünkben a használt mikrovezérlő alaplap kezelője URL-jének a beillesztése (vö. 23. ábra) az egyik feltétele a programozhatóságnak.



23. ábra Alaplap kezelő URL-jének a beállítása
Forrás: Saját képernyőkép

A másik feltétel két package telepítése az Arduino AVR Boards by Arduino az esp32 by Espressif Systems (vö. 24. ábra). Ezek telepítését követően elérhetővé és írhatóvá válik a szoftver számára a mikrovezérlő.

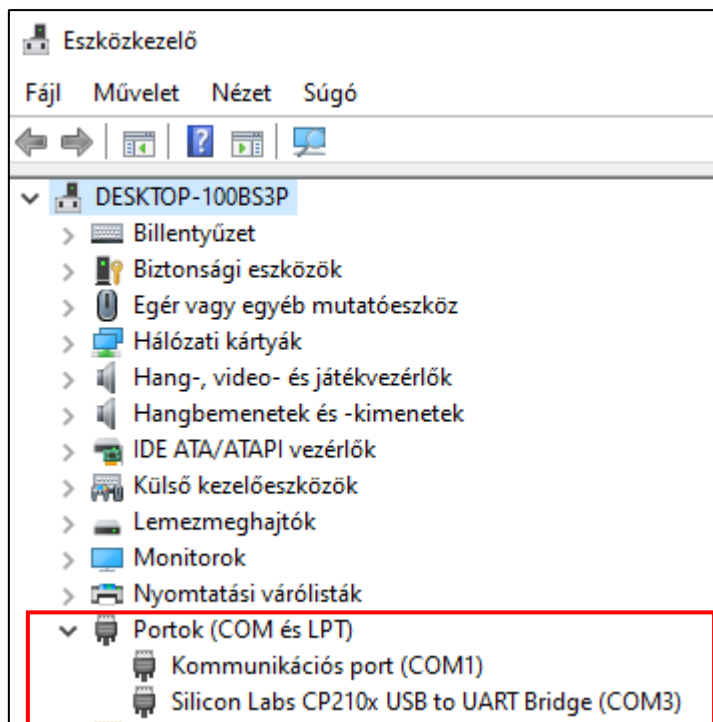


24. ábra Mikrovezérlő csomagjának telepítése
Forrás: Saját képernyőkép

A fejlesztői környezeten túl alapvető a mikrovezérlő driverének a feltelepítése is. A Windows10-et megelőző operációs rendszerek esetében szükséges az eszközben használt mikrochip (CP2102) illesztőprogramjának a feltelepítése is. Ez a gyártó honlapján (Silicon Labs)

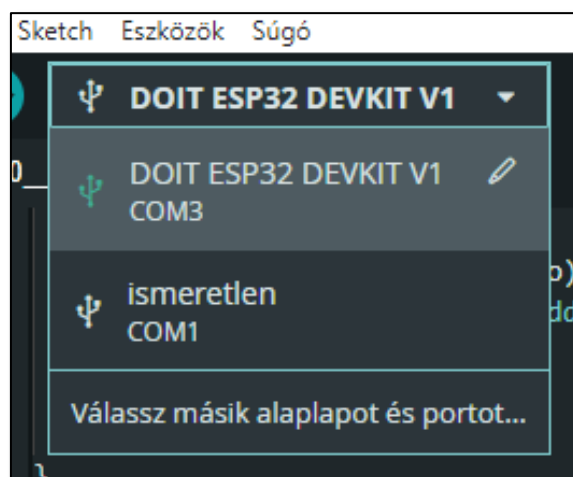
futtatható program formájában rendelkezésre áll. A Windows10 és magasabb verziói külön telepítést nem igényelnek ugyanis, már tartalmazzák ezt a meghajtó programot.

Egy számítógép és a mikrovezérlő közötti kapcsolatot egy Micro USB – USB A típusú kábellel valósítjuk meg soros porton keresztül. Ha élő kapcsolatunk van, az megtekinthető az operációs rendszer eszközkészletében a portok között.



25. ábra Kapcsolat ellenőrzés
Forrás: Saját képernyőkép

Fejlesztői környezet számára be kell azonosítani, hogy mely porton (vö. 25. ábra), mely eszközt csatlakoztattuk. Ki kell választanunk a megfelelő portot és a programozni kívánt eszközt, esetünkben COM3-as porton, a DOIT ESP32 DEVKIT V1 eszközt (vö. 26. ábra).

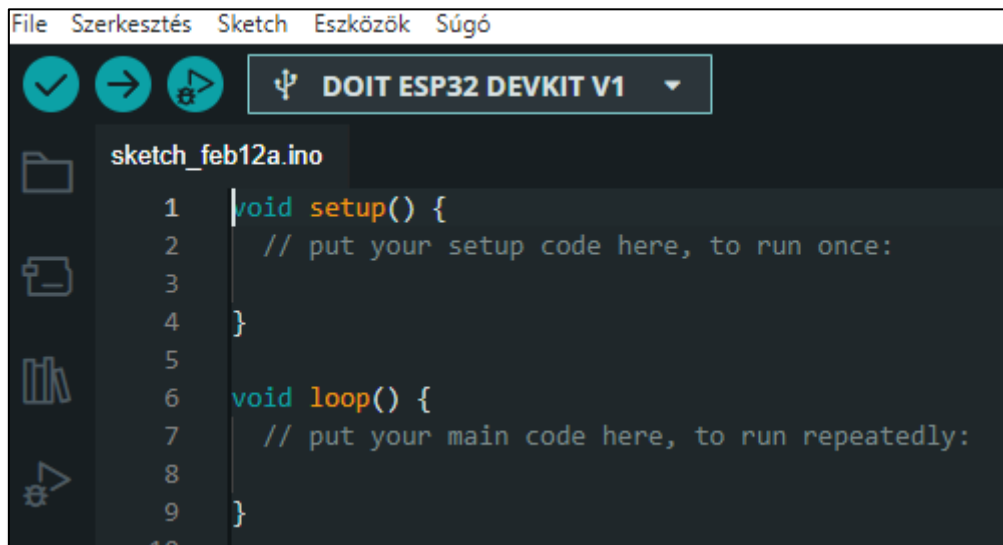


26. ábra Kapcsolat kiválasztása a fejlesztői felületen

Forrás: Saját képernyőkép

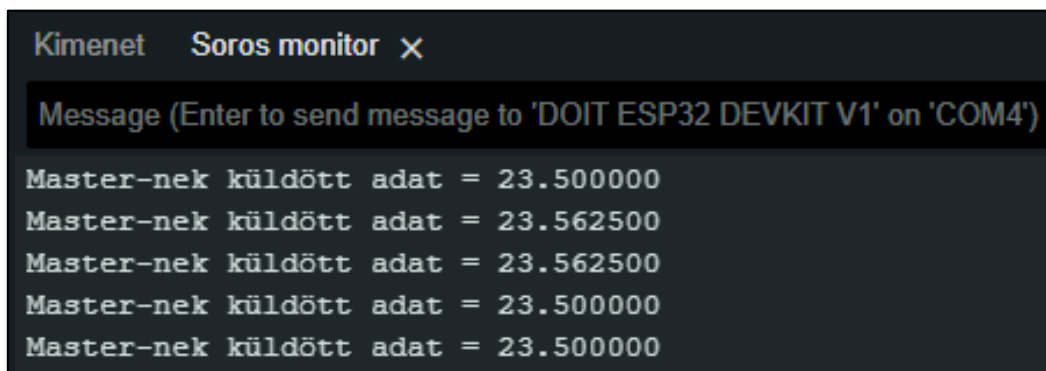
A program megírása lehetséges USB kapcsolat nélkül is, azonban az alaplap és a port meghatározása nélkül az eszközre feltölteni nem lehet, így annak futása sem ellenőrizhető.

A programnak két fő felülete van. Az egyik melyen a programkód megírható (vö. 27. ábra), a másik pedig a soros monitor (vö. 28. ábra), amelyen nyomon követhetők a folyamatok aktuális státuszai (pl.: programfeltöltés, hibakódok stb....) valamint az eszköz működése közben, az általa szolgáltatott adatok.



27. ábra Fejlesztői környezet alapképernyője

Forrás: Saját képernyőkép



28. ábra Soros monitor bemutatása a fejlesztői környezetben

Forrás: Saját képernyőkép

3.3.6. Webhelyen elhelyezett fájlok fejlesztői környezete

Webszerveren HTML és PHP kódok, illetve fájlok biztosítják az adatok lekérdezését és a megjelenítést. A projekt jelen állapotában nem rendelkezik összetett fájlkapcsolatokkal. Három PHP kiterjesztésű fájl oldja meg a teljes vizualizációt. A fájlok szerkesztésére Visual Studio Code (ver 1.98.2) program került felhasználásra. A szoftver alapbeállításai elegendőek volta a

szerkesztéshez, külön beépülő modulok telepítésére nem volt szükség. A kódolás után a fájlok webszerverre történő feltöltését – szintén alapbeállításokkal rendelkező – FileZilla 3.2.8.1. szoftverrel végeztem FTP kapcsolaton keresztül.

3.4. Forráskódok és működésük

A fejezetben ismertetem a mikrovezérlőkre feltöltött forráskódok, valamint a webszerveren futó fájlok tartalmát és azok működésének részleteit. Az eszközök közötti kommunikáció nulladik lépése az alárendelt eszköz MAC-címének a kiolvasása, aminek a programkódja is bemutatásra kerül. Ezek után az alárendelt eszköz, más néven a Slave eszköz, valamint a felsőbbrendű eszköz, azaz a Master eszköz feladatainak és forráskódjainak ismertetése is megtörténik. Részletesen bemutatásra kerül rendszerem adatbázisának kezelése és az adatok webes grafikus megjelenése, forráskóddal társítva.

3.4.1. MAC-cím kiolvasás

A Slave és a Master eszköz közti kapcsolat kialakításának alap pillére a Master eszköz MAC-címének meghatározása, melyre a Slave eszközben hivatkozva létrejön a kapcsolat az eszközök között. Fontos megjegyzendő információ ezzel kapcsolatban, hogy a küldő által meghatározott program struktúrájának megegyezőnek kell lennie a fogadó struktúrájával is. Ha nem egyeznek, a kapcsolat ugyan létrejön, de adatátvitelre nem fog sor kerülni.

(Rui Santos – ESP-NOW with ESP32: Receive Data from Multiple Boards (many-to-one) – <https://randomnerdtutorials.com/esp-now-many-to-one-esp32/> – Letöltés: 2025.02.28.)

A MAC-cím kiolvasásakor segítségünkre válik egy rövid és egyszerű szkript, melyet az eszközre feltöltve és futtatva, a soros monitor megmutatja a megadott eszköz MAC-címét.

```
#include <WiFi.h>
#include <esp_wifi.h>

// Függvény a MAC-cím kiolvasására és kiírására a soros monitorra
void readMacAddress() {
    uint8_t baseMac[6]; // 6 bájtos tömb deklarálása a MAC-cím tárolására

    // MAC-cím lekérése
    esp_err_t ret = esp_wifi_get_mac(WIFI_IF_STA, baseMac);

    if (ret == ESP_OK) {
        // Ha sikeres a lekérdezés, formázott módon kiírjuk a MAC-címet
        Serial.printf("%02X:%02X:%02X:%02X:%02X:%02X\n",
                      baseMac[0], baseMac[1], baseMac[2],
                      baseMac[3], baseMac[4], baseMac[5]);
    } else {
```

```

        // Hiba esetén hibaüzenetet írunk ki
        Serial.println("Failed to read MAC address");
    }
}

void setup() {
    // Soros kommunikáció indítása 115200 baud sebességgel
    Serial.begin(115200);

    WiFi.mode(WIFI_STA); // Az ESP32 WiFi módját állomás módra (STA) állítjuk
    WiFi.begin();         // WiFi kapcsolat indítása)

    Serial.print("[DEFAULT] ESP32 Board MAC Address: ");
    readMacAddress(); // MAC-cím kiolvasása és kiírása
}

void loop() {
    // Nincs kód, mert csak egyszer kell lefutnia a programnak.
}

```

Program: Arduino IDE

3.4.2. Master eszköz kódjának bemutatás

A forráskódba először be kell illeszteni a program futásához szükséges osztálykönyvtárakat, összesen kilencet, mert a feladatok nagy része ebben az eszközben valósul meg.

Az általános felépítésnek megfelelően, az első részben deklarálni kell a szükséges változókat és létre kell hozni az szükséges függvényeket. Az egyik ilyen az aktuális idő lekérdezésére (getTime()), a WiFi (void initWiFi()) és az ESP-NOW (void initEspNow()) inicializálására szolgáló függvény. Itt indul el a Slave eszközről az adatokat fogadó callback függvény is (void onReceive()). A program a következő részben beállítja soros port sebességét és meghívja a WiFi és a ESP-NOW függvényeit, lekérdezi az NTP időt és inicializálja a barométer szenzort. A program ismétlődő fő ciklusa (void loop()) végzi a folyamatos szenzoradat frissítéseket. A szenzorok adatainak tíz másodpercenként frissíti, amit harminc percenként elküld a webszervernek.

```

#include <Arduino.h>
#include <WiFi.h>
#include <esp_now.h>
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#include <WiFiClientSecure.h>
#include <HTTPClient.h>
#include "time.h"

```

```

// NTP szerver beállítása az idő lekéréséhez
const char* ntpServer = "pool.ntp.org"; // NTP szerver elérési útja
const long  gmtOffset_sec = 3600; // Időeltolódás másodpercben
const int   daylightOffset_sec = 3600; // Nyári időszámítás eltérése
unsigned long sensoradat_ido;

// Adatküldés webszerverre 30 percenként
unsigned long lastSendTime = 0; // Utolsó küldés ideje
const unsigned long interval = 30 * 60 * 1000; // 30 perc milliszekundumban

// Aktuális idő lekérése
unsigned long getTime() {
    time_t now;
    struct tm timeinfo;
    if (!getLocalTime(&timeinfo)) {
        return(0);
    }
    time(&now);
    return now;
}

// Wi-Fi hálózat adatai
constexpr char WIFI_SSID[] = "lokális SSID";
constexpr char WIFI_PASS[] = "*****";

// Webszerver beállítások
const char* serverName = "https://to.progsol.hu/post-data.php";
String apiKeyValue = "*****";

// Szélmérő szenzor adatai
const int anemometerPin = 34;
const float minVoltage = 0.0;
const float maxVoltage = 3.3;
const float maxWindSpeed = 70.0;
const float mps_to_kmh = 3.6; // Átváltás m/s-ról km/h-ra

// BME280 szenzor inicializálása
Adafruit_BME280 bme;
unsigned long delayTime;

volatile uint8_t vizhom = 0;
float vh_ki=0;

// Wi-Fi inicializálása
void initWiFi() {
    WiFi.mode(WIFI_MODE_APSTA);
    WiFi.begin(WIFI_SSID, WIFI_PASS);
}

```

```

Serial.printf("Connecting to %s .", WIFI_SSID);
while (WiFi.status() != WL_CONNECTED) { Serial.print("."); delay(200); }
Serial.println(" ok");

IPAddress ip = WiFi.localIP();
Serial.printf("SSID: %s\n", WIFI_SSID);
Serial.printf("Channel: %u\n", WiFi.channel());
Serial.printf("IP: %u.%u.%u.%u\n", ip & 0xff, (ip >> 8) & 0xff, (ip >> 16)
& 0xff, ip >> 24);
}

// ESP-NOW fogadás callback függvény
void onReceive(const esp_now_recv_info_t *mac_addr, const uint8_t *data, int
len) {
    if (len == sizeof(float)) {
        float vizhom;
        memcpy(&vizhom, data, sizeof(float));
        Serial.printf("Slave-ről érkező vízhőmérsékleti adat= %2f°C\n",
vizhom);
        vh_ki=vizhom;
    }
    else {
        Serial.println("Hiba a fogadott adatban!");
    }
}

// ESP-NOW inicializálása
void initEspNow() {
    if (esp_now_init() != ESP_OK) {
        Serial.println("Az ESP-NOW inicializálása nem sikerült");
        while (1);
    }
    esp_now_register_recv_cb(onReceive);
}

// Beállítások inicializálása
void setup() {
    Serial.begin(115200);
    delay(500);
    initWiFi();
    initEspNow();

    Serial.println(F("BME280 test"));

    configTime(gmtOffset_sec, daylightOffset_sec, ntpServer);
    bool status;

    status = bme.begin(0x76);
    if (!status) {

```

```

        Serial.println("Could not find a valid BME280 sensor, check wiring!");
        while (1);
    }
    delayTime = 10000;
    Serial.println();
}

// Fő programciklus
void loop() {
    Serial.println();
    sensoradat_ido = getTime();

    // Kiírás a soros monitorra
    Serial.print("Léghőmérséklet = ");
    Serial.print(bme.readTemperature());
    Serial.println(" *C");

    Serial.print("Légnyomás = ");
    Serial.print(bme.readPressure() / 100.0F);
    Serial.println(" hPa");

    Serial.print("Páratartalom = ");
    Serial.print(bme.readHumidity());
    Serial.println(" %");

    Serial.print("Víz hőmérséklet = ");
    Serial.print(vh_ki);
    Serial.println(" *C");

    // Szélsebesség mérése
    int sensorValue = analogRead(32);
    int szelseb = map(sensorValue, 0, 4095, 0, 70);
    float kmh = szelseb * mps_to_kmh;

    Serial.print("Szélsebesség = ");
    Serial.print(szelseb);
    Serial.print(" m/s, ");
    Serial.print(kmh);
    Serial.println(" km/h");

    // Adatküldés a webszerverre
    if (millis() - lastSendTime >= interval) {
        lastSendTime = millis(); // Frissítjük az időzítést

        if(WiFi.status()== WL_CONNECTED){
            WiFiClientSecure client;
            client.setInsecure();
            HTTPClient https;
            https.begin(client, serverName);

```

```

    https.addHeader("Content-Type", "application/x-www-form-urlencoded");
    String httpRequestData = "api_key=" + apiKeyValue + "&homers=" +
String(bme.readTemperature()) + "&paratart=" + String(bme.readHumidity()) +
"&legnyomas=" + String(bme.readPressure()/100.0F);
    int httpResponseCode = https.POST(httpRequestData);
    Serial.print("HTTP válaszkód = ");
    Serial.println(httpResponseCode);
    https.end();
  } else {
    Serial.println("WiFi Disconnected");
  }
}
delay(10000);
}

// Helyi idő kiírása
void printLocalTime() {
  struct tm timeinfo;
  if(!getLocalTime(&timeinfo)){
    Serial.println("Failed to obtain time");
    return;
  }
  Serial.println(&timeinfo, "%Y %B %d, %A %H:%M:%S");
}

```

Program: Arduino IDE

3.4.3. Slave eszköz kódjának bemutatás

Az eszköznek alapvetően két feladata van, a vízhőmérséklet mérő szenzor adatainak kiolvasása majd pedig továbbítása, közvetlenül a Master eszköz felé.

A kódot a feladat végrehajtásához szükséges osztálykönyvtárak meghívásával kezdjük.

Utána definiálni kell a szükséges globális változókat (hálózat SSID azonosítója, az időmérés változói) majd pedig azt, hogy fizikailag az eszköz (DS18B20 szenzort) a mikrovezérlő melyik, lábához csatlakozik. Jelen esetben ez a 4-es számú láb.

A következő szekció megkeresi a meghatározott SSID-vel rendelkező hálózatot és visszaadja a csatorna számát, vagy pedig a 0 értéket, ha nem találja meg.

Utána kerül sor az eszköz állomás módra állítása, illetve a WiFi és az ESP-NOW inicializálására. Fontos része ennek, a korábban említett (vö. 3.5.1 fejezet) Master eszköz, Slave eszközben történő MAC címének a beállítása, ugyanis ez a fajta kommunikációs protokoll azon alapszik, hogy a küldő egy előre meghatározott MAC címmel rendelkező eszköznek küldi az adatokat. A program a következő részben szintén beállítja soros port sebességét, elindítja a WiFi-t, az ESP-NOW-t, valamint a szenzorérzékelést. A program ismétlődő része (void loop()), először az

időt megvizsgálja, azaz, hogy eltelt-e 10 perc, ha igen lekéri a szenzor adatait és elküldi a Masternek, ha nem, akkor még várakozik.

```
#include <Arduino.h>
#include <WiFi.h>
#include <esp_wifi.h>
#include <esp_now.h>
#include <DallasTemperature.h>
#include <OneWire.h>

constexpr char WIFI_SSID[] = "lokális SSID";

unsigned long lastSendTime = 0; // Utolsó küldés ideje
const unsigned long interval = 10 * 60 * 1000; // 10 perc milliszekundumban

// Lábdefiniálás - erre csatlakozik a vízhőmérséklet mérő szenzor
const int oneWireBus = 4;
OneWire oneWire(oneWireBus);
DallasTemperature sensors(&oneWire);

// WiFi csatorna lekérdezése adott SSID alapján
int32_t getWiFiChannel(const char *ssid) {
    if (int32_t n = WiFi.scanNetworks()) { // Hálózatok keresése
        for (uint8_t i = 0; i < n; i++) {
            if (!strcmp(ssid, WiFi.SSID(i).c_str())) {
                return WiFi.channel(i); // Visszaadjuk a csatornaszámot
            }
        }
    }
    return 0; // Ha nem található a „lokális SSID”, visszatérünk 0-val
}

// WiFi inicializálása
void initWiFi() {
    WiFi.mode(WIFI_MODE_STA); // Állomás mód bekapcsolása
    int32_t channel = getWiFiChannel(WIFI_SSID); // WiFi csatorna lekérése
    esp_wifi_set_promiscuous(true);

    // Csatorna beállítása
    esp_wifi_set_channel(channel, WIFI_SECOND_CHAN_NONE);
    esp_wifi_set_promiscuous(false);
}

// ESP-NOW vevő eszköz MAC címének a beállítása
constexpr uint8_t ESP_NOW_RECEIVER[] = { 0x20, 0x43, 0xa8, 0x64, 0x33, 0x08 };
esp_now_peer_info_t peerInfo;

// ESP-NOW inicializálása
void initEspNow() {
```

```

    if (esp_now_init() != ESP_OK) { // Ha az inicializálás sikertelen
        Serial.println("ESP NOW failed to initialize");
        while (1); // Végtelen ciklus
    }

    // Partner eszköz adatainak beállítása
    memcpy(peerInfo.peer_addr, ESP_NOW_RECEIVER, 6);
    peerInfo.ifidx = WIFI_IF_STA;
    peerInfo.encrypt = false;

    // Partner eszköz hozzáadása
    if (esp_now_add_peer(&peerInfo) != ESP_OK) {
        Serial.println("ESP NOW pairing failure");
        while (1); // Végtelen ciklus hiba esetén
    }
}

// Beállítások inicializálása
void setup() {
    Serial.begin(115200); // Soros kommunikációs sebesség beállítása
    initWifi(); // WiFi inicializálása
    initEspNow(); // ESP-NOW inicializálása
    sensors.begin(); // Hőmérséklet érzékelő inicializálása
}

// Fő programciklus
void loop() {
    // Ha eltelt a beállított időintervallum
    if (millis() - lastSendTime >= interval) {
        lastSendTime = millis(); // Időzítés frissítése

        sensors.requestTemperatures(); // Hőmérséklet lekérése
        float temperatureC = sensors.getTempCByIndex(0);
        float data = temperatureC;

        // Adat küldése a Masternek
        esp_now_send(ESP_NOW_RECEIVER, (uint8_t *) &data, sizeof(float));
        // Adat kiírása a soros portra
        Serial.printf("Master-nek küldött adat = %2f", data);
        Serial.println();
    }

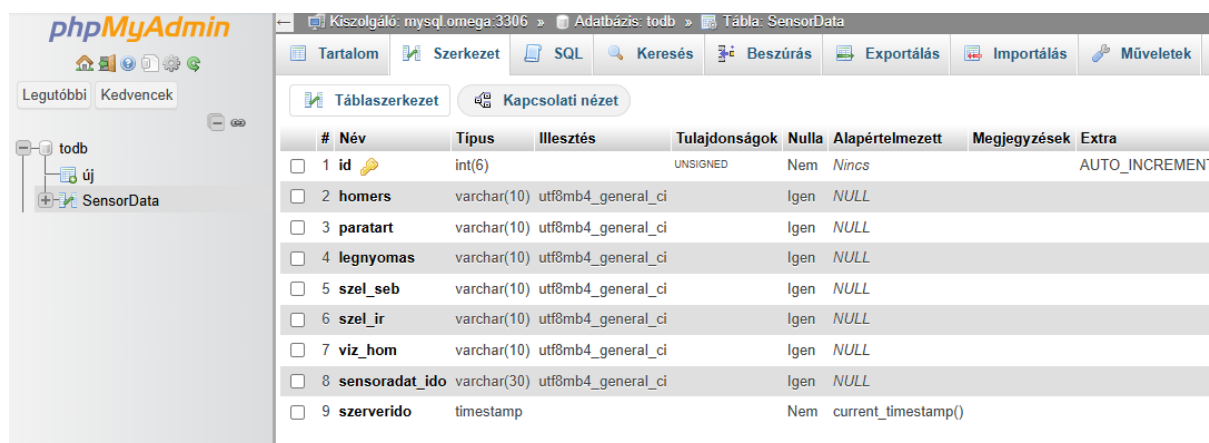
    delay(3000); // 3 másodperces késleltetés
}

```

Program: Arduino IDE

3.4.4. Webes adatbázis kezelése

Az adatbázisnak kialakításához meg kell határozni a megfelelő adatstruktúrát és azt létre kell hozni a webtárhely szolgáltató phpMyAdmin oldalán. A projekt igényei alapján elkészült egy „todb” nevű adatbázis, abban pedig a „SensorData” adattábla az alábbi adatszerkezettel (vö. 29. ábra):



#	Név	Típus	Illesztés	Tulajdonságok	Nulla	Alapértelmezett	Megjegyzések	Extra
1	id	int(6)		UNSIGNED	Nem	Nincs		AUTO_INCREMENT
2	homers	varchar(10)	utf8mb4_general_ci		Igen	NULL		
3	paratart	varchar(10)	utf8mb4_general_ci		Igen	NULL		
4	legnyomas	varchar(10)	utf8mb4_general_ci		Igen	NULL		
5	szel_seb	varchar(10)	utf8mb4_general_ci		Igen	NULL		
6	szel_ir	varchar(10)	utf8mb4_general_ci		Igen	NULL		
7	viz_hom	varchar(10)	utf8mb4_general_ci		Igen	NULL		
8	sensoradat_ido	varchar(30)	utf8mb4_general_ci		Igen	NULL		
9	szerverido	timestamp			Nem	current_timestamp()		

29. ábra: Adattábla

Forrás: Saját képernyőkép – Tárhelyszolgáltató phpMyAdmin felülete

A cél olyan kód írása mely képes a mért adatokat ebben az adattáblába beszúrni. Erre a megoldást a PHP nyelv kínálja.

A PHP-t „Rasmus Lerdorf indította útjára 1995-ben, ... ami akkoriban Personal Home Page Tools néven futott. ... Napjainkban a The PHP Group tagjai tartják fenn és fejlesztik”. (Wikipédia – PHP – <https://hu.wikipedia.org/wiki/PHP> – Letöltés: 2025.03.12).” A PHP alapvetően egy „szabad szoftver; de licence nem csereszabatos a GNU licenccel, mivel megkötések tartalmaz a PHP név használatára.” (WEBiskola– PHP Ismeretek –https://webiskola.hu/php-ismeretek/mi-az-a-php-fogalma-bemutatas/#google_vignette – Letöltés: 2025.03.13.) A PHP egy szerveroldali szkriptnyelv, mely dinamikus weboldalak készítésére használunk. Sajátos fejlesztő környezetének a segítségével adatbázis alapú weboldal létrehozására is kivitelezhetővé válik. A webszerveren található feldolgozómodul értelmezi a PHP nyelven fogadott kódokat. Ez az első olyan szkriptnyelv, mely HTML oldalakba ágyazható, egy külső fájlhasználat helyett. A webprogramozás a sajátosságát figyelembe véve először deklarálni kell a szükséges változókat. Megvizsgáljuk, hogy érkezik-e post üzenet, ha igen, akkor megnézzük, hogy érkezett-e vele API kulcs. Ha érkezett, azt betesszük egy változóba és összehasonlítjuk ebben a fájlban deklarált API kulcs értékével. Ha nem egyezik meg a kulcs, akkor hibaüzenetet kapunk, ha megegyezik a két kulcs akkor kerülhet sor majd az adatbázis műveletekre. De mielőtt ez megtörténik a „function test input ()” függvény biztonsági adattisztítást hajt végre. Az érkező

adatokat betesszük egy helyi változóba, utána következik az adatbázishoz való csatlakozás. Ha ez bármi oknál fogva nem lehetséges, akkor hibaüzenetet kapunk. Ha viszont a kapcsolódás sikeres, akkor történhet meg a helyi változóban lévő adatok beillesztése az adatbázisba az INSERT INTO parancs segítségével.

Jelen esetben az alábbi PHP kód hajtja végre a fenti műveleteket.

```
<?php
// Adatbázis kapcsolat adatai
$servername = "localhost"; // Szerver címe (helyi szerver esetén 'localhost')
$dbname = "todb";          // Adatbázis neve
$username = "todb";        // Adatbázis felhasználónév
$password = "*****";      // Adatbázis jelszó

// API kulcs, amelyet a bejövő adatok hitelesítésére használunk
$api_key_value = "0123456789ABCDEF";

// Inicializálás: üres változók létrehozása az adatok tárolására
$api_key = $homers = $paratart = $legnyomas = $szel_seb = $szel_ir = $viz_hom
= $sensoradat_ido = "";

// Ellenőrizzük, hogy a kérés POST típusú-e
if ($_SERVER["REQUEST_METHOD"] == "POST")

    // Beolvasás és tisztítás (validálás) az API kulcs esetén
    $api_key = test_input($_POST["api_key"]);

    // Ellenőrizzük, hogy a kapott API kulcs megegyezik-e a saját API
    kulcsunkkal
    if ($api_key == $api_key_value) {

        // Beolvasás és tisztítás (validálás) az egyes szenzoradatok esetén
        $value_homers = test_input($_POST["homers"]); // Hőmérséklet
        $value_paratart = test_input($_POST["paratart"]); // Páratartalom
        $value_legnyomas = test_input($_POST["legnyomas"]); // Légnyomás
        $value_szel_seb = test_input($_POST["szel_seb"]); // Szélsébség
        $value_szel_ir = test_input($_POST["szel_ir"]); // Szélirány
        $value_viz_hom = test_input($_POST["viz_hom"]); // Vízhőmérséklet
        $value_sensoradat_ido = date("Y-m-d H:i:s",
            test_input($_POST["sensoradat_ido"])); // Szenzor adatgyűjtésének
            ideje

        // Kapcsolódás az adatbázishoz
        $conn = new mysqli($servername, $username, $password, $dbname);

        // Ellenőrizzük, hogy sikerült-e csatlakozni
        if ($conn->connect_error) {
```

```

        die("Connection failed: " . $conn->connect_error); // Hiba esetén
        kilépünk és kiírjuk a hibát
    }

    // SQL parancs: beszúrjuk az adatokat az adatbázisba
    $sql = "INSERT INTO SensorData (homers, paratart, legnyomas, szel_seb,
szel_ir, viz_hom, sensoradat_ido)
        VALUES ('" . $value_homers . "', '" . $value_paratart . "', '" . $va-
lue_legnyomas . "', '" . $value_szel_seb . "', '" . $value_szel_ir . "', '" .
$value_viz_hom . "', '" . $value_sensoradat_ido . "')";

    // Lefuttatjuk az SQL parancsot és ellenőrizzük, hogy sikeres volt-e
    if ($conn->query($sql) === TRUE) {
        echo "New record created successfully"; // Sikeres beszúrás esetén
        üzenetet küldünk vissza
    }
    else {
        echo "Error: " . $sql . "<br>" . $conn->error; // Hiba esetén ki-
írjuk az SQL hibát
    }

    // Bezárjuk az adatbázis kapcsolatot
    $conn->close();
}
else {
    // Ha az API kulcs nem egyezik, visszautasítjuk a kérést
    echo "Wrong API Key provided.";
}
}
else {

    // Ha nem POST kérés érkezik, figyelmeztető üzenetet küldünk vissza
    echo "No data posted with HTTP POST.";
}

// Tisztító függvény a bemenő adatokhoz (alapvető védelem XSS és SQL Injection
ellen)
function test_input($data) {
    $data = trim($data); // Levágja a szóközöket az elejétől és vé-
géről
    $data = stripslashes($data); // Eltávolítja a "\" karaktereket
    $data = htmlspecialchars($data); // Átalakítja a HTML speciális karaktere-
ket (XSS támadások ellen)
    return $data; // Visszatér a tisztított értékkel
}

```

3.4.5. Adatok GUI-n keresztüli megjelenítése

A weboldal GUI felülete szolgál a szenzorok által küldött adatok vizuális megjelenítésre. Az adatok egy MySQL adatbázisban tároljuk, amelyből a weboldal annak betöltésekor vagy frissítésekor kéri le a legutolsó mérésnek megfelelő adatokat. Jelenleg az adatok megjelenítése két oldalon történik. Az egyik oldal a legutolsó értékeket jeleníti meg (vö. 30. ábra):

Mért adatok	Értékek
Léghőmérséklet	22.15 °C
Légnyomás	994.95 hPa
Páratartalom	42.08 %
Szélsébség	21.60 km/h
Szélirány	ÉSZAK
Víz hőmérséklet	22.06 °C
Mérés ideje	2025-03-30 18:33:29

30. ábra: Utolsó adatok megjelenítése a webfelületen

Forrás: Saját képernyőkép – Jelmagyarázat: a felület csak a legutolsó mért adatot jeleníti meg a léghőmérsékletet és a víz hőmérsékletet Celsius-fokban, a légnyomást hektopascalban, a páratartalmat százalékban, a szélsébséget kilométer/óránban, a szélirányt égtáj szerint, a mérés idejét pedig év-hónap-nap óra-perc-másodperc megjelölésével.

A másik (vö. 31. ábra) pedig az adatbázis teljes tartalmát, így a korábbi mérések is elérhetők és elemezhetők.

Mérés ideje	Léghőmérséklet °C	Légnyomás hPa	Páratartalom %	Szélsébség km/h	Szélirány	Víz hőmérséklet °C
2025-03-30 18:33:29	22.15	994.95	42.08	21.60	ÉSZAK	22.06
2025-03-30 18:33:18	22.16	994.94	42.19	18.00	DÉL-NYUGAT	22.06
2025-03-30 18:33:08	22.13	994.93	43.05	21.60	DÉL-KELET	22.06
2025-03-30 18:32:58	22.11	994.95	42.80	18.00	DÉL-KELET	22.06
2025-03-30 18:32:47	22.07	994.96	43.18	18.00	DÉL-KELET	22.06
2025-03-30 18:32:37	22.04	994.92	42.43	18.00	NYUGAT	22.13

31. ábra: Weboldalon történő teljes adatbázis megjelenítése

Forrás: Saját képernyőkép – Képernyőkép Jelmagyarázat: a GUI másik felületén valamennyi mért adat megtekinthető a mérés ideje szerint csökkenő sorrendben, azaz a legutolsó mért adat a legfelső sorban kerül megjelenítésre. Egy sor, a *Mérés ideje* időpontban gyűjtött adatok összessége.

3.5. Speciális aspektusok

Ez a szakdolgozati téma több IT biztonsági aspektust is érint, amelyeket figyelembe kell venni a tervezés és a megvalósítás során. Az alábbi főbb biztonsági szempontokat érdemes átgondolni: (vö. ChatGTP konverzációra 2025.02.22 – 10.6.2 fejezet)

3.5.1. Adatbiztonság és titkosítás

„Az időjárásállomás által küldött adatok titkosítása (pl.: TLS/SSL használata a kommunikáció során)” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

„A TLS (Transport Layer Security) és elődje, az SSL (Secure Sockets Layer) kriptográfiai protokollok, amelyek célja a hálózati kommunikáció biztonságának szavatolása. Csatornákat hoznak létre a szerver és a kliens között az adatok módosításának, lekérésének és ellopásának megakadályozása érdekében. Továbbá biztosítják még az adatok integrálásának ellenőrzését és a felek azonosítását is” (Wikipédia – Transport Layer Security https://en.wikipedia.org/wiki/Transport_Layer_Security – Letöltés: 2025.03.22.). A két protokoll közti fő különbség, hogy a SSL újabb és jelentősebb biztonságot nyújt (Kinsta – TLS vs SSL: What’s the Difference? Which One Should You Use? – <https://kinsta.com/knowledgebase/tls-vs-ssl/> – Letöltés: 2025.03.22.).

- *„Ha az adatokat hosszabb ideig tárolod, akkor megfelelő titkosítás és jogosultságkezelés szükséges.”* (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

Az eszköz ebben az állapotában egy darab adatbázist és egy darab adattáblát használ a web-szerveren. Az adatok helyi adattárolás nélkül közvetlenül az adatbázisba kerülnek. Az azokhoz való hozzáférés a webtárhely rendszergazdai jogosultságával lehetséges. Jelenleg nincs olyan korlátozott hozzáféréssel rendelkező adminisztrátor, aki adatmanipulációt hajthat végre.

- *„Az érzékeny adatok (pl.: földrajzi koordináták, szenzorok azonosítói) védelme.”* (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

Érzékeny adatok jelenleg nem kerülnek helyben tárolásra és feldolgozásra.

3.5.2. Hálózati biztonság

- „Az adatküldés biztonságos protokollokkal történjen (HTTPS, MQTT+TLS, vagy más titkosított csatorna).” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

A HTTPS (Hypertext Transfer Protocol Secure) az egy biztonságos web kommunikációs protokoll, amely titkosítja az adatokat a kliens és a szerver között. „HTTP protokoll biztonságos változata, amely titkosítást alkalmaz az adatok védelme érdekében a számítógépes hálózaton történő kommunikáció során” (Wikipédia – HTTPS – <https://en.wikipedia.org/wiki/HTTPS> – Letöltés: 2025.03.22.) Az SSL/TLS protokoll használata esetén a szerver és a kliens közötti kommunikáció titkosított csatornán keresztül folyik, ami megakadályozza az adatátvitel alatti támadásokat. Ez különösen fontos nyilvános Wi-Fi hálózatokon, ahol a felhasználók érzékeny információkat, például bejelentkezési adatokat vagy pénzügyi tranzakciókat végeznek. A Master eszköz HTTP protokollt használ az adatbázis elérésére, továbbá a megépített rendszer nem MQTT protokoll szerint működik, így biztonsági szempontból ez a téma nem releváns.

- „A hálózati eszközök (Wi-Fi modul, IoT hálózat) sebezhetőségeinek minimalizálása.” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

3.5.3. Hozzáférés-kezelés és autentikáció

- „A mikrovezérlő és a webszerver közötti kommunikáció autentikációja (pl.: JWT, HMAC vagy egyedi API-kulcs).” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

A projektet az API kulcs használata érinti, mivel az adatok forrásának a helye egy, a Master eszköz programjába épített, előre definiált 16 jegyű karaktersorozattal van azonosítva.

```
String apiKeyValue = "0123456789ABCDEF"; //arduino forráskód részlet
```

A webszerveren meghívott PHP fájlban szintén deklarálva van ugyanezt a kulcs.

```
$api_key_value = "0123456789ABCDEF"; //php forráskód részlet
```

A program összehasonlítja egy feltételvizsgálattal a rendelkezésre álló és a kapott kulcsot és az adatok adattáblába írása csak akkor valósul meg, ha a két kulcs megegyezik, különben hibaüzenetet íródik ki.

- „Webes felület esetén erős jelszókezelés és felhasználói jogosultságok beállítása.” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

Az erős jelszó és a felhasználók jogosultságának megfelelő beállítása a webes alkalmazások egyik alapvető eleme. Az erős jelszó használat megnehezíti a támadók (hackerek) számára a

fiók feltörését, ezért ajánlott a kis és nagy betű kombinálása, az egyediség, azaz minden felhasználó felületen új jelszó alkalmazása, és a hossz megfelelő megválasztása. Ajánlott legalább 12 karakteres jelszó használata (vö. 10.6.2. pont). Az igénybe vett tárhelyszolgáltató is legalább 12 jegyű jelszót ajánl. Ennek betartása erősen ajánlott a biztonsági kockázatok miatt.

- „Ha több eszköz csatlakozik, akkor eszközazonosítók egyedi kezelésének biztosítása.”
(OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22)

A projektben ez jelenen nem releváns. Amennyiben a jövőkép víziói megvalósulnak, akkor erre tekintettel kell lenni a fejlesztés során.

3.5.4. Adatintegritás és megbízhatóság

- *Adatok manipulálásának vagy hamis adatküldésnek a megakadályozása (pl.: digitális aláírás, checksum).* (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

A projektben ez jelenleg ezek a megoldások nem merültek fel.

- *Redundancia vagy hibatűrő mechanizmusok kialakítása (pl.: időbélyegzés, duplikált adatellenőrzés).* (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

Az IT biztonsági kockázatokat csökkenti az időbélyegek kezelése. A projektben két időadat került kezelésre. Az egyik Master eszköz által rögzített mérési idő, a másik pedig az adatbázis timestamp típusú adata.

Mivel a Master eszköz az internethez csatlakozik, így a `getTime()` függvény segítségével az NTP idő lekérhető. Ez az időadat elküldésre kerül a webszerverre a többi mért adattal együtt.

A timestamp időadat, az érkező adatokkal együtt kerül rögzítésre az adattáblában, így ellenőrizhetőbbé válik az adatbázis.

A 32. ábra egy részletet mutat a két idő közötti különbségről. Eszerint adatok beírása, szinte kivétel nélkül azonos időben megtörténik. Ritkán fordul elő egy másodperces késés.

sensoradat_ido	szerverido
2025-03-29 11:48:37	2025-03-29 11:48:37
2025-03-29 11:48:47	2025-03-29 11:48:47
2025-03-29 11:48:57	2025-03-29 11:48:58
2025-03-29 11:49:08	2025-03-29 11:49:08
2025-03-29 11:49:18	2025-03-29 11:49:19
2025-03-29 11:49:29	2025-03-29 11:49:29
2025-03-29 11:49:39	2025-03-29 11:49:39
2025-03-29 11:49:49	2025-03-29 11:49:50

32. ábra: MySQL adatbázis tábla részlet

Forrás: Saját képernyőkép– Jelmagyarázat: a tárhelyszolgáltató phpMyAdmin felülete, mely a szenzor által elküldött idő és az adatbázis időbélyegét mutatja év-hónap-nap óra-perc-másodpercben.

3.5.5. Webszerver és adatbázis biztonsága

- „SQL injekció, XSS, CSRF támadások elleni védelem a webalkalmazásban.” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

SQL injekció az egyik leggyakoribb webes sebezhetőség, amely jelentős kockázatot jelent az adatbiztonságra nézve. Lehetővé teszi a támadóknak, hogy rosszindulatú SQL parancsokat illesszenek be egy alkalmazás lekérdezésébe. Ez egy súlyos támadási forma mely lehetővé teszi az adatlopást, az adatbázis módosítását vagy esetleg törlését és akár az adatbázis szerver irányítást is átvehetik, mely további rendszer kompromitáláshoz is vezethet. (Wikipédia – SQL-injekció – <https://hu.wikipedia.org/wiki/SQL-injekci%C3%B3> – Letöltés:2025.03.22.), (Owasp – SQL Injection – https://owasp.org/www-community/attacks/SQL_Injection – Letöltés: 2025.03.22.)

A Cross-Site Scripting (XSS) lehetővé teszi a támadóknak, hogy rosszindulatú szkripteket töltsenek fel a weboldalra, amelyet a böngészőben futtat a felhasználók nevében, ami adatlopáshoz, módosításhoz vagy hamis tartalmak megjelenítéséhez vezethet. Az XSS elleni védekezés alapvető lépésnek számít, amiben a Content Security Policy (CSP) segít, ami korlátozza a böngészőben betöltött forrásokat. (Nette – Sebezhetőségi védelem – <https://doc.nette.org/hu/vulnerability-protection> – Letöltés:2025.03.22.)

A meghívott PHP kód rendelkezik egy biztonsági függvénnyel, mely védelmet nyújt az XSS támadások ellen. A PHP fájl az érkező adatot az alábbi kódok alapján vizsgálja át:

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $api_key = test_input($_POST["api_key"]);  
    if($api_key == $api_key_value) {  
        $value_homers = test_input($_POST["homers"]);  
        $value_paratart = test_input($_POST["paratart"]);  
        $value_legnyomas = test_input($_POST["legnyomas"]);  
        $value_szel_seb = test_input($_POST["szel_seb"]);  
        $value_szel_ir = test_input($_POST["szel_ir"]);  
        $value_viz_hom = test_input($_POST["viz_hom"]);  
    }  
}
```

Az érkező adatok átadásra kerülnek test_input() függvénynek, ami megtisztítja a veszélyt jelző karaktereket és visszaadja a változót. Ez valamennyi érkező adatnál végrehajtásra kerül.

```
function test_input($data) {  
    $data = trim($data); //Levágja a szóközöket, tabokat és új sorokat az ele-  
    jéről és végéről  
    $data = stripslashes($data); //Eltávolítja a fordított perjeleket (\),  
    hogy elkerüljük az esetleges rossz beviteleket  
    $data = htmlspecialchars($data); Hozzájárult ahhoz, hogy különböző nézőpon-  
    tokból elemezzem a problémákat.  
    return $data; // A megtisztított adat visszaadása  
}
```

A Cross-Site Request Forgery (CSRF) egy olyan támadás típus mely ráveszi a felhasználót, hogy nem kívánt művelet hajtson végre akaratlanul. Elég csak egy rosszindulatú linkre kattintani, ami lendít egy jogosulatlan tranzakciót, tehát a támadó kihasználja az aktív felhasználói felületet, legyen szó, jelszó változtatásról, pénztalásról vagy akármilyen egyéb műveletről. Az egyedi token (azonosító) használata, amit a szerver ellenőriz minden kérés feldolgozása során. (WebFox – Mi a CSRF elleni védelem? – <https://vexfox.com/mi-a-csrf-elleni-vedelem> – Letöltés: 2025.03.22.)

A projektben a GUI egyirányú kommunikációt végez, azaz adatokat mutat be a felhasználóknak, nincs interakció a userek felől. Így a fenti támadás típus nem jelent kockázatot.

„Tokenek működése: Egy oldalra való bejelentkezést követően a szerver egy token, titkos kódot generál, ha elküldünk egy műveletet, űrlapot a kódot is elküldjük a kéréssel. A szerver ellenőrzi a kódot, összehasonlítja őket, ha a kód egyezik, akkor elfogadja, ha pedig nem egyezik vagy nincs kód, akkor elutasítja a kérés”. (Európai Információs Technológia Hitelesítési Akadémia – Mik azok a CSRF tokenek, és hogyan védenek a CSRF támadások ellen? – <https://hu.eitca.org/cybersecurity/eitc-is-wasf-web-applications-security-fundamentals/server->

[security/server-security-safe-coding-practices/examination-review-server-security-safe-coding-practices/what-are-csrf-tokens-and-how-do-they-protect-against-csrf-attacks](https://www.chatgpt.com/security/server-security-safe-coding-practices/examination-review-server-security-safe-coding-practices/what-are-csrf-tokens-and-how-do-they-protect-against-csrf-attacks) – Letöltés: 2025.03.22.)

- „Rendszeres szoftverfrissítések és biztonsági patchek alkalmazása.” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22.)

A kockázatok folyamatos figyelése, és az azokra való reagálás – azaz a mikrovezérlők programjainak frissítése – elkerülhetetlen.

- „Naplózás és monitoring az esetleges behatolások észlelésére.” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22)

Azon túl, hogy a webtárhely szolgáltató folyamatosan felügyeletet gyakorol szerverek felett és monitorozza a működésüket és a futó folyamatokat, a megrendelő is felelős a szerverbiztonságáért. Köteles minden védelmi eszközt alkalmazni és tájékoztatni a szolgáltatót az esetlegesen észlelt támadásról.

3.5.6. Fizikai biztonság

Az eszközök fizikai védelme is hozzájárul a biztonsághoz. Egyrészt anyagi érdek kapcsolódik hozzá (eszközök ára, felprogramozási, telepítési és kiépítési költségek), másrészt az eszközök működésképtelensége miatt, az adatok kimaradása közvetve egyéb hátrányt okozhatnak (horgászok elmaradása, szolgáltatásminőség csökkenése).

Ezek kiküszöbölésére az eszközt célszerű legalább 2,8 méter magasba építeni, a tápellátását vandálbiztos kivitelben készíteni, kívülről nem hozzáférhető módon, valamint rejtett kábelezés kialakítani. Helyi viszonyoktól függően a kiépítést célszerű lehet kamerával megfigyelt területen elvégezni, épület tetején (ügyelve a karbantartási lehetőségek biztosítására) esetleg körberített zárt kertben.

Mivel nyílt eszközökről van szó, így a IP szabvány szerinti vízmentes kialakítás is fontos, melynek legalább IP65-ösnek kell lennie.

„Az IP-védetség (International Protection Marking vagy Ingress Protection Code) jelölése Nemzetközi Védetségjelölés. A műszaki berendezések áramköréit védő tokozás (készülékház) környezeti behatások elleni védetségét jelzik vele.” (Wikipédia – IP-védetség – <https://hu.wikipedia.org/wiki/IP-v%C3%A9de%C3%A9s> – Letöltés: 2025.03.30.)

3.5.7. DDoS és más szolgáltatásmegtagadási támadások elleni védelem

- „Gyanús aktivitások figyelése és blokkolása (pl.: IDS/IPS megoldások).” (OpenAI – IT biztonsági aspektusok – www.chatgpt.com – Letöltés: 2025.02.22)

„A behatolásérzékelő rendszer (IDS) egy olyan hálózati biztonsági megoldás, amely figyeli a forgalmat a gyanús tevékenységek szempontjából, és riasztja a biztonsági csapatokat, ha ilyen tevékenységet észlel” (Makay – Mi a különbség az IDS és az IPS között? – <https://makay.net/ids-ips-kulonbseg-intrusion-detection-prevention-system> – Letöltés: 2025.03.22.). „Az Intrusion Prevention System (IPS) egy hálózati biztonsági és fenyegetésmegelőző eszköz. Az IPS fő célja a potenciális fenyegetések azonosítása és gyors kezelése.” (Makay – Mi a különbség az IDS és az IPS között? – <https://makay.net/ids-ips-kulonbseg-intrusion-detection-prevention-system> – Letöltés: 2025.03.22.). Az IDS és IPS kombinálásával növelhető a hálózat biztonsága, lehetővé teszi a fenyegetések gyors és precíz észlelését elhárítását. A megfelelően implementált IDS/IPS rendszerek szinte elengedhetetlenek a modern hálózatok biztonságának fenntartásához. Ebben a projektben ezen a rendszerek működtetése, nem a megrendelőt terhelik.

3.5.8. Tesztelés

A projekt, illetve az alkalmazott eszközök méretei, bonyolultsága miatt kiterjedt tesztelési terv nem készült. Két alapelv került ezzel kapcsolatban figyelembevételre:

„A tesztelést kezdjük mindig kicsiben és haladjunk a nagyobb részek felé. Első lépésben tehát győződjünk meg az eljárások, függvények helyes működéséről, majd jöhetnek a modulok, al-rendszerek, és végül a teljes rendszer.” (Horváth László, 2007 – Szoftvertesztelés a gyakorlatban).

„Korai teszt: Érdemes a tesztelést az életciklus minél korábbi szakaszában elkezdni, mert minél hamar találunk meg egy hibát (mondjuk a specifikációban), annál olcsóbb javítani.”

(Ficsor Lajos, Dr. Kovács László, Krizsán Zoltán, Dr. Kúspér Gábor, 2011 – Szoftvertesztelés)

A program felépítése kis lépésekkel történt. Amíg egy-egy modul, szenzor nem volt képes a megfelelő módon működni, addig nem volt értelme tovább lépni a következő szenzorra vagy eszközre. Ennek tesztelésére az Arduino fejlesztői környezet rendelkezik egy soros monitorral, ahol vizuálisan megfigyelhető a soros portok folyamatai és a kapott hibaüzenetek is.

A program mikrovezérlőre történő feltöltése előtt, az Arduino lefordítja, egyben ellenőrzi a szintaktikai, illetve a szemantikai hibák egy részét is (pl.: változók hibás deklarálása).

Már a programírás során ügyelni kell a tesztelhetőségre, azaz egy-egy változó vagy eredmény kiíratásáról gondoskodni kell. Ha azok az elvárt eredményt adják, akkor lehet a következő folyamatra lépni. Ezek a program sorok a működést nem befolyásolják, viszont az esetleges hibák felderítését a későbbiekben is segítik.

Amikor a Master és a Slave eszközök külön-külön is az elvárt adatokat szolgáltatották, akkor következhetett a két mikrovezérlő közötti kommunikáció kialakítása az ESP-NOW segítségével.

Ennek működőképessége után a Master eszköz soros monitorán megjelentek a Slave által mért (vö. 3. ábra) és küldött adatok.

A következő fázis az összegyűjtött adatok webszerverre történő továbbításának megoldása. A vonatkozó programrészletnek is a részét képezi az adattovábbítás aktuális állapotának a monitorozása. Függvény segítségével kiírható az adatok adatbázisban való rögzítésnek sikerességéről lekért válasz üzenet is. Ha ez a függvény a 200-as értékkel tér vissza (vö. 34. ábra), akkor az adatok rögzítésre kerültek.

Az adatok továbbításának folyamatát az alábbi képek egy példán keresztül szemléltetik.

A Slave eszköz a tesztidőszakban 3 másodperces mérésre és adattovábbításra volt beállítva (vö. 33-35. ábra).

```
14:20:28.161 -> Master-nek küldött adat = 22.187500
14:20:31.177 -> Master-nek küldött adat = 22.187500
14:20:34.217 -> Master-nek küldött adat = 22.187500
14:20:37.291 -> Master-nek küldött adat = 22.250000
14:20:40.327 -> Master-nek küldött adat = 22.250000
14:20:43.367 -> Master-nek küldött adat = 22.125000
14:20:46.380 -> Master-nek küldött adat = 22.250000
```

33. ábra: A Slave soros monitorán megjelenő teszt adatok

Forrás: Saját képernyőkép – Jelmagyarázat: egy adatsor végig követése a GUI-n való megjelenítésig 1; a Slave eszköz mért adata

```
14:20:40.325 -> Slave-ről érkező vízhőmérsékleti adat= 22.250000°C
14:20:41.255 ->
14:20:41.255 -> 2025 March 30, Sunday 14:20:40
14:20:41.255 -> Léghőmérséklet = 22.62 *C
14:20:41.255 -> Légnyomás = 996.05 hPa
14:20:41.255 -> Páratartalom = 42.47 %
14:20:41.255 -> Vízhőmérséklet = 22.25 *C
14:20:41.255 -> Szélirány = 329 fok
14:20:41.255 -> Égtáj = ÉSZAK-NYUGAT
14:20:41.297 -> Szélsebesség = 4 m/s, 14.40 km/h
14:20:41.297 ->
14:20:41.297 -> Webszerverre küldött adatok:
14:20:41.297 -> api_key=[redacted]&homers=22.62&paratart=42.47&1
14:20:41.619 -> HTTP válaszkód = 200
14:20:41.619 ->
14:20:43.366 -> Slave-ről érkező vízhőmérsékleti adat= 22.125000°C
```

34. ábra: A Master soros monitorán megjelenő teszt adatok

Forrás: Saját képernyőkép – Jelmagyarázat: egy adatsor végig követése a GUI-n való megjelenítésig 2; Látható a Slave eszközről érkező adatsor (vö. 30. ábra), a Master eszköz mért adatai, az webszerverre küldendő összefűzött karaktersorozat, valamint a webszerverről visszaérkező 200-as válaszkód az adatbázisban való adatbeírásról. Biztonsági api key szürke színnel kitakarva.)

Kontrollt jelent a küldött paraméterek és azok értékének a kiírása:

`api_key=0123456789ABCDEF&homers=22.62¶tart=42.47&legnyomas=996.05&szel_seb=14.40&szel_ir=ÉSZAK-NYUGAT&viz_hom=22.25&sen-soradat_ido=1743337240`

Ezek után az adatok már GUI-n megjeleníthetők (vö. 43. ábra):

Mérési adatok

Mérés ideje	Lég hőmérséklet °C	Légnyomás hPa	Páratartalom %	Szélsebesség km/h	Szélirány	Víz hőmérséklet °C
2025-03-30 14:21:01	22.61	996.04	42.61	7.20	DÉL-NYUGAT	22.19
2025-03-30 14:20:50	22.62	996.07	42.35	18.00	NYUGAT	22.25
2025-03-30 14:20:40	22.62	996.05	42.47	14.40	ÉSZAK-NYUG	22.25

35. ábra: GUI-n megjelenő adatok

Forrás: Saját képernyőkép – Jelmagyarázat: egy adatsor végig követése a GUI-n való megjelenítésig 3; A GUI felületen vizualizált adatsor, melyen megjelenítésre került a Master eszköz által küldött valamennyi adat (vö. 29-30 ábra)

3.5.9. Mesterséges Intelligencia

Jövőbeni cél egy olyan adatvagyon létrehozása, melyből egy valós predikciót kapjon a felhasználó arra vonatkozólag, hogy érdemes-e útnak indulnia a horgászni, illetve milyen esélyekkel tudja azt megtenni.

Ehhez szükséges azt megvizsgálni, hogy az elképzelésben szereplő adatok nemcsak tapasztalati módon, hanem a Knuth-i elv „tudás/tudomány csak az, ami forráskódba átírható”, (MIAU–Napló – https://miau.my-x.hu/miau2009/index_tki.php3?_filterText0=*knuth. – Letöltés: 2025.03.28.) alapján a hasonlóságelemzést felhasználva relevánsak-e, így megalapozott, jó döntés a műszaki projekt megvalósítása.

„A Knuth-féle elvek a mesterséges intelligencia területén a kódolás szigorú formalizálását és a tudás strukturált megjelenítését hangsúlyozzák. Donald Knuth módszertana különösen hatékonyan alkalmazható az MI-rendszerek fejlesztésében” (Knuth elvek – [Perplexiti.ai](https://perplexiti.ai) – Letöltés: 2025.03.28.)

Ennek elemzésére a mesterséges intelligencia alapú, COCO (Component-based Object Comparison for Objectivity) elnevezésű online keretrendszert használtam, mely „Dobó Andor matematikus által kialakított hasonlóságok hasonlósága (Dobó Andor: A hasonlóságelmélet alkalmazása a Joker rendszerben 1992) gondolatvilág és a mesterséges neurális hálók előnyeinek integrálásából született meg.” (Pitlik Marcell, 2020 – Térfogati égés optikai vizsgálata és mesterséges intelligencia alapú elemzése Tudományos Diákköri Konferencia Budapest – chrome-

[extension://efaidnbnmnnibpcajpcglcleftindmkaj/https://www.combustioninstitute.hu/tdk_dij/2021/PitlikMarcell_TDK2020.pdf](https://efaidnbnmnnibpcajpcglcleftindmkaj/https://www.combustioninstitute.hu/tdk_dij/2021/PitlikMarcell_TDK2020.pdf) – Letöltés: 2025.03.28.)

A rendszernek több modulja van melyből a COCO_STD (standard) és a COCO_MCM (Monte-Carlo Method) került igénybevételre.

COCO_STD (standard): a COCO-alapú hasonlóságelemzés standard modulja, mely keretében kell, hogy legyen valós Y -változó (pl.: ár), mely az X -változók lépcsősfüggvényeként kerül közelítő jelleggel felépítésre a hasonlóságelemzés komponens-alapú logikája szerint.

COCO_MCM (Monte-Carlo Method): a COCO-alapú hasonlóságelemzés azon modulja, mely keretében nincs semmilyen korlátozó feltétel megadva az optimalizálás kapcsán, vagyis az attribútumok szintjei és az output szintek közötti tetszőleges alakzatú (polinomizálódó), legnagyobb magyarázó erejű összefüggés exploratív feltárása a cél” (Pitlik László, 2014 – Vidék- és területfejlesztési potenciálok elemzése, ezekhez kapcsolódóan a hasonlóságelemzés módszertan alkalmazása – Letöltés: 2025.03.25.)

A vizsgálat során először szükség volt egy táblázatra, mely soronként tartalmazza a kiértékelendő attribútumokat (vö. 36. ábra).

Szenárió ID	Víz hőmérséklet (°C)	Lég hőmérséklet (°C)	Szélesség (km/h)	Széllírány	Légnyomás (hPa)	Víz mélység (m)	Csali típusa	Napszak	Fogott halak száma (db)	Fogott halak össz tömege (kg)
1	23,7	26,9	5,28	Északi	1012	3,5	Fehérje	Reggel	3	18,5
2	22,4	27,4	7,77	Északi	994,5	2,9	Fehérje	Reggel	24	64,5
3	4,6	9,4	12,61	Déli	969,7	8,6	Szénhidrát	Délben	13	28
4	12,9	15,4	17,64	Nyugati	952,5	5,1	Fehérje	Délben	17	21,3
5	24,7	28,8	9,88	Keleti	954,5	4,7	Fehérje	Reggel	18	47,8
6	13,3	16,8	1,78	Északi	1010,8	8,3	Szénhidrát	Reggel	27	48,3
7	14,9	19,5	29,06	Nyugati	965,9	4,1	Szénhidrát	Délben	26	76,4
8	12,2	15,2	9,91	Északi	1003,1	9,4	Fehérje	Reggel	29	70
9	11,9	14,6	5,35	Keleti	1030,5	7,3	Szénhidrát	Reggel	29	86,7
10	20,3	23,9	0,59	Északi	1018,5	6,9	Szénhidrát	Reggel	25	50,6
11	26,4	28,8	9,61	Északi	981,7	4,3	Szénhidrát	Este	5	7

36. ábra: Adatok struktúrája (részlet, 11 sor a 300 sorból)

Forrás: Saját képernyőkép – Jelmagyarázat: az oszlopokban található az egyes mérési tulajdonságok és a hozzá tartozó mértékegységek, a sorokban pedig az egy-egy időpontban mért adatok

Ezt a kiindulási táblázatot szükséges átalakítani, olyan OAM (Objektum-Attribútum-Mátrix) mártixszá, mely alkalmas a COCO általi feldolgozásra. (vö. <https://miau.myx.hu/miau/323/horgaszat/horgaszat.xlsx>)

Feltételezés, hogy a 32. ábrában szereplő tulajdonságok, hatással vannak a fogott halak számára és tömegére. A továbbiakban szükséges számszerűsíteni a szöveges attribútumokat ahhoz, hogy a számítások elvégezhetőek legyenek. Továbbá a következmény változók értékének a kifejezésére, nagy értékeket kell választani a számítások pontosságának érdekében.

Hogy hogyan hatnak a tulajdonságok, arra nincs adat, így lényegében a hatásoknak nincs iránya, valamint az oszlop adatain belül sincs egyértelmű kapcsolat és nincs prioritás sem a

szélnél, sem a csalinál, sem a napszaknál. Megállítható, hogy lényegében végtelen sok nyitott kérdés van, ami befolyásolja a fogott halak számát és tömegét.

A fentiek és a 37. ábrában található segéd táblázat alapján, összeállítható volt az input tábla (vö. 38. ábra).

Szél:		
Északi	1	n/a
Déli	2	n/a
Nyugati	3	n/a
Keleti	4	n/a

Csali:		
Fehérje	1	n/a
Szénhidrát	2	n/a

Napszak:		
Reggel	1	n/a
Dél	2	n/a
Este	3	n/a

37. ábra: Segéd táblázat
Forrás: Saját kép – Képernyőkép

Szenárió ID	Vízhőmérséklet (°C)	Léghőmérséklet (°C)	Szélesség (km/h)	Szélirány	Légnyomás (hPa)	Vízmélység (m)	Csali típusa	Napszak	Fogott halak száma (db)	Fogott halak össz. tömege (kg)
1	23,7	26,9	5,28	1	1012	3,5	1	1	3000	18500
2	22,4	27,4	7,77	1	994,5	2,9	1	1	24000	64500
3	4,6	9,4	12,61	2	969,7	8,6	2	2	13000	28000
4	12,9	15,4	17,64	3	952,5	5,1	1	2	17000	21300
5	24,7	28,8	9,88	4	954,5	4,7	1	1	18000	47800
6	13,3	16,8	1,78	1	1010,8	8,3	2	1	27000	48300
7	14,9	19,5	29,06	3	965,9	4,1	2	2	26000	76400
8	12,2	15,2	9,91	1	1003,1	9,4	1	1	29000	70000
9	11,9	14,6	5,35	4	1030,5	7,3	2	1	29000	86700
10	20,3	23,9	0,59	1	1018,5	6,9	2	1	25000	50600
11	26,4	28,8	9,61	1	981,7	4,3	2	3	5000	7000

38. ábra: Input tábla (részlet, 11 sor a 300 sorból)

Forrás: Saját képernyőkép – Jelmagyarázat: az oszlopokban található az egyes mérési tulajdonságok és a hozzá tartozó mértékegységek, a sorokban pedig az egy-egy időpontban mért adatok, a segéd táblák (vö. 37. ábra) szerint átalakítva, azaz a szöveges adatok helyett számokkal kódolva

Mivel nem tudjuk attribútumok pontos hatásmechanizmust (irányt), így inverz adatok leképezésére is szükség van (vö. 39. ábra)

	inverz	inverz	inverz	inverz	inverz	inverz	inverz	inverz	n/a	n/a
Szenárió ID	Vízhőmérséklet (°C)	Léghőmérséklet (°C)	Szélesség (km/h)	Szélirány	Légnyomás (hPa)	Vízmélység (m)	Csali típusa	Napszak	Fogott halak száma (db)	Fogott halak össz. tömege (kg)
1	76	75	21	23	63	28	52	33	3000	18500
2	72	77	29	23	45	22	52	33	24000	64500
3	9	14	46	50	21	82	101	68	13000	28000
4	39	36	62	75	4	44	52	68	17000	21300
5	80	83	36	101	6	40	52	33	18000	47800
6	40	41	7	23	62	80	101	33	27000	48300
7	46	50	98	75	17	33	101	68	26000	76400
8	37	35	36	23	55	91	52	33	29000	70000

39. ábra: Inverz tábla (részlet, 11 sor a 300 sorból)

Forrás: Saját képernyőkép – Jelmagyarázat: az oszlopokban található az egyes mérési tulajdonságok és a hozzá tartozó mértékegységek, a sorokban pedig az egy-egy időpontban mért inverz adatok, melyek mértékegysége sorszám. A fogott halak száma esetén konverzációra került végrehajtásra: db*1000

Az így elkészült táblázat már alkalmas arra, hogy input adatai legyenek a COCO rendszernek. A modellben az egy-egy időponthoz, $O(n)$ -hez tartozó különböző attribútumok $X(An)$ -el kerülnek megjelölésre.

Elsőként standard modell (COCO-STD) futtatására került sor, melynek eredménye olyan termelési függvény (vö. 40. ábra), ami megmutatja, hogy hány darab halat fogunk fogni, illetve melyik attribútum, milyen hatással van az elvárt eredményre.

Lépcsők (2)	X(A1) Víz hőm.	X(A2) Lég hőm. (°C)	X(A3) Szél seb.	X(A4) Szél ir.	X(A5) Légnyo. (hPa)	X(A6) Víz mély. (m)	X(A7) Csali típusa	X(A8) Napszak	X(A9) Víz hőm.	X(A10) Lég hőm. (°C)	X(A11) Szél seb.	X(A12) Szél ir.	X(A13) Légnyo. (hPa)	X(A14) Víz mély. (m)	X(A15) Csali típusa	X(A16) Napszak
S1	4767,3	7481,9	5755,6	1236	5726,7	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	1512	0	1420,4
S2	4767,3	7481,9	5755,6	1236	2082,3	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	1512	0	1420,4
S3	4767,3	7481,9	5755,6	1236	2082,3	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	1512	0	1420,4
S4	4767,3	6180,7	5755,6	1236	2082,3	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	1512	0	1420,4
S5	4767,3	6180,7	5755,6	1236	2082,3	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	1512	0	1420,4
S6	4767,3	5692,2	5755,6	489,2	2082,3	852,2	0	0	3652,8	6084,1	6107,3	0	3203,8	571	0	1420,4
S7	4767,3	5692,2	5755,6	489,2	2082,3	852,2	0	0	3652,8	4639	6107,3	0	3203,8	571	0	1420,4
S8	4767,3	5692,2	4887,8	489,2	2082,3	852,2	0	0	3652,8	4639	6107,3	0	3203,8	571	0	1420,4
S9	2790	5692,2	4887,8	489,2	2082,3	852,2	0	0	3652,8	4639	4467	0	3203,8	571	0	1420,4
S10	2790	5692,2	4887,8	489,2	2082,3	852,2	0	0	3652,8	4639	4467	0	3203,8	571	0	1420,4
S11	2790	5692,2	4887,8	489,2	2082,3	852,2	0	0	3652,8	4639	4467	0	3203,8	571	0	1420,4
S12	2790	5692,2	4179,4	489,2	2082,3	777,5	0	0	3652,8	4639	3508,3	0	3203,8	571	0	1420,4
S13	2790	5692,2	4179,4	489,2	2082,3	777,5	0	0	3652,8	4639	3508,3	0	3203,8	571	0	1420,4
S14	2790	3558,4	4179,4	489,2	2082,3	777,5	0	0	3652,8	4639	3508,3	0	3203,8	571	0	1420,4
S15	2790	3558,4	4179,4	489,2	2082,3	777,5	0	0	3652,8	4248,5	3508,3	0	3203,8	551,9	0	1420,4
S16	2790	3558,4	4179,4	489,2	2082,3	777,5	0	0	3652,8	4248,5	3508,3	0	3203,8	551,9	0	1420,4
S17	2790	3558,4	4179,4	489,2	2082,3	777,5	0	0	3652,8	4248,5	3508,3	0	3203,8	551,9	0	1420,4
S18	2790	3558,4	4179,4	489,2	2082,3	777,5	0	0	3652,8	4248,5	3508,3	0	3203,8	551,9	0	1420,4

40. ábra: Termelési függvény (részlet)

Forrás: Saját képernyőkép – Jelmagyarázat: az oszlopokban található az egyes mérési tulajdonságok és a hozzá tartozó mértékegységek, a sorokban pedig az egy-egy időpontban mért adatok, melyek mértékegysége sorszám. A fogott halak száma esetén konverzációra került végrehajtásra: db*1000

A táblázatból levonható következtetés, hogy az X(A7)-es oszlopnak megfelelő tulajdonság (Csali típus) és annak inverze az X(A15)-ös nincs hatással a fogott halak mennyiségére, mert nullákat adott vissza a rendszer. Az X(A8)-as (Napszak) és annak inverze X(A16)-os esetében más a helyzet. A táblázat 1-8 oszlopa a „minél nagyobb annál jobb”, míg a 9-16 oszlopa „minél kisebb annál jobb” logikát követi. Mivel a 8. oszlopban 0 szerepel ezért nem érvényesül az előbbi logika, viszont a 16. oszlop magas értékei a második logikai irányt értékelik reálisnak. Azaz a segédtáblázat szerinti kisebb értékek preferáltabbak, vagyis figyelembe véve a segédtáblázat (vö. 37. ábra) adatait, reggel lesz nagyobb esély az eredményre.

A számítás eredményeként becsült és tény értékeket is kapunk a halak számának és a fogott halak mennyiségére vonatkozólag egyaránt. Ennek értékeléséhez a korrelációs számításokat kell végezni, ami a korrelációs együttható kiszámítását és vizsgálatát jelenti.

A korreláció fogalma „A matematikában (a statisztikában) a korreláció jelzi két tetszőleges érték közötti lineáris kapcsolat nagyságát és irányát (avagy ezek egymáshoz való viszonyát). Az általános statisztikai használat során a korreláció jelzi azt, hogy két tetszőleges érték nem független egymástól.” (17th International Congress on Artificial Intelligence August 21-22, 2024 – Türkiye by IKSAD Institute AI alapú levezetés attribútumok fontosságáról értékelési modellek esetén Bán Gyula. Rikk János Pitlik László)

A COCO rendszer által visszaadott mátrix feldolgozása manuálisan a MS-Excel-ben a KOR-ELL() függvénnyel történt, mely segítségével kiszámításra került két cellatartomány korrelációs együtthatója, így az a tartományok között jellemző összefüggést vizsgálja, azaz van-e kapcsolat közöttük, az mekkora és milyen irányú. Értéke -1 és +1 között lehet és az alábbi módon értelmezhető:

- „0 – Ha két változó között nem mutatható ki korreláció, kapcsolat. (...)
- 0 és -1 közötti érték – Ha két változó egymással ellentétes irányban mozog. (...)
- 0 és +1 közötti érték – Ha a két változó egymással megegyező irányban mozog.”

(Hold – Korreláció – <https://hold.hu/lexikon/korrelacio-jelentese-feltetelei-ertelmezese/> – Letöltés: 2025.03.28.)

A függőség fokát a 41. ábra szemlélteti.

„ r = korrelációs együttható

<i>Korrelációs együttható értéke</i>	<i>Kapcsolati jellemző</i>
$r = 1$	<i>tökéletesen pozitív kapcsolat, függvényszerű lineáris kapcsolat</i>
$0,7 \leq r < 1$	<i>erős pozitív kapcsolat</i>
$0,2 \leq r < 0,7$	<i>közepes pozitív kapcsolat</i>
$0 < r < 0,2$	<i>gyenge pozitív kapcsolat</i>
$r = 0$	<i>nincs lineáris kapcsolat</i>
$-0,2 < r < 0$	<i>gyenge negatív kapcsolat</i>
$-0,7 < r \leq -0,2$	<i>közepes negatív kapcsolat</i>
$-1 < r \leq -0,7$	<i>erős negatív kapcsolat</i>
$r = -1$	<i>tökéletes negatív kapcsolat függvényszerű lineáris kapcsolat”</i>

41. ábra Korrelációs együttható értékelése

Forrás: Dr. Csallner András Erik: bevezetés az SPSS statisztikai programcsomag használatába.

Korrelációt számítva a COCO output értékek között azt látjuk, hogy halak száma esetében 0,36 (vö. <https://miau.my-x.hu/miau/323/horgaszat/horgaszat.xlsx> / Munka3 munkalap A914:U1214 tartomány; S1216 cella), míg a halak mennyiségének tekintetében 0,25 a korrelációs együttható (vö. <https://miau.my-x.hu/miau/323/horgaszat/horgaszat.xlsx> / Munka5 munkalap A515:U815 tartomány; S817 cella), ami azt jelenti, hogy egy közepesen pozitív kapcsolat van attribútumok és az eredmény között. Ezek az értékek a COCO-STD alapján nem meggyőzőek az attribútumok és azok eredményre tett hatást illetően.

Az input tábla alkalmas a COCO-MCM-be való bevitelre is. Ebben a rendszerben a még kevesebb megszorításokkal végzi a rendszer a számításokat, vagyis tetszőlegesen veszi számításba bementi adatok hatásmechanizmusait (vö. <https://miau.my-x.hu/miau/323/horgaszat/horgaszat.xlsx> / Munka5 munkalap AA515:AU815 tartomány; AS817 cella). Itt a becslt és tény

adatok közötti korreláció 0,93-as értéket vett fel a halak mennyiségének a tekintetében, ami azt jelenti, hogy a lehetőség benne van az adatvagyonban. Az elemzése alapján minden adva van ahhoz, hogy az adatbázisba betárolt adatvagyonból – átalakítva azt megfelelő input adatamennyiséggé – egy cURL hívással reális predikciót kapjunk a várható eredményről a COCO segítségével előre, körülbelül 6-12 óra időtávra.

3.6. A fejlesztési perspektívák rövid elemzése

A fejezet tartalmazza a jövőbeli lehetséges fejlesztési irányokat, elképzeléseket, és azok értékeit a projektben, mint például a mért adatok mennyiségének a növelése, az adatgyűjtés hatékonyságának javítása és a felhasználói élmény bővítése hozzáadott érzékelőkkel és funkciókkal. A fejlesztésben nagy hozzáadott értéket képvisel a PH érték és a víz oxigén szintjének a mérése, a LoRa technológia alkalmazása nagyobb távolságok elérése végett, a rendszer energia fogyasztásának a minimalizálása és ezzel együtt az energiaellátásának kiépítése.

3.6.1. PH szenzor

Az eszközöm hatékonyságának növelése érdekében PH érték szenzort tervezek applikálni a Slave eszközbe, ami nagyban segíti a horgászokat a csali jellegének pontosításában „*A magas fehérjetartalommal rendelkező csalogatóanyagok az enyhén savas közegben fejtik ki leginkább hatásukat, addig a szénhidrátok kioldhatósága enyhén lúgos közegben tekinthető optimálisnak.*” (Haldorado –Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 4. rész – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-4-resz-a3700> – Letöltés: 2025.03.18.)

Ugyan a halak táplálkozásának jellege nehezen kalkulálható, függ a víz hőmérséklettől, vízoxigénszinttől, a halak szoktatásától és vízterülettől. A PH szenzor segít támpontot adni ebben. A nagy csalogató cégek termékeiket ugyanis főleg PH értékük alapján állítják be, és tökéletesítik. A PH érték mérése az elengedhetetlen adatok egyike lesz a rendszerem mesterséges intelligencián alapuló előrejelzésében.

3.6.2. A víz oldott oxigénszintjének mérése

A víz oldott oxigénszintje is fontos szerepet játszik a halak táplálkozásában, ugyanis csökkenti a halak anyagcseréjét és étvágyát. Merészen kimondható, hogy minél magasabb a víz oxigén szintje, annál nagyobb az esélyünk egy sikeres horgászatra. A víz oxigénszint mérő szenzor hasznos fejlesztés lehet a projekt számára, ugyan akkor ez valamennyire kikövetkeztethető a meglévő szenzorjaim értékeiből is. Minél melegebb a víz, annál kevesebb oxigént képes oldani,

a halaknak pedig annál nagyobb az oxigén igénye, ezért melegebb időben a felsőbb vízrétegekben helyezkednek el.

(Haldorado – Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-1-a-homerseklet-a-legnyomas-es-a-frontok-a2722?utm> – Letöltés: 2025.03.20.)

3.6.3. Vízhőmérséklet mérésének a kiterjesztése különböző helyeken, különböző vízmélységben

Napjaink fejlett technológia világában, már léteznek olyan mérőeszközök mely távolról, sőt akár víz rétegenként képesek mérni a víz hőmérsékletét. Ez a szenzor közkedveltnak számít a horgászok körében, ugyanakkor nagyon kevés ember engedheti meg magának az eszközre való beruházás, ugyanis még a horgászathoz mérten is meglehetősen drágának számít. Az eszközt legtöbbször radarba vagy szonárba szerelve forgalmazzák, főként csónakokhoz és etetőhajókhoz. Egy ilyen eszköz beszerelése a rendszerembe számottevően javítaná az eszközöm által szolgáltatott adatok részletességét, bár a magas ára végett egyelőre a beszerzése nem opcionális. Vannak viszont mérési alternatívák, amelyekkel kivitelezhető lehet a precíz részletesség. Tó vízfelületétől függően ki kell helyezniünk egy adott számú mérőeszközök a tó különböző pontjaira, melynek küldött jeleit egy pontban kell gyűjtenünk. Többféle megoldás is van ezen fejlesztési ötlet megvalósításához, melynek majdnem mindegyike a mesterséges intelligencia projektembe való implementálásán alapszik. A következő adatok megszerzéséhez lenne szükség pontos értékek meghatározásához:

1. Mérési adatok:

- Vízfelszíni hőmérséklet (a tó különböző pontjain)
- Vízmélységi hőmérséklet (pl.: 1 méterenként a tófenéig)
- Mérések távolsága (pl.: a parttól a túloldalig, minden méteren)

2. Környezeti tényezők

- Léghőmérséklet (a vízfelszínre gyakorolt hatás miatt)
- Napsugárzás erőssége (mennyire melegíti a vizet)
- Szélsebesség és -irány (a felszíni víz keveredését befolyásolja)
- Levegő páratartalma (befolyásolhatja a párolgás miatti hőveszteséget)

3. Hidrológiai tényezők

- Vízmélység és mederalakulatok (befolyásolja a víz áramlását és hőeloszlását)
- Áramlások és turbulencia (pl.: folyók befolyása, szél miatti áramlások)

- Csapadék és párolgás (friss víz beáramlása vagy vízvesztés)

4. Mérési módszerek

- Szenzorhálózat: Több mérési pont (pl.: bójákra vagy csónakra szerelt hőmérők)
- Távérzékelés: Hőkamerás drónok vagy műholdas adatok
- Fizikai mérések: Hőmérők leengedése különböző mélységekben

3.6.4. LoRa kommunikáció kialakítása nagyobb távolságok elérése érdekében;

Valós helyzetben a tavaknál a jelenlegi mérési távolság (a Master és Slave eszköz biztos adat-továbbítása kb. 140 méterig történt meg, amit tapasztalti úton határoztam meg) elenyészőnek számít. Ennek a távolságnak a növelése kritikus része a fejlesztésnek. Ezt a célt szolgálja LoRa kommunikáció kialakítása.

„A LoRa egy nagy hatótávolságú és kis teljesítményű vezeték nélküli kommunikációs rendszer, amely kis mennyiségű adatot képes nagy távolságokra küldeni.” (KnowHOW – Mi az a LoRa és a LoRaWAN technológia? – <https://knowhow.distrelec.com/hu/telekommunikacio/mi-az-a-lora-es-a-lorawan-technologia/> – Letöltés: 2025.03.22.) Ezen adottságai teszik közkedvelt alternatívává az IoT iparágakban.

3.6.5. Tápellátás fejlesztése napelemes rendszer kialakításával

Jelen esetben tápellátást a Slave eszköznél akkumulátor biztosítja a Master eszköznél pedig hálózati adapter. Az energiafelhasználás csökkentése érdekében jövőbeli cél lehet a napelemes akkumulátoros energiaellátás biztosítása mind a két eszköz esetében, mivel az áramfelvételük 0.15A. A fejlesztés reális rövidtávú cél mind a két esetben.

3.6.6. Szoftverfejlesztés az energiatakarékosság elérése érdekében

A mikrovezérlők alkalmasak a mélyalvás üzemmódra. Ez azt jelenti, hogyha feltételezzük, hogy 30 percenként szeretnék mérést végezni, adatot küldeni, akkor a két aktív időszak között a készülék mélyalvás üzemmódja aktiválható, ezáltal jelentős energiafelhasználás csökkenést érhetünk el. Ez a jövőbeli programfejlesztés során relatíve könnyen implementálható.

3.6.7. Hardveres fejlesztés élőkép stream-elése a felhasználók felé

Az élő kamerakép kiépítése nem csak a horgászat szempontjából kellemes hozzáadott érték. Az időjárási viszonyok GUI-n való vizualizálása egy igen értékes pont. A tópartra telepített kamerán keresztüli kép is nagyban hozzájárulhat a sikerünkhöz, feltéve, hogy nemcsak az időjárás viszonyokat tudjuk felmérni a saját szemeinkkel, hanem az esetleges ott lévő horgászok

tömegét is jobban be tudjuk azonosítani. Arról nem is beszélve, hogy egy horgásznak már csak a tópart helyzete, látványa is kellemes élményekre kerít sort.

3.6.8. Esőszenzor kiépítése

A projektekben használt mikrovezérlők képességeinek csak egy töredéke van felhasználva. Az eszközre számos érzékelhető csatlakoztatható lenne még, melyhez csak a program átírására és a fizikai illesztésre lenne szükség. Ilyen megoldás lehet többek között az eső szenzor beépítése, amely segíthetné a horgászok döntéshozatalát a komfort érzés növelése érdekében.

3.6.9. Online lehetőség biztosítása a felhasználóknak online fogási eredmények regisztrálása, annak érdekében, hogy a MI pontosabb predikciót készítsen

A mesterséges intelligencia beépítése a projektembe egy alapvető cél, amit rövidtávon tűztem ki magam elé. Az időjárási tényezők hatalmas befolyással vannak a halak táplálkozási szokásaira, ezzel együtt fontos támpontot tudnak adni a horgászoknak a halak eredményesebb becserkészése érdekében. Ámbár az adatok kiértékelése fontos és hasznos dolog, nem minden horgász ért ezen adatok megfelelő felhasználásához. Itt jönne közbe az MI, ugyanis megfelelő adatbázissal kialakítható és kiértékelhető, hogy a jelenlegi időjárási behatásokhoz milyen taktika társítása lehet a legsikeresebb. Ezzel a kezdő horgászokat is segíteni lehetne. Mivel a halak viselkedése és étkezési szokásai nagyban függenek a vízterületektől is, ezért a vízterületeket külön kell kezelni. Jelenleg ez sajnos kapacitásaimat tekintve megoldhatatlannak ítélnélhető, ugyanis a mesterséges intelligencia specifikus programozásához, elengedhetetlen tényezője és feltétele az adatbázis betáplálása. Az adatbázis megfelelő feltöltéséhez nem csak a mérőeszközöm által nyújtott adatok kellenek, hanem a lelkes horgászok együttműködésére is szükség van. Sajnos az általam készített eszköz ugyan méri az időjárási tényezőket, azonban a fogott halak mennyiségét, súlyát, fajtáját, fogott mélységét csali jellegét nem tudja behatárolni. Egy megfelelően működő kiértékeléshez pedig ezen adatok is létfontosságúak. A hosszútávú célom, a mesterséges intelligenciát a mért adatok pontosítására felhasználni.

4. Kérdőív

A szakdolgozatban többféle kvantitatív és kvalitatív adatgyűjtési módszert is használtam.

„A kvalitatív eszközök együttes jellemzője, hogy az innen származó megállapítások nem számszerűsíthetők”. (Sug Galaxis Fogalomtár – Kvalitatív Kutatás <https://galaxis.startupguide.hu/kvalitativ-kutatas> – Letöltés: 03.30.).

Ezek közül a megfigyelést, a terepkutatást alkalmaztam, mivel hosszú éveket töltöttem a horgászok között és jelenleg is egyik kedvelt hobbim.

„A legfontosabb kvantitatív módszerek a kísérlet, a kérdőíves adatfelvétel, a már létező statisztikák vagy adminisztratív adatbázisok elemzése, valamint a kvantitatív szöveg- és tartalomelemzés.” (Hajdu Gábor – A kvantitatív és a kvalitatív társadalomtudományi kutatás módszerei)

Ezek közül a kérdőív összeállítását találtam célszerűnek, mert ezzel gyors és széleskörű közvéleménykutatást tudtam megvalósítani. Fontos szempont volt, hogy kevés számú, célorientált, maximum 12-14 kérdést tegyek fel, hiszen saját tapasztalatom az, hogy én sem töltök ki hosszú, bonyolult kérdőíveket. Mivel a célközösségem a horgászok, akik többségét férfiak alkotják, s akik válaszolni sem nagyon szeretnek, ezért a kérdőívemet, zárt, feleletválasztós kérdésekből állítottam össze.

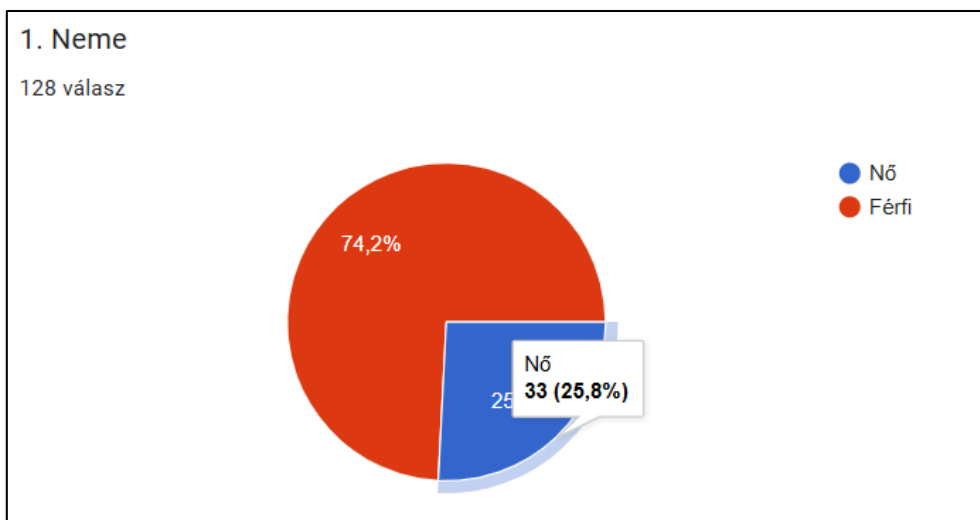
A horgászoldalakon megosztva, pár hét alatt összegyűlt a 128 kitöltő, amikor is lezártam a válaszadást, nem fogadtam több visszajelzést.

„Ha már megtörtént az összegyűjtött adatok feldolgozása, következhet az elemzés. A kutatási eredmények elemzése során vonhatunk le következtetéseket, melyek vagy megerősítik előzetes hipotéziseinket, vagy újabb kutatásra ösztönöznek bennünket.” (Earl Babbie, 2003 – A társadalomtudományi kutatás gyakorlata, Balassi Kiadó, Budapest)

Kérdéseim kapcsolódnak a tanult tantárgyaimhoz, a generációs összetétel, a mesterséges intelligencia és a modern eszközök használata révén, de kitérnek a horgászat módjára és a kifejlesztett időjárési adatgyűjtő állomás használatára is.

Az előzetes hipotéziseim szerint még mindig a férfiak horgásznak többször és manapság már gyakran használnak valamilyen digitális eszközt a horgászat során. A mesterséges intelligencia használata még nem terjedt el a magyar horgászok körében és ha használják is néhányan, akkor ők biztosan a fiatal korosztály tagjai. Az időjárési tényezők fontosságával nincs teljesen tisztában a horgászok egy része, néhány népi bölcsességet kivéve, mint, hogy esőben kapnak a halak, de egyre többen ismerik a többi befolyásoló tényezőt.

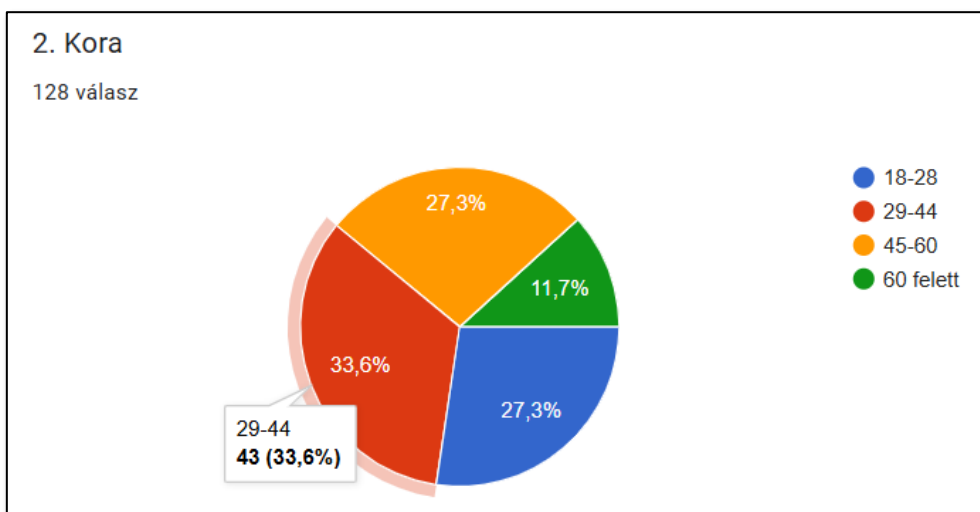
Nézzük a válaszokat a hipotézisek tükrében.



42. ábra: Nemek aránya

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók nemek közötti megoszlása százalékban kifejezve

A válaszadók többsége valóban férfi, ami csak a kérdőív témájának köszönhető, hiszen az ismerőseim kérdőíveit többségben nők töltötték ki, ők jobban hajlandóak erre áldozni az idejükből. A 128 kitöltőből 95 fő, 74,2 % férfi és 33 fő nő. Örömmel töltött el a női kitöltők aránylag nagy száma, hiszen egy másfajta szemléletet hoznak a téma elemzéséhez (vö. 42. ábra).

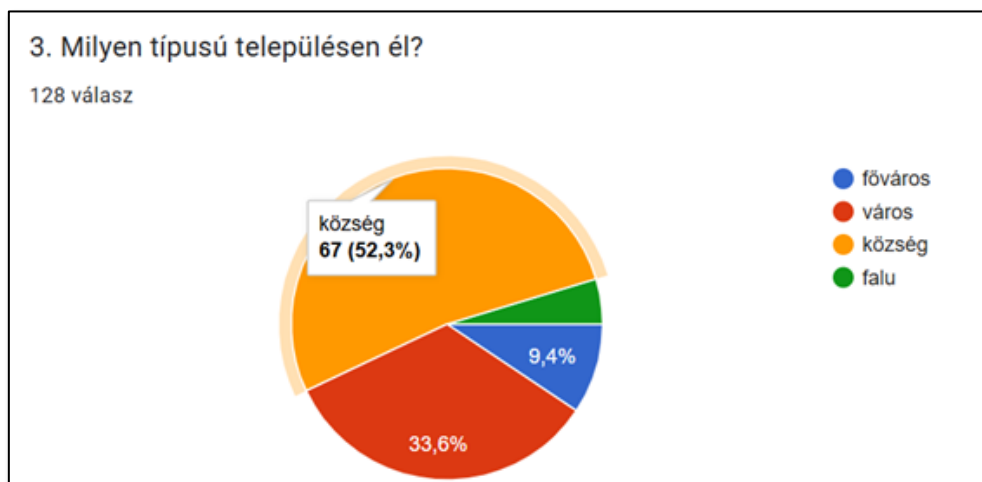


43. ábra: Életkor aránya

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók kor szerinti megoszlása százalékban kifejezve

A különböző generációk szép számmal képviseltették magukat a válaszadók között. Tehát a horgászat minden generációt érdekel. A Baby Boomerek közül (1946-1965), akik ma 60 év felettiek, 15 fő töltötte ki. Az X generáció (1966-1979) tagjai, a 45-60 év közöttiek közül 35-en válaszoltak. Az Y generáció (1980-1995) tagjai, a 29-44 év közöttiek voltak a legaktívabbak, a 128 főből negyvenhárom, azaz 33,6 % -a tartozott ebbe a generációba a válaszadók közül. A Z

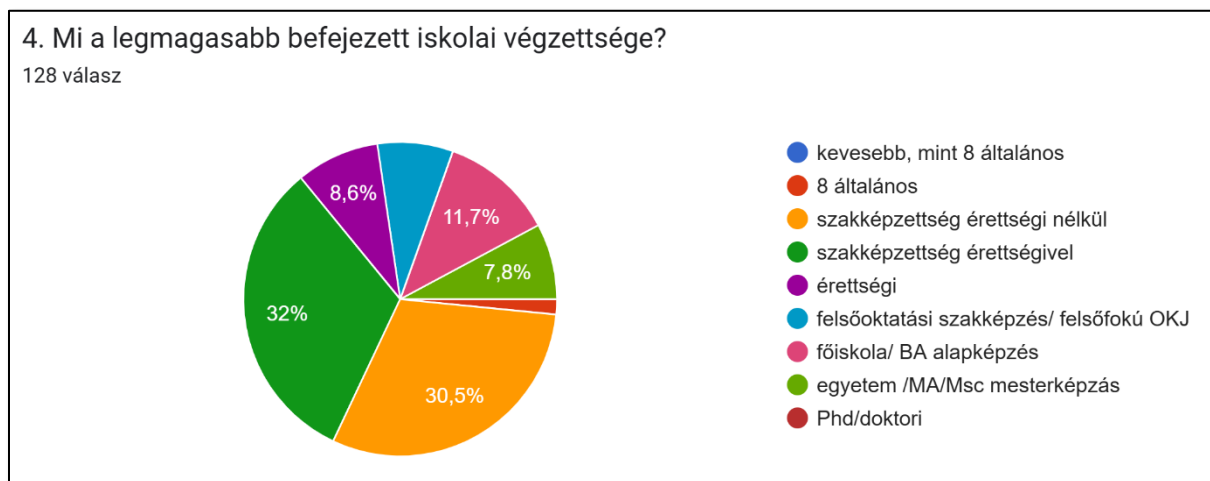
generáció (1996-2007), a 18-28 évesek, 35 fővel képviseltették magukat. Az Alpha-generáció tagjai még nem töltötték be a 18 évet, ezért nem tartoztak a célcsoportba (vö. 43. ábra).



44. ábra: Települések aránya

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók lakhely szerinti megoszlása százalékban kifejezve

A válaszadók többsége községekben él (67 fő), de jelentős a városokban (43 fő) és a fővárosban (12 fő) élők együttes aránya is. Kisebb falvakban mindössze 6 fő él. Ez azért érdekes, mert a falvakban és községekben lakókat még mindig természetközeli környezet jellemzi. Több természetes és mesterséges víz található a lakóhelyük közelében, így könnyebben és gyakrabban van lehetőségük horgászni (vö. 44. ábra).



45. ábra: Iskolai végzettségek

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók iskolai végzettség szerinti megoszlása százalékban kifejezve

Az iskolai végzettségnél, a két véglet nem található a válaszadók között. Sem a 8 általánosnál kevesebbet, sem pedig doktori képzést végzett személy nem válaszolt. Legtöbben érettségizett szakképzettek vettek részt a kitöltésben (41 fő), őket követte az érettségi nélküli szakképzettek (39 fő). 11 fő érettségizett szakképzés nélkül. A legkevesebb (2 fő) a 8 általánost végzett

személy volt. 35 fő valamilyen felsőfokú végzettséggel rendelkezik (FOSZK/BA/MA). A digitális ismeretek nagyban függhetnek az iskolázottság fokától is, nemcsak a kortól és generációbeli hovatartozástól. (vö. 45. ábra)



46. ábra: Vízterület típusok aránya

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók preferált vízterület szerinti megoszlása százalékban kifejezve

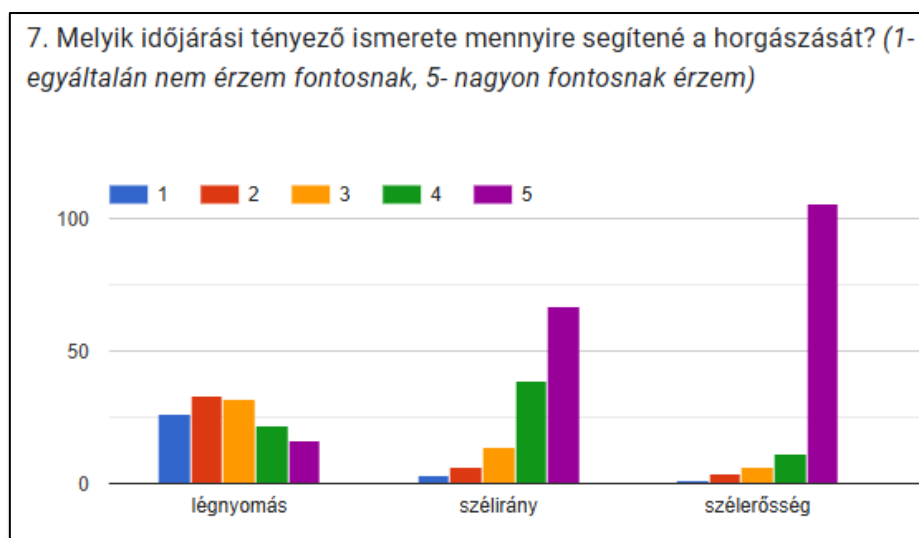
A legtöbb válaszadó a tavakat részesíti előnyben (114 fő) és itt horgászni szeretnek első sorban (111 fő). Az eszközöm használatának módját kis mértékben érintheti a vízterület típusa. (vö. 46. ábra)



47. ábra: Parton töltött tevékenység

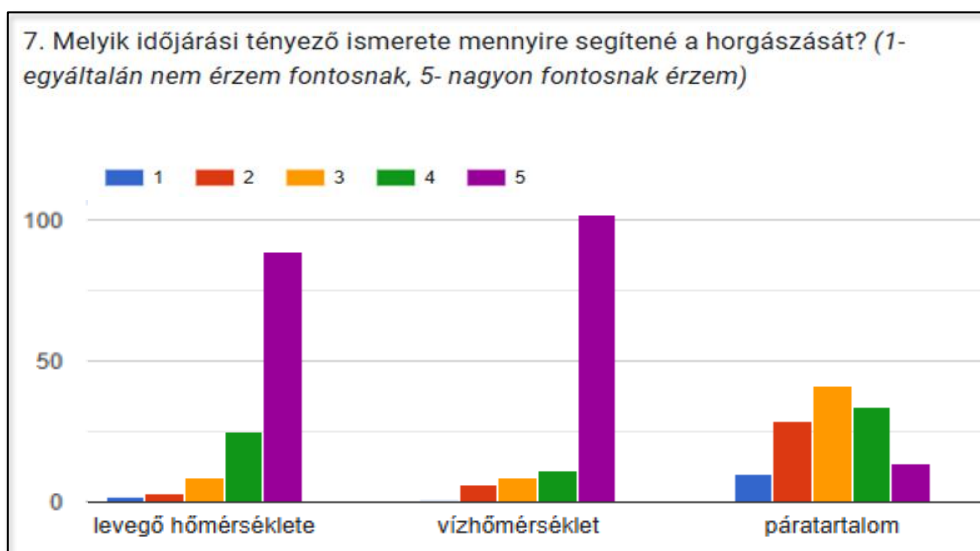
Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók vízparton töltött tevékenységek közötti megoszlása százalékban kifejezve

A fürdés, a vízisportok és a kirándulás is szóba jöhetnek egy-egy tóparton a válaszolók 13,3%-a szerint. Természetesen csak, ha megfelelő az idő. Bár a kifejlesztett rendszer célközönsége a horgászok, más célcsoport is hasznosíthatja az eredményeket. (vö. 47. ábra)



48. ábra: Időjárási tényezők 1.

Forrás: Saját képernyőkép – Jelmagyarázat: 1-től 5-ig skálán értékelték a válaszadók az időjárási tényezők ismeretének fontosságát, ahol az 1-es jelentette *nem fontos* attribútumot, az 5-ös pedig a *nagyon fontos*.

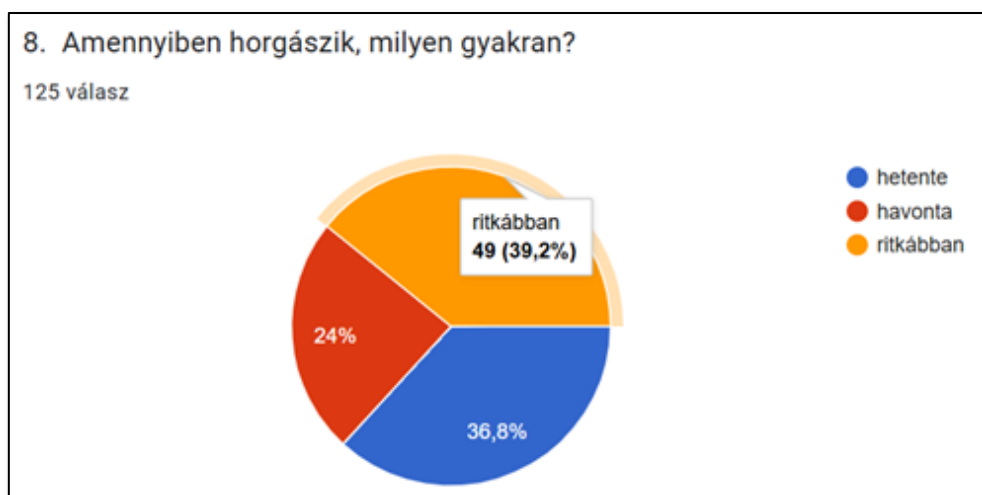


49. ábra: Időjárási tényezők 2.

Forrás: Saját képernyőkép – Jelmagyarázat: 1-től 5-ig skálán értékelték a válaszadók az időjárási tényezők ismeretének fontosságát, ahol az 1-es jelentette *nem fontos* attribútumot, az 5-ös pedig a *nagyon fontos*.

Az időjárási tényezők közül magasan a szélerősség adatait tartják a legfontosabbnak, de levegő- és vízhőmérséklet is az érdeklődés középpontjába került. A szélirány sajnos nem került be az első három helyezett közé. A páratartalmat és a légnyomást kissé alulértékelték, pedig szerepük a sikeres horgászatban igen jelentős.

A metán és a kén-hidrogén, vastag, oxigénhiányos iszapban keletkeznek és nagyon mérgezőek. „Főleg légnyomás csökkenésekor igen finom buborékok formájában szabadulnak fel az iszapból, és a tó vizében oldódva tömeges halpusztulást okozhatnak” (Lajkó István, 2003 – Korszerű halgazdálkodás, Szaktudás Kiadó Ház, Budapest) (vö. 48-49. ábra)



50. ábra: Horgász aktivitás gyakorisága

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók horgászatának gyakoriságát mutatja százalékban kifejezve

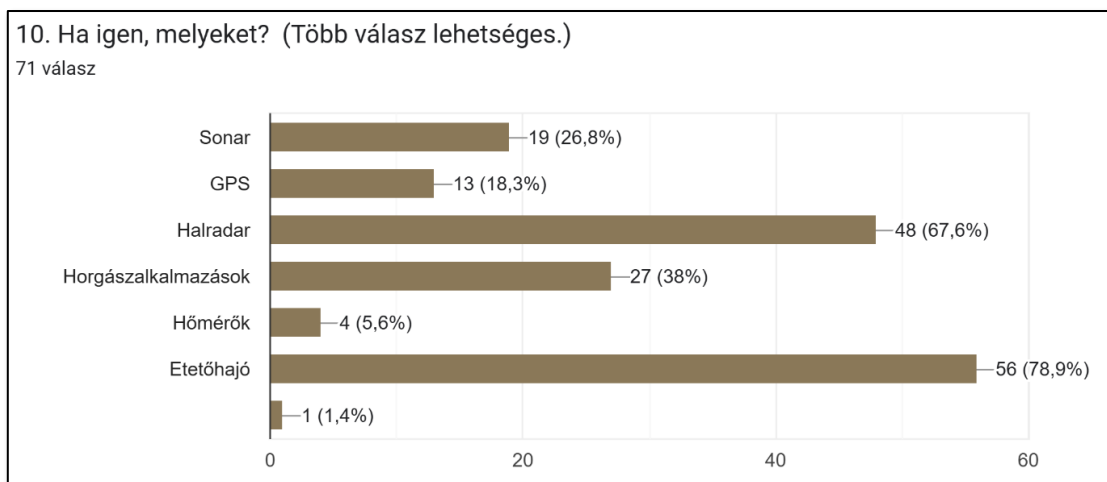
A válaszolók többsége ritkábban horgászik, mint havonta, de csaknem annyian horgásznak hetente is. Tehát gyakorolt szakemberekkel van dolgunk. Csak három kitöltő nem tartozik a horgászok közé. Ezek az adatok arról tájékoztathatnak, hogy a gyakorta horgászóknak több lehetőségük van az általam fejlesztett eszköz használatára. (vö. 50. ábra)



51. ábra: Modern eszközök használata

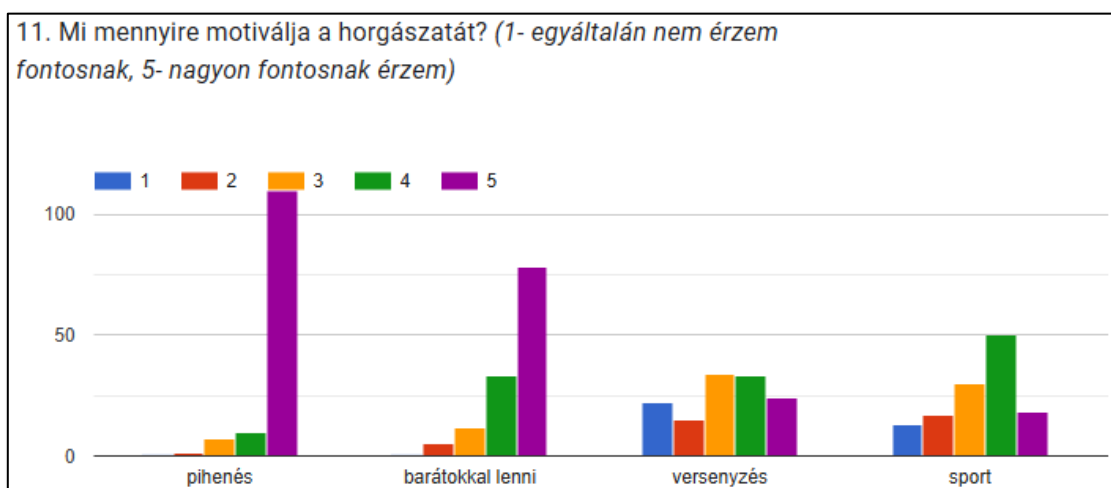
Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók megoszlását mutatja százalékban kifejezve, hogy használnak-e a horgászat során modern eszközöket (vö. 48. ábra)

Meglepően sokan, a kitöltők több, mint fele, használ a horgászathoz modern, technikai eszközöket. Közülük az etetőhajó és a halradar a legnépszerűbb (vö. 51. ábra).



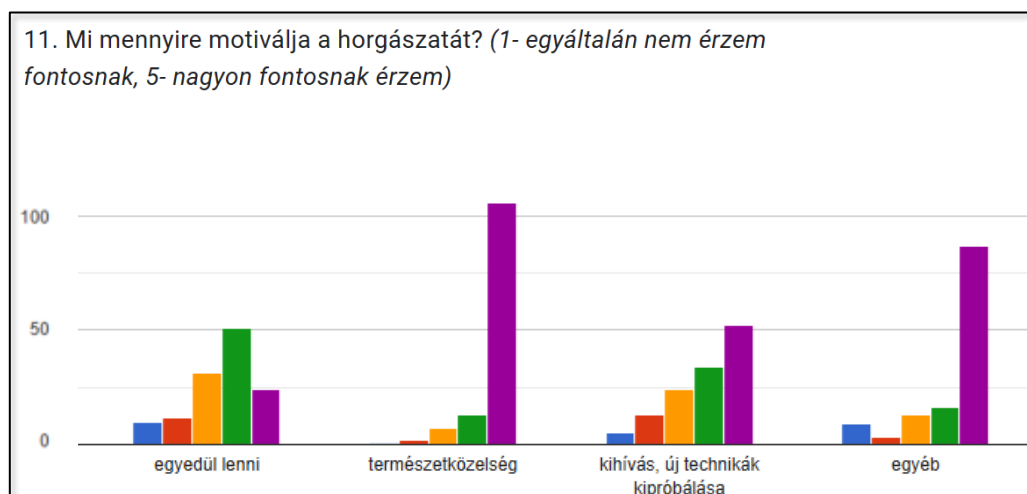
52. ábra: Modern eszközök használata a horgászatban

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók modern eszközök közötti használati megoszlását mutatja százaléokban kifejezve



53. ábra: Horgász motiváltság 1.

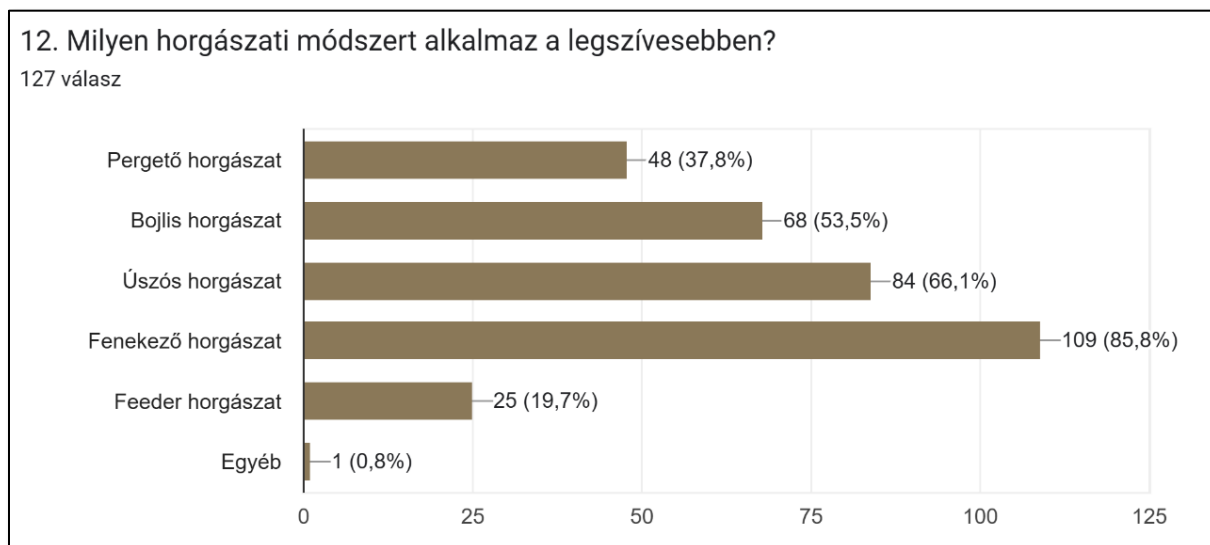
Forrás: Saját képernyőkép – Jelmagyarázat: az ábra mutatja, hogy különböző tényezőket mennyire ítélik fontosnak a horgászok, ahol az 1-es jelentette *nem fontos* attribútumot, az 5-ös pedig a *nagyon fontos*.



54. ábra: Horgász motiváltság 2.

Forrás: Saját képernyőkép – Jelmagyarázat: az ábra mutatja, hogy különböző tényezőket mennyire ítélik fontosnak a horgászok, ahol az 1-es jelentette *nem fontos* attribútumot, az 5-ös pedig a *nagyon fontos*.

A pihenés és a természet erős motiváló tényezők, viszont az egyéb tényezőknél többen a halfogás élményére gondoltak, ami a fő vonzóereje a horgászatnak és aminek sikeréhez hozzásegíthetnek a mért adatok. (vö. 52-54. ábra)



55. ábra: Alkalmazott horgászati módszer

Forrás: Saját képernyőkép– Jelmagyarázat: a válaszadók alkalmazott horgászati módszerei közötti megoszlást mutatja százalékban kifejezve

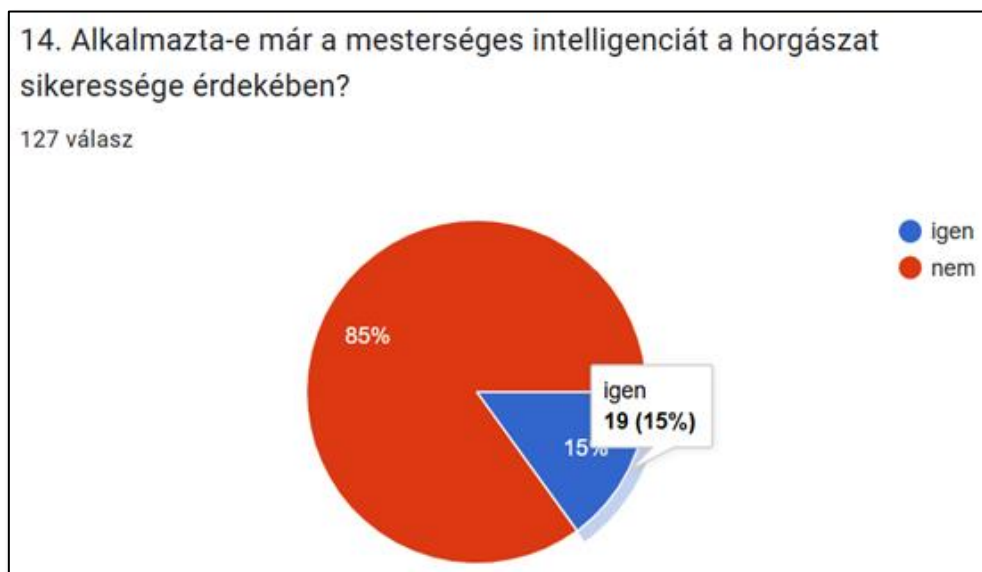
A módszerek is igen változatosak, nagyon sokan több módszert is kipróbálnak. A válaszolók többsége a fenekező horgászatot kedveli a leginkább. Ez azért fontos az ötletemet tekintve, mert minden módszernél egy kicsit másképpen, vagy másra használhatók, de mindenhol fontos szerephez jutnak az általam mért adatok (vö. 55. ábra).



56. ábra: Alkalmazás iránti érdeklődés

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók hajlandóságát mutatja, hogy használnának-e horgászathoz időjárás figyelő alkalmazást százalékban kifejezve. Figyelembe veendő a szoftverfejlesztési és licence díj kalkuláció számításánál. (vö. 1.6 alfejezet)

Ez az utolsó előtti kérdés a legfontosabb a szakdolgozatom szempontjából. Nagyon örültem neki, hogy minden kitöltő válaszolt erre a kérdésre, az is, aki nem horgászik és csak fürödni vagy a vízi sportok miatt látogatja vizeinket. 128-ból 106 fő használná az alkalmazást, ami azt jelenti, hogy hasznosnak tartaná. A tótulajdonosoknak ez jelentős bevételnövelő tényező lehetne, jó befektetést jelenthetne. A 69,5 százalékuk, 89 fő ingyenesen, de 13,3 %, 17 fő elő is fizetné. A bizonytalanok is meggyőzhetőek az eredményesség láttán. Csak 7%, mindössze 9 fő utasította el a vizsgált csoportból az alkalmazás lehetőségét (vö. 56. ábra).



57. ábra: Mesterséges intelligencia használata a horgászatban

Forrás: Saját képernyőkép – Jelmagyarázat: a válaszadók MI használatát (igen/nem) mutatja legalább egy alkalommal horgászat alkalmával, százalékban kifejezve.

19 főnek már eszébe jutott a mesterséges intelligencia használata a horgászathoz, a válaszolók korösszetétele és iskolai végzettsége miatt, nem is volt sokkal több várható. Az idősebb generáció tagjai nem nagyon szeretik és ismerik a digitális világot és a mesterséges intelligenciát. A fiatalabbak viszont tanulmányaik során már biztosan találkoztak vele

Összegezve az eredményeket, a válaszadók korbéli, nembéli és a technika iránti elkötelezettség sokszínűsége alapján feltételezhető, hogy szélesebb réteg is igényt tartana az általam kifejlesztett alkalmazásra (vö. 57. ábra).

5. Vita

Felmérésem megosztása kapcsán nemcsak pozitív megerősítés érkezett, bár az volt többségben. A visszajelzések között voltak, akik nem értették a lényegét. Voltak, akik nem hisznek az adatok értékében és volt olyan is, aki a tudományos megközelítést és a horgászat közti kapcsolatot vonta kérdőre. Sokakban felmerültek kérdések, kételyek és ennek köszönhetően különböző nézőpontokat is megismerhettem, amelyek nagy segítségemre fognak válni a későbbiekben. A fejezet célja, hogy bemutassam és megvizsgáljam a különböző, a projektemmel kapcsolatos alternatív nézőpontokat, kritikákat, melyekre a „Következtetések” fejezetben reagálok.

5.1. Az adatok megbízhatósága és pontossága

A rendszer különböző szenzorokból (vö. 3.2 alfejezet) mért adatokat dolgoz fel. A szenzorok pontossága ellenőrzött. Az adatok ember általi feldolgozása hamis eredményekhez vezethet, mert, hamis vagy véletlenszerű értékekből nem lehet helytálló következtetéseket levonni.

5.2. Az adattovábbítás és a rendszer biztonság

A rendszer alapvető működését képezi az adatok vizuális megjelenítése is. A Master eszköz, adatokat küld az adatbázisnak és a weboldalnak. Felmerül a kérdés, hogy bár ez a jelenlegi igény keretén belül működik, több tíz, száz vagy akár ezer felhasználó esetén, hogyan fog reagálni a rendszer és a kapcsolat? Készen áll a rendszer, hogy a felhasználók adatainak biztonságát garantálja?

5.3. Hozzájárul-e a horgászat sikeréhez és az élményhez?

Alapvetően a rendszerem célja, hogy a környezeti adatok mérésével és a vizualizációval segítse a horgászokat a megfelelő időpont és hely kiválasztásában. Felmerülhet a kérdés azonban, hogy gyűjtött adatok vajon tényleg hasznot hoznak-e a horgászat során, valamint, hogy a felhasználók mennyire nyitottak rá? Ha azok is, mennyire sikeresen tudják felhasználni az adatokat saját maguktól, azaz mennyire növekszik meg a halfogási esélyük, mennyire növekednek a sikereik, és milyen mennyiségben és darabszámban?

5.4. Van-e értékelhető piaci igény a rendszerre?

Egy innovatív technológiai megoldás sikere függ attól, hogy létezik-e már a termékhez hasonló eszköz, mennyire egyszerűsít le egy folyamatot és hogy van-e rá kereslet, illetve mennyire tekint a célközösség a megoldást korszerűnek, elfogadhatónak. Vajon bele illik-e a jelenlegi horgász szokásokba, trendekbe? Léteznek ugyan hasonló megoldások, mint például Fishing &

Hunting Solunar Time, ami egy időjárás, illetve fogási esélyt előrejelző alkalmazás. Nagyobb, híresebb tavakon (Balaton, Velencei tó, Tisza tó) már létezik valós idejű adatszolgáltatás, de csak a szélsébségről és a szélirányról, ami elsősorban nem a horgászok érdekeit szolgálja, hanem a strandolókét és a víziközlekedését. Ezek az alkalmazások használhatók horgászatra is, de közel sem olyan részletesek, mint a projektem.

6. Következtetés

A vita során több olyan kérdés is felmerült melyek a rendszer működésének pontosságát, biztonságát, stabilitását, értékét és piaci relevanciáját érintik. Ezen kérdések kiemelten fontosak. A projekt értékelése és a továbbfejlesztési lehetőségeinek, prioritásainak meghatározó részei a kritikák. Ebben a fejezetben, megvizsgálom a felvetett problémákat és megválaszolom őket. Az elemzés célja, hogy tisztázzam a rendszer valódi előnyeit és korlátait.

6.1. Az adatok megbízhatósága és pontossága

Tény: a szenzoroknak vannak csekély méretű kilengései a mért adatok és az esetleges tényleges értékek között, ugyanakkor ezek az eltérések olyan minimálisak, hogy nem okoznak változást az optimális taktika, hely és idő kiválasztásában. A rendszer megbízhatóságának és pontosságának biztosítása érdekében a karbantartás fontos szerepet játszik itt is. Az esetleges hibás adatok (pl.: nyáron negatív léghőmérsékleti érték megjelenítése vagy irreális adatkiugrás) adatok kiszűrése vagy a meghibásodás jelzése szűrési mechanizmusokkal (pl.: különböző vezérlési szerkezettel) és algoritmusokkal relatív egyszerűen megoldható, ezáltal tovább biztosítva a rendszer optimális működését.

6.2. Az adattovábbítás és a rendszerbiztonság

A rendszer biztonságos működése különösen fontos nagyobb (pl.: több ezres) felhasználói terhelés esetén. Az igények felmérése és a szerverkapacitás hozzáigazítása egyike azon fontos pontoknak melyeket rendszeresen (pl.: havonta) vizsgálni és módosítani kell. Jelenleg ugyan a rendszer nem tart abban a fázisában, hogy rendelkezne felhasználói fiókokkal, de a fejlesztése során, kiállítása szinte kötelezővé válik. A biztonsági protokollok, például titkosított adatkapcsolat (TLS/SSL) és megfelelő hitelesítési mechanizmusok (pl.: 2FA – Two factor authentication) alkalmazása segíthet minimalizálni a lehetséges támadásokat (pl.: adathalászat) és az adatvesztési kockázatokat (pl.: SQL injection).

6.3. Hozzájárul-e a horgászat sikeréhez és élményéhez?

A válasz rendkívül egyszerű. Igen! A rendszer által biztosított környezeti adatok maximálisan hozzájárulnak a horgászathoz a megfelelő (jó intuíciónal rendelkező) felhasználó kezében. Nagyon fontos kiemelni, hogy a jelenlegi rendszer, a szükséges adatok hiányában (horgászok által fogott halak, mennyisége és súlya), csak a környezeti tényezők mérésére képes. Az adatok feldolgozása, értelmezése és a mesterséges intelligencia alapú ajánlások megjelenítése egy hosszútávú cél, mely elérhető. Ehhez viszont szükség van a lelkes felhasználók

tisztességére és segítségére. Jelenleg a felhasználói élmény nagyban függ a felhasználó horgász tudásának kiterjedésétől, hogy mennyire értik és tudják megfelelően értelmezni az adatokat (vö. 3.4.5). Ezen problémák javításának érdekében a weboldal egy vizuálisan is megjelenő táblázata kínál majd némi támpontot a közeljövőben.

6.4. Van-e értékelhető piaci igény a rendszerre?

A rendszer piaci igénye nagymértékben függ attól, hogy a célközönség mennyire nyitott az ilyen típusú technológiai megoldásokra, illetve, hogy mennyire illeszkedik a jelenlegi horgászati trendekhez. Az igény a kérdőív alapján már pozitívnak bizonyult (vö. 4. fejezet). A rendszerterv elkészítési időszak végén kifejezetten igény felmérés céljából egy kérdőív készült, ami több (10 db) horgász közösségben is megosztásra kerül. Az eredmények még engem is megleptek. Bár a horgászok különböző korosztályból (vö. 43. ábra) származnak, különböző horgász módszereket alkalmaznak (vö. 50. ábra) és különböző szokásoknak a hívei, a visszajelzések jelentős része igen pozitív volt (vö. 56. ábra). Nem csak a rendszer használatára van igény, hanem a rendszer adatainak értelmezésében való segítségre is. A rendszerem publikálása és továbbfejlesztése kiemelkedően fontos újítás lehet a horgászok közösségében.

A vitában említett alkalmazások (Fishing & Hunting Solunar Time, nagy tavakon kihelyezett időjárás mérő), szintén széleskörű adatot szolgáltatnak (szélirány, szélsébség, léghőmérséklet, páratartalom, légnyomás, valamint holdfázis) azonban ezek nem közvetlen vízparti szenzorokból származó adatokat dolgoznak fel, hanem központi meteorológiai adatokat kérnek le. Az általam készített projekt lényege, hogy minden egyes vízpart valós idejű környezeti adatait mu-tassa meg, ami specifikusabb információ szolgáltatást nyújthat a horgászoknak.

7. Jövőkép

A dolgozatban bemutatott rendszer egy olyan innovatív megoldást kínál a horgászok számára, amely a modern technológia segítségével növeli a horgászatban elért sikereket. Az ESP32 mikrovezérlőre épített adatgyűjtő rendszer jelenlegi formájában is jelentős előnyöket biztosít, azonban a technológia további fejlesztése számos új lehetőséget rejt magában.

A jövőben a rendszer funkcionalitásának bővítése és pontosítása kulcsfontosságú lehet. Az egyik irány a mért adatok elemzése és az ezek alapján készített előrejelzések finomítása, körülbelül a következő 6-12 órára, csalitípusra és az elfoglalt horgász helyre vonatkozólag, melyet mesterséges intelligencia algoritmusok alkalmazásával lehetne elérni. Tapasztalat azt mutatja, hogy a 6-12 órás előrejelzés reális cél lehet. Az adatbázisokat folyamatosan bővíteni lehetne, ezáltal idővel egyre pontosabb előrejelzések lennének készíthetőek a halak mozgásáról, várható halfogas darabszámáról, tömegéről és aktivitásukról.

További fejlesztési lehetőséget jelent a hardvereszközök optimalizálása.

Egy interaktív mobilalkalmazás kifejlesztése lehetőséget teremtene arra, hogy a horgászok valós időben követhessék az érzékelők adatait, beépítésre kerülne egy funkció mellyel a horgászok fogásokat, csalikat és tapasztalatokat tudnának megosztani. Ez egy hatalmas segítség és egy elengedhetetlen funkció lenne az alkalmazás predikciók irányába való továbbfejlesztésében.

Végül a hosszabb távú cél a piaci bevezetés és a sorozatgyártás lenne (vö. 5.4 alfejezet). A horgászoknak, halöröknek és a vízterület fenntartóinak segítséget nyújtana a vízterület monitorozásának valós idejű nyomon követése. A vízterületek egészségének megőrzésének védelmében a fenntartók rendszeresen mérik a víz kémhatóságát, oxigénszintjét és még sok mást. Az alkalmazásomban ezek a paraméterek részben már nyomon követhetővé válnak, amivel nem csak a tavak felügyeletét könnyíthetjük meg, hanem a halak védelmét is egy új szintre emeljük. Amennyiben a fenntartóknál érdeklődés mutatkozik ezen lehetőségek után, akár egy olyan funkció is beépítésre kerülhet, mely a mért adatok alapján jelzést küld az esetleges beavatkozást igénylő veszélyekről. Ennek tudatában időben cselekedni tudnak megóva ezzel a halállomány egészségét! Ehhez azonban szükséges lenne a technológia további finomítása, valamint a felhasználói visszajelzések folyamatos figyelembevétele. A fejlesztésem akár lényegesen hozzájárulhat horgászati technológia fejlődéséhez, és az adat vezérelt horgászat megközelítéséhez is.

8. Összefoglalás

A fejezetben a dolgozat magyar, illetve angol nyelvű összefoglalása kerül bemutatásra.

8.1. Rezümé – HU

A projekt során sikeresen integráltam a kiválasztott szenzorokat (DS18B20, BME280, RS-FXJT-N01, RS-FSJT-No1) az ESP32 mikrovezérlővel, majd elkészítettem valamennyi programot az adatok begyűjtésére, feldolgozására, tárolására és vizualizálására. A rendszer képes valós időben adatokat küldeni egy webserverre (<https://to.progsol.hu/>), ahol a felhasználók grafikus felületen keresztül tekinthetik meg azokat.

A mérési adatok hosszabb távú elemzése segíthet megérteni egy adott tó halállományának a viselkedését (vö. https://miau.my-x.hu/miau/322/horgasziker_adatgyujtes.xlsx), amelyek a horgászok számára hasznos információkkal szolgálhatnak a megfelelő időpontok és helyszínek kiválasztásában.

A dolgozat készítése közben felfedeztem a további fejlesztési lehetőségeket is, mint például a PH-szenzor integrálása, a víz hőmérséklet mérésének kiterjesztése különböző mélységekben, vagy a LoRa-kommunikáció alkalmazása nagyobb hatótávolság elérése érdekében. Emellett a tápellátás napelemes rendszeren keresztül történő biztosítása is egy ígéretes irány lehet.

A szakdolgozat bizonyítja, hogy az IoT-technológiák alkalmazása a horgászatban nemcsak innovatív megoldásokat eredményezhet, hanem valós hasznosságot is nyújthat a felhasználók számára. A fejlesztett rendszer hozzájárulhat a modern horgászati technikák fejlődéséhez, valamint további kutatások alapjául szolgálhat a környezeti tényezők és a halak viselkedésének vizsgálatában.

8.2. Rezümé – EN

During the project, I successfully integrated the selected sensors (DS18B20, BME280, RS-FXJT-N01, RS-FSJT-No1) with the ESP32 microcontroller and developed all necessary programs for data collection, processing, storage, and visualization. The system is capable of transmitting data in real time to a web server <https://to.progsol.hu/>, where users can view it through a graphical interface.

The long-term analysis of the measurement data can help understand the behavior of a lake's fish population (vö. https://miau.my-x.hu/miau/322/horgasziker_adatgyujtes.xlsx), providing valuable insights for anglers in selecting the optimal times and locations for fishing.

While working on this thesis, I also identified further development opportunities, such as integrating a pH sensor, extending water temperature measurements to different depths, or utilizing

LoRa communication for extended range. Additionally, implementing a solar-powered energy supply could be a promising direction.

This thesis demonstrates that the application of IoT technologies in fishing can not only lead to innovative solutions but also provide real value to users. The developed system can contribute to the advancement of modern fishing techniques and serve as a foundation for further research on the relationship between environmental factors and fish behavior.

9. Felhasznált irodalom

Arduino hivatalos oldala:

Language Reference – Letöltés: 2025.03.13. – <https://docs.arduino.cc/language-reference/en/functions/math/map/>

Az Európai Unió Hivatalos Lapja:

A tanács 834/2007/EK rendelete – Letöltés: 2025. 04.05. – <https://eur-lex.europa.eu/legal-content/HU/TXT/HTML/?uri=CELEX:32007R0834>

Butor Klára:

Intézményi sport stratégia és hallgatói jól - lét program – Kodolányi János Főiskola PIQ&LEAD képzési modelljében Letöltés: 2025. 04. 05. – <https://www.kodolanyi.hu/kutatas-fejlesztes/kiadvanyaink>

In: Schumann Júlia: A közösségfejlesztés útjai, 2016 – Kodolányi János Főiskola

Centroszet hivatalos oldala:

Feszültségosztás – Letöltés: 2025.03.10. – https://centroszet.hu/tananyag/elektro_bovitett/343_feszltsgosztts.HTML

COCO:

Component-based object comparison for objectivity Letöltés: 2025.03.30. – <https://miau.my-x.hu/myx-free/>

Drexler Dóra, Varga Korinna, Allacherné Szépkuthy Katalin:

Az ökológiai gazdálkodás 2027-ig tartó cselekvési terve, agrarium7, 2022/07/03 – Letöltés: 2025.04.05. – <https://agrarium7.hu/cikkek/2190-az-okologiai-gazdalkodas-2027-ig-tarto-cselekvesi-terve>

Erik Brynjolfsson:

Artificial Intelligence and the Future of Work Congressional Testimony – Letöltés: 2025.03.28. – <https://www.congress.gov/116/meeting/house/109981/witnesses/HHRG-116-SY15-Wstate-BrynjolfssonE-20190924.pdf>

ESP-IDF oldala:

ESP-NOW – Letöltés: 2025.03.01. – https://demo-dijiudu.readthedocs.io/en/latest/api-reference/wifi/esp_now.HTML

Espressif hivatalos oldala:

ADC – Letöltés: 2025.03.13. – <https://espressif-docs.readthedocs-hosted.com/projects/arduino-esp32/en/latest/api/adc.HTML>

ESPRESSIF hivatalos oldala:

ESP-NOW – Letöltés: 2025.03.28. – https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_now.HTML

Európai Információs Technológiai Hitelesítési Akadémia hivatalos oldala:

Mik azok a CSRF tokenek, és hogyan védenek a CSRF támadások ellen? – Letöltés: 2025.03.22. – <https://hu.eitca.org/cybersecurity/eitc-is-wasf-web-applications-security-fundamentals/server-security/server-security-safe-coding-practices/examination-review-server-security-safe-coding-practices/what-are-csrf-tokens-and-how-do-they-protect-against-csrf-attacks>

Halcatraz hivatalos oldala:

Az időjárás és a halak táplálkozása – Letöltés: 2025.03.20. – https://www.halcatraz.hu/blog/2012/06/26/az-idojaras-es-a-halak-taplalkozasa?srsltid=AfmBO-ornd0EaaEemwAVWS9MaO0dEMIJWI9_bMyUOWFDfZi2IO48oL54f&utm

Halcatraz hivatalos oldala:

Elméletből gyakorlatba - Method gyorstalpaló –Letöltés: – 2025.03.16. – <https://www.halcatraz.hu/blog/2022/08/01/elmeletbol-gyakorlatba-method-gyorstal-palo>

Halcatraz hivatalos oldala:

Pontyozás hideg vízben – Letöltés: 2025.03.16. – <https://www.halcatraz.hu/blog/2023/01/11/pontyozas-hideg-vizben?srsltid=AfmBOooPs88751zWe3aUS-lxEZIo7BHGRArFTIh91ZH-XMbIZWxZDm8Tw>

Haldorádó hivatalos oldala:

Az abiotikus környezeti tényezők hatásai a pontyok táplálkozására 1. A hőmérséklet, a légnyomás és a frontok – Letöltés: 2025.02.20. – <https://www.haldorado.hu/topikok/horgasziskola-kezdoknek-c10/pontysuli-c24/az-abiotikus-kornyezeti-tenyezok-hatasai-a-pontyok-taplalkozasara-1-a-homerseklet-a-legnyomas-es-a-frontok-a2722?utm>

Horváth Balázs, Pestiné Rácz Éva Veronika:

Ökológia – Letöltés: 2025. 04.04. – https://dtk.tan-konyvtar.hu/xmlui/bitstream/handle/123456789/8106/0021_Okologia.pdf?sequence=1&isAllowed=y

Juhász Róbert honlapja:

Mikrovezérlők története – Letöltés: 2025.03.04. – <https://jrobi.hu/mikrovezerlok/mikro-vezerlok-toertenete>

Kinsta honlapja:

TLS vs SSL: What's the Difference? Which One Should You Use? – Letöltés: 2025.03.22. – <https://kinsta.com/knowledgebase/tls-vs-ssl/>

Kodolányi János Egyetem:

A KJE BPROF-képzésének szakdolgozati specialitásai — Letöltés: 2025.02.12. – https://www.kodolanyi.hu/ugyfelszolgalat/images/tartalom/File/informatika_tsz/kje_bprof_szakdolgozat_specialitasok.docx

Kodolányi János Egyetem:

Szakdolgozati szabályzat– Letöltés: 2025.02.12.–https://www.kodolanyi.hu/ugyfelszolgalat/images/tartalom/File/informatika_tsz/kje_egyseges_szakdolgozati_szabalyzat_2021marc17_hatalyos-1.pdf

Kodolányi János Egyetem:

Szakdolgozat minta – Letöltés: 2025.02.12.–https://www.kodolanyi.hu/konyvtar/images/tartalom/File/Honlapra/Feltoltes/szakedoli_minta.pdf

KnowHow oldala:

Mi az a LoRa és a LoRaWAN technológia? – Letöltés: 2025.03.22. – <https://knowhow.distrelec.com/hu/telekommunikacio/mi-az-a-lora-es-a-lorawan-technologia/>

Last Minute Engineers:

Analog-to-digital Converter; – Letöltés: 2025.03.13. – <https://lastminuteengineers.com/esp32-basics-adc/>

Last Minute Engineers:

Interface BME280 Temperature, Humidity & Pressure Sensor with Arduino – Letöltés: 2025.01.15. – <https://lastminuteengineers.com/bme280-arduino-tutorial/>

Last Minute Engineers:

Create A Simple ESP32 Weather Station With BME280 – Letöltés: 2025.01.15. – <https://lastminuteengineers.com/bme280-esp32-weather-station/>

Last Minute Engineers:

Display Values of Multiple DS18B20s on ESP8266 NodeMCU Web Server – Letöltés: 2025.01.15. – <https://lastminuteengineers.com/multiple-ds18b20-esp8266-nodemcu-tutorial/>

Last Minute Engineers:

Interfacing DS18B20 1-Wire Digital Temperature Sensor with Arduino – Letöltés: 2025.03.13. – <https://lastminuteengineers.com/ds18b20-arduino-tutorial/>

Makay oldalal:

Mi a különbség az IDS és az IPS között? – Letöltés: 2025.03.22. – <https://makay.net/ids-ips-kulonbseg-intrusion-detection-prevention-system>

Malatyinszki Szilárd:

Az új generáció, mint erőforrás – In: Multidiszciplináris kihívások, sokszínű válaszok, ISSN 2630-886X, 2020. (9. évf.), 1. sz. — Letöltés: 2025. 04. 05. – http://epa.oszk.hu/03400/03448/00019/pdf/EPA03448_multidiszciplinaris_2020_1_090-101.pdf

Miau hivatalos oldala:

Matematikai alapok (ea) – Letöltés: 2025.03.19. – https://miau.myx.hu/mediawiki/index.php/BROFtant%C3%A1lrgyak#Matematikai_alapok_.28ea.29

Miau hivatalos oldala:

NAPLÓ / HÍREK – Letöltés: 2025.03.21. – https://miau.my-x.hu/miau2009/index_tki.php3?_filterText0=*knuth

Miau hivatalos oldala:

Szemelvények az információbiztonság teréről – Letöltés: 2025.03.22. – <https://miau.my-x.hu/miau/322/itb/>

Modulshop webáruház:

Leírás, paraméterek. – Letöltés: 2025.03.06. – <https://modulshop.hu/akku-shileld-5-v-es-3-v-kimenet-1-db-18650-cella-mikro-usb?keyword=Akku>

Modulshop webáruház:

Leírás, paraméterek. – Letöltés: 2025.03.06. – <https://modulshop.hu/antenna-24-ghz-wifi-bluetooth-zigbee-sma-to-ipex-1779?keyword=antenna>

Mohosz Tisza Tavi kirendeltségének hivatalos oldala:

A ponty biológiája és táplálkozása 1. rész – Letöltés: 2025.03.18. <https://sporthorgasz.eu/2020/08/26/a-ponty-biologiaja-es-taplalkozasa-1-resz>

Nash Magyarország hivatalos youtube csatornája:

Letöltés: 2025.03.16. – https://www.youtube.com/@nashtackle_hungary/videos

Nette oldala:

Sebezhetőségi védelem – Letöltés:2025.03.22. – <https://doc.nette.org/hu/vulnerability-protection>

Nemzeti Jogszabálytár:

A szerzői jogokról – Letöltés: 2025.03.22. – <https://njt.hu/jogszabaly/1999-76-00-00.24>

NKP hivatalosoldala:

Ellenállás, Ohm törvénye – Letöltés: 2025.03.19. – https://nat2012.nkp.hu/tankonyv/fizika_8/lecke_01_006

NKP hivatalosoldala:

V. Árpád népe – Letöltés: 2025.03.18. – https://nat2012.nkp.hu/tankonyv/tortenelem_5/lecke_05_040

OTRS:

Mit jelent az ITIL? – Letöltés: 2025.03.08. – <https://otrs.com/hu/blog/itsm/mi-az-til/>

Owasp oldala:

SQL Injection – Letöltés: 2025.03.22. – https://owasp.org/www-community/attacks/SQL_Injection

Rui Santos hivatalos oldala:

DIY Cloud Weather Station with ESP32/ESP8266 (MySQL Database and PHP) – Letöltés: 2025.01.15. – <https://randomnerdtutorials.com/cloud-weather-station-esp32-esp8266/>

Rui Santos hivatalos oldala:

ESP-NOW Two-Way Communication Between ESP32 Boards – Letöltés: 2025.01.15. – <https://randomnerdtutorials.com/esp-now-two-way-communication-esp32/>

Rui Santos hivatalos oldal:

ESP-NOW with ESP32: Receive Data from Multiple Boards (many-to-one) – Letöltés: 2025.02.28. – <https://randomnerdtutorials.com/esp-now-many-to-one-esp32/>

Rui Santos hivatalos oldala:

ESP32 with BME280 Sensor using Arduino IDE (Pressure, Temperature, Humidity) – Letöltés: 2025.01.15. – <https://randomnerdtutorials.com/esp32-bme280-arduino-ide-pressure-temperature-humidity/>

Rui Santos hivatalos oldala:

ESP32 DS18B20 Temperature Sensor with Arduino IDE (Single, Multiple, Web Server) – Letöltés: – 2025.01.15. – <https://randomnerdtutorials.com/esp32-ds18b20-temperature-arduino-ide/>

Rui Santos hivatalos oldala:

Getting Started with ESP-NOW (ESP32 with Arduino IDE) – Letöltés: 2025.01.15. – <https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/>

Rui Santos hivatalos oldala:

Guide for DS18B20 Temperature Sensor with Arduino – Letöltés: – 2025.01.15. – <https://randomnerdtutorials.com/guide-for-ds18b20-temperature-sensor-with-arduino/>

Rui Santos hivatalos oldala:

Installing ESP32 Board in Arduino IDE 2 (Windows, Mac OS X, Linux) – Letöltés: 2025.03.20. – <https://randomnerdtutorials.com/installing-esp32-arduino-ide-2-0/>

Sarkadi Nagy István (2002):

ITIL - az informatikaszolgáltatás módszertana – Letöltés: 2025.04.05. http://www.itsmf.hu/documents/itil2modszertan_osszefoglalo_v3.1.pdf

Sug Galaxis Fogalomtár:

Kvalitatív Kutatás – Letöltés: 2025.03.30. – <https://galaxis.startupguide.hu/kvalitativ-kutatas>

Studiopieters:

ESP32 – PinOut – Letöltés: 2025.01.15. – <https://www.studiopieters.nl/esp32-pinout/>

Webiskola:

PHP Ismeretek – Letöltés: 2025.03.13. – https://webiskola.hu/php-ismeretek/mi-az-a-php-fogalma-bemutatasa/#google_vignette

VexFox hivatalos oldala:

Mi a csrf elleni védelem? – Letöltés: 2025.03.22. – <https://vexfox.com/mi-a-csrf-elleni-vedelem>

Wikipédia (EN):

1-Wire – Letöltés: 2025.03.13. – <https://en.wikipedia.org/wiki/1-Wire>

Wikipédia (EN):

HTTPS – Letöltés: 2025.03.22. – <https://en.wikipedia.org/wiki/HTTPS>

Wikipédia (EN):

I²C – Letöltés: 2025.03.13. – <https://hu.wikipedia.org/wiki/I%C2%B2C>

Wikipédia (EN):

Internet Protocol – Letöltés: 2025.03.30. – https://en.wikipedia.org/wiki/Internet_Protocol

Wikipédia (EN):

Transport Layer Security – Letöltés: 2025.03.22. – https://en.wikipedia.org/wiki/Transport_Layer_Security

Wikipédia (HU):

IP-védettség – <https://hu.wikipedia.org/wiki/IP-v%C3%A9dets%C3%A9g> – Letöltés: 2025.03.30.

Wikipédia (HU):

MySQL – <https://hu.wikipedia.org/wiki/MySQL> – Letöltés: 2025.04.05.

Wikipédia (HU):

Objektumorientált programozás – http://hu.wikipedia.org/wiki/Objektumorient%C3%A1lt_programoz%C3%A1s – Letöltés: 2025.03.08.

Wikipédia (HU):

Operációs rendszer – Letöltés: 2025.03.19. – https://hu.wikipedia.org/wiki/Oper%C3%A1ci%C3%B3s_rendszer#cite_note-1,

Wikipédia (HU):

PHP – Letöltés: 2025.03.12. – <https://hu.wikipedia.org/wiki/PHP>

Wikipédia (HU)

Python (programozási nyelv) – Letöltés: 2025.03.08 – [https://hu.wikipedia.org/wiki/Python_\(programoz%C3%A1si_nyelv\)](https://hu.wikipedia.org/wiki/Python_(programoz%C3%A1si_nyelv))

Wikipédia (HU):

SQL-injekció – Letöltés: 2025.03.22. – <https://hu.wikipedia.org/wiki/SQL-injekci%C3%B3>

Wikipédia (HU):

Számítógép-hálózat – Letöltés: 2025.03.22. – <http://hu.wikipedia.org/wiki/Sz%C3%A1m%C3%ADt%C3%B3g%C3%A9ph%C3%A1l%C3%B3zat>

Wikipédia (HU):

Unified Modeling Language – Letöltés: 2025.03.21. – https://hu.wikipedia.org/wiki/Unified_Modeling_Language

Wikipédia (HU):

Analóg-digitális átalakító – Letöltés: 2025.03.10. – https://hu.wikipedia.org/wiki/Anal%C3%B3g-digit%C3%A1lis_%C3%A1talak%C3%ADt%C3%B3

Xfish hivatalos oldala:

A halak étvágyára ható időjárási tényezők – Letöltés: 2025.03.20. – https://www.xfish.hu/horgaszmagazin/a-halak-etvagya-hato-idojarasi-te-nyezok/980?srsltid=AfmBOopQ6oGeY_KjKFwV9qe2SaV-CEbj39FNwYrYVXH8RUP_108X5sRZ_&utm

Amaël Cattaruzza (2019):

A digitális adatok geopolitikája, Pallas Athéné Könyvkiadó KFT., Budapest.

Adonyi Róbert (2011):

Adatstruktúrák és algoritmusok – Typotex kiadó

Bergmann Gábor, Darvas Dániel, Molnár Vince, Szárnyas Gábor, Tóth Tamás (2024):

Rendszermodellezés, Kritikus Rendszerek Kutatócsoport.

Csorbai Balázs, Dr. Bokor Zoltán, Dr. Ferincz Árpád, Dr. Hegyi Árpád, Dr. Eszterbauer Edit, Dr. Mézes Miklós, Dr. Staszny Ádám, Dr. Urbányi Béla (2020):

Halászat, haltenyésztés, Hermann Ottó Intézet Nonprofit KFT., Budapest.

Dr. Balázs Péter, egyetemi docens Dr. Németh Gábor, adjunktus (2019)

Adatbázisok a tervezéstől az alkalmazásfejlesztésig, Szegedi Tudományegyetem Természettudományi és Informatikai Kar Informatikai Intézet

Dr. Halasi-Kovács Béla (2019):

A magyarországi vízfolyások halközösségeinek ökológiai szempontú elemzése –Fazekas Nyomda, Szarvas

Dr. Izsó Lajos, Dr. Antalovits Miklós (2000):

Bevezetés az információ-ergonómiába, Emberi tényezők az információs technológiák bevezetésében és alkalmazásában.

Dr. Mojzes Imre (2005):

Mikroelektronika és technológia; Műegyetemi kiadó

Dr. Juhász Lajos (2011):

Magyar horgászvizeken, Tóth Könyvkereskedés és Kiadó KFT., Debrecen.

Dr. Kusper Gábor és Dr. Radványi Tibor (2012):

Jegyzet a projekt labor című tárgyhöz.

Earl Babbie (2003):

A társadalomtudományi kutatás gyakorlata, Balassi Kiadó, Budapest

Hajdu Gábor:

A kvantitatív és a kvalitatív társadalomtudományi kutatás módszerei

Harsányi Réka, Juhász Márton András (2014):

Fizikai számítástechnika: elektronikai alapok és Arduino programozás, Typotex Kiadó, Budapest.

Horváth László (2007):

Szoftvertesztelés a gyakorlatban.

Horváth László (2007):

Szoftvertesztelés a gyakorlatban.

Horváth László, Pirkó József (2001):

Informatikai Tudástár, Kiskapu Kiadó, Budapest.

Illés Zoltán (2005):

Programozás C# nyelven, Jedlik Oktatási Stúdió, Budapest.

Komló Csaba (2013):

Információs rendszerek tervezésének módszertana.

Kristály Mátyás (2001):

Minőségirányítási alapismeretek, Okker Kiadó, Budapest.

Lajkó István, (2003):

Korszerű halgazdálkodás, Szaktudás Kiadó Ház, Budapest

Magyar Gábor, Nemeslaki András, Syi (2021):

A digitális transzformáció kérdései, Gondolat Kiadó, Budapest.

Martin Ford (2022):

Jövönk a robotok korában, HVG Könyvek, Budapest

Peter Hobbs (2000):

Projektmenedzsment, Sclar Kiadó.

Pitlik László (2014):

Vidék- és területfejlesztési potenciálok elemzése, ezekhez kapcsolódóan a hasonlóság-elemzés módszertan alkalmazása.

Pitlik Marcell (2020):

Térfogati égés optikai vizsgálata és mesterséges intelligencia alapú elemzése, Tudományos Diákköri Konferencia, Budapest.

Ramez Elmasri és Shamkant B. Navathe (2011):

Fundamentals of Database Systems című könyve alapján készült: Fejezetek az adatbázisrendszerek elméletéből.

Reiter István (2012):

C# programozás lépésről lépésre – Jedlik Oktatási Stúdió

Rui Santos & Sara Santos:

Learn ESP32 with Arduino IDE 3rd Edition (Version 3.1).

Simon Gyula (2011):

A programozás alapjai.

Szabó Endre Győző (2022):

A védelmi lépcső elmélete, Ludovika Egyetemi Kiadó, Budapest.

Timár Gigi, Szennay Áron, Csákné Filep Judit:

A vállalkozásoktatás hatása a vállalkozói aktivitásra a közép-kelet-európai tranzíciós országokban – In: Köz-gazdaság - Gazdaságelméleti és -politikai szemle 2025. évf. 1. szám.

Zachary Wong (2019):

Nyolc nélkülözhetetlen humán készség a projektmenedzsmentben, Pallas Athéné
Könyvkiadó KFT., Budapest.

10. Mellékletek

Ebben a fejezetben azok a projektet támogató segédanyagok kerülnek tételesen felsorolásra, amelyeket terjedelmi okokból, célszerűtlen lett volna a megelőző fejezetekben kifejteni.

10.1. A projektben felhasznált programok letöltési helye és verziószáma:

<https://www.arduino.cc/en/software> (v: 2.3.4)

<https://www.drawio.com/> (v: 26.0.9)

<https://code.visualstudio.com/> (v: 1.98.2)

<https://visualstudio.microsoft.com/> (v: 17.13.5)

<https://filezilla-project.org/> (v: 3.68.1)

<https://www.arduino.cc/en/software>

https://www.adobe.com/hu/creativecloud/buy/students.HTML?mv=search&mv2=paidsearch&sdid=9DJJ4F4Q&ef_id=Cj0KCQjw16O_BhDNARIsAC3i2GDIY5u09kn75EN-LuDw9hlzTUyfgWB9Kb3BDHVVW1mLGjwcJORfKSzzAa-App8EALw_wcB:G:s&s_kwid=AL!3085!3!649502098319!p!!g!!adobe%20ingyen!18254414329!143422047320&gad_source=1&gclid=Cj0KCQjw16O_BhDNA-RIsAC3i2GDIY5u09kn75ENLuDw9hlzTUyfgWB9Kb3BDHVVW1mLGjwcJORfKSzzAa-App8EALw_wcB (v: 2.18.1)

ESP-32 driver:

<https://www.silabs.com/developer-tools/usb-to-uart-bridge-vcp-drivers>

Programozás segédanyagok:

<https://docs.arduino.cc/programming/>

<https://www.arduino.cc/reference/en/>

QR kód generátor:

<https://online-qr-generator.com/hu>

Webhely szolgáltató:

<https://www.nethely.hu/>



10.2. Ábrajegyzék

1. ábra: Többletérték becslés	13
2. ábra: Fix költségek	14
3. ábra: Havi költségek.....	14
4. ábra Feszültségosztás alapelve	22
5. ábra Feszültségosztás törvénye	23
6. ábra: A projekt megvalósítás szakaszai	25
7. ábra: Szerkezeti modell	39
8. ábra: Összeállított rendszer működő asztali verziója	40
9. ábra: ESP32-WROOM-32U.....	41
10. ábra Mikrovezérlő és annak lábkiosztása.....	43
11. ábra: Jelmagyarázat (vö. 10. ábra.).....	43
12. ábra: Szemléltetés analód és digitális jelekről.....	45
13. ábra: DS18B20 szenzor	46
14. ábra: Akkumulátor burok	47
15. ábra: Antenna.....	48
16. ábra: BME280 szenzor	48
17. ábra: RS-FXJT-N01 szélirány mérő	49
18. ábra: RS-FSJT-N01 szélsébség mérő	50
19. ábra. Slave eszköz felépítése	51
20. ábra Master eszköz felépítése.....	52
21. ábra Hálózati csatlakozás	53
22. ábra: Fejlesztői környezet letöltése	54
23. ábra Alaplap kezelő URL-jének a beállítása	55
24. ábra Mikrovezérlő csomagjának telepítése	55
25. ábra Kapcsolat ellenőrzés.....	56
26. ábra Kapcsolat kiválasztása a fejlesztői felületen	56
27. ábra Fejlesztői környezet alapképernyője	57
28. ábra Soros monitor bemutatása a fejlesztői környezetben	57

29. ábra: Adattábla.....	66
30. ábra: Utolsó adatok megjelenítése a webfelületen	69
31. ábra: Weboldalon történő teljes adatbázis megjelenítése	69
32. ábra: MySQL adatbázis tábla részlet.....	73
33. ábra: A Slave soros monitorán megjelenő teszt adatok	77
34. ábra: A Master soros monitorán megjelenő teszt adatok.....	77
35. ábra: GUI-n megjelenő adatok	78
36. ábra: Adatok struktúrája (részlet, 11 sor a 300 sorból).....	79
37. ábra: Segéd táblázat.....	80
38. ábra: Input tábla (részlet, 11 sor a 300 sorból)	80
39. ábra: Inverz tábla (részlet, 11 sor a 300 sorból)	80
40. ábra: Termelési függvény (részlet)	81
41. ábra Korrelációs együttható értékelése.....	82
42. ábra: Nemek aránya.....	88
43. ábra: Életkor aránya	88
44. ábra: Települések aránya	89
45. ábra: Iskolai végzettségek	89
46. ábra: Vízterület típusok aránya.....	90
47. ábra: Parton töltött tevékenység	90
48. ábra: Időjárási tényezők 1.	91
49. ábra: Időjárási tényezők 2.	91
50. ábra: Horgász aktivitás gyakorisága.....	92
51. ábra: Modern eszközök használata.....	92
52. ábra: Modern eszközök használata a horgászatban	93
53. ábra: Horgász motiváltság 1.	93
54. ábra: Horgász motiváltság 2.	93
55. ábra: Alkalmazott horgászati módszer	94
56. ábra: Alkalmazás iránti érdeklődés	94
57. ábra: Mesterséges intelligencia használata a horgászatban.....	95

10.3. Definíció jegyzék

Adatok: Strukturált vagy strukturálatlan információk, amelyeket tárolnak, feldolgoznak és elemznek, hogy értelmezhetővé és felhasználhatóvá váljanak.

Arduino: Egy könnyen programozható nyílt forráskódú hardver- és szoftverplatform, amelyet mikrovezérlő projektekhez fejlesztettek ki.

Benchmarking: összehasonlítás, mely inkább egy gondolkodásmódot, mintsem egy algoritmus jelent, tehát a benchmarking az összehasonlító módszertanok egy fajta gyűjtő kifejezése.

Deep sleep: (Mélyalvás mód) Egy energiatakarékos üzemmód mikrovezérlők.

Dobós horgászat: Egy horgász módszer, mely a végszerelék dobással való bejuttatását jelenti.

ESP-NOW protokoll: Egy vezeték nélküli, alacsony késleltetésű kommunikációs protokoll.

Externális: Külső tényezőkre vagy erőforrásokra utal.

Exploratív: Feltáró jellegű, azaz egy adott témát vagy jelenséget előzetes hipotézisek nélkül vizsgál, új összefüggéseket és mintázatokat keresve.

Mikrovezérlő: (Microcontroller) Egy apró számítógép egyetlen chipen, amely tartalmazza a processzort (CPU), memóriát (RAM, ROM/Flash) és perifériákat.

Predikció: (Előrejelzés) A predikció egy adott rendszer jövőbeli állapotának vagy eseményeinek előrejelzését jelenti.

Rapid horgászat: Olyan horgász módszer, melynek alapja a kevés idő alatti eredményes horgászat. (pl.: 1-2, pár órás horgászat amikor az idő engedi).

Real time: Olyan rendszerek vagy folyamatok, amelyek azonnali (vagy közel azonnali) választ igényelnek és biztosítanak.

PVA bag: (Polivinil-alkoholból) készült kis tasak, mely koncentrált etetőanyag bejuttatására szolgál dobós horgászat esetén.

Wireless: (Vezeték nélküli) Olyan kommunikációs technológia, amely nem igényel fizikai kábeleket az adatok továbbításához. (pl.: Wi-Fi, Bluetooth).

Freemium licence modell: alap szolgáltatás ingyenes, extra funkciók fizetősek

Üzleti terv: Egy írásos dokumentum, amely összefoglalja egy vállalkozás céljait, működését, pénzügyi terveit és stratégiáit a sikeres működés érdekében.

10.4. Rövidítések jegyzéke

2FA: (Two Factor Authentication) Két faktoros hitelesítés.

ADC: (Analog-to-Digital Converter) Átalakító, amely az analóg jeleket (pl.: feszültséget) digitális értékkel alakítja, hogy a mikrovezérlő feldolgozhassa.

AI: Artificial Intelligence

COCO: Component-based object comparison for objectivity (<https://miau.my-x.hu/myx-free/>)

GND: (Ground – Földelés) Az áramkörökben a GND a referenciafeszültség vagy a negatív pólus.

GPIO: (General-Purpose Input/Output) Egy mikrovezérlő vagy számítógép által vezérelhető csatlakozópont, amely bemenetként vagy kimenetként használható különféle eszközök vezérlésére.

GUI: (Graphical User Interface – Grafikus felhasználói felület) Egy olyan vizuális interfész, amely lehetővé teszi a felhasználók számára, hogy grafikus elemek (pl.: gombok, ikonok, menük) segítségével interakcióba lépjenek egy rendszerrel vagy alkalmazással, szemben a parancssoros (CLI) vezérléssel.

I²C: Egy kétvezetékes kommunikációs protokoll, amellyel egy vezérlő több eszközzel tud adatot cserélni egyszerűen és hatékonyan.

IoT: (Internet of Things) vagyis "Dolgok Internete" egy olyan technológiai koncepció, ahol különböző eszközök kapcsolódnak az internethez, és képesek adatokat gyűjteni, megosztani és akár önállóan működni.

MQTT: (Message Queuing Telemetry Transport), amely egy könnyű, publish-subscribe alapú hálózati protokoll, elsősorban IoT (Internet of Things) eszközök közötti kommunikációhoz.

NTP: Az NTP (Network Time Protocol) egy hálózati protokoll, amely biztosítja az eszközök pontos időszinkronizálását egy központi időforráshoz.

OAM: objektum-attribútum-mátrix, azaz más szóval tanulási minta, melyben immár tradicionálisan a sorok az objektumok (esetek), az oszlopok ezek jellemzői (attribútumok, szempontok, változók stb.).

PVA: (Polivinil-alkohol): nedvesség hatására feloldódó anyag.

SCL: (Serial Clock Line – Soros órajel vonal) Az I²C (Inter-Integrated Circuit) kommunikáció egyik vonala, amely az adatátvitelt vezérlő órajelet biztosítja.

SDA: (Serial Data Line – Soros adatvonall) Az I²C kommunikáció másik vonala, amelyen keresztül az eszközök egymás között adatokat küldenek és fogadnak.

ITIL: Az ITIL (Information Technology Infrastructure Library) egy olyan keretrendszer, amely bevált gyakorlatokat nyújt az informatikai szolgáltatásmenedzsmenthez, hogy az IT-szolgáltatások hatékonyan és ügyfélközpontúan működjenek.

10.5. ChatGPT – konverzációk

Ebben a fejezetben a ChatGPT névvel rendelkező Mesterséges Intelligenciával folytatott párbeszélések láthatók, annak érdekében, hogy az olvasó egy tisztát képet kapjon az AI által nyújtott támogatásról, melyek a dolgozat írása közben átgondolást igényeltek.

10.5.1. IEEE 802.11b/g/n/e/i szabvány leírása

Ebben a fejezetben a ChatGPT fejt ki a véleményét az IEEE 802.11b/g/n/e/i szabványt és erre ad részletes leírását.

Kérdés: Kérem fejtse ki a IEEE 802.11b/g/n/e/i szabvány leírását

ChatGPT válasza:

Az IEEE 802.11 egy vezeték nélküli hálózati (Wi-Fi) szabványcsalád, amelyet az Institute of Electrical and Electronics Engineers (IEEE) fejlesztett ki. Az általad említett 802.11b/g/n/e/i különböző verziói ennek a szabványnak, mindegyik egy adott fejlesztést vagy új funkciót ad hozzá a Wi-Fi hálózatokhoz.

IEEE 802.11b/g/n/e/i szabványok részletes leírása:

1. IEEE 802.11b – Az első elterjedt Wi-Fi szabvány (1999):

Jellemzők

- Frekvencia: 2,4 GHz
- Maximális sebesség: 11 Mbps
- Moduláció: DSSS (Direct Sequence Spread Spectrum)
- Hatótávolság: ~100 méter (optimális környezetben)
- Biztonság: WEP (gyenge titkosítás)

Előnyei:

- Nagyobb hatótávolság az alacsony frekvencia miatt
- Széles körben elterjedt, olcsó eszközök támogatják

Hátrányai:

- Lassú sebesség (max. 11 Mbps)
- Nagyon érzékeny az interferenciára, mert a 2,4 GHz-es sávon sok más eszköz is működik (Bluetooth, mikrohullámú sütők stb.)

- A használt WEP titkosítás könnyen feltörhető

2. IEEE 802.11g – Gyorsabb Wi-Fi a 2,4 GHz-es sávban (2003)

Jellemzők:

- Frekvencia: 2,4 GHz
- Maximális sebesség: 54 Mbps
- Moduláció: OFDM (Orthogonal Frequency-Division Multiplexing)
- Hatótávolság: Hasonló, mint az 802.11b (~100 méter)
- Biztonság: WPA, WPA2 támogatás

Előnyei:

- 5x gyorsabb, mint az 802.11b (54 Mbps vs. 11 Mbps)
- Visszafelé kompatibilis a 802.11b eszközökkel
- Jobb titkosítás (WPA/WPA2 támogatás)

Hátrányai:

- Még mindig csak 2,4 GHz-es sávot használ, tehát interferencia léphet fel
- Az elméleti 54 Mbps sebesség a valóságban sokkal alacsonyabb (~20-25 Mbps)

3. IEEE 802.11n – A Wi-Fi sebességének és hatótávolságának ugrásszerű növelése (2009):

Jellemzők:

- Frekvencia: 2,4 GHz és 5 GHz (dual-band)
- Maximális sebesség: 600 Mbps (4 MIMO antennával)
- Moduláció: OFDM
- Hatótávolság: 150+ méter
- Technológiák:
 - MIMO (Multiple Input Multiple Output) → Több antenna egyidejű használata a nagyobb sebesség és jobb lefedettség érdekében
 - Channel Bonding → Két 20 MHz-es csatornát kombinál, így akár 40 MHz-es csatornaszélességet is lehet használni
 - Beamforming (irányított jelátvitel)

Előnyei:

- Sokkal gyorsabb (akár 600 Mbps)
- 5 GHz-es sáv támogatása, kevesebb interferencia
- MIMO technológia → stabilabb kapcsolat

Hátrányai:

- Nem minden eszköz támogatja az 5 GHz-es sávot
- A nagyobb sebességhez több antenna szükséges, ami növeli a költségeket

4. IEEE 802.11e – Minőségjavítás (QoS – Quality of Service) (2005):

Jellemzők

- Célja: A Wi-Fi hálózatok minőségét (QoS) javítja azáltal, hogy prioritást ad egyes adatoknak
- Fontos fejlesztések:
 - WMM (Wi-Fi Multimedia) → Prioritás a hang- és videóátvitelnek
 - EDCA (Enhanced Distributed Channel Access) → Különböző forgalomtípusokhoz eltérő csatornakezelés
 - TSPEC (Traffic Specification) → Az eszközök előre kérhetik az erőforrásokat

Előnyei:

- Jobb hang- és videóminőség (pl.: VoIP hívások, videostreaming)
- Csökkenti a késleltetést és a csomagvesztést

Hátrányai:

- Nem növeli a sebességet, csak a kapcsolat minőségét javítja
- Nem minden régi eszköz támogatja

5. IEEE 802.11i – Biztonsági fejlesztések (2004):

Jellemzők

- Célja: A Wi-Fi hálózatok biztonságának növelése
- Fontos újítás:
 - WPA2 titkosítás (AES-alapú CCMP)
 - WPA (Wi-Fi Protected Access) → WEP helyett
 - TKIP (Temporal Key Integrity Protocol) → Dinamikus kulcsváltás az adatlopás ellen

Előnyei:

- Erősebb titkosítás (AES) → Nem törhető fel könnyen
- Megakadályozza az illetéktelen hozzáférést
- Megszüntette a WEP gyenge pontjait

Hátrányai:

- Nagyobb számítási teljesítményt igényel az AES titkosítás miatt
- Régebbi eszközök nem támogatják

10.5.2. IT Biztonsági aspektusok

A kérdés feltétel célja, hogy az AI szerint milyen IT biztonsági körülményekre kell tekintettel lenni a projekt során.

Kérdés: Milyen IT biztonsági aspektusok merülnek fel a következő szakdolgozati téma esetén? Nyílt területi időjárás állomás építése mikrovezérlő segítségével, adatküldése webszerverre, adatelemzés.

ChatGPT válasza:

Ez a szakdolgozati téma több IT biztonsági aspektust is érint, amelyeket érdemes figyelembe venni a tervezés és a megvalósítás során. Az alábbi főbb biztonsági szempontokat érdemes átgondolni:

1. Adatbiztonság és titkosítás

- Az időjárásállomás által küldött adatok titkosítása (pl.: TLS/SSL használata a kommunikáció során).
- Ha az adatokat hosszabb ideig tárolod, akkor megfelelő titkosítás és jogosultságkezelés szükséges.
- Az érzékeny adatok (pl.: földrajzi koordináták, szenzorok azonosítói) védelme.

2. Hálózati biztonság

- Az adatküldés biztonságos protokollokkal történjen (HTTPS, MQTT+TLS, vagy más titkosított csatorna).
- A mikrovezérlő és a szerver közötti kapcsolat megfelelő autentikációja (pl.: API-kulcs, tanúsítvány).
- A hálózati eszközök (Wi-Fi modul, IoT hálózat) sebezhetőségeinek minimalizálása.

3. Hozzáférés-kezelés és autentikáció

- A mikrovezérlő és a webszerver közötti kommunikáció autentikációja (pl.: JWT, HMAC vagy egyedi API-kulcs).
- Webes felület esetén erős jelszókezelés és felhasználói jogosultságok beállítása.
- Ha több eszköz csatlakozik, akkor eszközazonosítók egyedi kezelésének biztosítása.

4. Adatintegritás és megbízhatóság

- Adatok manipulálásának vagy hamis adatküldésnek a megakadályozása (pl.: digitális aláírás, checksum).
- Redundancia vagy hibatűrő mechanizmusok kialakítása (pl.: időbélyegzés, duplikált adatellenőrzés).

5. Webszerver és adatbázis biztonsága

- SQL injekció, XSS, CSRF támadások elleni védelem a webalkalmazásban.

- Rendszeres szoftverfrissítések és biztonsági patchek alkalmazása.
- Naplózás és monitoring az esetleges behatolások észlelésére.

6. Fizikai biztonság:

- Az időjárásállomás illetéktelen hozzáféréseinek megakadályozása (pl.: zárt doboz, védelem vandalizmus ellen).
- Ha nyilvános helyen van, akkor adatlopás és manipuláció elleni védelem.

7. DDoS és más szolgáltatásmegtagadási támadások elleni védelem

- Limitált API-hívások és forgalomszabályozás (rate limiting, IP szűrés).
- Gyanús aktivitások figyelése és blokkolása (pl.: IDS/IPS megoldások).

Ha konkrét implementációs terv van, akkor érdemes mélyebben átgondolni az adott technológiákra szabott védelmi intézkedéseket. Van már konkrét technológiai stack, amit tervezel használni?

Kérdés: Hány karakteresek a biztonságos API kulcsok webes felületen?

ChatGPT válasza:

A biztonságos API kulcsok hossza változó, de általában 32 és 64 karakter között mozognak.

Néhány ismert API szolgáltató kulcshossza:

- Google API kulcsok → 39-40 karakter
- OpenAI API kulcsok → 51 karakter
- Stripe API kulcsok → 32–64 karakter
- AWS Access Key ID → 20 karakter, Secret Key → 40 karakter
- JWT tokenek → Általában 128-256 karakter vagy még hosszabb

Biztonsági szempontok:

- Legalább 32 karakter hosszú legyen az erős védelem érdekében.
- Használjon véletlenszerű alfanumerikus és speciális karaktereket (ha támogatott).
- Ne tárold API kulcsokat frontend kódban vagy nyilvánosan elérhető helyen.

Ha egy konkrét API szolgáltató kulcsának hosszára vagy kíváncsi, meg tudom nézni!

10.5.3. Számítógép hálózatok

A következő kérdés összefoglalja a hálózatba kötött hardver eszközök rövid előnyeit, mivel a projekt eszközei jelentősen támaszkodnak a hálózati alapelvekre.

Kérdés: Kérem ismertesd a Számítógép hálózatok előnyei:

ChatGPT válasza:

A számítógép-hálózatok számos előnyt kínálnak mind a magánfelhasználók, mind a vállalatok számára. Néhány fő előny:

1. Erőforrás-megosztás

- Nyomtatók, szkennerek, tárhelyek és egyéb eszközök közösen használhatók.
- Költséghatékonyabb, mivel nem kell minden géphez külön erőforrást biztosítani.

2. Hatékony adatmegosztás és kommunikáció

- Dokumentumok, fájlok gyors és biztonságos megosztása a hálózaton keresztül.
- E-mail, csevegőalkalmazások és videóhívások segítségével gyorsabb kommunikáció.

3. Központi adatkezelés és biztonság

- Központi szerveren tárolt adatok könnyebben kezelhetők és biztonsági mentések készíthetők róluk.
- Hálózati biztonsági megoldások (pl.: tűzfalak, vírusvédelem) hatékonyabb védelmet biztosítanak.

4. Távoli elérés és mobilitás

- A felhasználók távolról is elérhetik az adataikat (VPN, felhőszolgáltatások).
- Lehetővé teszi az otthoni munkavégzést és a rugalmas munkakörnyezetet.

5. Költségcsökkentés és hatékonyság

- Egyetlen internetkapcsolatot több eszköz is használhat.
- Automatizált rendszerek és hálózati szoftverek javítják a munkafolyamatokat.

6. Könnyű skálázhatóság

- A hálózat egyszerűen bővíthető új eszközökkel és felhasználókkal.
- Növekvő vállalkozások számára ideális megoldás.

10.5.4. Többllet-érték becslés

A következő kérdésfeltevésnek a célja egy rövid kalkuláció készítése, mely összességében kimutatja a projekt hozzáadott értékét.

Kérdés: Hogyan lehet forint-pontosságú információs többlletérték-becslés célcsoportonként (profí és amatőr horgászok) és mindösszesen a horgászatban, ha mérjük az időjárási adatokat, és a vízhőmérsékletet?

ChatGPT válasza:

Nagyon jó kérdés! Egy forint-pontosságú információs többlletérték-becslés (vagyis annak megbecslése, hogy mennyit érnek az egyes információk célcsoportonként) a horgászatban komoly lehetőségeket rejt, különösen, ha megbízható környezeti adatokat (pl.: időjárás, vízhőmérséklet)

tudsz szolgáltatni. Az alábbi lépések egy gyakorlati megközelítést mutatnak be profi és amatőr horgászok szerinti bontásban.

1. Célcsoportok azonosítása

- Profi horgászok: versenyzők, intenzív sporthorgászok, sok időt és pénzt fektetnek bele.
- Amatőr horgászok: hobbihorgászok, családoso, hétvégi pecások.

2. Milyen adatok teremtenek értéket?

- Vízparaméterek (vízhőmérséklet, oxigénszint stb.): hatással van a halfajok aktivitására, táplálkozására.
- Időjárás (légnomás, szélirány, páratartalom): sok horgász ez alapján tervezi meg mikor és hova megy.

Ezek alapján segíteni lehet jobb fogási döntések meghozatalát, azaz több hal, kevesebb felesleges horgászidő.

3. Információ értékének becslése (forintban)

Példa (egyszerűsített számítás):

Profi horgász:

- Egy nap horgászat költsége (utazás, etetőanyag, idő, verseny): 15 000 Ft
- Ha a pontos vízhőmérséklet/időjárás alapján elkerül egy "rossz napot", vagy megfog egy versenyt, az értéke akár 10–50 000 Ft/alkalom lehet.
- Éves szinten: 50–100 000 Ft érték, amit a megbízható információval nyerhet.

Amatőr horgász:

- Egy nap költsége: kb. 5000–8000 Ft
- Ha jobban tudja mikor érdemes menni, évente 3-4 felesleges pecát elkerül: 20–30 000 Ft megtakarítás
- Extra: több sikeres nap, elégedettség, élmény → nem forintosítható, de növeli az értéket

4. Milyen formában található az adat?

- Alkalmazás vagy weboldal, amely:
- Élő vízhőmérsékletet, szélirányt, nyomást mutat.
- "Ideális pecaidő" előrejelzést ad.
- Testreszabott ajánlást ad a célzott halfaj és helyszín alapján.

5. Modellezés ötlet:

Használható gazdasági modellek:

- Contingent Valuation Method (CVM) – megkérdezed, mennyit fizetnének a pontos adatokért.
- Revealed Preference – megnézed, hogy mennyi pénzt költ most a peca előkészületeire, tanácsadókra, appokra.

Összefoglalva

Ha egy rendszer megbízható időjárási és vízhőmérsékleti adatokkal látja el a horgászokat, az információs többlet több tízezer forintnyi értéket is jelenthet évente egy-egy felhasználó számára. Ez a becsült érték szegmentálható célcsoportonként és akár üzleti modell alapjául is szolgálhat (pl.: előfizetéses alkalmazás).

10.5.5. Licence díj kalkuláció

Eben a fejezetben elsődleges költség, illetve a konkrét projekttel kapcsolatos licence díj kalkulációjával kapcsolatos információt láthatjuk.

Kérdés: Egyszerűbb szoftverek licence díjait hogyan lehet kikalkulálni?

A ChatGPT válasz:

Nagyon jó kérdés! Az egyszerűbb szoftverek licencdíjainak kikalkulálása többféleképpen történhet, attól függően, hogy milyen üzleti modellt, funkciókat vagy célcsoportot céloz meg a szoftver. Nézzük végig a legegyszerűbb és leggyakoribb módszereket:

1. Költségalapú árazás (Cost-based pricing)

Módszer: Összeadod a fejlesztés, karbantartás, support stb. költségeit, majd erre ráteszel egy nyereséghányadot.

Képlet: $\text{Licencdíj} = \text{Összköltség} / \text{várható felhasználók száma} + \text{haszonkulcs}$

Példa:

- Fejlesztési költség: 1.000.000 Ft
- Éves karbantartás: 200.000 Ft
- Várt felhasználók: 100 fő
- Haszonkulcs: 30%

$\text{Licencdíj} = (1.200.000 / 100) \times 1,3 = 15.600 \text{ Ft/fő/év}$

2. Értékalapú árazás (Value-based pricing)

Módszer:

Megnézed, mennyi értéket teremt a szoftver a felhasználónak (pl.: mennyi időt vagy pénzt takarít meg), és ahhoz mérten állapítod meg az árat.

Tényezők:

- Mennyi pénzt takarít meg?

- Mennyi időt spórol meg?
- Milyen problémát old meg?

Példa: Ha a szoftver havonta 10 órát spórol, és 1 óra értéke 5.000 Ft → akkor akár 50.000 Ft/hó árat is elkérhetsz – vagy annak egy részét, pl.: 10–15.000 Ft/hó.

3. Piaci összehasonlítás (Competitor-based pricing)

Módszer: Megnézed, hogy a hasonló szoftverek mennyibe kerülnek, és az alapján pozicionálsz a saját áradat – olcsóbban, ugyanúgy, vagy drágábban.

Példa:

- Versenytárs A: 2.000 Ft/hó
- Versenytárs B: 4.000 Ft/hó → Ha te valami pluszt adsz (pl.: magyar nyelv, jobb support), mehetsz a magasabb ár felé.

4. Licencmodellek típusai – mit válassz?

Licenc típus	Jelentése	Példa
Egyszeri díj	Csak egyszer fizet a vásárló	9.990 Ft / örökre
Előfizetés	Havi vagy éves rendszeres fizetés	1.500 Ft / hó
Freemium	Alap ingyenes, extra funkciók fizetők	Pl.: Dropbox modell
Per user / gép	Felhasználónként vagy eszközönként fizetnek	5.000 Ft / user / év
Open source + support	Ingyen a szoftver, de a támogatás fizetős	Pl.: Red Hat modellek

Melyiket érdemes használni?

- Egyszerű hobbiapphoz: egyszeri díj vagy freemium
- Vállalati eszközhöz: előfizetés / felhasználónkénti licenc
- Egyedi fejlesztésű apphoz: értékalapú árazás

Ha szeretnéd, kiszámolom egy konkrét szoftveredhez a **javasolt licencdíjat** is. Csak mondd el:

- Mi a szoftver célja?
- Kik használják?
- Milyen költségeid vannak vele?
- Hány felhasználóra számítasz?

Szívesen segítek végigvezetni az árazási döntésen is!

Kérdés: Mi a szoftver célja? időjárásadatok gyűjtése konkrét horgász tavaknál. Kik használják? sport és hobbi horgászok Milyen költségeid vannak vele? egyszeri Szoftverfejlesztés 250.000 Ft, eszközök 150.000 Ft/db, havi fix költségek 10.000 Ft/eszköz, marketing 200.000 Ft/összes

eszköz, terméktámogatás 10.000 Ft/eszköz. Hány felhasználóra számíthat? körülbelül 50 db horgásztónál 5.000 fő, Kérem szövegesen is elemezze ki az eredményt

A ChatGPT válasza:

Rendben, íme a szöveges elemzés a megadott bemenő adatok alapján:

Cél: A szoftver célja: időjárásadatok gyűjtése konkrét horgásztavaknál, melyet sport- és hobbi horgászok használnak döntéstámogatásra, például mikor érdemes horgászni.

Költségösszesítés

1. Egyszeri költségek:

Kölcségtípus	Mennyiség	Egységköltség	Összesen
Szoftverfejlesztés	1 db	250.000 Ft	250.000 Ft
Eszközbeszerzés	50 eszköz	150.000 Ft	7.500.000 Ft
Marketing	1 kampány	200.000 Ft	200.000 Ft
Összesen (egyszeri)			7.950.000 Ft

2. Havi fenntartási költségek (50 eszközre):

Kölcségtípus	Egységköltség / eszköz	Összesen / hó
Fix költség	10.000 Ft	500.000 Ft
Terméktámogatás	10.000 Ft	500.000 Ft
Havi összesen		1.000.000 Ft

Felhasználói adatok

- Várható felhasználószám: 5.000 fő
- Eszközök száma: 50 db (tóhelyenként 1 db)

Kalkulált fajlagos költségek

Egyszeri költség / felhasználó:

$$7.950.000 \text{ Ft} / 5.000 \text{ fő} = 1.590 \text{ Ft} / \text{fő}$$

Havi működési költség / felhasználó:

$$1.000.000 \text{ Ft} / 5.000 \text{ fő} = 200 \text{ Ft} / \text{fő} / \text{hó}$$

Következtetések és javaslatok:

- A rendszer kölcséghatékony, ha eléri az 5.000 fő felhasználóbázist, hiszen havi 200 Ft/fő fenntartási díj nem magas.
- Egyszeri belépési díj (pl.: éves előfizetésbe építve) érdemes lehet, pl.: 2.000–2.500 Ft/fő, amivel a kezdeti beruházás 1 év alatt megtérülhet.

- Célszerű lehet freemium modellt alkalmazni: alapszolgáltatás ingyenes, prémiumért (pl.: részletes elemzések, múltbeli trendek, fogási előrejelzések) fizetnek a felhasználók.
- A profi horgászok célcsoportja hajlandó lehet akár 500–1000 Ft/hó díjat is fizetni a plusz adatokért.
- Az amatőrök esetében inkább az egyszerűség, élményfaktor a döntő – itt lehet alacsonyabb havi díjjal számolni.

10.6. Licence

ESP-NOW network + WiFi gateway to the Internet

<https://github.com/mlcr0lab-esp32/esp-now-network-and-wifi-gateway/blob/Master/Readme.md>

MIT Licence

Copyright (c) 2020 Stéphane Calderoni

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

<https://github.com/mlcr0lab-esp32/esp-now-network-and-wifi-gateway/blob/master/LI-CENSE>

10.7. Műszaki adatlapok

Az alábbi linkek a projekt során alkalmazott eszközök műszaki adatlapjainak elérhetőségeire mutatnak.

10.7.1. esp32-wroom-32d_esp32-wroom-32u_datasheet_en

https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf

10.7.2. DS18B20

<https://www.analog.com/media/en/technical-documentation/data-sheets/ds18b20.pdf>
<https://www.farnell.com/datasheets/3759119.pdf>

10.7.3. Battery Shield V3

<https://modulshop.hu/akku-shield-5-v-es-3-v-kimenet-1-db-18650-cella-mikro-usb>

10.7.4. bst-bme280-fl000

<https://cdn-learn.adafruit.com/downloads/pdf/adafruit-bme280-humidity-barometric-pressure-temperature-sensor-breakout.pdf>

10.7.5. RS-FXJT-N01

<https://instrucenter.com/wp-content/uploads/2022/03/RS-FXJT-N01.pdf>

10.7.6. RS-FSJT-No1

<https://instrucenter.com/wp-content/uploads/2022/03/RS-FSJT-No1.pdf>

Kodolányi János Egyetem

NYILATKOZAT

Alulírott Major Levente, U53V9Z üzemmérnök informatikus hallgató nyilatkozom, hogy a szakdolgozat saját munkám terméke, a felhasznált irodalmat a tudományterületnek megfelelő módon kezeltem, az erre vonatkozó jogszabályokat betartottam.

Az elektronikusan benyújtott szakdolgozat tartalmában és formájában is egyaránt megegyezik a konzulens által jóváhagyott példánnyal.

Budapest 2025. 04. 16.

.....
Major Levente
aláírás

Kodolányi János Egyetem NYILATKOZAT

Alulírott (név)
(beosztás) nyilatkozom, hogy a szakdolgozat a Kodolányi János Egyetem Szakdolgozati szabályzata előírásainak tartalmi és formai szempontból egyaránt megfelel.

Indoklás: Ellenőrző lista

A dolgozat tartalmazza az alábbi kötelező elemeket	IGEN✓
1. Borítólap vagy külső borító (Szakdolgozat/Portfólió)	
2. Címlap (cím, és ha van alcím)	
3. Tartalomjegyzék	
4. Oldalszám	
5. Irodalomjegyzék/Felhasznált irodalom	
6. Egyetlen PDF fájl (hang- és képfájlok külön feltöltve)	
Egyéb fontos adatok	
7. Megnevezés Kodolányi János Egyetem	
8. Témát gondozó tanszék/egység neve	
9. Évszám/elkészítés éve	
10. Képzési helyszín	
11. Konzulens neve	

Témavezető/konzulens által ellenőrzött tartalmi elemek, nyilatkozat:

A dolgozat tartalmazza az alábbi kötelező elemeket	IGEN✓
1. Tartalomjegyzék	
2. Irodalomjegyzék/Felhasznált irodalom	
3. Hivatkozásokkal ellátva (Szövegközi vagy lábjegyzet)	
4. Szövegközi hivatkozás esetén hivatkozásjegyzék	
5. A konzultációk megtörténtek	

A dolgozat végleges változatát áttanulmányoztam, annak benyújtásához hozzájárulok.

Kelt.: Budapest, 20.....

.....
Konzulens aláírása