

BESZÁMOLÓ
a 2024-2-HU01-KA210-VET-000271388 számú projekt 6. találkozásáról

A megvalósított tevékenységek bemutatása és a projekt szakmai kiegészítése

A projekt 2025. november 17–21. között valósult meg Nagyváradon, a debreceni és nagyvárad partneriskolák részvételével. A kezdeményezés célja az volt, hogy a tanulók gyakorlati környezetben sajátítsák el az Ipar 4.0 és IoT technológiák alkalmazását, és közösen hozzanak létre olyan épületfelügyeleti rendszereket, amelyek valódi okosotthon-funkciókat demonstrálnak. A program a két intézményben előzetesen elkészült, 3D nyomtatott házmodulokra és ESP32 alapú elektronikai egységekre épült.

Szakmai háttér és a platformváltás indoklása

A projekt kiindulópontját a SmartFarm oktatókészlet jelentette, amelynek szoftvere ESP32 mikrokontrolleren fut. Ez a megoldás alkalmas alapvető szenzoradat-kezelésre és egyszerű logikák futtatására, azonban az ESP32 **korlátozott erőforrásai** (processzor- és memória-kapacitás, valamint a beépített webszerver teljesítménye) csak korlátozott funkcionalitást tesznek lehetővé. A készlet eredeti felépítése szerint az ESP32 látja el a backend és frontend kiszolgálását is, ami egy komplex, stabil és valós idejű adatkezelést végző épületfelügyeleti rendszer esetén már nem bizonyult elegendőnek.

A projekt ezért egy új, ipari mintát követő **master–slave rendszerarchitektúrát** vezetett be:

- **Master: Raspberry Pi**
 - a teljes backend szolgáltatás (Flask),
 - a frontend webalkalmazás kiszolgálása,
 - adatgyűjtés, naplózás, felhasználói interfész,
 - okostelefonos elérés biztosítása.
- **Slave: ESP32**
 - szenzoradatok gyűjtése,
 - aktorvezérlés,
 - kommunikáció **Modbus TCP** protokollon keresztül.

Beszámoló a 6. találkozásról

Nagyvárad, 2025.11.17-2025.11.21.

A megvalósított tevékenységek bemutatása és a projekt szakmai kiegészítése

A projekt 2025. november 17–21. között valósult meg Nagyváradon, a debreceni és nagyváradai partneriskolák részvételével. A kezdeményezés célja az volt, hogy a tanulók gyakorlati környezetben sajátítsák el az Ipar 4.0 és IoT technológiák alkalmazását, és közösen hozzanak létre olyan épületfelügyeleti rendszereket, amelyek valódi okosotthon-funkciókat demonstrálnak. A program a két intézményben előzetesen elkészült, 3D nyomtatott házmodulokra és ESP32 alapú elektronikai egységekre épült.

Szakmai háttér és a platformváltás indoklása

A projekt kiindulópontját a SmartFarm oktatókészlet jelentette, amelynek szoftvere ESP32 mikrokontrolleren fut. Ez a megoldás alkalmas alapvető szenzoradat-kezelésre és egyszerű logikák futtatására, azonban az ESP32 korlátozott erőforrásai (processzor- és memória-kapacitás, valamint a beépített webserverteljesítménye) csak korlátozott funkcionalitást tesznek lehetővé. A készlet eredeti felépítése szerint az ESP32 látja el a backend és frontend kiszolgálását is, ami egy komplex, stabil és valós idejű adatkezelést végző épületfelügyeleti rendszer esetén már nem bizonyult elegendőnek.

A projekt ezért egy új, ipari mintát követő master–slave rendszerarchitektúrát vezetett be:

- | | | |
|--|------------------|--------------|
| ● Master: | Raspberry | Pi |
| – a teljes backend szolgáltatás (Flask), | | |
| – a frontend webalkalmazás kiszolgálása, | | |
| – adatgyűjtés, naplózás, felhasználói interfész, | | |
| – okostelefonos elérési biztosítása. | | |
| ● Slave: | | ESP32 |
| – szenzoradatok gyűjtése, | | |
| – aktorvezérlés, | | |
| – kommunikáció Modbus TCP protokollon keresztül. | | |

A Modbus TCP alkalmazása különösen fontos volt, mivel ez ipari környezetben kvázi szabványos megoldásnak számít, és lehetővé teszi a tanulók számára, hogy valódi PLC–HMI rendszerlogikát ismerjenek meg.

A platformváltás tehát nem eltérésként, hanem szakmailag indokolt fejlesztési lépésként jelent meg: a cél egy olyan skálázható, stabil és ipari jellegű megoldás létrehozása volt, amely túlmutat a SmartFarm kit eredeti keretein.

A találkozó részletes bemutatása

2025.11.17. (Hétfő) – 1. nap: Utazás, moduláris 3D elemek összeszerelése

A debreceni tanulók és tanárok megérkeztek Nagyváradra, majd megkezdődött a két intézményben korábban legyártott 3D nyomtatott házmodulok összeállítása. A diákok a moduláris elemekből 10 darab szerkezetkész házat hoztak létre, miközben a hiányzó vagy hibás elemeket helyben, 3D nyomtatóval pótolták. A nap célja az volt, hogy minden épület konstrukciós szempontból készen álljon a szenzorok és elektronikai egységek integrálására.

2025.11.18. (Kedd) – 2. nap: Raspberry Pi, Linux és hálózati alapok elsajátítása

A második napon a tanulók a fejlettebb rendszerirányítási platformot, a Raspberry Pi-t ismerték meg. Telepítették és konfigurálták a Raspberry Pi OS-t, majd létrehozták az SSH alapú távoli elérést, ami az ipari rendszerekben megszokott, közvetlen megjelenítőeszközt nem igénylő, távoli elérésen alapuló működés előfeltétele.

A nap során elsajátították a legfontosabb Linux-parancsokat és rendszerszintű ismereteket:

- fájlrendszer-kezelés, jogosultságok,
- csomagkezelés és szoftvertelepítés,
- hálózati beállítások és diagnosztika,
- folyamatkezelés és rendszerfelügyelet.

Ez a tudás később fontos alapot jelentett a backend fejlesztéséhez és a Modbus kommunikáció beállításához.

2025.11.19. (Szerda) – 3. nap: Saját épületfelügyeleti szoftver fejlesztésének megkezdése

A harmadik napon megkezdődött a teljesen új, Raspberry Pi alapú épületfelügyeleti rendszer fejlesztése. A tanulók vegyes (debreceni–nagyvárad) csoportokban dolgoztak, így megtapasztalták a közös fejlesztés valós kihívásait.

A feladatok:

- a frontend webalkalmazás alapjainak kialakítása (HTML, CSS, JavaScript),
- a felügyeleti panel funkcionális tervezése,
- a Raspberry Pi-ESP32 kommunikációs logika előkészítése,
- a fejlesztési munkafolyamatok felosztása és összehangolása.

A diákok ekkor már kifejezetten a mobilról is vezérelhető, gyorsabb és stabilabb rendszer létrehozását célozták meg.

2025.11.20. (Csütörtök) – 4. nap: A webalkalmazás továbbfejlesztése, a házak „felokosítása”

A negyedik napon megkezdődött az elektronikai integráció. A tanulók az ESP32 alapú 24-in-1 szenzorkészlet elemeit építették be a 3D házakba, majd kábelezési és illesztési feladatokat végeztek. Kiválasztották a megvalósítandó funkciókhoz szükséges érzékelőket (mozgás-, fény-, hőmérséklet-, páratartalom-érzékelők stb.), és elkészítették az adatkommunikációs struktúrát.

Ezzel párhuzamosan a frontend alkalmazás is bővült: létrejöttek a szenzoradatok valós idejű megjelenítésére szolgáló panelek, valamint a felhasználói vezérlőelemek is.

2025.11.21. (Péntek) – 5. nap: Tesztelés, validálás, értékelés és hazautazás

Az utolsó napon a tanulók elvégezték a rendszer teljes körű tesztelését:

- szenzoradatok valós idejű kiolvasása és validálása,
- aktorok (relék, LED-ek, motorok) működésének ellenőrzése,
- Modbus TCP kommunikáció stabilitásának vizsgálata,
- hibakeresés és finomhangolás.

Megkezdődött a saját épületfelügyeleti alkalmazás funkcionális bővítése is: a rendszer már képes volt a mért szenzoradatok alapján automatizált aktorvezérlést végezni. A diákok így egy teljesen működő IoT alapú okosház-rendszert hoztak létre, amelyet akár mobiltelefonról is kezelhettek.

A nap végén a résztvevők online kérdőíven keresztül értékelték a mobilitást, annak szakmai, pedagógiai és szervezési színvonalát, majd a debreceni csoport visszautazott Magyarországra.

Összegzés

A projekt eredményesen valósult meg, és több területen a tervezettnél is magasabb szintű szakmai mélységet ért el. A tanulók komplex IoT-rendszer létrehozásában vettek részt, amely a 3D modellezéstől kezdve az elektronikai illesztésen át egészen a webes felügyeleti platform fejlesztéséig terjedt. A SmartFarm kit korlátait felismerve egy új, Raspberry Pi alapú master–ESP32 slave architektúrát valósítottak meg, amely közelebb áll az ipari automatizálási rendszerek logikájához.

A projekt hozzájárult ahhoz, hogy a diákok megértsék az innováció jelentőségét, és megtapasztalják: fejlettebb, hatékonyabb és skálázhatóbb rendszerek mindig létrehozhatók – különösen az Ipar 4.0 technológiáinak alkalmazásával.

Melléklet: A rendszer architektúrájának és működésének szakmai leírása

1. Rendszeráttekintés

A megvalósított rendszer egy többkomponensű, IoT alapú épületfelügyeleti megoldás, amely a résztvevő tanulók által készített 3D nyomtatott házmodellek szenzoros és elektronikai integrációjára épül. A rendszer célja a valós ipari környezetben alkalmazott architektúrák és kommunikációs protokollok modellezése, különös tekintettel a decentralizált adatgyűjtésre, a távoli felügyeletre és a webalapú vezérlésre.

Az elkészült platform az Ipar 4.0 szemléletmódnak megfelelően **elosztott intelligenciájú** megoldást valósít meg, melyben az adatgyűjtés és a központi feldolgozás szerepei elkülönülnek.

2. Rendszer architektúra

A rendszer két fő komponensből épül fel:

2.1. Master egység – Raspberry Pi

A Raspberry Pi tölti be a központi vezérlő (master) szerepét. Fő feladatai:

- Backend szolgáltatás biztosítása (Flask keretrendszerrel),
- Frontend webfelület kiszolgálása (HTML/CSS/JavaScript),
- Felhasználói interakciók kezelése (vezérlőgombok, állapotlekérdezés),
- Szenzoradatok fogadása és naplózása,

- Aktorvezérlő parancsok továbbítása az ESP32 felé,
- Távoli elérés biztosítása mobiltelefonról vagy számítógépről,
- A rendszer üzemi állapotának vizuális megjelenítése.

A Raspberry Pi erősebb processzora és nagyobb memóriakapacitása lehetővé teszi komplexebb szoftverretek egyidejű futtatását, így alkalmas a webes és adatfeldolgozó funkciók együttes biztosítására.

2.2. Slave egység – ESP32

Az ESP32 mikrokontroller feladata az érzékelők és aktorok közvetlen kezelése. Az eszköz a SmartFarm készlet szenzoraival együtt működve végzi:

- szenzoradatok folyamatos olvasását,
- bemenetek előfeldolgozását,
- aktorok vezérlését (pl. LED-ek, relék, motorok),
- adatok továbbítását **Modbus TCP** protokollon keresztül.

Az ESP32 lényegében *mezőszintű eszközként* működik, amely fizikai kapcsolatban áll a beépített elektronikai elemekkel, és adatokat szolgáltat a master egység felé.

3. Kommunikáció és adatáramlás

A master és a slave egységek közötti kommunikációt a **Modbus TCP** protokoll biztosítja. A protokoll ipari környezetben elterjedt megoldás, amely egyszerű, megbízható és determinisztikus adatcserét tesz lehetővé.

A kommunikáció fő folyamatai:

1. **Szenzoradat-gyűjtés** (ESP32)
A mikrokontroller beolvassa a csatlakoztatott érzékelők mért értékeit (pl. hőmérséklet, páratartalom, mozgás, fényintenzitás).
2. **Adatátadás** **Modbus** **regiszterekbe**
Az ESP32 a beolvasott adatokat Modbus regiszterekbe helyezi.
3. **Backend** **lekérdezés** **(Raspberry Pi)**
A Flask alkalmazás rendszeres időközönként lekérdezi a Modbus regisztereket.

- | 4. Adatok | megjelenítése | frontend | felületen |
|--|---------------|----------|-----------|
| Az adatok a felügyeleti panelen valós időben frissülnek. | | | |
5. **Aktuátorvezérlés**
A felhasználói műveletek (pl. „STOP”, világítás be/ki) a backendben keletkeznek, majd Modbus TCP-n keresztül visszakérülnek az ESP32-be, amely végrehajtja az utasítást.

4. Szenzorok és aktorok integrációja

A 3D nyomtatott házmodellekbe a következő érzékelők és aktorok kerültek integrálásra:

- Hőmérséklet- és páratartalom-érzékelők (DHT, SHT típusok),
- Fényérzékelők (LDR, fotodióda),
- Mozgásérzékelők (PIR szenzor),
- LED-es visszajelzők,
- Minirelék vagy tranzisztoros meghajtók,
- Egyéb, projektfüggő elektronikai elemek.

A szenzorok pozicionálását a házmodell funkcionális használati módjaihoz igazítottuk (pl. bejárat mozgásérzékelő, beltéri klímaszenzor).

5. A webalapú felügyeleti rendszer felépítése

A felhasználói felület egyszerű, letisztult webalkalmazás, amely a következő funkciókat biztosítja:

- valós idejű szenzoradat-megjelenítés,
- vizuális állapotjelzés (indikátorok),
- aktív vezérlési lehetőségek (pl. világítás kapcsolása),
- STOP funkció a gyors beavatkozáshoz,
- mobileszközről is elérhető felület.

A frontend és a backend egyaránt a Raspberry Pi-n fut, amely így stabil, lokalizált szolgáltatási réteget biztosít.

6. Összegzés: A rendszer előnyei és szakmai értéke

Az elkészült IoT alapú épületfelügyeleti rendszer jelentős pedagógiai és szakmai értéket képvisel. A platform valós ipari gyakorlatot modellez:

- elosztott intelligenciát alkalmaz (master–slave modell),
- szabványos ipari protokollt használ (Modbus TCP),
- képes valós idejű adatgyűjtésre és vizualizációra,
- webalapú, távolról elérhető felügyeleti felületet biztosít,
- moduláris és bővíthető felépítésű.

A megoldás túlmutat az eredeti oktatókészlet képességein, és olyan ipari szemléletű rendszerarchitektúrát ad a tanulók kezébe, amely a későbbi tanulmányok és gyakorlati alkalmazás szempontjából is kiemelten értékes.