



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

# Jó Gyakorlat Gyűjtemény: Mikrokontrolleres Időjárás-állomások

A gyűjtemény a mikrokontrollerek és a hálózatba kapcsolt eszközök területén (IoT – Internet of Things) nyújt gyakorlati útmutatást szakképzésben részt vevő diákok és tanárok számára. A projektek során a résztvevők adatgyűjtést, adatátvitelt, adatbázis-kezelést és webes vizualizációt érintő készségeket fejlesztenek.

## 1. Jó Gyakorlat: ESP32 – IoT Időjárás-állomás WiFi-vel

Ez a jó gyakorlat az ESP32 mikrokontroller Wi-Fi képességeit kihasználva valósít meg egy teljesen önálló, hálózatba kapcsolt adatgyűjtő rendszert.

### Óravázlat (Magyar)

<b>Cím</b>	<b>ESP32 – Komplex IoT Időjárás-állomás Wi-Fi-vel</b>
<b>STE(A)M</b>	Technológia, Mérnöki tudományok, Adattudomány, Informatika <b>Tárgyak</b>
<b>Korosztály</b>	16+ (Szakképzés, Felsőfokú képzés)
<b>Időtartam</b>	6 óra (3 x 2 órás alkalom)
<b>Kulcskérdés</b>	Hogyan hozhatunk létre egy teljesen digitális, energiatakarékos rendszert a környezeti adatok gyűjtésére és megjelenítésére?

### Rövid bemutatás

A projekt egy **ESP32 mikrokontrollert** használ a DHT11 hőmérséklet- és páratartalomérzékelő adatainak olvasására. Az ESP32 Wi-Fi kapcsolaton keresztül elküldi a mért adatokat egy távoli webszerverre **HTTP POST** kérések formájában. A szerveroldalon a **PHP** kód fogadja az adatokat, SQL adatbázisban tárolja, és egy weblapon (HTML/CSS) jeleníti meg. Ez a gyakorlat az **adatgyűjtéstől a webes megjelenítésig** lefedi az IoT rendszer teljes láncát.



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

## Célok/Tanulási Eredmények ☞ Kategória

### Tanulási Eredmények (LOs)

Képesek lesznek egy komplett IoT rendszer általános kutatási tervének

**Diákoknak** felvázolására (A1). Képesek lesznek mikrokontrollert Wi-Fi hálózathoz csatlakoztatni, valamint a HTTP protokoll használatával adatot küldeni (B2, C2). Képesek lesznek a mérnöki tervezést a rendelkezésre álló infrastruktúrához igazítani (A3). Megértik az adatok fogadásának (POST) és lekérdezésének (GET) elvét a webes kommunikációban.

Képesek lesznek egy munkaalapú tanulási forgatókönyv megtervezésére a

**Tanároknak** digitális energiahatékonyság jegyében (A2). Képesek lesznek a gyakorlati projekt adatainak vizualizálásával motiválni a diákokat a STEAM tárgyak iránt.

### Eszközök és Adatok ✂

#### Kategória

Megnevezés

#### Hardver

ESP32 mikrokontroller, DHT11 hőmérséklet/páratartalom érzékelő, bekötő vezetékek.

#### Szoftver/Infrastruktúra

Arduino IDE (C++ kódhoz), PHP webservert SQL adatbázissal (pl. MySQL), Wi-Fi hálózat.

#### Adatok

Hőmérsékleti és páratartalmi adatok (mért értékek), HTTP kódok és szerver válaszüzenetek (hibaelemzéshez).

### Iránymutatások (Óravázlat) ?

#### Rész

#### Tevékenység

A DHT11 szenzor és az ESP32 alapjainak bemutatása, adatgyűjtés

#### 1. Rész: Bevezetés és Hardver (2 óra)

lokálisan (soros monitoron). A hálózati kapcsolódás elméleti alapjai: SSID, jelszó, `connectWiFi()` funkció.

Az  
ESP  
32  
HT  
TP  
PO  
ST  
kódj  
ának  
meg  
értés  
e



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

<p><b>2. Rész: Adatátvitel és Szerverkommunikáció</b> (2 óra)</p> <p><b>Vizualizáció és Következtetés</b> (2 óra)</p> <p><b>Pedagógiai Fókusz (Források alapján)</b> <i>Flipped Classroom</i> megközelítés: A diákok előre elsajátítják az alapismereteket (pl. a DHT11 működését), a fókusz az</p>	<p>(<code>http.POST(postData)</code>) és a szükséges HTTP fejlécek beállítása. A PHP/SQL kód áttekintése az adatok fogadására és adatbázisba írására (tranzakciókezelés/táblatörlés 10 rekord után).</p> <p>A weboldal (HTML/PHP) felépítésének megértése, amely a <b>3. Rész: Webes GET</b> metódussal lekérdezi az adatokat és táblázatban jeleníti meg. Adatgyűjtési terv felülvizsgálata (A2): Mi történik, ha 10 rekord eléri a limitet?. A projekt finomítása és az eredmények bemutatása.</p>	<p>osztályban a kód és a kapcsolódás gyakorlati megvalósításán van.</p> <p>Kutatási módszertan: Az <b>adatok gyűjtésének és kezelésének infrastruktúrája</b> (B2). <i>Problémamegoldás (RBT)</i>: Hibakódok (pl. <code>httpCode</code>) elemzése a sikertelen kérések okainak feltárására. Kollaboratív tanulás (Collaborative Learning): A diákok megosztják munkájukat, és megvitatják, hogyan lehetne az adatok vizualizációját (pl. grafikonok) javítani (A4). A megszerzett tudás újrahasznosíthatóságának (OER) ösztönzése.</p>
---	--	---

## 2. Jó Gyakorlat: Arduino UNO – Költséghatékony Wi-Fi Megoldás (ESP-01S)

Ez a gyakorlat bemutatja, hogyan valósítható meg az időjárás-állomás egy gyengébb, de olcsóbb mikrokontrollerrel (Arduino UNO), kihasználva az ESP-01S modult Wi-Fi modemként.

### Óravázlat (Magyar)

<b>Cím</b>	<b>Arduino UNO + ESP-01S: Wi-Fi Adaptáció Korlátok Kezelésével</b>
<b>STE(A)M</b>	Technológia, Mérnöki tudományok, Fizika, Problémamegoldás
<b>Tárgyak</b>	
<b>Korosztály</b>	14–18 (Szakképzés)
<b>Időtartam</b>	6 óra (3 x 2 órás alkalom)
<b>Kulcskérdés</b>	Hogyan használhatunk fel korlátozott erőforrásokkal (memória, sebesség) rendelkező eszközöket hálózati kommunikációra?



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

## Rövid bemutatás

Ez a projekt az Arduino UNO-t használja a szenzoradatok olvasására (DHT11), de az **ESP01S modul** Wi-Fi adapterként. A kommunikáció a két egység között szoftveres soros porton (SoftwareSerial) keresztül történik, **AT parancsok** használatával a Wi-Fi vezérlésére. Ez rávilágít a korlátozott erőforrások kezelésének, a programozási technikák adaptálásának (pl. **késleltetések** beépítése) és a meglévő webinfrastruktúra (PHP/SQL) változatlan használatának fontosságára.

## Célok/Tanulási Eredmények ☞ Kategória

### Tanulási Eredmények (LOs)

Képesek lesznek felismerni a hardveres korlátokat (pl. memória/sebesség).

Képesek lesznek egy kutatási terv (research design) adaptálására a

**Diákoknak** rendelkezésre álló infrastruktúrához (A3). Képesek lesznek szoftveres soros kommunikáció beállítására és AT parancsok használatára. Képesek lesznek egy meglévő webes felülethez HTTP POST kéréseket küldeni a mikrokontrolleren keresztül.

Képesek lesznek a mérnöki tervezés és a programozás kihívásainak

**Tanároknak** bemutatására egy alacsony költségvetésű, de funkcionális megoldáson keresztül.

### Eszközök és Adatok ✂

#### Kategória

#### Megnevezés

Arduino UNO, ESP-01S Wi-Fi modul, DHT11 érzékelő, bekötő **Hardver** vezetékek (különösen a TX/RX csatlakozások).

**Szoftver/Infrastruktúra** Arduino IDE (SoftwareSerial könyvtár), PHP webservert SQL adatbázissal (változatlanul használható).

Hőmérsékleti és páratartalmi adatok, AT parancsok soros

**Adatok** kommunikáción keresztül.

### Iránymutatások (Óravázlat) !



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

Rész	Tevékenység	Pedagógiai Fókusz (Források alapján)
	Az Arduino és az ESP-01S bekötésének áttekintése (VCC, RX, TX). A soros kommunikáció (SoftwareSerial) hogyan befolyásolja a hardver kommunikáció (2 beállítása és tesztelése. Elméleti	
óra)	áttekintés a teljesítménybeli különbségekről az ESP32-höz összetettségét és megbízhatóságát. képest. A <code>connectWiFi()</code> funkció	kiválasztása a projekt Kutatásalapú tanítás (RBT): A
<b>2. Rész: Wi-Fi Kapcsolódás AT Parancsokkal (2 óra)</b>	megvalósítása AT parancsok használatával (pl. <code>AT+CWMODE=1</code> , diákok megtanulják dokumentálni <code>AT+CWJAP</code> ). A <code>delay()</code> funkció a munkafolyamatokat, például a késleltetések pontos idejét, és használatának szükségességének megindokolják a tervezési elemzése az AT parancsokra várva, a megbízható működés érdekében. döntéseket (A4).	
<b>3. Rész: HTTP POST AT Parancsokkal (2 óra)</b>	A <code>sendToServer()</code> funkció létrehozása, amely manuálisan (AT Problémamegoldó készségek: A parancsokkal) felépíti a HTTP diákok megtanulják, hogy miként kérést, beleértve a TCP kapcsolatot lehet kézzel (alacsony szinten) és a Content-Length fejlécet. Az megvalósítani azokat a adatátvitel tesztelése, feladatokat, amelyeket egy összehasonlítva az ESP32-es erősebb eszköz automatikusan megoldással (teljesítménybeli (magasabb szinten) kezel. eltérések).	

---

### 3. Jó Gyakorlat: Arduino UNO – Vezetékes Ethernet Megoldás

Ez a gyakorlat egy harmadik megközelítést mutat be: a hálózati stabilitásra helyezi a hangsúlyt a vezetékes Ethernet Shield W5100 segítségével, megkerülve a Wi-Fi-t.

#### Óravázlat (Magyar)

Cím	Arduino UNO – Vezetékes Stabilitás Ethernet Shield W5100-zal
STE(A)M	Technológia, Hálózati ismeretek, Mérnöki tudományok, Fizika
Tárgyak	
Korosztály	14–18 (Szakképzés)



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

**Időtartam** 6 óra (3 x 2 órás alkalom)

**Kulcskérdés** Mikor van szükség vezetékes kapcsolatra az adatgyűjtéshez, és hogyan konfiguráljuk az alacsony szintű hálózati beállításokat (pl. IP címeket)?

### Rövid bemutatás

Ez a projekt az Arduino UNO-t és a DHT11 érzékelőt az **Ethernet Shield W5100** segítségével köti össze a hálózattal. Mivel az Arduino UNO nem rendelkezik beépített Wi-Fi-vel, vezetékes kapcsolaton keresztül éri el az internetet. A kód a `Ethernet.h` könyvtárat használja, és bemutatja, hogyan kell konfigurálni a hálózati paramétereket, például a **statikus IP-címet** és a MAC-címet.

### Célok/Tanulási Eredmények ☞ Kategória

#### Tanulási Eredmények (LOs)

Képesek lesznek a hardveres infrastruktúrától függő könyvtárak kiválasztására és használatára (pl. `Ethernet.h` a `WiFi.h` helyett). Képesek lesznek egy helyi

**Diákoknak** hálózati interfész inicializálására (MAC, IP cím beállítása). Képesek lesznek vezetékes és vezeték nélküli adatátviteli megoldások előnyeinek/hátrányainak megvitatására (C2). Képesek lesznek a hálózati kliens objektumok kezelésére (`client.connect()`, `client.println()`, `client.stop()`).

**Tanároknak** Képesek lesznek a hálózati technológiák mélyebb megértésére (IP címzés, DHCP vs. statikus beállítás).

#### Eszközök és Adatok ✂

Kategória	Megnevezés
<b>Hardver</b>	Arduino UNO, Ethernet Shield W5100, DHT11 érzékelő, Ethernet kábel.
<b>Szoftver/Infrastruktúra</b>	Arduino IDE (Ethernet.h és SPI.h könyvtárak), PHP webszerver SQL adatbázissal (a kód a <code>example.com</code> címet használja mint helyettesítő szerver).
<b>Adatok</b>	MAC-cím, statikus IP-cím, hőmérsékleti és páratartalmi adatok.



Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Tempus Public Foundation. Neither the European Union nor the granting authority can be held responsible for them.

## Iránymutatások (Óravázlat) ?

Rész	Tevékenység
<b>1. Rész: Hálózati Inicializálás (2 óra)</b>	Az Ethernet Shield bekötése és inicializálása. A MAC-cím és a statikus IP-cím fogalmának bemutatása, és ezek jelentőségének megértése a hálózati kommunikációban. A DHT11 adatainak olvasása. A HTTP POST kérés felépítése a <code>client.println()</code> parancsokkal (a <b>Vezetékes (2 óra)</b>
<b>2. Rész: Adatküldés (2 óra)</b>	HTTP/1.1 protokoll manuális Adatküldés implementálása). A TCP kapcsolat indításának és lezárásának fontossága ( <code>client.connect()</code> , <code>client.stop()</code> ). A szerver válaszáinak kezelése ( <code>client.readString()</code> ). A vezetékes
<b>3. Rész: Hibaelemzés és Portolás (2 óra)</b>	megoldás portolásának képessége: a Hibaelemzés szerveroldali PHP/SQL kód változatlanul használható. A vezetékes/vezeték nélküli megoldások összehasonlítása, a projekt lezárása.

Átváltható készségek: Az analitikus és problémamegoldó készségek fejlesztése a hálózati hibák diagnosztizálása során (pl. a `Connection failed` üzenet elemzése).

## Pedagógiai Fókusz (Források alapján)

Kutatási módszertan (RBT): A diákoknak meg kell érteniük a *változókat* (pl. IP-cím, port, szerver) és a feltételeket (pl. a DHCP hibája esetén a statikus IP-re való váltás logikája) a kutatási terv részeként (A2.1).

Mérnöki tudományok: A vezetékes protokollok stabilitásának és megbízhatóságának elemzése, szemben a Wi-Fi-vel járó esetleges hibákkal.