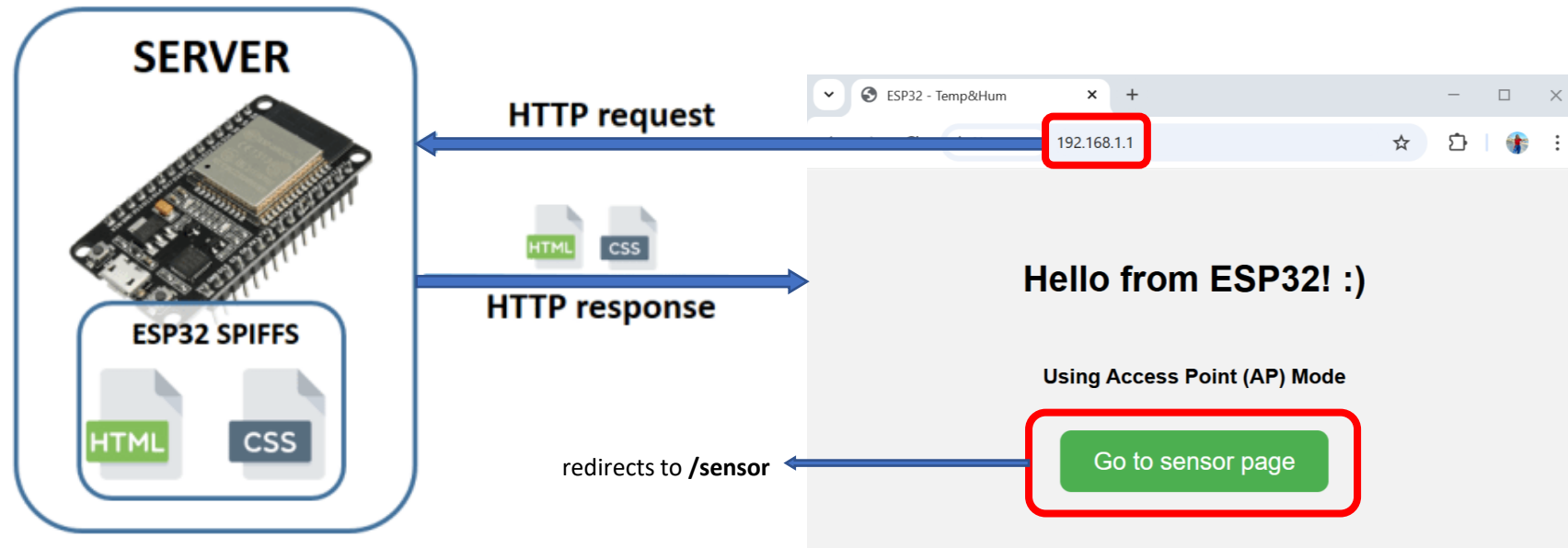


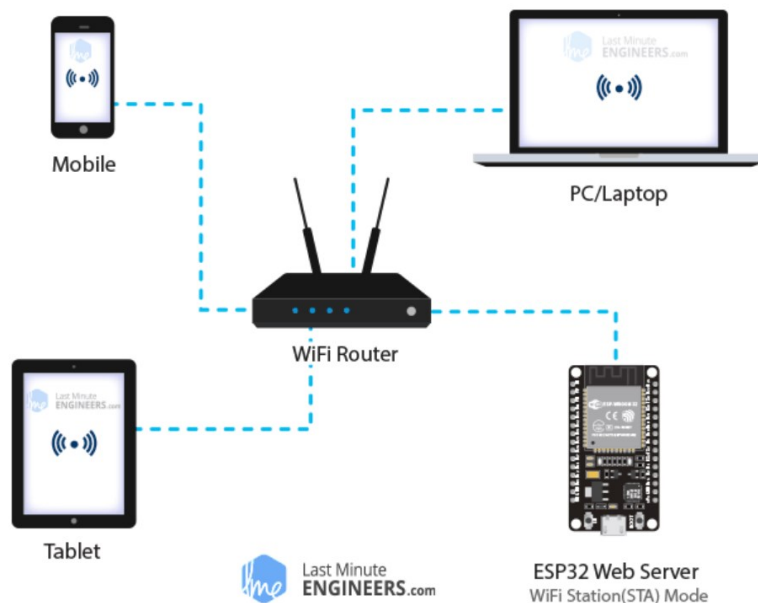
# ESP32 kao web server



# ESP32 WiFi režimi rada

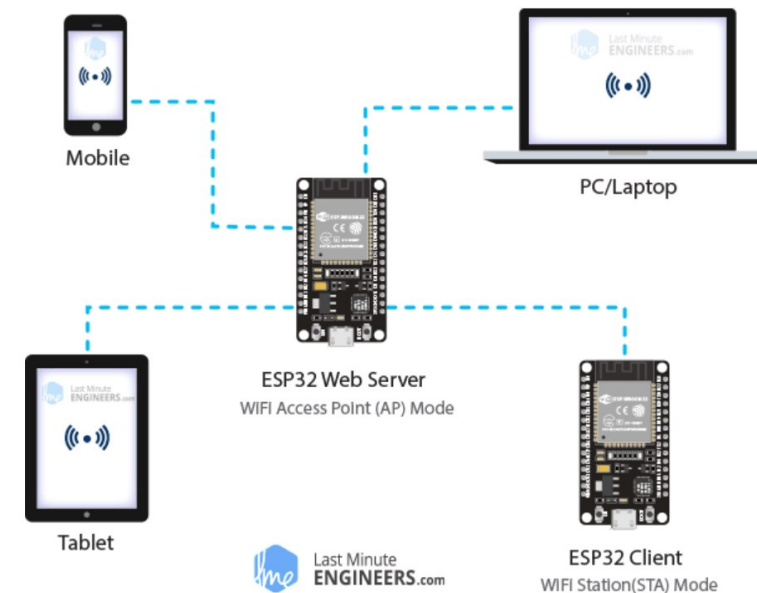
## Station (STA) Mode

- ESP32 se povezuje na postojeću WiFi mrežu



## Soft Access Point (AP) Mode

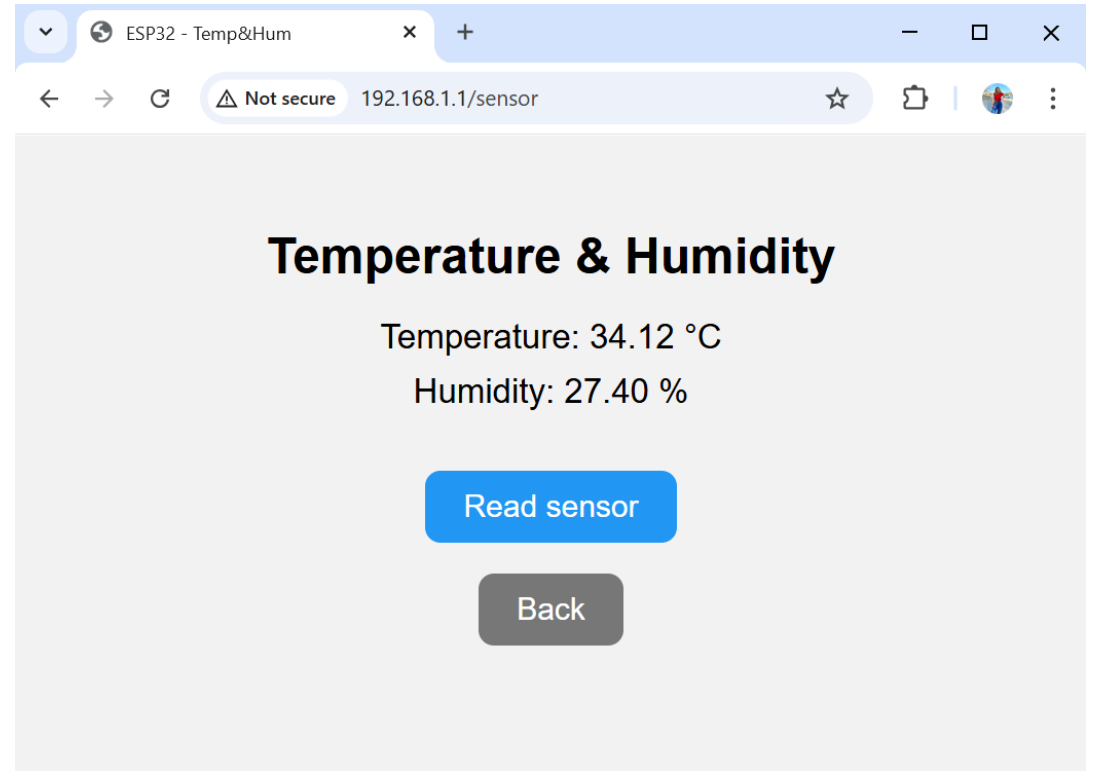
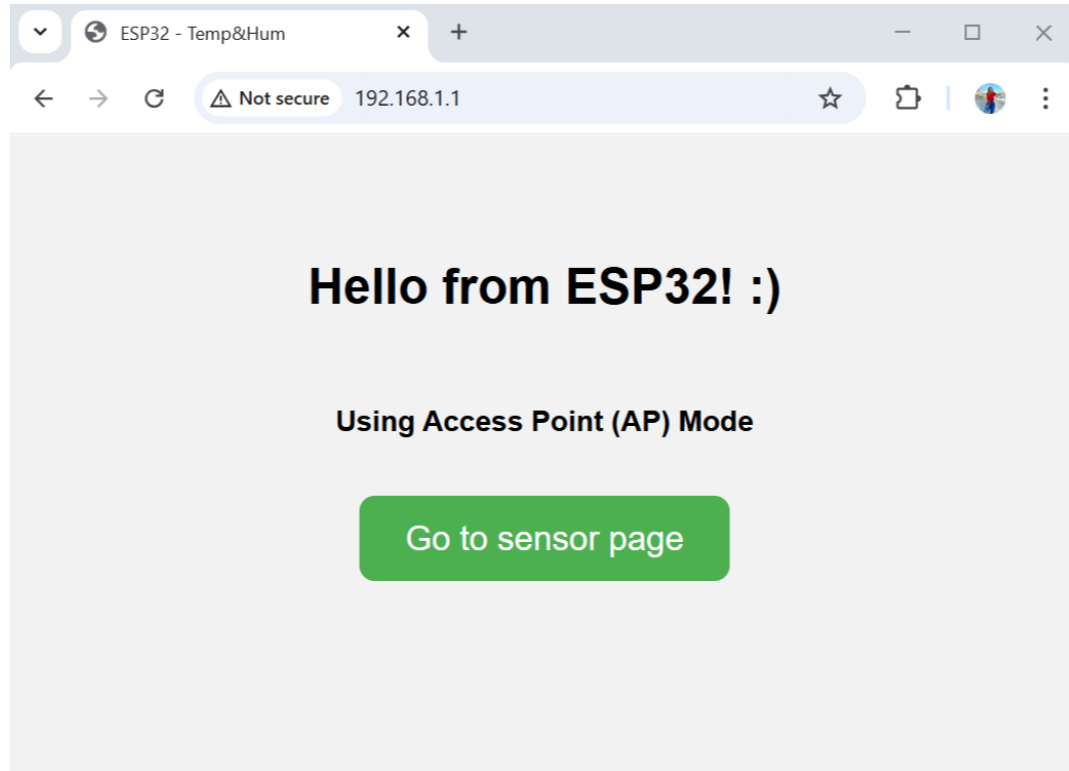
- ESP32 kreira svoju privatnu WiFi mrežu u kojoj preuzima ulogu pristupne tačke (AP)
- Za razliku od klasičnog WiFi rutera, u ovom slučaju ESP32 kao pristupna tačka nema priključak na WAN (tj. pristup internetu)
- Maksimalan broj uređaja koji mogu da se priključe na mrežu je ograničen na 5



# Primer 1: Web server u AP režimu

- Očitavanje temperature i vlažnosti vazduha pomoću web servera
- *ESP32\_AP\_server.ino*
- Nakon upisa serverske aplikacije u ESP32, uređaj na kojem se izvršava klijentska aplikacija (npr. telefon ili laptop) treba povezati na novokreiranu WiFi mrežu
  - SSID: ESP32
  - Pass: 12345678
- Iz web browsera kao klijentske aplikacije povezati se na web stranicu <http://192.168.1.1>

# Primer 1: Web server u AP režimu



# SPIFFS (SPI Flash File System)

- **SPIFFS** (SPI Flash File System) organizuje memorijski prostor eksterne flash memorije na sličan način kao na USB memorijskim modulima, ali jednostavnije i sa više ograničenja (npr. nisu podržani direktorijumi)
- Osnovne funkcionalnosti uključuju čitanje, upis, zatvaranje i brisanje datoteka
- Najčešće primene SPIFFS na ESP32 ploči (kapaciteta 4 MB):
  - Kreiranje konfiguracionih datoteka sa podešavanjima
  - Logovanje podataka
  - Čuvanje HTML i CSS datoteka za realizaciju web servera
  - Čuvanje slika i ikonica

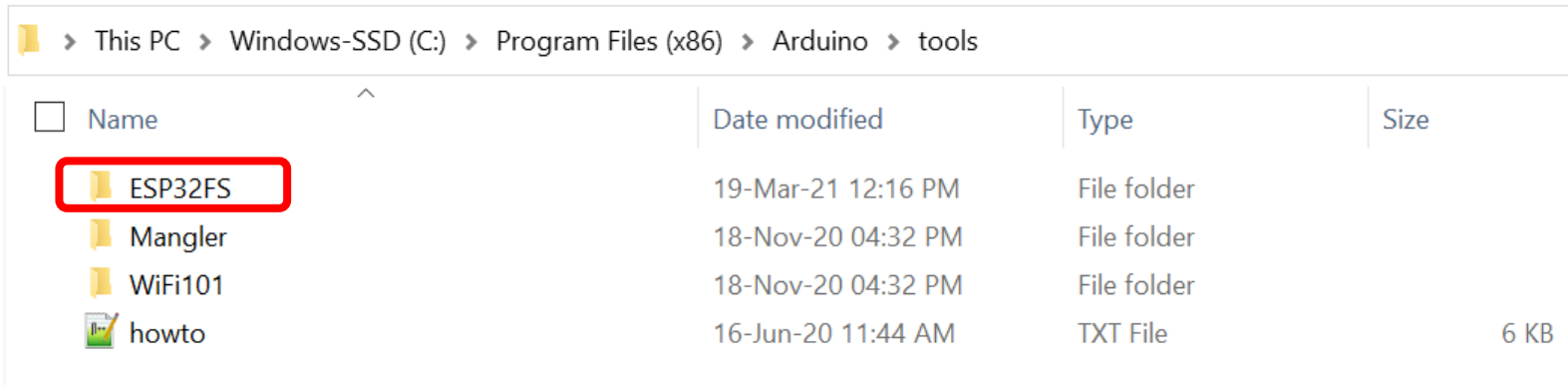
# Instalacija dodatka: Arduino ESP32 Filesystem Uploader

Dodatak za Arduino IDE dozvoljava direktno učitavanje datoteka sa foldera na PC na SPIFFS. Instalacija se obavlja prateći sledeće korake:

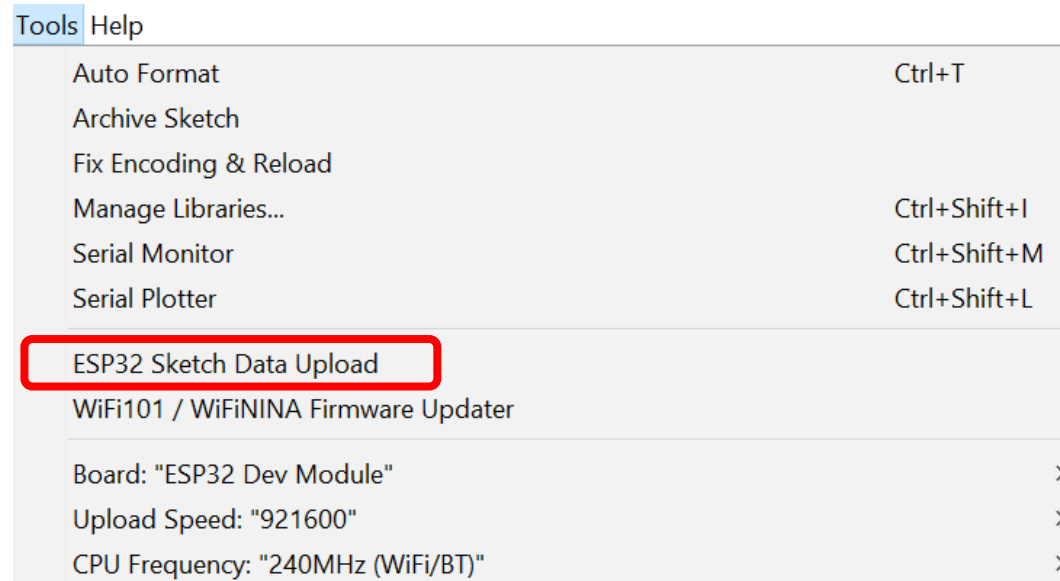
1. Kliknuti na link <https://github.com/me-no-dev/arduino-esp32fs-plugin/releases/> i ispod **Release for esptool\_py** preuzeti *ESP32FS-1.0.zip*



- Otići u folder gde je instaliran Arduino IDE i raspakovati sadržaj .zip datoteke u **tools** folder:

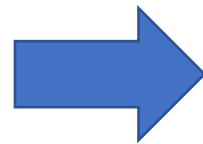
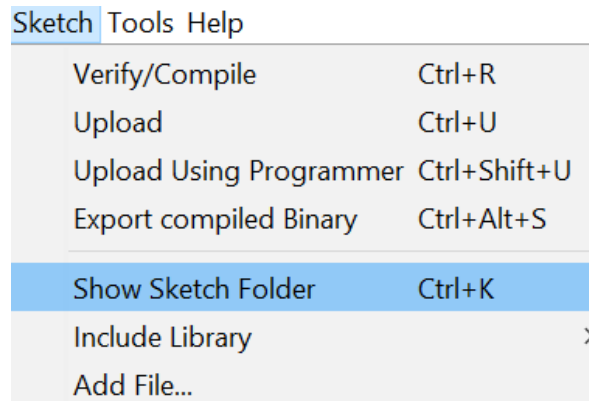


- Restartovati Arduino IDE. Ukoliko je dodatak korektno instaliran, u meniju Tools pojavíće se opcija **ESP32 Sketch Data Upload**.



# Primer 2: Učitavanje datoteka na ESP32 SPIFFS

- Kreirati nov prazan Arduino program (sketch) i snimiti ga pod nazivom *SPIFFS\_demo*
- Pomoću opcije Sketch -> **Show Sketch Folder** otvoriti folder u kojem se nalazi program. U okviru foldera gde je *SPIFFS\_demo.ino* potrebno je kreirati folder *data* koji će biti korišćen za smeštanje datoteka koje se učitavaju u SPIFFS



data



SPIFFS\_demo

- U folderu *data* kreirati novu datoteku pod nazivom *primer.txt* i u nju uneti proizvoljan sadržaj
- Pomoću opcije Tools -> **ESP32 Sketch Data Upload** učitati sadržaj foldera *data* na SPIFFS

```

SPIFFS Image Uploaded
Configuring flash size...
Auto-detected Flash size: 8MB
Compressed 1638400 bytes to 3095...
Writing at 0x00670000... (100 %)
Wrote 1638400 bytes (3095 compressed) at 0x00670000 in 0.0 seconds (effective 2
Hash of data verified.

Leaving...
Hard resetting via RTS pin...

```

- Iskopirati programski kod i spustiti ga na ESP32 ploču 

```

#include "SPIFFS.h"

void setup()
{
  Serial.begin(115200);

  if(!SPIFFS.begin(true))
  {
    Serial.println("Greska prilikom
inicijalizacije SPIFFS");
    return;
  }

  File file = SPIFFS.open("/primer.txt");
  if(!file)
  {
    Serial.println("Greska prilikom otvaranja
datoteke");
    return;
  }

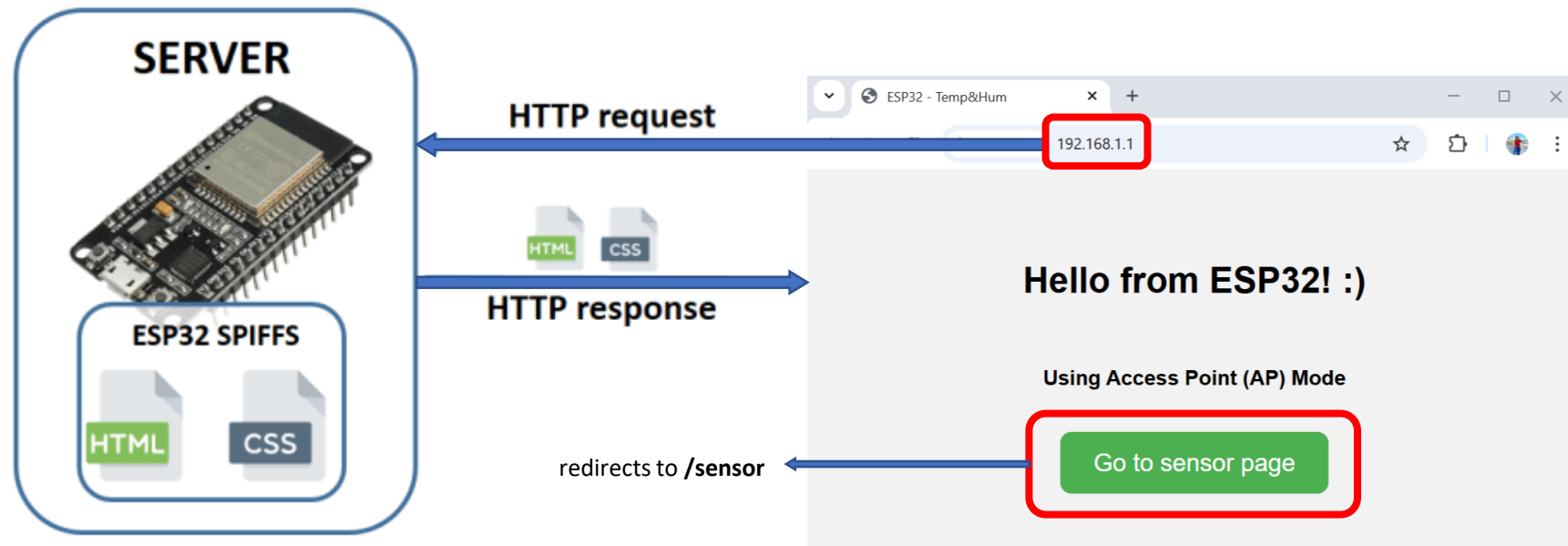
  Serial.println("Sadrzaj datoteke:");
  while(file.available())
    Serial.write(file.read());
  file.close();
}

void loop()
{
}

```

# Primer 3: Web server u STA režimu uz podršku SPIFFS

- Slično primeru 1, serverska aplikacija očitava temperaturu i vlažnost vazduha pomoću senzora.
- Web interfejs se sastoji od dve stranice: početne stranice uređaja i posebne stranice za prikaz podataka sa senzora, između kojih se prelazi pritiskom na odgovarajuća dugmad.
- U ovom slučaju umesto kreiranja AP, ESP32 se povezuje na lokalnu WiFi mrežu u STA režimu.



# Instalacija potrebnih biblioteka

## Instalacija biblioteke **ESPAsyncWebServer**:

- Klikom na link preuzeti biblioteku [ESPAsyncWebServer](#)
- Raspakovati sadržaj .zip datoteke
- Preimenovati folder iz *ESPAsyncWebServer-master* u *ESPAsyncWebServer*
- Prebaciti folder u *libraries*, na lokaciji gde je instaliran Arduino IDE

## Instalacija biblioteke **AsyncTCP**:

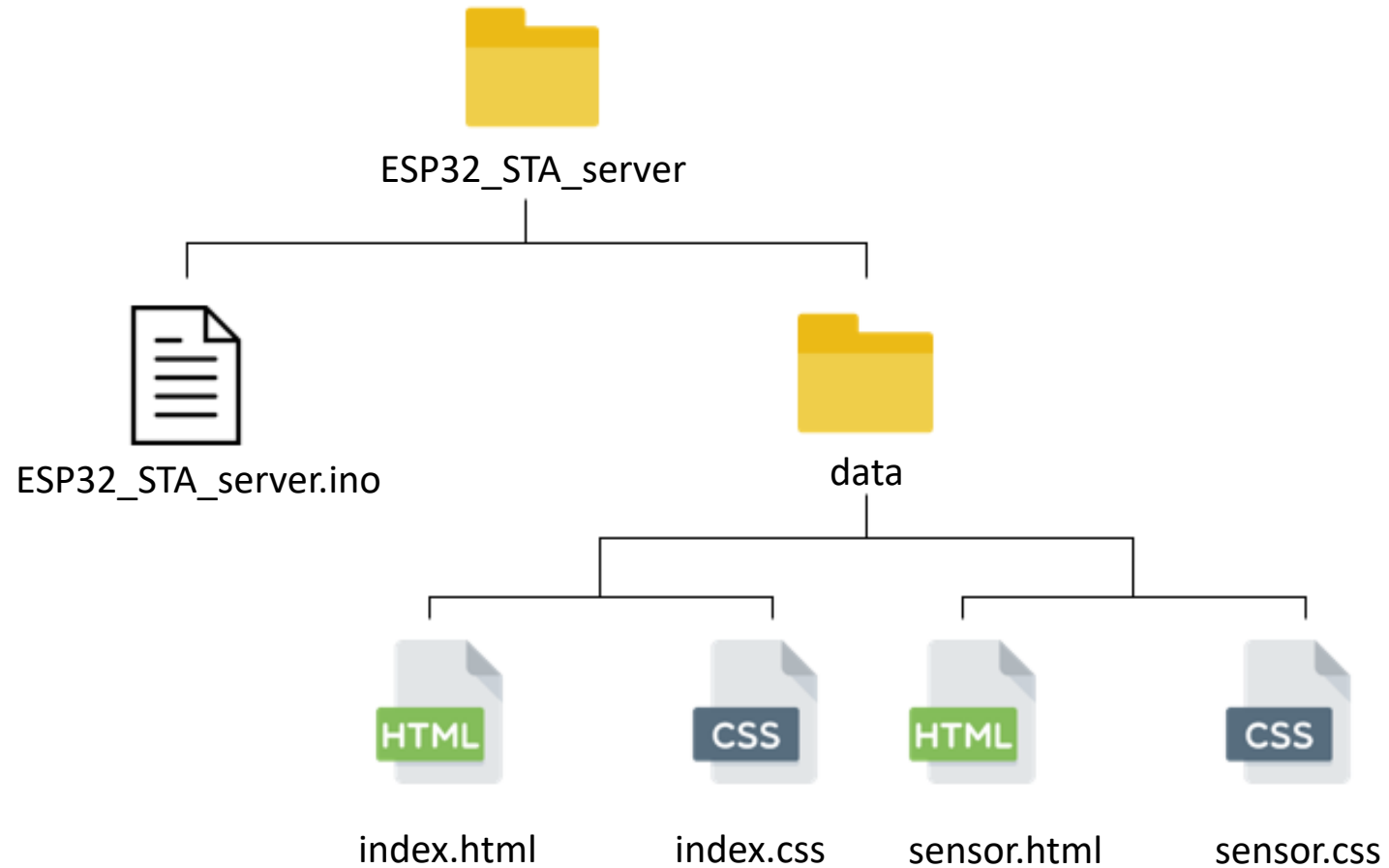
- Klikom na link preuzeti biblioteku [AsyncTCP](#)
- Raspakovati sadržaj .zip datoteke
- Preimenovati folder iz *AsyncTCP-master* u *AsyncTCP*
- Prebaciti folder u *libraries*, na lokaciji gde je instaliran Arduino IDE

Restartovati Arduino IDE

# Logika aplikacije

- ESP32 izvršava serversku aplikaciju baziranu na biblioteci [ESPAsyncWebServer library](#)
- HTML i CSS datoteke su smeštene na ESP32 SPIFFS
- Upućivanjem zahteva serveru preko određenog URL iz klijentske aplikacije (web browser), ESP32 u odgovoru šalje zahtevane datoteke
- U okviru web stranice postoje tzv. *placeholders* za stanje sa senzora i ono se upisuje direktno u HTML kod, između znakova '%', odnosno **%TEMP%** i **%HUM%**
- *ESP32\_STA\_server*

# Organizacija datoteka



# Zadatak

- Napisati program koji proširuje postojeći ESP32 web server za očitavanje temperature i vlažnosti tako što osim stranice “/sensor”, sadrži i posebnu web stranicu “/time” za prikaz trenutnog vremena, pomoću NTP servera pool.ntp.org.
- Na stranici “/time” prikazati dugme za očitavanje vremena. Nakon pritiska na dugme, potrebno je prikazati trenutno vreme. Obezbediti dugme za povratak na početnu stranicu.
- ESP32 je potrebno konfigurisati u station (STA) modu.