



Univerzitet u Novom Sadu
Fakultet tehničkih nauka

Dokumentacija za projektni
zadatak

**Web Site-Live Sensor Data
Charting via MicroC**

Predmet: WEB Bazirani Merno-Akvizicioni Sistemi

Student:
Marko Bjelica EM 32/2017

Mentori:
Profesor Platon Sovilj
Asistent Ivan Gutai

SADRŽAJ

SADRŽAJ

1. UVOD.....	2
2. MOTIVACIJA	2
3. ALAT.....	2
4. POJEDINAČNI IZGLED SVAKE STRANE I NJIHOVA NAMJENA	3
4.1 POČETNA STRANA	3
4.2 STRANA: O SISTEMU REGISTRACIJE	4
4.3 STRANA: O MIKROKONTROLERU.....	5
4.4 STRANA: GRAFIK	6

1. UVOD

Na predmetu WEBMAS je napravljen projekat koji sadrži stranicu realizovanu u HTML kodu, registraciju korisnika i grafički prikaz podataka sa mikro-kontrolera. Projekat je realizovan uz pomoć C programskega jezika, ESP-WROOM-32 i potenciometra od $100\text{K}\Omega$. Na ovom sajtu korinsink dobija mogućnost da prati i kontroliše podatke sa senzora uz pomoć samog senzora, Arduino IDE, mikrokontrolera i stranice.

Na sajtu je realizovan grafik koji prikazuje napon regulisan potenciometrom i sistemom za upozorenje na greške, iskačući prozori i bojom koordinisani podaci.

2. MOTIVACIJA

Korišćenje embedded elektronskih sistema i naročito, mikrokontrolera opšte namjene se povećala zadnjih nekoliko godina zahvaljujući dostupnosti integriranih kola, mala cijena i jednostavnost u programiranju. Mikrokontroler je veoma jaka alatka koja omogućuje dizajneru da napravi skoro neograničeno varijacija jednostavno modifikujući program.

Zbog velikog značaja u industriji i zbog ostalih predmeta koji se dosta oslanjaju na mikrokontrolere, ovaj projekat pruža jako dobru osnovu za bilo koju nadogradnju.

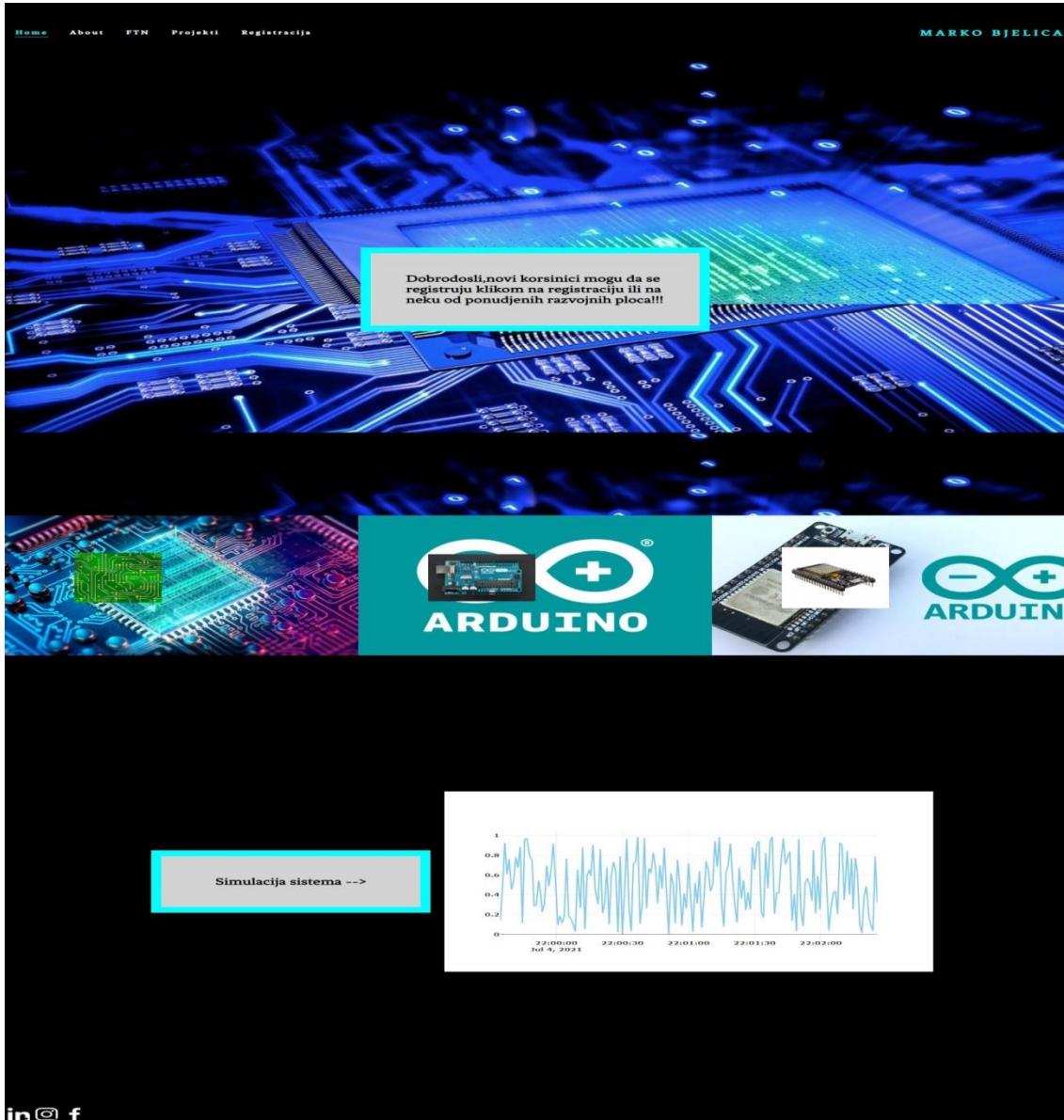
3. ALAT

Za izradu veb sajta korišćeni su HTML, CSS i JavaScript .

- Osnovna struktura stranice je napravljena pomoću HTML-a, dalje je nadogradjena uz pomoć CSS-a i JavaScript-a .
- Za dalju dekoraciju i izgled sajta korišćen je css kao i biblioteke kao što su Bootstrap, cdnjs, svg...
- Programski jezik JavaScript je korišćen za sve funkcionalnosti stranice i uz njega je korišćena biblioteka jQuery i biblioteke kao što su ChartJS, HighCharts, PlotlyJS. Uz pomoć svih navedenih stvari napravljeni su grafici za prikazivanje željenih podataka .
- Za registraciju/login korišćene su sledeće stvari:
- Node.js – višeplatformsko JavaScript okruženje koje je korišćeno za izvršavanje JavaScript-a na serverskoj strani
- Npm init – komanda u JSON jeziku koja kreira package.json fajl za naš projekat koji sadrži informacije projekta
- Express – Node.js biblioteka koja pruža razvijanje web aplikacije
- EJS(Embedded JavaScript Templating) – predstavlja šablon koji koristi Node.js za kreiranje HTML šablona sa minimalno koda
- --save-dave nodemon – nadgleda promjene u kodu i automatski restartuje server da ih primjeni
- Bcrypt(password-hashing) – funkcija enkripcije šifre, algoritam koji maskira šifru
- Passportjs – biblioteka autentikacionih funkcija, autentikacioni proces koji štiti privatni sadržaj
- Session – radi kao kolačići sajta, čuva informaciju i prenosi kroz stranice
- Flash – renderuje i preusmjerava, prikazuje poruke greške
- Method-override – omogućava upotrebu HTTP glagola kao PUT i DELETE na mjestima gdje ih klijent ne podržava

4.POJEDINAČNI IZGLED SVAKE STRANE I NJIHOVA NAMJENA

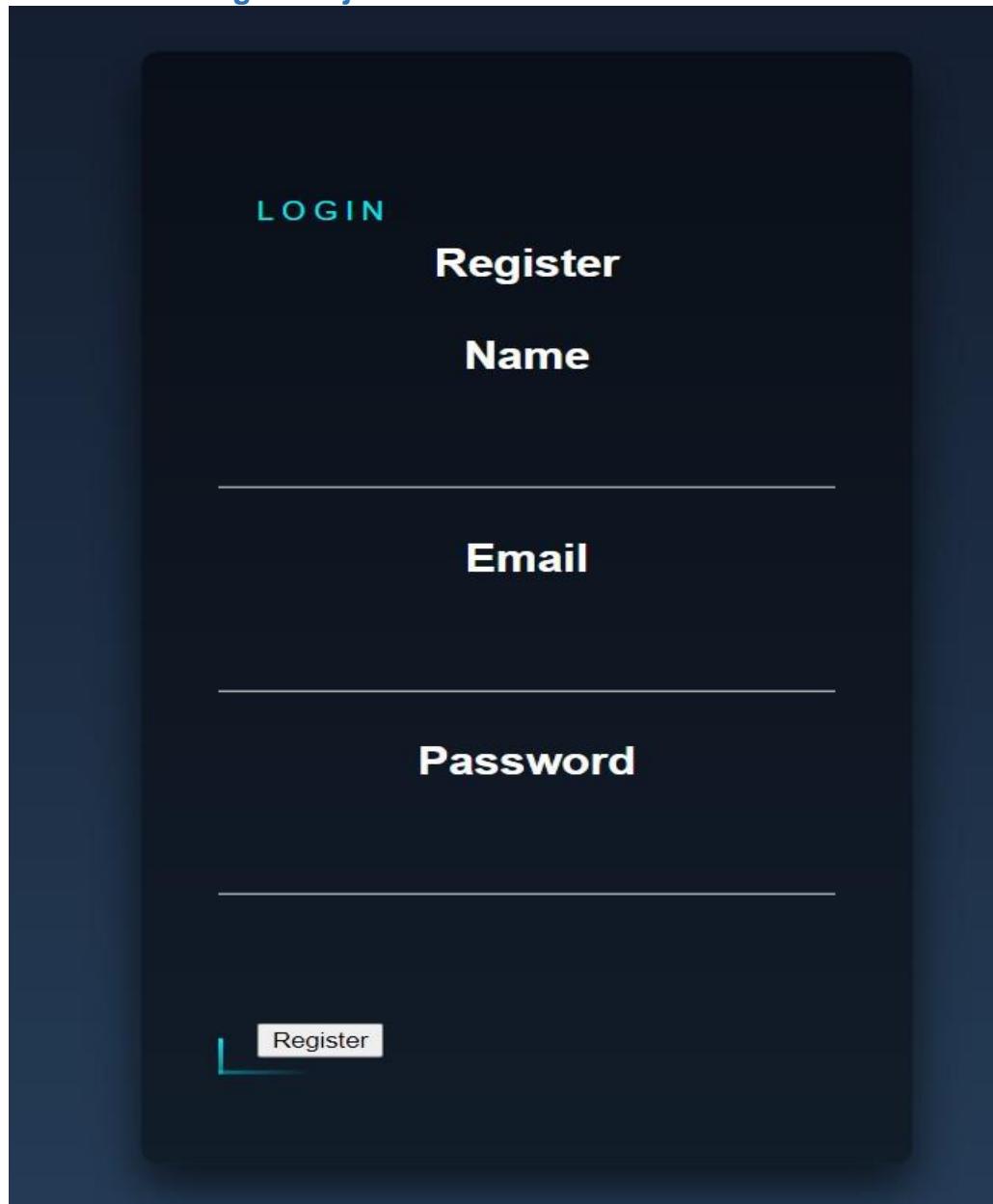
4.1 Početna strana



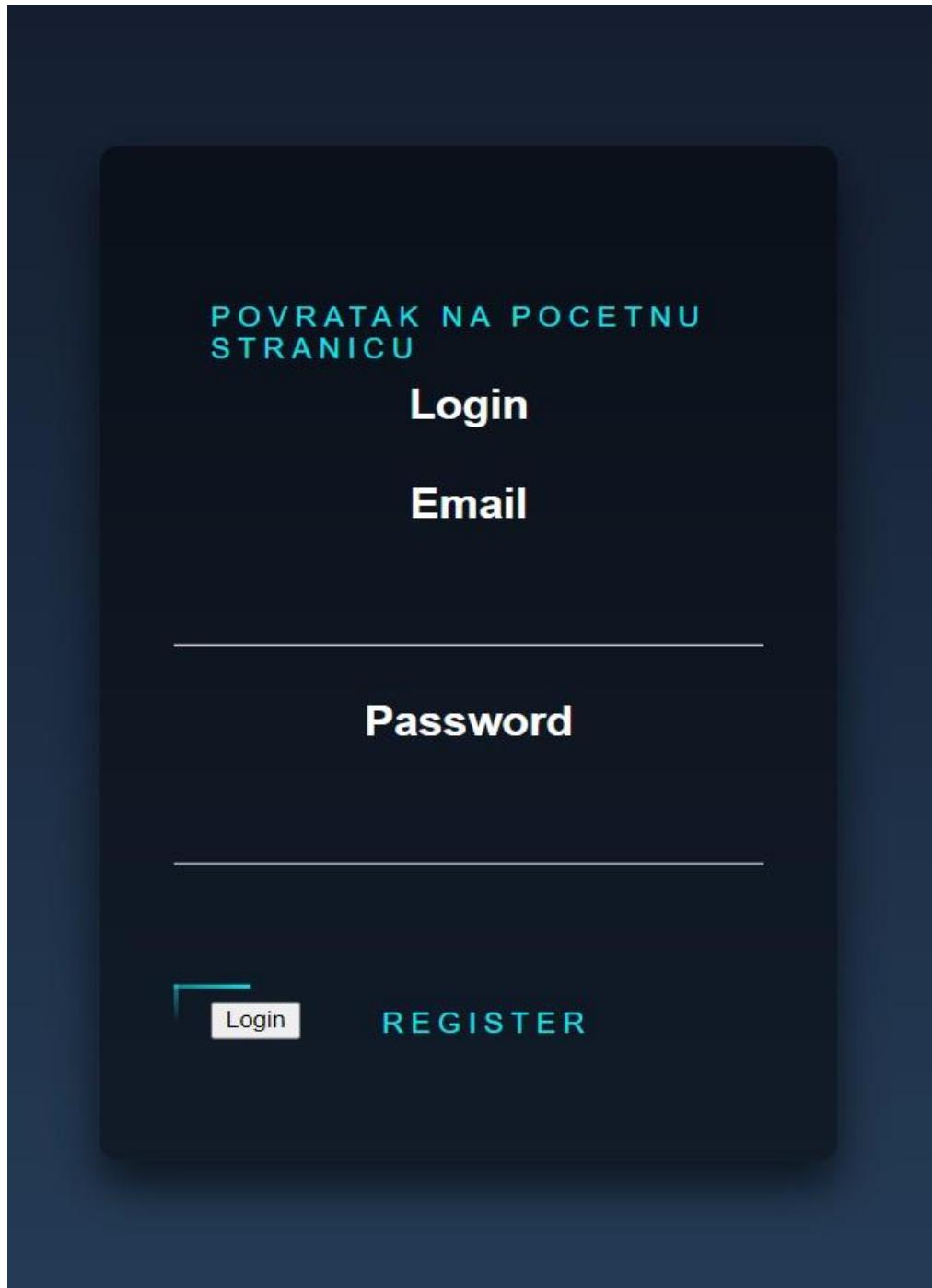
Slika 1 - Početna strana

- Početna strana sajta izgleda kao na slici.
- Na stranici se nalazi navigaciona linija(Home,About,Projekti,FTN,Registracija).
- Pritisak na About,vodi do stranice koja ukratko opisuje sistem.
- Pritisak na Registraciju, vodi na stranicu koja nudi formualr za popunjavanje i registraciju novih klijenata.
- Prelaskom miša preko slika, pojavljuju se linkovi koji vode do About stranice i Registracije.

4.2 Strana: O sistemu registracije

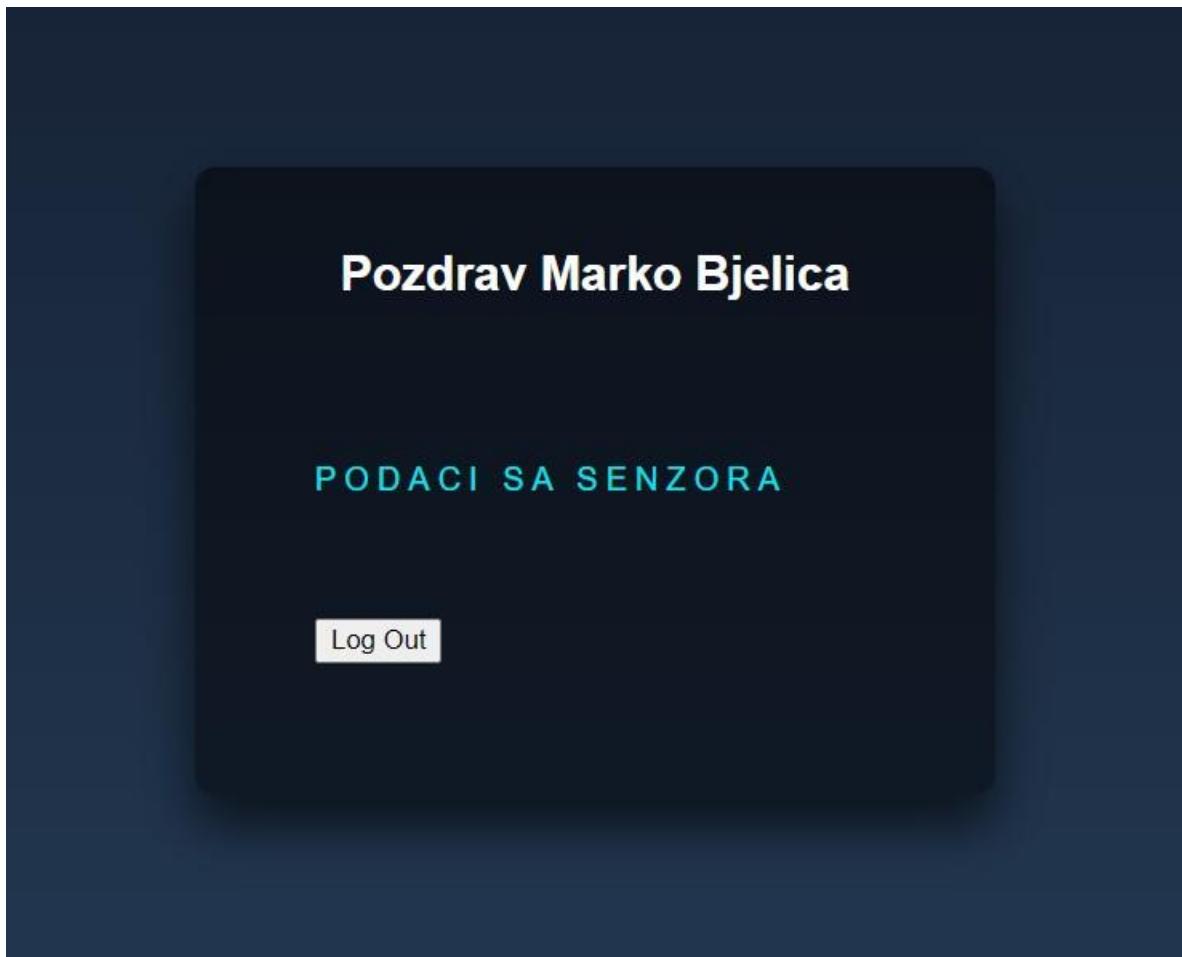


Slika 2 – Registracija



Slika 3 - LogIn

- Kada se pristupi registracionom linku, on vodi do registracionog formulara.
- Prvo se pojavljuje stranica sa formularom za registraciju, kada se popune sva navedena polja, prebacuje korisnika na LogIn formular.
- LogIn formular se popunjava istim informacijama kao i za registraciju i ukoliko se poklapaju informacije on vas salje dalje, u suprotnom dobija se poruka o grešci.
- U svakom momentu registracije i logovanja omogućen je povratak na početnu stranicu i prekid registracije.



Slika 4 – Uspješno logovanje i pristup podacima

[WEBMAS-Projekat\uspjesna-registracija.mp4](#)

[WEBMAS-Projekat\pogresan-unos-podataka.mp4](#)

4.3 Strana: O mikrokontroleru



Slika 5 – MicroC ESP-WROOM-32

- Na slici je prikazan mikrokontroler koji je zadužen za upravljanje vanjskih periferija, senzora, motora, potenciometara...
- ESP32 je serija jeftinih sistema male snage sa čipovima mikrokontrolera koji posjeduju integriran Wi-Fi i dual-mode Bleutooth.
- ESP32 je kreirao i razvio Espressif Systems sa sjedištem u Šangaju.



Slika 6 – Potenciometar $100\text{K}\Omega$

- Na slici je prikazan potenciometar koji je korišćen zajedno sa mikrokontrolerom.
- Povezan je na mikrokontroler sa 3 žice, jedna na napajanje(3V3), druga na uzemljenje(GND) i treća na pin 34.



Slika 7 – Potenciometar I ESP32

- Da bi ovo sve funkcionalo, potreban je odgovarajući firmware.

```
// Import required libraries
#ifndef ESP32
    #include <WiFi.h>
    #include <ESPAsyncWebServer.h>
    #include <SPIFFS.h>
#else
    #include <Arduino.h>
    #include <ESP8266WiFi.h>
    #include <Hash.h>
    #include <ESPAsyncTCP.h>
    #include <ESPAsyncWebServer.h>
    #include <FS.h>
#endif
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

// Replace with your network credentials
const char* ssid = "ssid";
const char* password = "sifra";

// Create AsyncWebServer object on port 80
AsyncWebServer server(80);

const int potPin = 34;

float PotValue = 0;

String readPotValue = String(float(analogRead(potPin)/1170.0));

void setup() {
    // Serial port for debugging purposes
    Serial.begin(115200);

    // Initialize SPIFFS
    if(!SPIFFS.begin()){
        Serial.println("An Error has occurred while mounting SPIFFS");
        request->send(SPIFFS, "/index.html");
    };
    server.on("/napon", HTTP_GET, [] (AsyncWebRequest *request){
        request->send_P(200, "text/plain", readPotValue.c_str());
    });

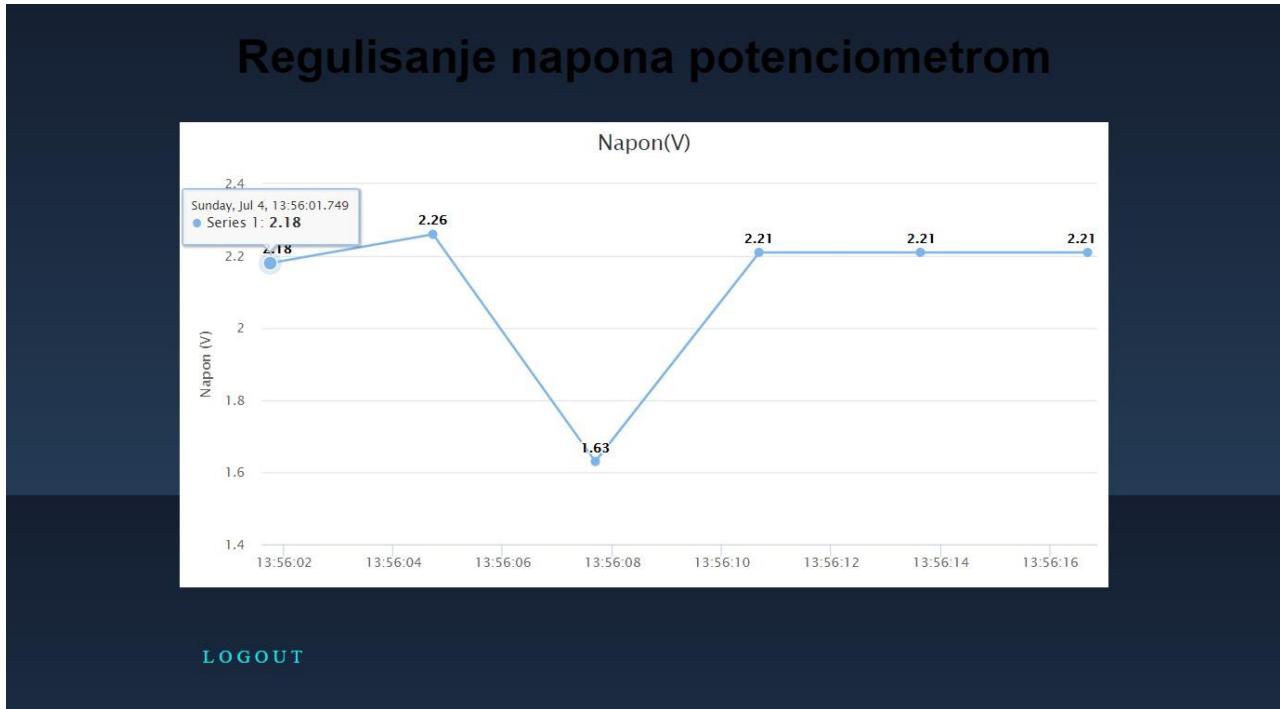
    // Start server
    server.begin();
}

void loop(){
    readPotValue = analogRead(potPin)/1170.0;
    Serial.println(readPotValue);
    delay(200);
}
```

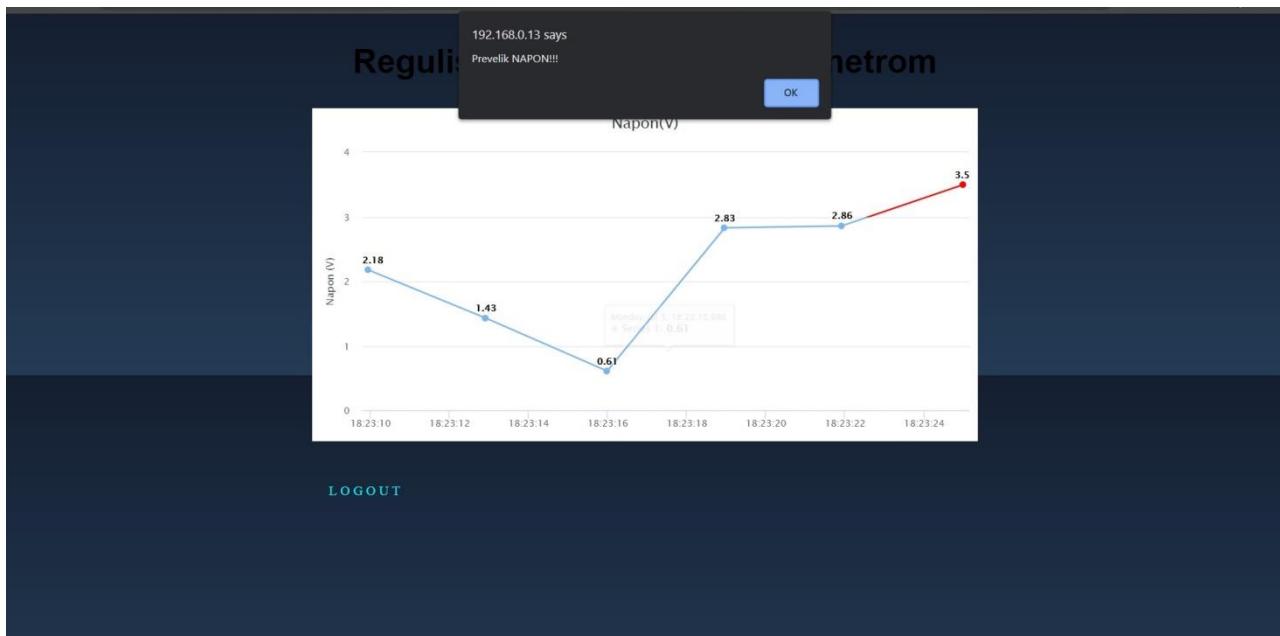
Slika 8 – Prikaz koda u ArduinoIDE

- Na slici je prikazan firmware koji je napravljen u aplikaciji ArduinoIDE.
- Zadužen je za čitanje vrijednosti sa pina 34 koje se menjaju putem potenciometra.
- Predstavlja kod koji je zadužen da napravi Web Server koji će napraviti most izmedju mikrokontrolera i web sajta.
- Povezuje se na WiFi, zauzima određeni port i pravi IP adresu, pokreće asinhroni HTTP zahtjev čitanja informacija.
- Putem SPIFFS sistema, čuva fajlove na samom mikrokontroleru koji su zaduženi za izgled stranice koja grafički dostavlja rezultate.

4.4 Strana: Grafik



Slika 9 – Grafički prikaz podataka



Slika 10 – Prikazivanje greški