#include <WiFi.h>

#include <HTTPClient.h>

#include <ArduinoJson.h>

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

#include <WiFiClientSecure.h>

#include <base64.h>

#define SCREEN\_WIDTH 128

#define SCREEN\_HEIGHT 64

#define OLED\_RESET -1

Adafruit\_SSD1306 display(SCREEN\_WIDTH, SCREEN\_HEIGHT, &Wire, OLED\_RESET);

const char\* ssid = "SSID";

const char\* password = "password";

String payload = "";

const char\* smtpServer = "smtp.gmail.com";

const int smtpPort = 465;

const char\* emailSender = "dharani2004d@gmail.com";

const char\* emailSenderPassword = "Dharneeee@2004";

const char\* emailRecipient = "sivanikrish65@gmail.com";

const char\* subject = "Stock Alert";

const char\* message = "Alert: Stock price dropped below threshold!";

WiFiClientSecure client;

void connectWiFi() {

display.clearDisplay();

display.setTextSize(2);

display.setTextColor(SSD1306\_WHITE);

display.setCursor(0, 0);

display.println("Connecting to WiFi...");

display.display();

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

}

display.clearDisplay();

display.setCursor(0, 0);

display.println("Connected to WiFi");

display.display();

delay(2000);

}

void resetDisplay() {

display.clearDisplay();

display.setTextColor(SSD1306\_WHITE);

display.setCursor(0, 0);

}

void readPrice(int x, int y, const String& stockName) {

String httpRequestAddress = "https://finnhub.io/api/v1/quote?symbol=" + stockName + "&token=c1tjb52ad3ia4h4uee9g";

HTTPClient http;

int httpCode;

http.begin(httpRequestAddress);

httpCode = http.GET();

if (httpCode > 0) {

DynamicJsonDocument doc(1024);

String payload = http.getString();

Serial.println(payload);

deserializeJson(doc, payload);

float previousClosePrice = doc["pc"];

float currentPrice = doc["c"];

float differenceInPrice = ((currentPrice - previousClosePrice) / previousClosePrice) \* 100.0;

resetDisplay();

display.setTextSize(2);

display.setCursor(x, y);

display.println(stockName);

if (differenceInPrice < 0.0) {

display.setTextColor(SSD1306\_WHITE);

} else {

display.setTextColor(SSD1306\_WHITE);

}

display.setTextSize(2);

display.setCursor(x, y + 25);

display.print(currentPrice, 2);

display.println(" USD");

display.setTextSize(2);

display.setCursor(x, y + 50);

display.print(differenceInPrice, 2);

display.println("%");

display.display();

// Email sending code

if (client.connect(smtpServer, smtpPort)) {

client.println("EHLO client");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

String auth = "AUTH LOGIN " + base64::encode(String(emailSender)) + " " + base64::encode(String(emailSenderPassword));

client.println(auth);

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.println("MAIL FROM:<" + String(emailSender) + ">");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.println("RCPT TO:<" + String(emailRecipient) + ">");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.println("DATA");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.println("Subject: " + String(subject));

client.println();

client.println(message);

client.println(".");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.println("QUIT");

delay(100);

if (client.available()) {

Serial.println(client.readString());

}

client.stop();

} else {

Serial.println("Connection to email server failed");

}

} else {

resetDisplay();

display.setTextSize(1);

display.setCursor(0, 0);

display.println("Error in HTTP request");

display.display();

}

http.end();

}

void setup() {

display.begin(SSD1306\_SWITCHCAPVCC, 0x3C);

display.clearDisplay();

display.setTextColor(SSD1306\_WHITE);

display.setTextSize(1);

display.setCursor(0, 0);

display.println("Stock Prices Tracker");

display.display();

connectWiFi();

}

void loop() {

readPrice(0, 0, "AAPL");

delay(3000);

display.clearDisplay();

readPrice(0, 0, "AMZN");

delay(3000);

display.clearDisplay();

readPrice(0, 0, "TSLA");

delay(3000);

display.clearDisplay();

readPrice(0, 0, "MSFT");

delay(3000);

readPrice(0, 0, "PFE");

delay(3000);

display.clearDisplay();

readPrice(0, 0, "OXY");

delay(3000);

display.clearDisplay();

readPrice(0, 0, "EBAY");

delay(3000);

display.clearDisplay();

readPrice(0,0, "FDX");

delay(3000);

display.clearDisplay();

}