Elektrotehnička škola ZAGREB

Konavoska 2

10000 Zagreb

ČUVAR BLAGA



Projektni tim: Dominik Ležaić, Roko Bogdanović, David Krznar

Mentor: Danijel Eskeričić, mag.ing.el., nastavnik-mentor

**Opis ideje:**

Ideja je razvijena tijekom osvajanja nagrada naše škole na smotrama inovativnih učeničkih radova kroz zadnje tri godine. Realizacijom ovoga projekta osvojeni trofeji, odnosno pehari, izloženi su u ioT učionici odnosno laboratoriju uz sigurnosni inovativni sustav.

**Opis projekta:**

Čuvar trofeja služi čuvanju trofeja odnosno pehara koje su osvojili učenici u ime škola, škola kao najbolji izlagač na smotrama inovativnih radova učenika te nagrade osvojene za najboljega mentora, koji mentorira i ovaj projekt. Čuvar trofeja sastavljen je od mikroupravljača ESP32 koji pomoću magnetskih senzora/prekidača aktivira alarm ako ih netko pokuša ukrasti s postolja i šalje upozorenje o krađi na pametni mobitel pomoću WhatsApp poruke. Mikroupravljač spojen je na bežičnu mrežu koju korisnik upiše u programu za programiranje (Arduino program) uz upisanu i lozinku. Projektom je predviđeno postavljanje osam trofeja (četiri po redu police), a sada se koristi samo četiri. Čuvar trofeja ima kreiranu vlastitu lozinku koja služi za isključivanje mogućnosti uključenja alarma pri uzimanju trofeja s police. Ako korisnik unese točnu lozinku može uzeti trofej i pogledati ga, u suprotnom se oglašava alarm i istovremeno dolazi pisano upozorenje autoru projekta na mobitel putem WhatsApp aplikacije.

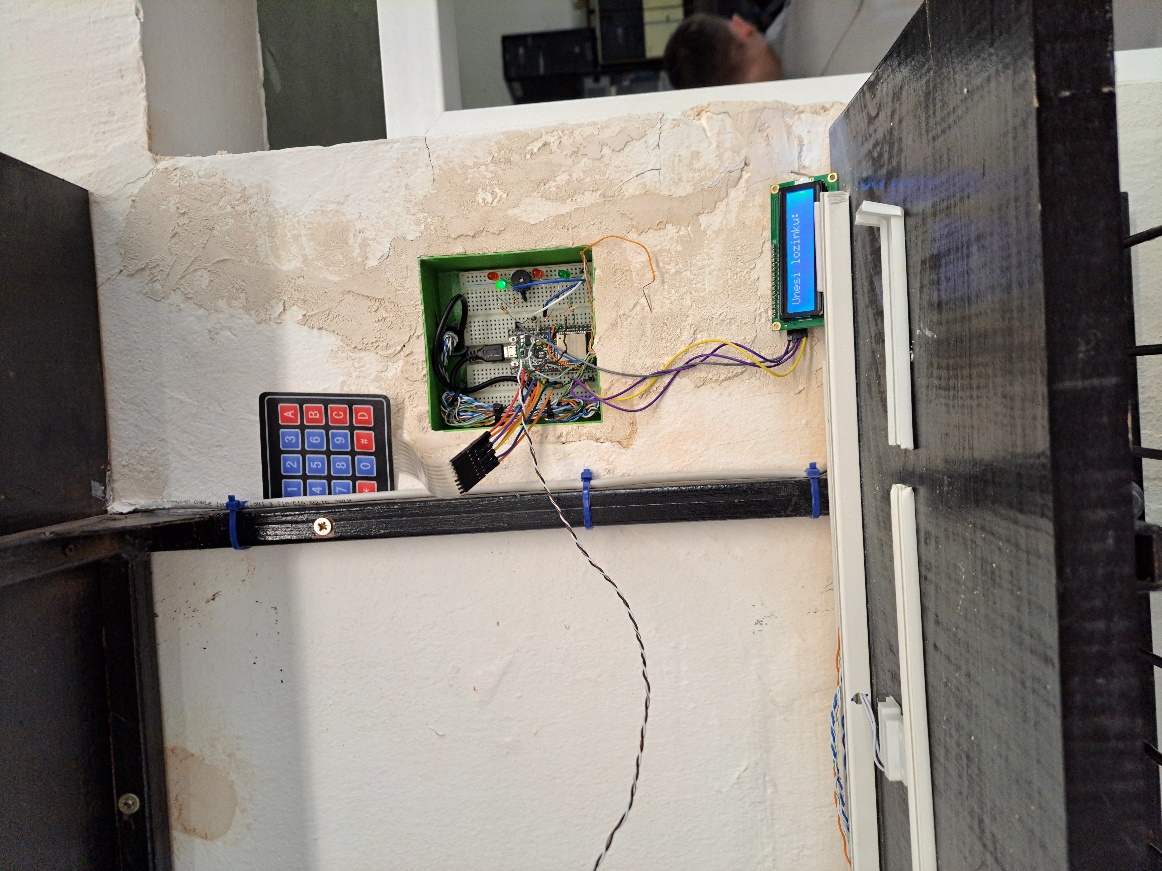
**Tijek izrade:**

****



Postavljanje automatskih osigurača (lijevo)

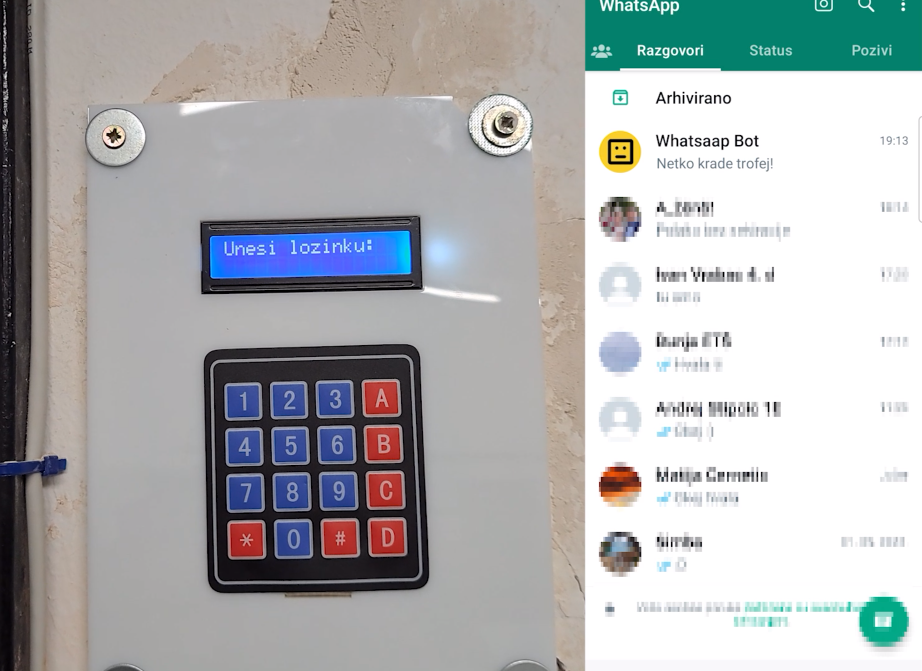
Postavljanje mikroupravljača ESP32 u 3D printanu kutijicu koja je ugrađena u zid (desno)

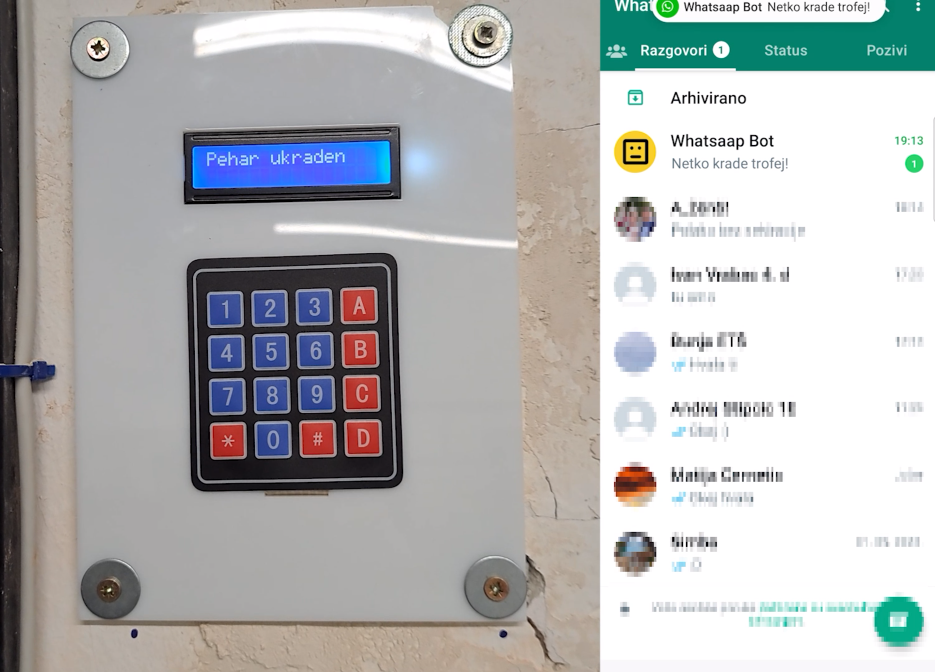
 





**Završni izgled projekta s prikazom aplikacije WhatsApp:**





**Video prezentacija funkcionalnosti projekta:**

<https://youtu.be/l5GDuI3l9Bs>

ili putem QR koda...

Slika na kojoj se prikazuje uzorak, kvadrat, simetrija, dizajn

Opis je automatski generiran

**Programski kod:**

//Sigurnosni sustav za trofeje pomoću WhatsApp-a

#include <WiFi.h> //učitavanje knjižnice za bežičnu WiFi komunikaciju

#include <HTTPClient.h> //učitavanje knjižnice za HTTP protokol

#include <Wire.h> //učitavanje knjižice za I2C komunikaciju

#include <LiquidCrystal\_I2C.h> //pozivanje na knjižicu za LCD s I2C modulom

#include <Keypad.h> //pozivanje na knjižicu za membransku tipkovnicu 4 x 4 (vanjski library)

#define Password\_Length 4 //definiranje duljine lozinke

char Data [Password\_Length]; //definiranje data kao lozinka koja se unosi

char Master [Password\_Length] = "123"; //kreiranje lozinke "123" //PROMIJENITI PO ŽELJI

byte data\_count = 0; //stanje prije nego li se upise lozinka

char customKey; //varijable lozinke koja ce biti upisana

const byte ROWS = 4; //konstanta za redak

const byte COLS = 4; //konstanta za stupac

char hexaKeys [ROWS][COLS] = { //definiranje rasporeda tipki membranske tipkovnice

{'1', '2', '3', 'A'},

{'4', '5', '6', 'B'},

{'7', '8', '9', 'C'},

{'\*', '0', '#', 'D'}

};

byte rowPins[ROWS] = {18, 5, 17, 16}; //pinovi tipkovnice i njenih redova

byte colPins[COLS] = {4, 0, 2, 15}; //pinovi tipkovnice i njenih stupaca

Keypad customKeypad = Keypad (makeKeymap (hexaKeys), rowPins, colPins, ROWS, COLS); //deklariranje tipkovnice

LiquidCrystal\_I2C lcd (0x27, lcdColumns, lcdRows); //deklariranje 16 x 2 zaslona s I2C modulom

int lcdColumns = 16; //broja stupca lcd ekrana

int lcdRows = 2; //broja redova lcd ekrana

const char\* ssid = "upisi WIFI mrezu"; //(!)UPIŠITE IME WIFI MREŽE

const char\* password = "lozinka WIFI mreze"; //(!)UPIŠITE LOZINKU WIFI MREŽE

String apiKey = "upisi token od WhatsApp bota"; //(!)UPIŠITE BROJ TOKENA I BROJ OD WHATSAPP BOTA. KAKO INSTALIRATI WHATSAAP BOT I DOBITI WHATSAPP TOKEN I BROJ:

String phone\_number = "+upisi broj mobitela"; //https://www.techtonions.com/esp32-sending-alert-message-to-whatsapp/#Setting-Up-A-WhatsApp-bot

String url;

int sifrazelena = 26; //zelena ledica koja pokazuje stanje tipkovnice ako je lozinka ispravno upisana

int sifracrvena = 34; //crvena ledica koja pokazuje stanje tipkovnice ako je lozinka neispravno upisana

int zelena = 12; //zelena ledica koja pokazuje stanje trofeja ako ga nitko ne krade

int crvena = 13; //crvena ledica koja pokazuje stanje trofeja ako ga netko krade

int buzzer = 14; //zvučnik koji se aktivira ako netko krade trofej

int sensor1 = 27; //magnetski senzor 1

int sensor2 = 25; //magnetski senzor 2

int sensor3 = 33; //magnetski senzor 3

int sensor4 = 32; //magnetski senzor 4

int state1; //stanje magnetskog senzora 1

int state2; //stanje magnetskog senzora 2

int state3; //stanje magnetskog senzora 3

int state4; //stanje magnetskog senzora 4

void setup() //pokreni kod jednom

{

Serial.begin(115200); //brzina prijenosa

lcd.init(); //inicijalizacija lcd ekrana

lcd.backlight (); //uključenje pozadinskog osvjetljenja LCD-a

pinMode (sifrazelena, OUTPUT); //ledica sifrazelena postavljena kao output

pinMode (sifracrvena, OUTPUT); //ledica sifracrvena postavljena kao output

pinMode(buzzer, OUTPUT); //zvučnik postavljen kao output

pinMode(zelena, OUTPUT); //ledica zelena postavljena kao output

pinMode(crvena, OUTPUT); //ledica crvena postavljena kao output

pinMode(sensor1, INPUT\_PULLUP); //senzor1 postavljen kao input\_pullup te se aktivira kada se odvoji metalni senzor od postolja

pinMode(sensor2, INPUT\_PULLUP); //senzor2 postavljen kao input\_pullup te se aktivira kada se odvoji metalni senzor od postolja

pinMode(sensor3, INPUT\_PULLUP); //senzor3 postavljen kao input\_pullup te se aktivira kada se odvoji metalni senzor od postolja

pinMode(sensor4, INPUT\_PULLUP); //senzor4 postavljen kao input\_pullup te se aktivira kada se odvoji metalni senzor od postolja

WiFi.begin(ssid, password); //Spajanje na WiFi pomoću SSID i PSA

Serial.println("Connecting to WiFi");

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

delay(500);

Serial.print(".");

}

Serial.println();

}

void loop(){ //pokreći kod stalno

state1 = digitalRead(sensor1); //očitavanje stanja magnetski senzora

state2 = digitalRead(sensor2);

state3 = digitalRead(sensor3);

state4 = digitalRead(sensor4);

//ukoliko je state==HIGH znači da je magnetski senzor odvojen od postolja te da netko krade trofej

if (state1 == HIGH || state2 == HIGH || state3 == HIGH || state4 == HIGH) { //ukoliko je magnetski senzor odvojen postolja

message\_to\_whatsapp("Netko krade trofej!"); //pošalji poruku na WhatsApp

digitalWrite(buzzer, HIGH); //uključi zvučnik

digitalWrite(zelena, LOW); //isključi zelenu ledicu

digitalWrite(crvena, HIGH); //uključi crvenu ledicu

lcd.clear (); //resetiran lcd zaslon

lcd.setCursor (0, 0); //postavljanje pokazivača LCD-a na 1. red i 1. mjesto

lcd.print ("Pehar ukraden"); //ispis teksta na zaslonu

lcd.setCursor (0, 1); //postavljanje pokazivača LCD-a na 2. red i 1. mjesto

lcd.print ("Unos nedopustiv"); //ispis teksta na zaslonu

digitalWrite(zelena, LOW); //isključi zelenu ledicu

digitalWrite(crvena, HIGH); //isključi crvenu ledicu

Serial.print("Pehar ukraden"); //ispis teksta na monitoru

for (int i = 0; i < 35; i++) { //35 puta pali i gasi zvučnik da imitira alarm kada netko krade trofej i onda ga nakon 35. puta ugasi

digitalWrite(buzzer, HIGH); //uključi zvučnik

delay(500); //pričekaj 0.5 sekundi

digitalWrite(buzzer, LOW); //isključi zvučnik

delay(500); //pričekaj 0.5 sekundi

if (!(digitalRead(sensor1) == HIGH || digitalRead(sensor2) == HIGH || digitalRead(sensor3) == HIGH || digitalRead(sensor4) == HIGH)){ //kada se pehar vrati vrati na postolje prekini kod i vrati na početno stanje

break;

}

}

}

digitalWrite(buzzer, LOW); //isključi zvučnik

digitalWrite(zelena, HIGH); //isključi zelenu ledicu

digitalWrite(crvena, LOW); //isključi crvenu ledicu

lcd.setCursor (0, 0); //postavljanje pokazivača LCD-a na 1. red i 1. mjesto

lcd.print ("Unesi lozinku:"); //ispis teksta na zaslonu

customKey = customKeypad.getKey (); //stavlja varijablu lozinke kao lozinku koja je upisana na tipkovnici

if (customKey) { //provjera ako je lozinka upisana

Data [data\_count] = customKey;

lcd.setCursor (data\_count, 1);

lcd.print (Data [data\_count]); //ispiši na zaslonu lozinku koja se upisuje na tipkovnici

data\_count++; //stanje nakon što se upiše lozinka

}

if (data\_count == Password\_Length - 1){ //provjera ako je lozinka točna

lcd.clear (); //brisanje teksta sa zaslona

if (!strcmp (Data, Master)){ //provjera ako je trenutačno upisana lozinka točna

lcd.print ("Lozinka tocna"); //ispis teksta na zaslonu

digitalWrite (sifrazelena, HIGH); //uključi zelenu ledicu za tipkovnicu

digitalWrite (sifracrvena, LOW); //isključi crvenu ledicu za tipkovnicu

digitalWrite(buzzer, LOW); //isključi zvučnik

delay (1000); //pričekaj 1 sekundu

digitalWrite (sifrazelena, LOW); //isključi zelenu ledicu

digitalWrite (sifracrvena, HIGH); //uključi crvenu ledicu natrag

}

else{ //ukoliko lozinka nije točna

lcd.print ("Lozinka netocna"); //ispis teksta na zaslonu

digitalWrite(buzzer, HIGH); //uključi zvučnik

delay (1000); //pričekaj 1 sekundu

digitalWrite(buzzer, LOW); //isključi zvučnik

}

lcd.clear (); //brisanje teksta sa zaslonu

clearData (); //izbriši podatke o tipkovnici i omogući ponovno upisivanje šifre

}

}

void clearData(){ //funkcija za brisanje podataka o upisu na tipkovnicu

while (data\_count != 0) {

Data [data\_count--] = 0;

}

return;

} //povratak

void message\_to\_whatsapp(String message) //funkcija za slanje poruka na WhatsApp

{

//dodavanje svih brojeva, vašeg API ključa, vaše poruke u jedan potpuni url

url = "https://api.callmebot.com/whatsapp.php?phone=" + phone\_number + "&apikey=" + apiKey + "&text=" + urlencode(message);

postData(); //pozivanje postData za pokretanje iznad generiranog url-a jednom tako da ćete primiti poruku.

}

void postData() //korisnički definirana funkcija koja se koristi za pozivanje api (POST podaci)

{

int httpCode; //varijabla koja se koristi za dobivanje http koda odgovora nakon api poziva

HTTPClient http; //deklariranje objekta klase HTTPClient

http.begin(url); //pokreni HTTPClient objekt s generiranim url-om

httpCode = http.POST(url); //postavi URL s ovom funkcijom koja će pohraniti http kod

if (httpCode == 200) //pogledaj ako je response kod 200

{

Serial.println("Sent ok."); //ispis na monitoru

}

else //ako response nije 200 postavi stanje kao pogrešno

{

Serial.println("Error."); //ispis na zaslonu

}

http.end(); //završi HTTP

}

String urlencode(String str) //funkcija za enkodiranje URL-a

{

String encodedString = "";

char c;

char code0;

char code1;

char code2;

for (int i = 0; i < str.length(); i++) {

c = str.charAt(i);

if (c == ' ') {

encodedString += '+';

} else if (isalnum(c)) {

encodedString += c;

} else {

code1 = (c & 0xf) + '0';

if ((c & 0xf) > 9) {

code1 = (c & 0xf) - 10 + 'A';

}

c = (c >> 4) & 0xf;

code0 = c + '0';

if (c > 9) {

code0 = c - 10 + 'A';

}

code2 = '\0';

encodedString += '%';

encodedString += code0;

encodedString += code1;

//encodedString+=code2;

}

yield();

}

return encodedString;

}