

Proiect Informatica Aplicata

1.Introducere

Proiectul se bazeaza pe implementarea unui program care sa ofere printr-un API interogat prin cereri http datele unei liste. Prin folosirea metodei de comunicatie Bluetooth Classic si a unui microcontroller ESP32 putem afisa datele listei pe aplicatia support de pe telefon (ProiectIA). API este acronimul pentru Application Programming Interface, care este un intermediar software care permite a două aplicații să comunice între ele.

Tematica proiectului este jocul digital de carti de colectie online, HearthStone. Trebuie sa afisam pe aplicatia support caracteristicile minionilor, care sunt: ID-ul , numele, setul din care face parte, tipul cartii, cat costa cartea pentru a fi convocata, atacul, viata, si o imagine cu caracterul de pe carte.

2.Considerente teoretice

Modulul ESP32 este un modul SoC (System on Chip) fabricat de compania Espressif Systems, bazat pe microprocesorul Tensilica Xtensa LX6 cu unul sau două nuclee și o frecvență de lucru de între 160 și 240MHz precum și un coprocesor ULP (Ultra Low Power). Suplimentar, acesta dispune de comunicație WiFi și Bluetooth (clasic și low-energy) integrate, precum și de o gamă largă de interfețe periferice:

- 34 pini programabili GPIO (General Purpose Input/Output)
- 18 canale de conversie analog-digitală (ADC) cu rezoluție de 12 biți
- 2 canale de conversie digital-analogică (DAC) cu rezoluție de 8 biți
- 16 canale de ieșire PWM (Pulse Width Modulation)
- 10 senzori interni capacitivi
- 3 interfețe SPI (Serial Peripheral Interface)
- 3 interfețe UART (Universal Asynchronous Receiver-Transmitter)
- 2 interfețe I2C (Inter-Integrated Circuit)
- 2 interfețe I2 S (Inter-IC Sound)
- 1 interfață CAN 2.0 (Controller Area Network)
- controler pentru conectarea dispozitivelor de stocare (carduri de memorie)

Modulul ESP32 poate fi integrat în plăci de dezvoltare ce pot expune toți pinii/interfețele modulului sau doar o parte din ele. Cele mai des întâlnite tipuri de plăci de dezvoltare bazate pe modulul ESP32 sunt cele cu 30 sau 38 de pini. Programarea modulelor ESP32 se poate realiza folosind diverse medii de dezvoltare, cele mai populare fiind Arduino IDE si Visual Studio Code împreună cu extensia PlatformIO.

Un protocol de comunicatie folosit este Bluetooth Classic. Aceasta este o tehnologie wireless care permite schimbul de date între diferite dispozitive. Diferenta între Bluetooth Classic si Bluetooth Low Energy este ca cel clasic poate gestiona o mulțime de date, dar consumă rapid durata de viață a bateriei și costă mult mai mult, pe cand cel low energy este folosit pentru aplicații care nu au nevoie să facă schimb de cantități mari de date și, prin urmare, pot funcționa pe baterie ani de zile la un cost mai ieftin.

Wi-Fi este tehnologia wireless folosită pentru a conecta computere, tablete, smartphone-uri și alte dispozitive la internet. De asemenea, este semnalul radio trimis de la un router wireless către un dispozitiv din apropiere, care traduce semnalul în date pe care le puteți vedea și utiliza. Dispozitivul transmite un semnal radio înapoi către router, care se conectează la internet prin fir sau cablu.

Hypertext Transfer Protocol (HTTP) este fundamentul World Wide Web și este folosit pentru a încărca pagini web folosind link-uri hipertext. HTTP este un protocol de nivel de aplicație conceput pentru a transfera informații între dispozitivele din rețea și rulează peste alte straturi ale stivei de protocoale de rețea. Un flux tipic prin HTTP implică o mașină client care face o solicitare către un server, care trimite apoi un mesaj de răspuns.

JSON este un acronim în limba engleză pentru *JavaScript Object Notation*, și este un format de reprezentare și interschimb de date între aplicații informatice. Este un format text, inteligibil pentru oameni, utilizat pentru reprezentarea obiectelor și a altor structuri de date și este folosit în special pentru a transmite date structurate prin rețea, procesul purtând numele de serializare. JSON este alternativa mai simplă, mai facilă decât limbajul XML. Eleganța formatului JSON provine din faptul că este un subset al limbajului JavaScript, fiind utilizat alături de acest limbaj.

3. Implementare

Am folosit următoarele biblioteci:

```
#include <Arduino.h>

#include <BluetoothSerial.h>

#include <ArduinoJson.h>

#include <WiFi.h>

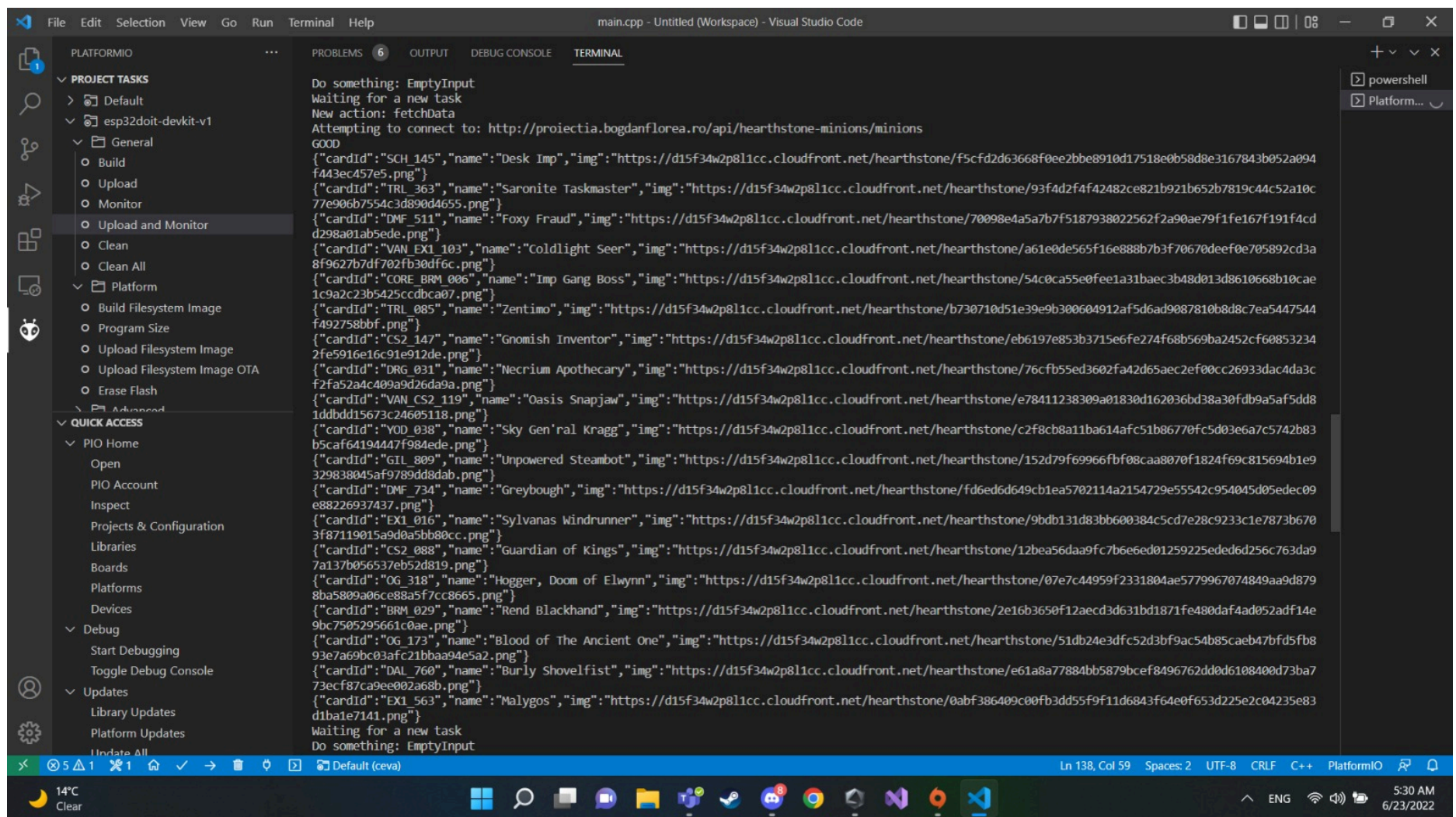
#include <HTTPClient.h>
```

Am început prin a conecta microcontroller-ul la wifi, apoi folosind tehnologia Bluetooth clasic am conectat smartphone-ul.

Am creat funcția `receivedData` care ne returnează acțiunea pe care o îndeplinește microcontroller-ul în momentul respectiv, spre exemplu după apăsarea butonului „Get data” funcția va returna acțiunea „fetchData”, pe aplicație va apărea lista cu id, nume și imagine pentru fiecare minion, iar în consola serială vor apărea aceste detalii ale fiecărui minion; ulterior funcția va returna „empty” și va aștepta următoarea comandă.

Am inițializat string-urile corespunzătoare caracteristicilor cartilor cu datele din fișierul JSON.

Am creat funcția `sendDataToApp` pentru a lua toate detaliile cartilor(id, name, img) ca într-o listă, incrementând un index pe care îl atribuim fiecărui obiect și a le trimite către telefon pentru a fi afișate în aplicație.



4. Concluzii

Gama de aplicații a modulelor ESP32 este foarte largă. Acestea sunt deosebit de bine adaptate aparatelor pentru automatizare a casei, electronicelor uzate și Internetului Lucrurilor (IoT). Un procesor eficient și numeroase periferice vă permit să implementați proiecte foarte complexe și avansate. Utilizarea ESP32 se dovedește a fi surprinzător de simplă grație numeroaselor soluții la gata. Toate software-urile necesare sunt disponibile gratuit și, în special, pe Internet, veți găsi numeroase biblioteci și numeroase proiecte finalizate și gata de folosit, care pot servi drept sursă de inspirație sau drept bază pentru propriile dumneavoastră studii.

5. Bibliografie

- <https://rapidapi.com/omgvamp/api/hearthstone/>
- <https://medium.com/@akash.kandhare/bluetooth-vs-bluetooth-low-energy-whats-the-difference-74687afcedb1#:~:text=Bluetooth%20can%20handle%20a%20lot,you're%20trying%20to%20accomplish.>
- <https://www.json.org/json-en.html>
- https://www.w3schools.com/whatis/whatis_json.asp
- <https://www.cloudflare.com/learning/ddos/glossary/hypertext-transfer-protocol-http/>
- <https://www.youtube.com/watch?v=RStncO3zb8g>
- <https://www.youtube.com/watch?v=xRrTneY-BdE>
- <https://www.verizon.com/info/definitions/wifi/>
- [https://catalog.us-east-1.prod.workshops.aws/workshops/5b127b2f-f879-48b9-9dd0-35aff98c7bbc/en-US/module1/esp32 modulul esp32](https://catalog.us-east-1.prod.workshops.aws/workshops/5b127b2f-f879-48b9-9dd0-35aff98c7bbc/en-US/module1/esp32%20modulul%20esp32)
- [ESP32 Bluetooth Low Energy \(BLE\) on Arduino IDE | Random Nerd Tutorials \(randomnerdtutorials-com.translate.goog\)](https://ESP32 Bluetooth Low Energy (BLE) on Arduino IDE | Random Nerd Tutorials (randomnerdtutorials-com.translate.goog))
- <https://en.wikipedia.org/wiki/ESP32>
- <https://arduinojson.org/v6/example/parser/>
- <https://arduinogetstarted.com/tutorials/arduino-http-request>
- <https://arduinojson.org/v6/assistant>

FileEditSelectionViewGoRunTerminalHelpmain.cpp - Untitled (Workspace) - Visual Studio Code

PLATFORMIO

PROJECT TASKS

Default

esp32doit-devkit-v1

General

Build

Upload

Monitor

Upload and Monitor

Clean

Clean All

Platform

Build Filesystem Image

Program Size

Upload Filesystem Image

Upload Filesystem Image OTA

Erase Flash

Advanced

QUICK ACCESS

PIO Home

Open

PIO Account

Inspect

Projects & Configuration

Libraries

Boards

Platforms

Devices

Debug

Start Debugging

Toggle Debug Console

Updates

Library Updates

Platform Updates

Update All

PROBLEMS6

OUTPUT

DEBUG CONSOLE

TERMINAL

Do something: EmptyInput

Waiting for a new task

New action: fetchData

Attempting to connect to: http://proiectia.bogdanflorea.ro/api/hearthstone-minions/minions

GOOD

{ "cardId": "SCH_145", "name": "Desk Imp", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/f5cfd2d63668f0ee2bbe8910d17518e0b58d8e3167843b052a094f443ec457e5.png" }

{ "cardId": "TRL_363", "name": "Saronite Taskmaster", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/93f4d2f4f42482ce821b921b652b7819c44c52a10c77e906b7554c3d890d4655.png" }

{ "cardId": "DMF_511", "name": "Foxy Fraud", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/70098e4a5a7b7f5187938022562f2a90ae79f1fe167f191f4cd298a01ab5ede.png" }

{ "cardId": "VAN_EX1_103", "name": "Coldlight Seer", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/a61e0de565f16e888b7b3f70670deef0e705892cd3a8f9627b7df702fb30dfcc.png" }

{ "cardId": "CORE_BRM_006", "name": "Imp Gang Boss", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/54c0ca55e0fee1a31baec3b48d013d8610668b10cae1c9a2c23b5425ccdbca07.png" }

{ "cardId": "TRL_085", "name": "Zentimo", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/b730710d51e39e9b300604912af5d6ad9087810b8d8c7ea5447544f492758bbf.png" }

{ "cardId": "CS2_147", "name": "Gnomish Inventor", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/eb6197e853b3715e6fe274f68b569ba2452cf608532342fe5916e16c91e912de.png" }

{ "cardId": "DRG_031", "name": "Necrium Apothecary", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/76cfb55ed3602fa42d65aec2ef00cc26933dac4da3cf2fa52a4c409a9d26da9a.png" }

{ "cardId": "VAN_CS2_119", "name": "Oasis Snapjaw", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/e78411238309a01830d162036bd38a30fdb9a5af5dd81ddbdd15673c24605118.png" }

{ "cardId": "YOD_038", "name": "Sky Gen'ral Kragg", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/c2f8cb8a11ba614afc51b86770fc5d03e6a7c5742b83b5caf64194447f984ede.png" }

{ "cardId": "GIL_809", "name": "Unpowered Steambot", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/152d79f69966fbf08caa8070f1824f69c815694b1e9329838045af9789dd8dab.png" }

{ "cardId": "DMF_734", "name": "Greybough", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/fd6ed6d649cb1ea5702114a2154729e55542c954045d05edec09e88226937437.png" }

{ "cardId": "EX1_016", "name": "Sylvanas Windrunner", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/9bdb131d83bb600384c5cd7e28c9233c1e7873b6703f87119015a9d0a5bb80cc.png" }

{ "cardId": "CS2_088", "name": "Guardian of Kings", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/12bea56daa9fc7b6e6ed01259225eded6d256c763da97a137b056537eb52d819.png" }

{ "cardId": "OG_318", "name": "Hogger, Doom of Elwynn", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/07e7c44959f2331804ae5779967074849aa9d8798ba5809a06ce88a5f7cc8665.png" }

{ "cardId": "BRM_029", "name": "Rend Blackhand", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/2e16b3650f12aecdd3d631bd1871fe480daf4ad052adf14e9bc7505295661c0ae.png" }

{ "cardId": "OG_173", "name": "Blood of The Ancient One", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/51db24e3dfc52d3bf9ac54b85caeab47bfd5fb893e7a69bc03afc21bbaa94e5a2.png" }

{ "cardId": "DAL_760", "name": "Burly Shovelfist", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/e61a8a77884bb5879bcef8496762dd0d6108400d73ba773ecf87ca9ee002a68b.png" }

{ "cardId": "EX1_563", "name": "Malygos", "img": "https://d15f34w2p8l1cc.cloudfront.net/hearthstone/0abf386409c00fb3dd55f9f11d6843f64e0f653d225e2c04235e83d1ba1e7141.png" }

Waiting for a new task

Do something: EmptyInput

Ln 138, Col 59

Spaces: 2

UTF-8

CRLF

C++

PlatformIO

14°C

Clear

5:30 AM

6/23/2022

05:31



BTC Device

Bonbardieriii

58:BF:25:17:45:0A

**-58
dBm**

Get data

58:BF:25:17:45:0A





List

Desk Imp
ID:

Details

Saronite Taskmaster
ID:

Details

Foxy Fraud
ID:

Details

Coldlight Seer
ID:

Details

Imp Gang Boss
ID:

Details

Zentimo
ID:

Details

Gnomish Inventor
ID:

Details

Necrium Apothecary
ID:

Details

Oasis Snapjaw
ID:

Details

Sky Gen'ral Kragg
ID:

Details

Unpowered Steambot
ID:

Details

Greybough
ID:

Details

Sylvanas Windrunner
ID:

Details

Guardian of Kings
ID:

Details

Hogger, Doom of Elwynn
ID:

Details

Rend Blackhand
ID:

Details

Blood of The Ancient One
ID:

Details

Burly Shovelfist
ID:

Details

Malygos
ID:

Details

Colossus of the Moon
ID:

Details