

**PROIECT ISW**

**Conducător științific:**

**Ciobanu Cătălin**

**Studenți:**

**Enache Mihai Sebastian**

**Ciocoiu Radu-Mihail**

**Grupa: 4LF691**

**BRAȘOV, 2023**

**Departamentul Electronică și Calculatoare**

**Programul de studii: Tehnologii și Sisteme de Telecomunicații**

***Manager,Documetație-ENACHE Mihai Sebastian***

***Developer,Tester-CIOCOIU Radu-Mihail***

OCR Scanner

**Conducător științific:**

Ciobanu Cătălin

Brașov, 2023

# Sumar

**Obiectivul proiectului**

Obiectivul proiectului nostru este să dezvoltăm un scanner compact și cost-efectiv, capabil să recunoască textul din imagini și să îl convertească în format digital. Acest proiect își propune să ofere o soluție de scanare ușor de utilizat și accesibilă pentru persoanele care doresc să digitalizeze documente și să le stocheze sau să le distribuie prin intermediul tehnologiei digitale.

**Funcționalități cheie**

* Scanarea documentelor
* Recunoașterea caracterelor
* Pre-procesare a imaginilor
* Exportarea datelor
* Rezultate precise
* Integrare cu alte aplicații

**Beneficii**

* Accesibilitate
* Economie de timp și efort
* Precizie îmbunătățită
* Spațiul de stocare mai mic
* Flexibilitate

# Cuprins

[Sumar 3](#_Toc133437748)

[Cuprins 4](#_Toc133437749)

[Listă de acronime 5](#_Toc133437750)

[1 Introducere 6](#_Toc133437751)

[2 Descriere Aplicație 7](#_Toc133437752)

[3 Tutorial de folosire a aplicației 8](#_Toc133437753)

[4 Tehnologii folosite 9](#_Toc133437754)

[5 GitHub Repository 17](#_Toc133437755)

[6 Testare 19](#_Toc133437756)

[7 Management 22](#_Toc133437757)

[8 Lecții Învățate 25](#_Toc133437758)

[9 Concluzii 27](#_Toc133437759)

[Bibliografie 28](#_Toc133437760)

[Appendix 29](#_Toc133437761)

[Codul Sursă 30](#_Toc133437762)

# Listă de acronime

IoT – Internet of things

WiFi- Wireless fidelity

DIY – Do it yourself

USB – Universal serial bus

IC- integrated circuit

OCR – Optical character recognition

# Introducere

Recunoașterea optică a caracterelor (OCR) este o tehnologie care permite procesarea automată a textului din imagini și documente scanate. Această tehnologie utilizează algoritmi specializați pentru a identifica caracterele din imagini și pentru a le converti în text, permițând astfel utilizatorilor să își digitalizeze documentele și să le facă căutabile și editabile. În ultimii ani, OCR a devenit din ce în ce mai important într-o gamă largă de aplicații, de la digitalizarea arhivelor istorice la procesarea facturilor și a chitanțelor. Tehnologia este utilizată în domenii precum afaceri, cercetare, biblioteci, arhive și multe altele. Într-un proiect de OCR, sunt implicați algoritmi de procesare a imaginilor, rețele neuronale, biblioteci software și hardware specializate. Proiectul poate fi implementat pe o gamă largă de platforme, de la calculatoare și telefoane mobile la microcontrolere și alte dispozitive integrate. În general, un proiect de OCR presupune dezvoltarea și testarea algoritmilor de procesare a imaginilor, integrarea cu biblioteci software și hardware specializate, și optimizarea performanței pentru a obține o soluție robustă și eficientă.

Pentru a aborda această provocare, am ales să dezvoltăm acest proiect de implementare a tehnologiei OCR pe un dispozitiv ESP32. Motivația noastră pentru acest proiect a fost de a oferi o soluție accesibilă și ușor de utilizat pentru integrarea tehnologiei OCR pe dispozitive IoT și microcontrolere, deschizând astfel calea pentru o varietate de aplicații noi și interesante.

# 2 Descriere Aplicație

Aplicația este un sistem de recunoaștere optică a caracterelor (OCR) pentru ESP32. Acesta utilizează o cameră ESP32 pentru a captura imagini, care sunt apoi procesate prin intermediul bibliotecilor OCR pentru a detecta și recunoaște caracterele. Aplicația este capabilă să detecteze caracterele în diferite limbi și să furnizeze textul recunoscut într-un format ușor de citit. Cu această aplicație, utilizatorii pot scana și converti documente fizice în format digital, ceea ce face posibilă căutarea și editarea conținutului. De asemenea, poate fi folosită pentru a recunoaște caracterele în imagini, cum ar fi numere de înmatriculare sau numere de telefon, făcând astfel mai ușoară și mai rapidă introducerea acestora într-un format digital. În general, aplicația este utilă pentru oricine are nevoie să convertească documente fizice în format digital sau să recunoască caractere în imagini.

Despre arhitectura aplicației , putem spune că aceasta este construită pe baza platformei Arduino și am utilizat biblioteci specifice pentru a controla placa ESP32 și camera atașată

Procesarea imaginilor se face prin intermediul bibliotecilor OCR , care sunt integrate în aplicație și sunt responsabile de detectarea și recunoașterea caracterelor .

De asemenea pentru partea de procesare am folosit limbajul de programare Python care permite o eficiență ridicată în procesarea datelor și în interacțiunea cu placa ESP32.

În plus , proiectul nostru poate fi folosit pe mai multe platforme , inclusiv Windows , Linux , MacOs. De asemenea , pentru a face mai ușoară procesul de dezvoltare și de mentenanță am folosit GitHub și poate fi descărcat și modificat de dezvoltatorii interesați.

În ceea ce privește costul proiectului OCR Scanner , acesta a fost dezvoltat ca parte a unui proiect de cercetare și dezvoltare și nu a fost destinată să fie un produs comercial. Din acest motiv , costul total al proiectului a fost relativ mic , estimat la 60 de lei , care inclus costurile pentru achiziționarea plăcii ESP32-CAM și alte componente necesare pentru a construi prototipul , precum și costurile pentru resursele de dezvoltare software utilizate în proiect.

Este important de menționat că acest cost este aproximativ și variază în funcție de resurse și materialele folosite în proiect.

# 3 Tutorial de folosire a aplicației

1. Conectarea plăcii ESP32-CAM: Pentru a utiliza OCR Scanner, trebuie să conectați placa ESP32-CAM la un port USB de pe calculatorul dumneavoastră. Asigurați-vă că ați instalat driverele necesare pentru placa ESP32-CAM înainte de a începe.
2. Pornirea aplicației OCR Scanner: După ce ați conectat placa ESP32-CAM la calculator, deschideți OCR Scanner de pe desktop-ul sau din folderul în care ați salvat aplicația. În momentul în care aplicația este pornită, veți fi întâmpinat de interfața grafică a acesteia.
3. Configurarea setărilor camerei: Pentru a începe scanarea, trebuie să configurați setările camerei prin selectarea opțiunilor potrivite. Alegeți tipul de imagine (color sau alb-negru), dimensiunea imaginii și setați opțiunea de auto-focus în funcție de preferințele dumneavoastră.
4. Face poză: După ce ați setat setările camerei, plasați documentul sub camera plăcii ESP32-CAM. Apăsați butonul "Scan" din interfața aplicației pentru a scana imaginea.
5. Salvează: După ce imaginea a fost scanată, aplicația va începe să proceseze textul din imaginea respectivă. Așteptați câteva momente până când procesul este finalizat.
6. Procesează: Aplicația va începe procesarea textului din imaginea respectivă. În această etapă , OCR Scanner utilizează algoritmi de recunoaștere optică a caracterelor pentru a detecta și extrage textul din imagine
7. Salvare în urma procesării: După ce textul a fost procesat , aplicația OCR Scanner vă permite să salvați rezultatele scanării sub forma unui fișier text. Puteți alege să salvați fișierul în orice locație de pe computer.
8. Afișează textul: După ce rezultate scanării s-au salvat , puteți deschide fișierul text pentru a vizualiza textul extras din imagine . În acest fel , puteți utiliza OCR Scanner pentru a prelucra conținutul acestora într-un mod eficient.

# 4 Tehnologii folosite

Proiectul de scanner OCR folosind modulul ESP32-CAM și Python implică utilizarea mai multor tehnologii:

ESP32 este un microcontroller dual-core de la Esressif Systems, ce include un processor Tensillica Xtensa LX6, memorie flash , memorie SRAM și difertie periferice, cum ar fi interfața WiFi, Bluetooth si Ethernet.

Este capabil să gestioneze o varietate de aplicații , de la Internet of Things (IoT) la dispoziție de control industrial și aplicații multimedia. ESP32 oferă o gamă largă de caracteristici și funcții, precum și o gamă largă de biblioteci și instrumente de dezvoltare disponibile, cee ace face ca dezvoltarea de aplicații cu ESP32 să fie relative ușoară ăi accesibilă.

Niște informații adiționale din datasheet-ul ESP32-CAM

* Procesor dual-core Tensilica LX6 de 32 biți cu frecvență de până la 240 MHz
* Co-procesorul ultra-low-power ULP cu frecvență de până la 200 MHz
* 448 KB de memorie ROM pentru boot, 520 KB de SRAM și 4 MB de memorie flash SPI QIO
* Conectivitate WiFi 802.11b/g/n, Bluetooth 4.2 BR/EDR și BLE (Bluetooth Low Energy)
* Interfață de cameră OV2640 care poate captura imagini de până la 2 megapixeli (UXGA) și suportă ieșire JPEG și RAW
* Suportă procesare de imagine prin intermediul blocurilor hardware dedicate, cum ar fi corectarea culorilor, zoom-ul digital și detecția fețelor
* Suportă criptarea hardware pentru AES, SHA2 și criptarea RSA
* Conține mai multe periferice, cum ar fi SPI, I2S, I2C, UART, ADC și DAC, precum și un interfață SD/MMC
* Funcționează la temperaturi cuprinse între -40 și 85 grade Celsius

De ce ESP32-CAM:

1. Utilizarea modulului ESP32-CAM pentru construirea unui scanner OCR (recunoaștere optică a caracterelor) prezintă următoarele avantaje:
2. Cost-efectiv: Modulul ESP32-CAM este un dispozitiv accesibil și ieftin care poate fi utilizat pentru construirea unui scanner OCR la un preț redus.
3. Dimensiune compactă: Modulul ESP32-CAM are o dimensiune mică și poate fi ușor integrat într-un design compact de scanner OCR.
4. Camera încorporată: Modulul ESP32-CAM vine cu o cameră încorporată, eliminând necesitatea unui modul de cameră extern, ceea ce reduce costul și complexitatea designului.
5. Conectivitate wireless: Modulul ESP32-CAM vine cu conectivitate WiFi încorporată, ceea ce permite transmiterea fără fir a imaginilor și datelor către alte dispozitive pentru procesarea OCR.
6. Flexibilitate: Modulul ESP32-CAM este un dispozitiv flexibil care poate fi programat utilizând diferite limbaje de programare, cum ar fi Arduino și Python, permițând o gamă largă de aplicații pentru scanner OCR.
7. În ansamblu, utilizarea modulului ESP32-CAM pentru construirea unui scanner OCR este o soluție rentabilă și flexibilă care poate fi ușor personalizată pentru a se potrivi diferitelor cerințe ale scanner-ului OCR.
8. Câteva dezavantaje ale utilizării modulului ESP32-CAM pentru construirea unui OCR Scanner :
9. Performanță limitată: Deoarece modului ESP32-CAM este un microcontroller cu o putere de procesare limitată , performanța OCR poate fi limitată în comparație cu soluțille de calcul mai puternice.
10. Memorie limitată: Modulul ESP32-CAM are o memorie limitată , cee ace poate restricționa dimensiunea și comeplexitatea .
11. Viteza de scanare limitată: Din cauza limitărilor hardware ale modulului, viteza de scanare poate fi limitată în comparație cu soluțiile mai puternice de OCR
12. Limitarea numărului de caractere: Numărul de caractere detectabile poate fi limitat în comparație cu alte soluții OCR mai avansate , care poate detecta caractere în diferite limbi sau fonturi.

Python este un limbaj de programare interpretat, interactive și orientat pe obiect. A fost creat în 1989 de Guido van Rossum și este unul dintre cele mai populare limbaje de programare din lume. Python se remarcă prin simplitatea și ușurința de învățare , precum și prin eficiența și putera sa. Este folosit într-o varietate de domenii, de la dezvotarea de aplicații web și desktop la analiză de date , inteligență artificială și machine learning.

Python are o sintaxă simplă și ușor de înțeles , cee ace îl face potrivit pentru programatorii începători. De asemenea, este un limbaj flexibil și extensibil, cu o gamă largă de biblioteci și module disponibile, cee ace îl face ideal pentru proiecte complexe. Python rulează pe mai multe platforme, inclusive Windows, Linux și macOS și este utilizat de companii mari precum Google , Facebook, Dropbox și multe altele.

Unul dintre avantajele majore ale Python este faptul că are o comunitate mare și activă de dezvoltatori și utilizatori care contribuie cu biblioteci și module noi , Acest lucru face ca Python să fie foarte popular pentru analiza de date și machine learning, deoarece există o varietate mare de instrumente disponibile pentru a ajuta la procesarea și analiza datelor

În concluzie , Python este un limbaj de programare puternic, ușor de învățat și flexibil, cu o gamă largă de aplicații și o comunitate active de dezvoltatori și utilizatori.

Arduino C++

Arduino C++ este un limbaj de programa derivate din C++ și este folosit pentru dezvoltarea de aplicații pentru platforma hardware Arduino. Limbajul este ușor de învățat și are o sintaxă simplă , fiind potrivit pentru programatori începători sau experimentați

Printre caractersticile sale se numără gestionarea automata a memoriei, suport pentru programare orientate pe obiecte și abstracții hardware pentru dispozitivele Arduino. Codul scris în Arduino C++ este compilat și transferat către placa Arduimo prin intermediul mediului de dezvoltare Arduino IDE

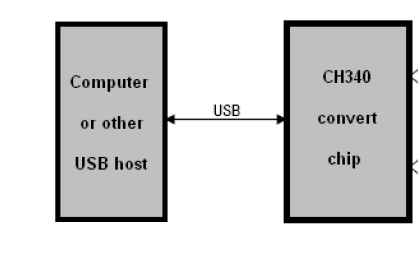
Cu ajutorul acestui limbaj , programatorii pot să crezze o gamă largă de aplicații , de la simplele proiecte cu LED-uri și senzori, până la proiecte mai complexe , cum ar fi drone sau roboți autonomi.

Arduino C++ este un instrument puternic pentru dezvoltarea de proiecte de electronica , avand o comunitate active și numeroase resurse disponibile online.

USB serial chip CH340

CH340 este un circuit integrat (IC) USB-serial care poate fi utilizat pentru a oferi o interfață serial la dispozitivele care nu au o interfață USB sau care necesită conversia semnalului USB la un semnal serial. Este un IC popular și ieftin, folosit frecvent în proiectele de electronica DIY (Do it yourself).

Acesta poate fi integrat în plăci de dezvoltare sau poate fi utilizat pentru a face conversia USB-serial pentru alte dispositive care utilizează semnalul serial. De asemenea, este compatibil cu o varietate de sisteme de operare , inclusive Windows , MacOs și Linux, și este ușor de utilizat și de integrat în proiectele de electronică.



Figură 1 - USB serial CH340

CH340 are următoarele caracteristici:

1. Suportă transfer de date la o viteză mare de până la 2Mbps
2. Suportă funcționalitatea plug-and-play, eliminând necesitatea instalării manual a driver-ului
3. Are o interfață USB 2.0 de mare viteză, ceea ce permite transferul rapid al datelor între dispozitive.
4. Este compatibil cu mai multe sisteme de operare, inclusive Windows , Linux , MacOs.
5. Suportă funcționalitatea de auto-alimentare , eliminând necesitatea unei surse de alimentare externe pentru dispozitivul conectat.
6. Are o construcție compactă și ușoară , fiind ideală pentru aplicații și proiecte.
7. ESP32-CAM: Modulul ESP32-CAM este componenta hardware principală utilizată în acest proiect. Este un dispozitiv versatil, compact și cu cost redus, care vine cu conectivitate WiFi încorporată, cameră și poate fi programat utilizând diferite limbaje de programare.
8. Tesseract OCR: Tesseract OCR este un motor OCR open-source care poate fi utilizat pentru recunoașterea caracterelor în imagini. În acest proiect, Tesseract OCR este utilizat pentru a efectua procesarea OCR pe imaginile capturate.
9. OpenCV: OpenCV este o bibliotecă open-source de computer vision care oferă diferite funcții de procesare și analiză a imaginilor. În acest proiect, OpenCV este utilizat pentru manipularea și preprocesarea imaginilor înainte de a fi trecute la motorul OCR.
10. Numpy: Numpy este o bibliotecă Python specializată în calculul matricelor și algebră liniară. Aceasta poate fi foarte utilă în cazul proiectului nostru , deoarece procesează și manipulează matrici de pixel din imagine.
11. Matplotlib: Matplotlib este tot o librarie Python pentru vizualizarea datelor și crearea de grafice. Am folosit această bibliotecă în proiectul nostru deoarece este o librărie puternică și versatilă capabilă de a crea un graphic care prezintă frecvența caracterelor detectate într-un document scanat.
12. Esp\_camera.h: esp\_camera.h este o bibliotecă inclusă în platforma ESP32 , care oferă funcționalitatea necesară pentru controlul camerei incorporate pe modulul ESP32-CAM. Am folosit această bibliotecă deoarece este capabilă de a face capture imaginilor care sunt apoi procesate prin intermediul bibliotecilor OCR pentru a detecta și recunoaște caracterele
13. WiFi.h: WiFi.h este o bibliotecă standard din ARDUINO IDE care permite comunicarea fără fir prin intermediul rețelelor WiFI.

În ansamblu, acest proiect implică utilizarea unei combinații de tehnologii hardware și software, inclusiv modulul ESP32-CAM, limbajul de programare Python, motorul OCR Tesseract, biblioteca de procesare a imaginilor OpenCV pentru a construi un scanner OCR funcțional.

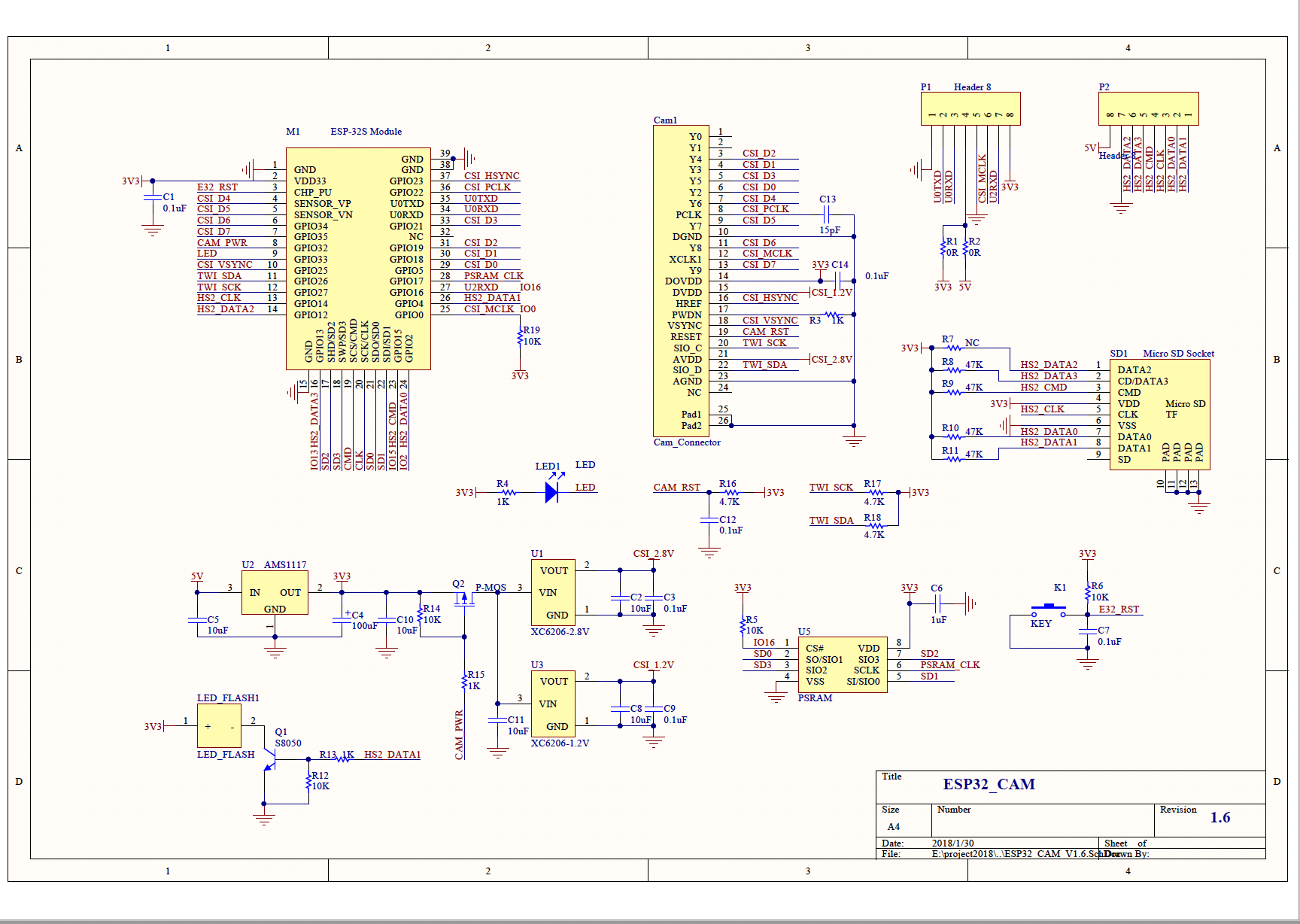


Figura 2 - Schema electrică a modului ESP32-CAM

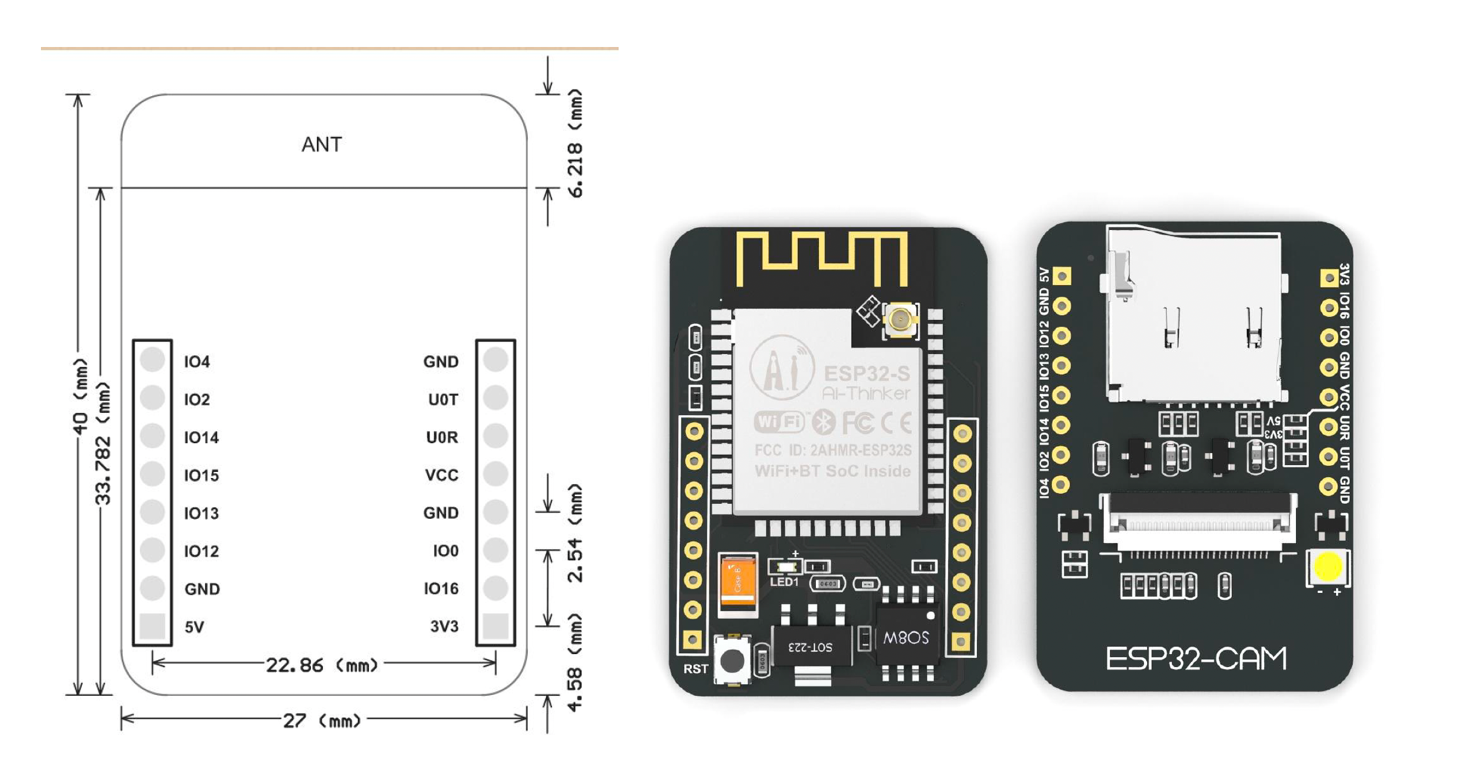
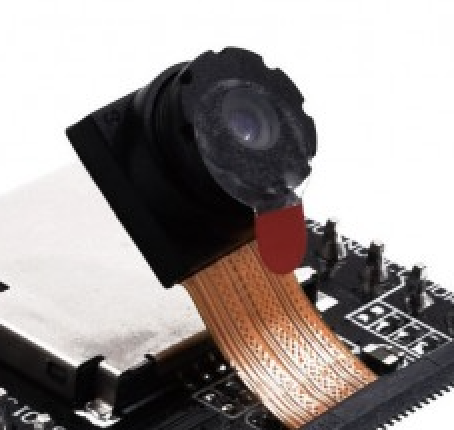


Figura 3 - MODULUL ESP 32-CAM



Figură 4 - Camera

# GitHub Repository

<https://github.com/21salyut12/OCR-Scanner> - Link-ul către github repository .

Acesta este un repostiory GitHub care conține codul sursă pentru aplicația noastră OCR Scanner .

Deschiderea unui repository pe GitHub a fost o parte importantă a dezvoltării proiectului nostru OCR Scanner . Acesta a oferit un mediu centralizat pentru gestionarea codului și colaborarea între membrii echipei.

Pentru a începe , am creat un nou repository , am configurat setările de bază. Apoi am adăugat membrul echipei ca și colaboratori pentru a putea contribui la dezvoltarea proiectului.

În ceea ce privește structura codului , acesta este împărțit în mai multe fișiere, fiecare cu o funcționalitate specifică. Fișierul principal este main.cpp , care conține codul principal al aplicației.

De asemenea , există fișiere separate pentru inițializarea camerei , pentru procesarea imaginilor și pentru integrarea cu Tesseract OCR.

În plus acest repository conține și documentația proiectului. De asemenea, există și un fișier README care conține informații utile .

Acest repository poate fi util pentru dezvoltatorii care doresc să construiască o aplicație OCR folosind modulul ESP32-CAM.

Pe măsură ce proiectul nostru a continuat să se dezvolte , am actualizat și întreținut repository-ul nostru, asigurându-ne că este mereu actualizat cu cele mai recente modificări și că funcționează corect.

# 6 Testare

Testarea a reprezentat o parte importanta a procesului de dezvoltare a aplicației OCR cu utilizarea modulului ESP32-CAM. Scopul testelor a fost de a verifica funcționarea corectă a tuturor componentelor aplicației , de la modulul de captare a imaginilor și până la procesul de recunoaștere a caracterelor.

Testele au fost realizate în mai multe etape . În primul rând s-au realizat teste unitare pentru fiecare componenta a aplicației cu scopul verificării funcționalității corecte a acestora . Au fost utilizate biblioteci și framework-uri de testare , precum unittest .

În al doilea rând , s-au realizat teste de integrare pentru a verifica cum interacționează componentele între ele și pentru a identifica orice problemă de comunicare. Testele de integrare au implicat utilizarea modulului ESP32-CAM pentru captarea imaginilor și aplicarea procesului de OCR pentru a recunoaște caracterele.

Unittest este un framework de testare unitară pentru limbajul de programare Python. Acesta este inclus în biblioteca standard Python, ceea ce înseamnă că nu trebuie instalat separat. Unittest oferă un cadru robust pentru scrierea și rularea testelor unitare, care sunt utilizate pentru a verifica comportamentul individual al componentelor software.

Testele unitare sunt utilizate pentru a verifica ca fiecare parte a codului funcționează corect individual. Aceste teste sunt esențiale pentru dezvoltarea de software de înaltă calitate, deoarece identifică probleme la nivelul componentelor individuale, înainte ca acestea să fie integrate în întregul sistem. Unittest oferă diverse metode pentru a testa diverse tipuri de funcții și clase, precum și pentru a verifica dacă o anumită excepție este aruncată sau nu.

Unittest are și capacitatea de a crea suite de testare, ceea ce permite gruparea testelor într-un singur loc și rularea lor într-un singur pas. Acest lucru este util atunci când există mai multe teste care trebuie rulate împreună. De asemenea, Unittest are suport pentru fixture, ceea ce permite crearea de date de testare comune pentru mai multe teste. Acest lucru poate economisi timp și efort în dezvoltarea de teste unitare.

În concluzie, Unittest este un framework puternic și flexibil de testare unitară pentru Python, care oferă o gamă largă de funcționalități și opțiuni pentru a scrie și rula teste de înaltă calitate pentru codul dvs.

Pe parcurs am întâmpinat mai mai multe probleme de dezvoltare care au necesitat soluționare imediată pentru a putea continua cu proiectul. Pentru a face față acestor provocări, am urmat următorii pași:

1. Identificarea problemei: Înainte de a încerca să rezolvăm o problemă, am petrecut timpul necesar pentru a înțelege clar problema și pentru a identifica factorii care o cauzează. Această etapă ne-a permis să găsim soluții potențiale și să evităm presupunerile greșite.
2. Testarea și validarea soluțiilor: După identificarea soluțiilor potențiale, am testat fiecare soluție într-un mediu de dezvoltare separat, utilizând metode de testare adecvate. Acest lucru ne-a permis să validăm eficacitatea fiecărei soluții și să selectăm soluția cea mai bună.
3. Implementarea și testarea finală: După ce am identificat și validat soluția optimă, am implementat soluția și am testat-o în mediul de dezvoltare final. Am asigurat că soluția implementată este scalabilă, flexibilă și îndeplinește cerințele și specificațiile proiectului.
4. Îmbunătățirea continuă: După implementarea soluției și finalizarea testelor finale, am continuat să îmbunătățim codul și să căutăm metode de optimizare a soluției noastre. Această abordare ne-a permis să îmbunătățim în mod constant soluția și să îndeplinim cerințele.

Pentru a asigura calitatea proiectului , s-au realizat și teste de performanță pentru a verifica timpul de răspuns al aplicației și consumul de resurse a modulului ESP32-CAM. Aceste teste au fost realizate utilizând seturi de date de dimensiuni diferite și au demonstrat faptul că aplicația funcționează eficient și rapid.

În concluzie , testarea a reprezentat o parte importantă a dezvoltării aplicației OCR cu modulul ESP32-CAM și ne-am asigurat că aplicația funcționează corespunzător și eficient. Testele de integrare și performanță au demonstrat că aplicația este capabilă să recunoască caracterele cu o precizie ridicată și să ofere o soluție viabilă pentru recunoașterea optică a caracterelor în diverse situații de utilizare .

# 7 Management

În timpul implementării proiectului , s-au întâmpinat anumite provocări de management care au necesitat o abordare atentă și soluționarea lor în mod eficient pentru a se asigura succesul proiectului. Unele dintre lecțiile importante învățate în timpul managementului proiectului includ :

1. Planificarea și organizarea – A fost important să se planifice și să organizeze fiecare etapă a proiectului în mod atent, inclusive definirea obiectivelor, stabilirea termenelor limită, alocarea resurselor , precum și definirea sarcinilor și responsabilităților . O bună planificare și organizare au contribuit semnificativ la atingerea obiectivelor proiectului în timp și cu resurse minime.
2. Comunicarea eficientă – Comunicarea eficientă între membrii echipei a fost crucială în reușita proiectului. A fost important să se stabiliească metode de comunicare clare și să se comunice constant cu toți membrii echipei pentru a se asigura că toată lumea este pe aceeași lungime de undă și se lucrează la obiectivele comune.
3. Managementul riscului – Identificarea și gestionarea riscurilor proiectului a fost o parte important a procesului de management . Au fost luate în considerare riscurile posibile și s-au luat măsuri preventive pentru a minimiza impactul acestora în cazul apariției lor.
4. Evaluarea continuă - A fost important să se efectueze evaluări continue ale proiectului pentru a identifica problemele și pentru a implementa soluții imediate. Aceste evaluări m-au ajutat atât pe mine ca manager cât și pe membrii echipei și să ajustăm strategia și planurile pentru a îndeplini obiectivele proiectului.

Jira este o platformă de management al proiectelor dezvoltată de compania Atlassian. Este utilizată de organizații din întreaga lume pentru a gestiona sarcinile, problemele, proiectele și rapoartele. Jira oferă un mediu centralizat în care echipele pot colabora, urmări progresele și lua decizii informate pe baza datelor și analizelor în timp real.

Funcționalitățile cheie ale Jira includ: managementul proiectelor și al sarcinilor, urmărirea problemelor și a bug-urilor, managementul timpului și al costurilor, raportarea și analiza datelor, colaborarea și comunicarea în cadrul echipei. Utilizatorii pot personaliza fluxurile de lucru, tablourile Kanban și rapoartele pentru a se potrivi nevoilor lor specifice de afaceri.

Jira oferă integrări cu alte instrumente și platforme de dezvoltare, cum ar fi GitHub, Bitbucket și Confluence, pentru a facilita colaborarea și managementul într-un mod mai eficient. În plus, Jira oferă o gamă largă de aplicații și plug-in-uri pentru a adăuga funcționalități suplimentare și pentru a îmbunătăți performanța.

În general, Jira este o platformă puternică și scalabilă pentru managementul proiectelor, utilizată de companii de toate dimensiunile și din toate industriile.

Despre metodologia de dezvoltare , am ales să folosim o abordare Agile pentru a putea gestiona mai eficient proiectul și pentru a putea face față oricăror schimbări sau cerințe care apar pe parcurs , de aceea am ales să

folosim platforma Jira pentru a ne organiza backlog-ul , pentru a urmări progresul sarcinilor și a prioritiza activitățile . Am împărțit proiectul în sprint-uri , astfel încât să putem avea un obiectiv clar pentru fiecare săptămână și să ne asigurăm că progresăm în mod eficient .

Am ținut legătura in fiecare zi cu membrul echipei , pentru a monitoriza progresul și face modificări , ajustări acolo unde a fost necesar.

Prin utilizarea metodei Agile , am reușit să livrăm o aplicație funcțională într-un timp destul de scurt și cu o echipă restrânsă.

În concluzie, partea de management a proiectului OCR Scanner ESP32-CAM a fost una esențială pentru finalizarea acestuia. Am reușit să ne coordonăm , să ne organizăm eficient activitățile.



Figura 5 - Metoda Agile

# 8 Lecții Învățate

Desfășurarea acestui proiect a implicat provocări și obstacole dar și o serie de lecții învățate care ne-au ajutat să ne îmbunătațim abilitățile și să facem față cu succes altor proiecte în viitor.

În cele ce urmează , voi enumera cele mai importante lecții pe care le-am învățat în timpul dezvoltării acestui proiect:

1. Planificarea este cheia succesului. Am învățat că planificarea este esențială pentru a ne asigura că toate sarcinile sunt realizate în timp util și că echipa lucrează într-un mod coerent și organizat. Prin crearea unui plan de proiect clar și detaliat, am reușit să evităm confuzia și să ne asigurăm că toate sarcinile au fost realizate într-un mod eficient.
2. Comunicarea este esențială. Am învățat că comunicarea deschisă și eficientă este esențială pentru succesul unui proiect. Am avut întâlniri regulate în care am discutat despre progresul proiectului, problemele întâmpinate și soluțiile găsite. Prin aceasta am reușit să ne menținem la unison și să ne asigurăm că toți membrii echipei știu exact ce trebuie să facă și când.
3. Învățarea continuă este importantă. Am învățat că este important să fim deschiși la noi idei și să ne îmbunătățim constant abilitățile. În timpul dezvoltării proiectului, am descoperit noi tehnologii și abordări pe care le-am integrat în proiect. De asemenea, am participat la cursuri și am citit documentații pentru a ne îmbunătăți cunoștințele.
4. Gestionarea timpului este cheia. Am învățat că gestionarea timpului este una dintre cele mai importante aspecte ale dezvoltării unui proiect. Prin planificarea eficientă și distribuirea sarcinilor în mod adecvat, am reușit să ne îndeplinim obiectivele în timp util.
5. Flexibilitatea este importantă. Am învățat că trebuie să fim flexibili și să ne adaptăm la schimbările care apar în timpul dezvoltării proiectului. Uneori, unele sarcini pot fi mai dificile decât altele, sau pot fi necesare modificări ale planului de proiect. În aceste situații, trebuie să fim deschiși la schimbare și să ne adaptăm pentru a atinge obiectivele noastre.
6. Încrederea în echipă este crucială. Am învățat că încrederea în membrii echipei este crucială pentru a realiza cu succes un proiect. Prin asigurarea că fiecare membru al echipei are o sarcină clară și este responsabil pentru realizarea acesteia, am putut să ne bazăm unii pe al

# 9 Concluzii

După finalizarea acestui proiect , am ajuns la următoarele concluzii :

* Tehnologia OCR ( Optical Character Recognition) este un domeniu complex și interesant , cu o varietate de aplicații practice și un potențial enorm de dezvoltare.
* Implementarea unui sistem OCR pe o placă de dezvoltarea ESP32 este o provocare tehnică dar și o oportunitate de a învăța și de a experimenta cu tehnologii noi.
* Utilizarea unei camere integrate în ESP32 pentru capturarea imaginilor poate fi o soluție eficientă pentru proiecte care necestiă un echipament compact și portabil.
* Integrarea bibliotecilor OCR existente în proiectul nostru a permis implementarea unei funcționaliăți de recunoaștere a textului pe imagini , oferind utilizatorilor un instrument puternic și ușor de utilizat.
* În timpul dezvoltării proiectului, am învățat să lucrăm cu tehnologii și instrumente noi, precum ESP-IDF, Tasseract OCR , GIT , GITHUB și altele.
* Am învățat că documentarea și planificarea sunt esențiale pentru un proiect de success și că comunicarea eficientă între membrii echipei este cheia pentru a evita erorile și pentru a accelera dezvoltarea.

Desfășurarea proiectului OCR Scanner ESP32-CAM a fost o experiență interesantă și provocatoare, dar și extrem de satisfăcătoare pentru mine și pentru echipa mea. În urma acestui proiect, am învățat multe lucruri noi și ne-am dezvoltat abilitățile de programare, de management de proiect și de colaborare în echipă.

Am avut parte de provocări și de momente de frustrare, dar am învățat să le gestionăm eficient și să găsim soluții în timp util. Colaborarea în echipă a fost foarte importantă pentru reușita proiectului și am învățat să lucrăm împreună pentru a atinge obiectivele setate.

De asemenea, am învățat să lucrăm cu tehnologii noi și să le integrăm în proiectul nostru, cum ar fi biblioteca ESP Camera, Numpy sau Matplotlib. Acest lucru ne-a adus o experiență valoroasă și ne-a îmbunătățit cunoștințele în domeniul programării.

Un alt aspect important pe care l-am învățat în cadrul acestui proiect este importanța testării și a documentației. Am învățat să testăm eficient codul nostru și să scriem o documentație clară și cuprinzătoare pentru a ne ajuta colegii și utilizatorii să înțeleagă mai bine proiectul și să lucreze cu el.

În concluzie, dezvoltarea proiectului OCR Scanner ESP32-CAM a fost o experiență foarte valoroasă și am învățat multe lucruri noi, atât din punct de vedere tehnic, cât și din punct de vedere al managementului de proiect și al colaborării în echipă. Sperăm ca proiectul nostru să fie util pentru alți utilizatori și să aducă o contribuție semnificativă în domeniul tehnologiei.

# Bibliografie

[https://github.com](https://github.com/)

[https://www.w3schools.com](https://www.w3schools.com/)

<https://www.espressif.com/en/products/socs/esp32>

<https://numpy.org/>

<https://matplotlib.org/>

<https://www.arduino.cc/en/Reference/WiFi>

<https://www.atlassian.com/software/jira>

# Appendix

# Codul Sursă