Aplicație Android pentru identificarea obiectelor în timp real utilizând machine learning

# Documentul de proiectare

Cuprins

[1. Introducere 1](#_Toc160527836)

[1.1 Scopul documentului 1](#_Toc160527837)

[2. Prezentare generală și abordări de proiectare 2](#_Toc160527838)

[2.1 Prezentare generală 2](#_Toc160527839)

[2.2 Presupuneri/ Constrângeri/ Riscuri 2](#_Toc160527840)

[2.2.1 Presupuneri 2](#_Toc160527841)

[2.2.2 Constrângeri 2](#_Toc160527842)

[2.2.3 Riscuri 3](#_Toc160527843)

[3. Considerațiii de proiectare 4](#_Toc160527844)

[3.1 Obiective și linii directoare (ghiduri) 4](#_Toc160527845)

[3.2 Metode de dezvoltare 4](#_Toc160527846)

[3.3 Strategii de arhitectură 4](#_Toc160527847)

[4. Arhitectura Sistemului și Proiectarea Arhitecturii 6](#_Toc160527848)

[4.1 Vedere logică 6](#_Toc160527849)

[4.2 Arhitectură hardware 6](#_Toc160527850)

[4.3 Arhitectură software 6](#_Toc160527851)

[4.4 Arhitectura informațiilor 7](#_Toc160527852)

[4.5 Arhitectura de comunicații interne 7](#_Toc160527853)

[4.6 Diagrama de arhitectură a sistemului 8](#_Toc160527854)

[5. Proiectarea sistemului 9](#_Toc160527855)

[5.1 Proiectarea bazei de date 9](#_Toc160527856)

[5.1.1 Obiecte de date și structuri de date rezultante 9](#_Toc160527857)

[5.1.2 Fișiere și baze de date 9](#_Toc160527858)

[5.2 Conversii de date 9](#_Toc160527859)

[5.3 Interfețe utilizator 10](#_Toc160527860)

[5.3.1 Intrări 10](#_Toc160527861)

[5.3.2 Ieșiri 10](#_Toc160527862)

[5.4 Proiectarea interfețelor cu utilizatorul 10](#_Toc160527863)

[6. Scenarii de utilizare 11](#_Toc160527864)

[7. Proiectare de detaliu 12](#_Toc160527865)

[7.1 Proiectare hardware de detaliu 12](#_Toc160527866)

[7.2 Proiectare software de deatliu 12](#_Toc160527867)

[7.3 Proiectare detaliată de securitate 13](#_Toc160527868)

[7.4 Proiectare de detaliu pentru performanța sistemului 14](#_Toc160527869)

[7.5 Proiectare detaliată a comunicațiilor interne (între componente) 14](#_Toc160527870)

[8. Controale pentru verificarea integrității sistemului 15](#_Toc160527871)

[Anexa A: Gestiunea modificărilor documentului 16](#_Toc160527872)

[Anexa B: Acronime 17](#_Toc160527873)

[Anexa C Documente la care se face referire 18](#_Toc160527874)

## Introducere

Acest Document de Proiectare a Sistemului (DPS) se referă la proiectul de dezvoltare a unei aplicații de identificare a obiectelor pentru platforma Android și descrie modul în care cerințele funcționale și non-funcționale stabilite în Documentul de Specificare a Cerințelor (SRS) sunt transpuse în specificații tehnice de proiectare a sistemului. Aplicația propusă are scopul de a permite utilizatorilor să identifice obiecte din mediul lor înconjurător folosind camera dispozitivului lor mobil.

**Numele proiectului:** Aplicație Android pentru identificarea obiectelor în timp real utilizând machine learning

**Acronimul proiectului:** ID\_OB

**Evoluția documentului:** Acest Document de Proiectare a Sistemului va continua să fie actualizat pe măsură ce se dezvoltă proiectul și pe măsură ce se fac modificări și îmbunătățiri ale sistemului. Versiunile ulterioare vor include specificații tehnice detaliate pentru fiecare componentă a sistemului și vor reflecta orice schimbări semnificative aduse arhitecturii și funcționalităților aplicației.

**Considerații referitoare la securitate și confidențialitate:** Implementarea măsurilor adecvate de securitate și respectarea normelor și reglementărilor relevante în acest domeniu sunt prioritare pentru asigurarea protecției adecvate a informațiilor personale și a datelor de identificare a utilizatorilor. Aceste aspecte sunt tratate în detaliu în secțiunile ulterioare ale documentului.

### Scopul documentului

Scopul acestui Document de Proiectare a Sistemului (DPS) este de a oferi o structură și un cadru clar pentru definirea arhitecturii și designului aplicației Android pentru identificarea obiectelor în timp real utilizând machine learning (ID\_OB). Obiectivele specifice ale acestui document sunt:

1. **Definirea arhitecturii sistemului:** Documentarea arhitecturii aplicației AOI, inclusiv structura sa generală, modulele componente și interacțiunile între acestea.
2. **Specificarea detaliată a designului:** Prezentarea specificațiilor tehnice detaliate pentru fiecare componentă a aplicației, inclusiv interfețe, algoritmi și logica de programare.
3. **Asigurarea coerenței și clarității:** Furnizarea unei descrieri clare și coerente a arhitecturii și designului aplicației, astfel încât echipele de dezvoltare să aibă o înțelegere comună a direcției proiectului.
4. **Ghidare pentru echipa de dezvoltare:** Furnizarea de îndrumări și recomandări pentru echipa de dezvoltare în implementarea efectivă a sistemului conform specificațiilor stabilite.
5. **Sursă de referință:** Servirea drept sursă de referință pentru toate părțile interesate în dezvoltarea și implementarea aplicației AOI, inclusiv managerul de proiect, echipa de proiect și clientul/utilizatorul.

## Prezentare generală și abordări de proiectare

Această secțiune descrie principiile și strategiile care vor fi utilizate ca ghiduri în momentul proiectării și implementării sistemului.

### Prezentare generală

Aplicația pentru idenficarea obiectelor în timp real (ID\_OB) are ca scop principal furnizarea unei soluții eficiente pentru identificarea obiectelor din mediul înconjurător folosind dispozitivele mobile cu sistem de operare Android. Arhitectura sistemului este concepută pentru a facilita procesul de recunoaștere a obiectelor în timp real și pentru a oferi utilizatorilor informațiile necesare despre obiectele identificate.

Principalele componente ale arhitecturii sistemului ID\_OB includ:

1. **Interfața cu utilizatorul (UI):** Este componenta prin intermediul căreia utilizatorii interacționează cu aplicația. Interfața utilizatorului este proiectată pentru a fi intuitivă și ușor de navigat, oferind funcționalități precum încărcarea imaginilor, activarea camerei pentru recunoașterea obiectelor și vizualizarea rezultatelor identificării.
2. **Modul de recunoaștere a obiectelor:** Acest modul este responsabil pentru procesarea imaginilor și identificarea obiectelor din fluxul video sau din imaginile încărcate de utilizator. Algoritmii de recunoaștere a obiectelor sunt implementați aici pentru a detecta și a clasifica obiectele prezente în imagine.
3. **Modul de identificare a culorilor:** Acest modul permite utilizatorilor să exploreze și să identifice culorile din imagini. Prin intermediul acestui modul, utilizatorii pot obține informații despre tonalitățile de culoare și combinațiile de culori prezente în imagine.

### Presupuneri/ Constrângeri/ Riscuri

#### Presupuneri

**Hardware-ul dispozitivului:** Se presupune că dispozitivele mobile pe care va rula aplicația ID\_OB dispun de o cameră foto de calitate și de suficientă performanță pentru a captura imagini clare și detaliate. De asemenea, se presupune că dispozitivele sunt compatibile cu versiunile recente ale sistemului de operare Android.

**Utilizatori finali:** Se presupune că utilizatorii aplicației ID\_OB au un nivel de confort minim în utilizarea dispozitivelor mobile și în navigarea interfeței cu utilizatorul. De asemenea, se presupune că aceștia sunt interesați de identificarea și explorarea obiectelor din mediul lor înconjurător.

**Actualizări ale sistemului de operare:** Se presupune că aplicația va fi adaptată la modificările și actualizările viitoare ale sistemului de operare Android, pentru a asigura compatibilitatea și funcționarea corectă a sistemului pe termen lung.

**Servicii de recunoaștere a obiectelor:** Se presupune că serviciile de recunoaștere a obiectelor, utilizate de aplicație, sunt disponibile și funcționale, și că pot furniza rezultate precise și relevante în timp real.

#### Constrângeri

**Limitări ale capacității de procesare a dispozitivelor mobile:** Capacitatea de procesare a dispozitivelor mobile poate fi limitată, ceea ce poate afecta performanța aplicației ID\_OB, în special în ceea ce privește timpul de răspuns al recunoașterii obiectelor. Acest lucru poate necesita optimizarea algoritmilor de recunoaștere și utilizarea eficientă a resurselor hardware disponibile.

**Cerințe de stocare a datelor:** Datele colectate și procesate de aplicația ID\_OB trebuie să fie stocate în mod eficient pe dispozitivul mobil al utilizatorului. Limitările de spațiu de stocare pot impune restricții asupra cantității de date care pot fi reținute local și pot necesita o strategie adecvată de gestionare a memoriei pentru a evita suprascrierea sau pierderea datelor.

**Cerințe de interoperabilitate:** Aplicația ID\_OB trebuie să fie interoperabilă cu diverse dispozitive mobile și sisteme de operare Android, pentru a asigura compatibilitatea și funcționarea corectă pe diferite platforme și configurații hardware.

**Cerințe de performanță:** Performanța aplicației ID\_OB trebuie să fie optimizată pentru a asigura o experiență fluidă și fără întreruperi pentru utilizatori. Limitările de performanță pot afecta timpul de răspuns al aplicației și capacitatea acesteia de a gestiona sarcini complexe, cum ar fi recunoașterea obiectelor în timp real.

#### Riscuri

**Riscul de performanță scăzută:** Există riscul ca aplicația ID\_OB să nu funcționeze la nivelul de performanță așteptat, determinând întârzieri în recunoașterea obiectelor sau încetinirea generală a aplicației, ceea ce ar putea duce la o experiență negativă a utilizatorului.

* **Strategie de reducere a riscului:** Pentru a reduce acest risc, se vor efectua teste ample de performanță pentru a evalua și optimiza timpul de răspuns al aplicației. De asemenea, se va face o analiză constantă a posibilităților de optimizare a codului și a algoritmilor pentru a asigura o performanță optimă.

**Riscul de erori de recunoaștere a obiectelor:** Există riscul ca algoritmii de recunoaștere a obiectelor să nu fie suficient de preciși sau să nu recunoască corect anumite tipuri de obiecte, ceea ce ar putea duce la furnizarea de informații inexacte sau incomplete utilizatorilor.

* **Strategie de reducere a riscului:** Pentru a minimiza acest risc, se vor efectua teste extinse de validare a algoritmilor de recunoaștere a obiectelor, folosind seturi de date variate și representative.

## Considerațiii de proiectare

**Utilizabilitatea și experiența utilizatorului:** Interfața cu utilizatorul trebuie să fie intuitivă și ușor de înțeles pentru utilizatori. Se vor lua în considerare fluxurile de lucru ale utilizatorilor și se va optimiza navigarea în aplicație pentru a oferi o experiență plăcută și eficientă.

**Eficiența și performanța:** Trebuie să se proiecteze aplicația astfel încât să ofere o performanță rapidă și fără probleme, în special în ceea ce privește recunoașterea obiectelor în timp real. Este esențial să se optimizeze algoritmii și gestionarea resursele hardware disponibile pentru a asigura o funcționare fluentă și rapidă a aplicației.

**Dependențele de resurse externe:** Trebuie să se evalueze dependențele aplicației față de serviciile sau resursele externe, cum ar fi serviciile de recunoaștere a obiectelor bazate pe modele deja antrenate. Trebuie să existe un plan de rezervă în cazul în care aceste servicii devin indisponibile sau sunt modificate, pentru a asigura că aplicația rămâne funcțională și utilă pentru utilizatori.

### Obiective și linii directoare (ghiduri)

**Performanță optimă:** Prioritatea principală este asigurarea unei performanțe optime a aplicației, în special în ceea ce privește recunoașterea obiectelor în timp real. Accentul se pune pe minimizarea timpului de răspuns al aplicației și maximizarea eficienței utilizării resurselor hardware disponibile.

**Utilizabilitate și experiență a utilizatorului:** Aplicația trebuie să ofere o interfață cu utilizatorul intuitivă și ușor de folosit, care să permită utilizatorilor să navigheze fără dificultăți și să obțină rezultatele dorite cu ușurință. Accentul se pune pe crearea unei experiențe de utilizare plăcute și fără probleme.

**Flexibilitate și extensibilitate:** Arhitectura sistemului trebuie să fie flexibilă și modulară, permițând extinderea și adaptarea ulterioară a funcționalităților în funcție de nevoile și cerințele viitoare. Accentul se pune pe crearea unei platforme care să permită adăugarea ușoară a noi caracteristici și funcționalități.

**Fiabilitate și stabilitate:** Aplicația trebuie să fie fiabilă și stabilă, minimizând erorile și defectele de funcționare. Accentul se pune pe testarea riguroasă a aplicației și identificarea și remedierea eficientă a oricăror probleme sau erori.

### Metode de dezvoltare

Pentru designul sistemului și al software-ului aplicației ID\_OB, se aplică o abordare orientată pe obiecte, îmbinată cu principii ale dezvoltării agile. Această metodă permite crearea unui sistem modular și flexibil, care să permită dezvoltarea iterativă și adaptarea la schimbările în cerințe și tehnologie.

**Orientarea pe obiecte:** Utilizând o abordare orientată pe obiecte, se dezvoltă aplicația ID\_OB în jurul unor entități distincte și interconectate, numite obiecte. Acest lucru permite structurarea aplicației într-un mod clar și modular, cu fiecare obiect responsabil pentru o anumită funcționalitate sau aspect al aplicației.

**Metodologii și tehnologii:** Se utilizează tehnologii moderne și instrumente de dezvoltare pentru a construi aplicația ID\_OB, inclusiv limbaje de programare precum Kotlin și Flutter pentru dezvoltarea aplicației Android, și biblioteci și framework-uri pentru funcționalități specifice, cum este TensorFlow pentru recunoașterea obiectelor.

**Contingente și planuri alternative:** În timpul procesului de dezvoltare, este posibil să se întâmpine contingente cum ar fi schimbări în cerințe sau tehnologii, sau lipsa acordurilor de interfață cu serviciile externe. Pentru a gestiona aceste situații, se dezvoltă planuri de rezervă și soluții alternative, inclusiv explorarea de alternative tehnologice și comunicarea deschisă cu clienții și alte părți interesate pentru a găsi cea mai bună cale de acțiune.

### Strategii de arhitectură

**Utilizarea limbajului de programare Flutter:** S-a decis utilizarea limbajului de programare Flutter datorită capacității sale de a crea interfețe utilizator moderne și consistente pe multiple platforme, inclusiv Android și iOS. Alegerea acestui limbaj este justificată de performanța sa ridicată, flexibilitatea în dezvoltarea interfeței utilizator și suportul puternic pentru personalizare.

**Reutilizarea componentelor software existente:** S-au identificat și reutilizat pachete și biblioteci Flutter existente atunci când a fost posibil, pentru a minimiza efortul de dezvoltare și pentru a asigura consistența și eficiența sistemului. Această abordare are ca scop reducerea timpului și costurilor de dezvoltare și promovarea dezvoltării sustenabile a aplicației.

**Extensibilitatea și îmbunătățirea continuă:** S-a planificat o arhitectură care să permită extinderea și îmbunătățirea continuă a aplicației în viitor. Astfel, s-au adoptat practici și modele de proiectare care facilitează adăugarea ulterioară de funcționalități și îmbunătățirea performanței și scalabilității sistemului pe măsură ce cerințele și tehnologiile evoluează.

**Interfața utilizatorului intuitivă:** S-a ales un model de interfață utilizator intuitiv și ușor de folosit, bazat pe principii moderne de design. Acesta include utilizarea widget-urilor Flutter corespunzătoare și crearea unui flux de lucru fluid și intuitiv pentru a oferi utilizatorilor o experiență plăcută de utilizare.

**Utilizarea pachetelor de ML Flutter:** Pentru funcționalitatea de recunoaștere a obiectelor, se vor folosi pachete de machine learning compatibile cu Flutter, adaptate pentru procesarea imaginilor și recunoașterea obiectelor în timp real. Alegerea acestor pachete este justificată de compatibilitatea lor cu limbajul de programare și de performanța lor ridicată în contextul aplicației ID\_OB.

## Arhitectura Sistemului și Proiectarea Arhitecturii

**Alcătuirea sistemului:**

* Interfața Utilizator (UI): Această componentă este responsabilă pentru prezentarea interfeței utilizatorului și pentru interacțiunea cu utilizatorii. Ea cuprinde elemente de design și interacțiuni utilizator-centrat, precum și logica necesară pentru a gestiona evenimentele de interfață.
* Motorul de Recunoaștere a Obiectelor: Acest subsistem este dedicat procesării fluxului video și recunoașterii obiectelor din imagini. Folosind tehnici de învățare automată și rețele neurale, acesta analizează datele de intrare și identifică obiectele relevante.
* Serviciul de Comunicare cu Baza de Date: Această componentă asigură interacțiunea cu baza de date locală pentru stocarea și gestionarea datelor. Ea gestionează cererile de acces la date și asigură consistența și securitatea datelor.
* Modulul de Interfață cu Alte Aplicații: Acest subsistem facilitează comunicația și integrarea cu alte aplicații sau servicii externe, permițând schimbul de date și funcționalități între diferite platforme sau sisteme.

**Colaborarea între componente:**

Interacțiunea între componente este realizată prin intermediul unor interfețe bine definite și a unor protocoluri de comunicare standardizate. De exemplu, interfața utilizatorului interacționează cu motorul de recunoaștere a obiectelor pentru a primi și afișa rezultatele identificării obiectelor. Apoi, motorul de recunoaștere a obiectelor comunică cu serviciul de comunicare cu baza de date pentru a stoca și recupera informații relevante despre obiecte.

Această secțiune conturează designul arhitecturii sistemului și hardware-ului.

### Vedere logică

Vederea logică poate fi regăsită în detaliile implementării, în codul sursă al fiecărei componente și în interfețele de comunicare definite între ele. Acestea se regăsesc în Documentul de Specificare a Cerințelor (SRS).

### Arhitectură hardware

Detaliile hardware ale proiectului depind de dispozitivul utilizatorului, astfel cerințele minime ale acestuia au fost descrise in Documentul de Specificare a Cerințelor.

### Arhitectură software

Arhitectura software a aplicației ID\_OB cuprinde mai multe componente și tehnologii care colaborează pentru a oferi funcționalitatea necesară. Acestea includ:

1. **Framework Flutter:** Folosit pentru dezvoltarea interfeței utilizatorului (UI), framework-ul Flutter oferă un set de instrumente și biblioteci pentru crearea unei interfețe moderne și interactive. Acesta include widget-uri predefinite și capacitatea de a crea widget-uri personalizate pentru a satisface cerințele specifice ale aplicației.
2. **TensorFlow Lite:** Utilizat pentru recunoașterea obiectelor din fluxul video al dispozitivului, TensorFlow Lite oferă capacități avansate de învățare automată și recunoaștere a modelelor. Acesta este integrat în aplicație pentru a identifica și clasifica obiectele detectate în timp real.
3. **Limbaj de programare Dart:** Limbajul de programare principal utilizat în dezvoltarea aplicației este Dart. Acesta este utilizat pentru scrierea codului sursă al aplicației și pentru definirea logicii de afaceri, inclusiv a algoritmilor de recunoaștere a obiectelor și a interacțiunilor cu baza de date.
4. **Biblioteci externe:** În plus față de componentele menționate mai sus, aplicația poate utiliza diverse biblioteci externe și pachete Flutter pentru a extinde funcționalitățile sale și pentru a îmbunătăți experiența utilizatorului. Aceste biblioteci pot include soluții pentru gestionarea stării aplicației, manipularea imaginilor sau lucrul cu datele geografice.

### Arhitectura informațiilor

**Date despre obiectele identificate:** Acestea sunt datele referitoare la obiectele detectate în fluxul video al dispozitivului utilizatorului. Aceste informații pot include caracteristici precum numele obiectului, categoria sau tipul obiectului, coordonatele sale în imagine și alte atribute relevante.

**Date de configurare și setări:** Aceste date sunt utilizate pentru a gestiona configurația și preferințele aplicației, cum ar fi limba preferată, tema sau modul de afișare a informațiilor.

**Date referitoare la interacțiunea utilizatorului:** Acestea sunt date despre modul în care utilizatorii interacționează cu aplicația, cum ar fi istoricul acțiunilor, preferințele de utilizare și alte metrici de angajament.

### Arhitectura de comunicații interne

În cadrul aplicației ID\_OB, nu există o infrastructură de rețea complexă, deoarece sistemul este în principal autonom și incorporat în dispozitivele mobile. Cu toate acestea, putem privi interfața dintre componentele hardware și software ca pe o formă simplă de comunicare în rețea.

Arhitectura de comunicații interne a aplicației ID\_OB implică interacțiunea dintre diferitele componente ale aplicației, precum camera dispozitivului mobil, modulele de procesare a imaginilor, interfața utilizatorului și alte servicii necesare pentru funcționarea corectă a aplicației. Aceasta include:

1. **Modulul de captare a imaginilor:** Acesta este responsabil pentru accesarea camerei dispozitivului mobil și capturarea imaginilor din mediu. Comunicarea cu camera este realizată prin intermediul API-urilor specifice platformei Android.
2. **Modulele de prelucrare a imaginilor:** Imaginile capturate sunt trimise către modulele de prelucrare a imaginilor, unde sunt procesate pentru identificarea obiectelor sau a culorilor. Aceste module utilizează algoritmi de învățare automată sau alte tehnici de prelucrare a imaginilor pentru a realiza aceste operațiuni.
3. **Interfața utilizatorului (UI):** Rezultatele prelucrării imaginilor sunt afișate utilizatorului prin intermediul interfeței utilizatorului. Utilizatorul poate interacționa cu aplicația, de exemplu, prin intermediul butoanelor pentru activarea funcțiilor de identificare a obiectelor sau a culorilor.
4. **Comunicarea între module:** Modulele din aplicație comunică între ele pentru a transfera date și comenzi necesare pentru funcționarea corespunzătoare a aplicației. De exemplu, imaginea capturată de modulul de captare a imaginilor este trimisă către modulele de prelucrare a imaginilor pentru analiză.

### Diagrama de arhitectură a sistemului

A white rectangular sign with black text

Description automatically generated

## Proiectarea sistemului

### Interfețe utilizator

**Utilizatori finali:**

* Descriere: Acești utilizatori sunt persoanele obișnuite care folosesc aplicația pentru a identifica obiecte din mediul lor.
* Responsabilități comune: Utilizarea interfeței aplicației pentru identificarea și explorarea obiectelor.
* Nivel de competență: Diferit, de la începători până la avansați în utilizarea tehnologiei.
* Activități de lucru: Utilizarea funcțiilor aplicației pentru a recunoaște și a obține informații despre obiectele din jur.
* Număr de utilizatori externi: Numărul utilizatorilor care utilizează aplicația prin intermediul rețelei de internet.

**Administratori:**

* Descriere: Acești utilizatori au privilegii speciale pentru gestionarea și administrarea aplicației.
* Responsabilități comune: Administrarea conturilor utilizatorilor, gestionarea bazei de date, monitorizarea performanței și a securității aplicației.
* Nivel de competență: Avansat în ceea ce privește cunoștințele tehnice și administrarea sistemului.
* Activități de lucru: Configurarea și întreținerea aplicației, rezolvarea problemelor și asigurarea securității.
* Număr de utilizatori externi: Nu este aplicabil.

#### Intrări

**Interfața utilizatorului (UI) pe dispozitivele mobile:**

* Utilizatorii introduc date folosind ecrane tactile și tastaturile virtuale ale dispozitivelor lor mobile.
* Ecranele de introducere a datelor includ vizualizarea camerei pentru capturarea imaginilor și butoanele pentru activarea funcțiilor, cum ar fi recunoașterea obiectelor sau identificarea culorilor.
* Utilizatorii pot selecta opțiuni din meniuri derulante, butoane de comutare și alte elemente de interfață grafică.

#### Ieșiri

**Ecrane de afișare a rezultatelor:**

* Ecranele de afișare prezintă utilizatorilor rezultatele procesării, cum ar fi obiectele recunoscute din imagini sau identificarea culorilor. Acestea sunt ecranele dispozitivelor folosite de utilizatori.
* Aceste ecrane includ imagini sau informații textuale relevante, precum etichete pentru obiecte recunoscute sau denumiri pentru culori identificate.
* Utilizatorii pot interacționa cu aceste ecrane pentru a vizualiza detalii suplimentare sau pentru a efectua acțiuni suplimentare, cum ar fi salvarea sau partajarea rezultatelor.

### Proiectarea interfețelor cu utilizatorul

**Interfața principală:**

* Acesta este ecranul de pornire al aplicației, unde utilizatorii sunt întâmpinați cu opțiuni pentru a începe procesul de recunoaștere a obiectelor sau identificare a culorilor.
* Interfața principală este simplă și clară, cu butoane mari și distincte pentru fiecare funcționalitate principală a aplicației.

**Ecranul de recunoaștere a obiectelor:**

* Acest ecran permite utilizatorilor să încarce sau să captureze imagini pentru a recunoaște obiectele prezente în acestea.
* Există un buton de captură foto și un buton de încărcare a imaginii pentru a oferi utilizatorilor opțiuni multiple pentru introducerea imaginilor.

**Ecranul de identificare a culorilor:**

* Acest ecran permite utilizatorilor să selecteze sau să identifice culori dintr-o paletă sau dintr-o imagine.
* Interfața include o paletă de culori interactivă și un instrument de pipetă pentru a selecta culorile direct dintr-o imagine.
* După ce o culoare este identificată, utilizatorii pot vedea numele și codul RGB asociat cu aceasta.

## Scenarii de utilizare

Funcționalitatea generală a aplicației constă în furnizarea unei platforme interactive pentru recunoașterea și clasificarea culorilor utilizând camera dispozitivului mobil. Utilizatorii pot accesa aplicația pentru a identifica culorile din mediul lor înconjurător și pentru a primi informații detaliate despre acestea. Interfața aplicației este concepută pentru a fi intuitivă, facilitând utilizatorilor procesul de recunoaștere a culorilor și de navigare prin diferite opțiuni și setări disponibile.

Scenariile de funcționare ale sistemului pot fi:

1. **Recunoașterea obiectelor din fluxul video:**
   * Utilizatorul deschide aplicația și selectează opțiunea de recunoaștere a obiectelor.
   * Utilizatorul apasă butonul de captură foto pentru a încărca o imagine sau folosește camera dispozitivului pentru a captura în timp real.
   * Aplicația procesează imaginea și identifică obiectele prezente în aceasta.
   * Rezultatele recunoașterii sunt afișate utilizatorului sub formă de liste sau diagrame.
2. **Identificarea culorilor din imagini:**
   * Utilizatorul selectează opțiunea de identificare a culorilor din meniul principal al aplicației.
   * Utilizatorul poate selecta o imagine din galeria dispozitivului sau poate utiliza camera pentru a fotografia o scenă.
   * După selectarea sau capturarea imaginii, utilizatorul poate folosi pipeta de culoare pentru a identifica culorile din imagine.
   * Numele și codurile RGB ale culorilor identificate sunt afișate utilizatorului pentru referință.
3. **Interacțiunea cu rezultatele:**
   * După ce sunt afișate rezultatele recunoașterii obiectelor sau identificării culorilor, utilizatorul poate interacționa cu acestea.
   * Utilizatorul poate selecta un obiect sau o culoare pentru a vedea mai multe detalii sau pentru a efectua acțiuni suplimentare.

## Proiectare de detaliu

În această secțiune, voi furniza informațiile necesare pentru a construi și integra efectiv componentele hardware, pentru a codifica și integra componentele software și pentru a interconecta segmentele hardware și software într-un produs funcțional. De asemenea, voi aborda procedurile detaliate pentru combinarea pachetelor separate externe într-un singur sistem. Aceast lucru este crucial pentru realizarea cu succes a produsului final și pentru asigurarea coerenței și integrității acestuia. Prin detalierea proceselor și procedurilor, imi propun sa facilitez dezvoltarea eficientă și integrarea fluidă a tuturor componentelor, asigurând că fiecare aspect al sistemului este gestionat cu precizie și că toate părțile se sincronizează armonios pentru a crea un produs final de înaltă calitate.

### Proiectare hardware de detaliu

Pentru a realiza proiectarea de detaliu a aplicației ID\_OB (identificare obiecte), este necesar să se furnizeze informații esențiale echipei de dezvoltare în următoarele domenii:

**Componente Hardware:**

* Specificațiile hardware necesare pentru funcționarea aplicației mobile pe diverse dispozitive Android, cum ar fi telefoanele inteligente și tabletele.
* Cerințele de securitate pentru protejarea datelor de identificare a obiectelor și a datelor utilizatorilor pe dispozitivele mobile.
* Integrarea componentelor hardware, cum ar fi camera dispozitivului pentru capturarea fluxului video în timp real în vederea identificării obiectelor.

**Interconectarea Hardware și Software:**

* Detalii despre modul în care aplicația mobilă interacționează cu hardware-ul dispozitivului, cum ar fi utilizarea camerei pentru captarea fluxului video și identificarea obiectelor în timp real.
* Configurarea și integrarea bibliotecilor de procesare a imaginilor și a altor componente software esențiale pentru funcționarea corectă a aplicației de identificare a obiectelor.

**Proceduri de Integrare a Componentelor Externe:**

* Documentarea procedurilor pentru integrarea bibliotecilor de procesare a imaginilor, API-urilor de identificare a obiectelor și a altor servicii externe necesare pentru funcționarea aplicației de identificare a obiectelor.
* Testarea și validarea corectitudinii integrării pachetelor separate externe în aplicația finală.

**Securitatea:**

* Implementarea protocoalelor de securitate pentru protejarea datelor de identificare a obiectelor și a datelor utilizatorilor.
* Proceduri detaliate pentru testarea securității aplicației și a infrastructurii subiacente.

### Proiectare software de deatliu

**Identificator serviciu:** ID\_OB\_Serv

**Clasificare:** Serviciu de aplicație

**Definiție:** Acest serviciu se ocupă de procesarea imaginilor provenite de la camera dispozitivului pentru identificarea obiectelor în timp real. Utilizează algoritmi de recunoaștere vizuală pentru detectarea și clasificarea obiectelor din imagini.

**Cerințe:** Satisfacerea cerințelor funcționale pentru identificarea obiectelor în imagini, precum și cerințele nonfuncționale legate de performanță și stabilitate.

**Structuri de date interne:** Matrice de pixeli pentru reprezentarea imaginilor, liste de obiecte identificate.

**Constraingeri:** Limitarea timpului de răspuns pentru identificarea obiectelor, precum și limitările de memorie pentru manipularea imaginilor și a datelor asociate.

**Compoziție:** Acest serviciu utilizează subservicii pentru preprocesarea imaginilor, extragerea caracteristicilor și clasificarea obiectelor.

**Utilizatori/Interacțiuni:** Interacționează cu interfața utilizatorului pentru a primi imagini de la camera dispozitivului și afișează rezultatele identificării obiectelor.

**Procesare:** Folosește algoritmi de preprocesare a imaginilor pentru îmbunătățirea calității acestora, extrage caracteristicile relevante și aplică modele de învățare automată pentru clasificarea obiectelor.

**Interfețe/Exporturi:** Oferă funcții pentru prelucrarea imaginilor și identificarea obiectelor, inclusiv metode pentru încărcarea imaginilor, extragerea caracteristicilor și returnarea rezultatelor identificării obiectelor.

### Proiectare detaliată de securitate

**Autentificare:** Autentificarea nu este necesară, deoarece aplicația nu are acces la internet și este proiectată pentru o utilizare ușoară.

**Criptare:** Criptarea va fi utilizată pentru a proteja integritatea bibliotecii de date a modulului machine learning, asigurând că aceasta rămâne intactă și nemanipulată pentru a asigura o identificare corectă a obiectelor. Datele stocate offline vor fi criptate pentru a preveni accesul neautorizat și manipularea lor.

**Utilizarea porturilor de rețea:** Porturile de rețea folosite de aplicație vor fi limitate și securizate pentru a asigura că numai traficul autorizat este permis. Deși aplicația funcționează offline, este important să se prevină orice expunere la riscuri de securitate atunci când este conectată la internet pentru actualizări sau alte scopuri.

**Detectare și prevenire a intruziunilor:** Se vor implementa mecanisme de detectare a intruziunilor pentru a monitoriza și a preveni accesul neautorizat la aplicație și la datele utilizatorilor. Aceste mecanisme vor asigura integritatea și securitatea datelor stocate offline, precum și protejarea împotriva oricăror încercări de manipulare a procesului de identificare a obiectelor.

Este esențial să se mențină actualizată aplicația și infrastructura subiacentă pentru a preveni și a remedia orice vulnerabilități de securitate care ar putea apărea în timpul utilizării sau în urma actualizărilor.

## Controale pentru verificarea integrității sistemului

1. **Securitate internă:**

Accesul la datele critice va fi restricționat astfel încât doar tipurile necesare de acces să fie permise utilizatorilor și operatorilor. Aceasta include definirea unor roluri și permisiuni clare pentru utilizatori, astfel încât fiecare utilizator să aibă acces numai la datele relevante pentru sarcinile lor specifice. De asemenea, vor fi implementate tehnologii de criptare și autentificare pentru a proteja datele împotriva accesului neautorizat.

1. **Proceduri de audit:**

Se vor implementa proceduri de audit pentru a îndeplini cerințele de control, raportare și perioade de reținere pentru rapoartele operaționale și de management. Aceste proceduri vor asigura o monitorizare adecvată a activității din sistem și vor facilita o reacție rapidă la orice incidente de securitate sau anomalii. De asemenea, vor fi stabilite politici și protocoale clare pentru gestionarea și analizarea datelor de audit.

1. **Piste de auditare a aplicațiilor:**

Aplicația va fi dotată cu piste de auditare pentru a înregistra dinamic accesul și modificările la datele critice designate. Aceste piste de auditare vor oferi o înregistrare detaliată a activității utilizatorilor și vor permite investigarea oricăror activități suspecte sau neautorizate. În plus, vor fi implementate mecanisme de alertare pentru a semnala imediat orice activități neobișnuite sau potențial periculoase.

1. **Tabele standard pentru validare:**

Se vor utiliza tabele standard pentru a valida câmpurile de date și pentru a asigura integritatea și consistența datelor stocate în sistem. Aceste tabele standard vor servi ca punct de referință pentru verificarea datelor introduse sau modificate în aplicație. De asemenea, vor fi stabilite reguli și restricții clare pentru datele introduse de utilizatori, pentru a preveni introducerea eronată sau incorectă a informațiilor.

1. **Procese de verificare a datelor critice:**

Vor fi stabilite procese clare și bine definite pentru adăugarea, ștergerea sau actualizarea datelor critice. Aceste procese vor include etape de verificare și aprobare pentru a asigura exactitatea și acuratețea datelor și vor contribui la prevenirea erorilor sau a manipulării neautorizate a datelor. În plus, vor fi implementate mecanisme de monitorizare a modificărilor pentru a urmări orice schimbare în datele critice și a preveni accesul neautorizat.

1. **Capacitatea de identificare a informațiilor de auditare:**

Sistemul va fi dotat cu capacitatea de a identifica toate informațiile de auditare prin identificarea utilizatorului, identificarea terminalului de rețea, dată, oră și datele accesate sau modificate. Aceasta va permite o urmărire precisă a activității și va facilita investigarea oricăror incidente sau anomalii de securitate. De asemenea, vor fi implementate tehnologii de monitorizare a accesului pentru a detecta și preveni utilizarea neautorizată a datelor și a resurselor.

Anexa A: Gestiunea modificărilor documentului

Tabel 1 – Înregistrarea modificărilor asupreaa documentului curent

| versiune | Data | Autorul/Deținătorul | Descriere |
| --- | --- | --- | --- |
| 1.0 | 05.03.2024 | Bratu Raluca-Ioana | Adaugare sectiune 4 |
| 1.1 | 08.04.2024 | Bratu Raluca-Ioana | Adaugare restul sectiunilor |

Anexa C Documente la care se face referire

Tabel 3 – Documente la care se facce referire

| Nume document | Locație sau URL | Dată emitere document |
| --- | --- | --- |
| Document de specificare a cerintelor | https://github.com/ralucaib/ASIPSI-Aplicatie-identificare-obiecte | 05.03.2024 |