

Titlu Proiect: Dermify – Centru de sănătate

Elevii:

Ionescu Marius Eduard și Turigioiu Raluca Ana Maria

Profesor îndrumător:

Ionică Daniela



Dermify: Intelligent Skincare Analysis Platform

Capitolul I. Alegerea tematicii

Secțiunea I.1. Analiza pieței

Dermify a fost dezvoltat pentru a răspunde unei nevoi reale în piața produselor de îngrijire a pielii: lipsa unei platforme care să combine analiza imagistică a pielii prin inteligență artificială cu recomandări personalizate de ingrediente și produse.

****Elemente distinctive față de aplicațiile existente:****

1. ****Analiză avansată prin YOLO v8****: Spre deosebire de alte aplicații care oferă doar analize superficiale bazate pe chestionare, Dermify utilizează un model YOLOv8 personalizat pentru a detecta cu precizie problemele pielii din imagini.
2. ****Bază de date comprehensivă de ingrediente****: Majoritatea aplicațiilor de skincare se concentrează pe produse, în timp ce Dermify include o bibliotecă extensivă de ingrediente cu explicații detaliate privind beneficiile și potențialele riscuri.
3. ****Rutine personalizate****: Platforma nu doar recomandă produse individuale, ci creează rutine complete adaptate tipului de piele și problemelor specifice ale utilizatorului.
4. ****Abordare holistică****: Dermify îmbină expertiza dermatologică cu tehnologia AI pentru a oferi o experiență educațională utilizatorilor, ajutându-i să înțeleagă nevoile pielii lor.

****Publicul țintă:****

Dermify se adresează mai multor segmente de utilizatori:

- Persoane cu probleme specifice ale pielii (acnee, hiperpigmentare, îmbătrânire etc.)
- Utilizatori interesați de ingrediente și compoziția produselor de skincare
- Entuziaști ai domeniului skincare care doresc să-și optimizeze rutina de îngrijire
- Profesioniști din domeniul frumuseții care caută un instrument de recomandare bazat pe evidențe

Secțiunea I.2. Planificarea dezvoltării

Dezvoltarea platformei Dermify a urmat un plan bine structurat:

****Faza 1: Cercetare și conceptualizare****

- Studiu de piață pentru identificarea nevoilor utilizatorilor
- Consultarea cu experți dermatologi pentru validarea conceptului
- Definirea arhitecturii și selectarea stack-ului tehnologic

****Faza 2: Dezvoltare MVP (Minimum Viable Product)****

- Implementarea funcționalităților de bază pentru autentificare și profil utilizator
- Dezvoltarea componentei de analiză a pielii cu YOLO v8
- Crearea bazei de date pentru ingrediente și produse
- Dezvoltarea algoritmului de recomandare

****Faza 3: Testare și optimizare****

- Testare internă pentru identificarea și rezolvarea bugurilor
- Optimizare pentru performanță și compatibilitate cross-browser

- Testare cu utilizatori reali pentru validare și feedback

****Faza 4: Lansare și dezvoltare continuă****

- Lansarea versiunii beta publice
- Implementarea de noi funcționalități bazate pe feedback
- Extinderea bazei de date cu produse și ingrediente
- Îmbunătățirea algoritmilor de analiză și recomandare

Capitolul II. Implementarea aplicației

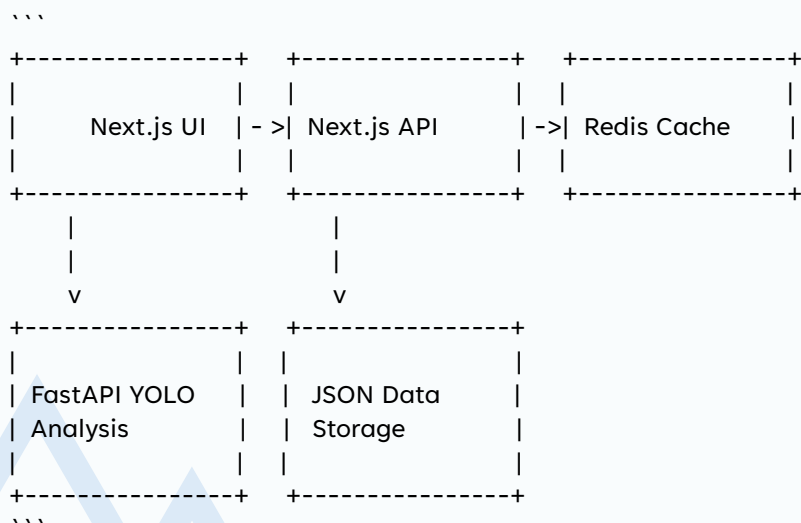
Secțiunea II.1. Proiectarea arhitecturală

Arhitectura Dermify a fost concepută pentru a fi modulară, scalabilă și ușor de extins. Am adoptat o arhitectură bazată pe microservicii pentru a permite dezvoltarea independentă a diferitelor componente și pentru a facilita scalarea.

****Componente principale:****

1. ****Frontend:**** Aplicație Next.js (React) care asigură o interfață rapidă și responsivă, utilizând Tailwind CSS pentru stilizare.
2. ****Backend API:**** Servicii API RESTful dezvoltate cu Next.js API routes pentru gestionarea datelor utilizatorilor, produselor și ingredientelor.
3. ****Serviciul de analiză a pielii:**** Microserviciu Python bazat pe FastAPI care utilizează modelul YOLO pentru analiza imaginilor.
4. ****Baze de date:****
 - Redis pentru caching și sesiuni
 - JSON pentru stocarea datelor despre produse și ingrediente
5. ****Sistem de autentificare:**** Implementat cu NextAuth.js pentru securitate și flexibilitate.

****Diagrama arhitecturală simplificată:****



Această arhitectură permite:

- ****Separarea responsabilităților**** între componente
- ****Dezvoltare independentă**** a diferitelor module
- ****Scalabilitate**** la nivel de componentă, în funcție de necesități

- **Extensibilitate** pentru adăugarea de noi funcționalități sau integrarea cu alte servicii

Secțiunea II.2. Tehnologiile folosite

Selecția tehnologiilor pentru Dermify a fost făcută cu atenție, ținând cont de cerințele specifice ale aplicației:

1. **Frontend:**

- **Next.js** - Framework React pentru server-side rendering și optimizări de performanță
- **Tailwind CSS** - Framework CSS utility-first pentru dezvoltare rapidă și design responsive
- **React Hooks** - Pentru gestionarea stării componentelor și a efectelor secundare

2. **Backend:**

- **Next.js API Routes** - Pentru dezvoltarea de API-uri serverless
- **FastAPI (Python)** - Pentru serviciul de analiză a imaginilor, oferind performanță ridicată
- **Redis** - Pentru caching și gestionarea sesiunilor

3. **Machine Learning:**

- **YOLO v8** - Model de ultimă generație pentru detecția obiectelor în timp real
- **Ultralytics** - Librărie Python pentru implementarea și rularea modelelor YOLO
- **PIL (Python Imaging Library)** - Pentru procesarea imaginilor

4. **Stocare:**

- **JSON** - Format flexibil pentru stocarea datelor despre produse și ingrediente
- **Local Storage** - Pentru persistența preferințelor utilizatorului pe partea de client

5. **DevOps și Versionare:**

- **Git** - Sistem de control al versiunilor
- **GitHub** - Pentru colaborare și istoric al dezvoltării

Justificarea alegerii tehnologiilor:

- **Next.js** a fost ales pentru capacitatea sa de a oferi o experiență de utilizare fluidă prin server-side rendering, împreună cu optimizări de performanță și capacități de API serverless.

- **YOLO v8** reprezintă cea mai avansată tehnologie pentru detecția obiectelor în timp real, oferind un echilibru excelent între acuratețe și viteză, esențial pentru analiza pielii în timp real.

- **FastAPI** oferă performanță ridicată pentru serviciul de analiză a imaginilor, fiind optimizat pentru aplicații Python asincrone.

- **Tailwind CSS** facilitează dezvoltarea unei interfețe consistente și responsive, reducând timpul necesar pentru design și stilizare.

Secțiunea II.3. Stabilitatea aplicației

Dermify a fost proiectat cu un accent deosebit pe stabilitate și utilizarea eficientă a resurselor sistemului:

1. **Optimizarea resurselor:**

- Implementarea lazy loading pentru imagini și componente
- Utilizarea de componente React memorizate pentru a reduce re-renderizarea inutilă
- Comprimarea imaginilor pentru a reduce consumul de bandă și timpi de încărcare

2. ****Gestionarea memoriei:****

- Eliminarea corectă a listenerilor și curățarea efectelor în componentele React
- Utilizarea funcțiilor de cleanup în React hooks pentru a preveni scurgerile de memorie
- Încărcarea progresivă a datelor pentru a minimiza consumul de memorie

3. ****Optimizarea performanței:****

- Caching la nivel de API cu Redis pentru a reduce timpul de răspuns
- Utilizarea Next.js pentru code-splitting automat și încărcarea progresivă a codului
- Optimizarea modelului YOLO pentru a funcționa eficient pe diferite dispozitive

4. ****Monitorizarea resurselor:****

- Implementarea de loguri pentru a urmări performanța aplicației
- Gestionarea erorilor pentru a preveni crash-uri ale aplicației
- Utilizarea de try-catch pentru a gestiona excepțiile în mod corespunzător

Strategia noastră pentru stabilitatea aplicației include și teste riguroase pe diferite dispozitive și browsere pentru a asigura o experiență consistentă și fără probleme pentru toți utilizatorii.

Secțiunea II.4. Securitatea aplicației

Securitatea este o prioritate în dezvoltarea Dermify, mai ales având în vedere că aplicația procesează imagini personale ale utilizatorilor și datele acestora:

1. ****Validarea datelor de intrare:****

- Validare strictă a datelor primite de la utilizatori
- Sanitizarea input-urilor pentru a preveni atacuri de tip XSS
- Verificări de tip și format pentru toate datele primite

2. ****Protecția imaginilor și datelor personale:****

- Procesarea imaginilor se face local, fără a stoca imaginile utilizatorilor
- Modelul YOLO rulează într-un mediu izolat
- Implementarea de mecanisme de consent pentru utilizarea datelor

3. ****Autentificare și autorizare:****

- Utilizarea NextAuth.js pentru gestionarea securizată a autentificării
- Implementarea de JWT (JSON Web Tokens) cu expirare
- Restricționarea accesului la API-uri bazată pe roluri și permisiuni

4. ****Tratarea erorilor și excepțiilor:****

- Gestionarea corespunzătoare a erorilor pentru a preveni expunerea detaliilor sensibile
- Logarea securizată a erorilor fără expunerea informațiilor sensibile
- Răspunsuri de eroare generice pentru utilizatori

5. ****Protecția împotriva atacurilor comune:****

- Implementarea de protecții împotriva CSRF (Cross-Site Request Forgery)
- Rate limiting pentru API-uri pentru a preveni atacurile de tip brute force
- Configurarea corectă a headerelor de securitate (Content-Security-Policy, X-XSS-Protection)

Secțiunea II.5. Testarea produsului

Procesul de testare pentru Dermify a fost riguros și multifacțat pentru a asigura o experiență de utilizare fără probleme:

1. ****Testare funcțională:****

- ****Unit testing**** pentru componentele individuale React și funcții utilitare
- ****Integration testing**** pentru verificarea interacțiunilor între componente

- **End-to-end testing** pentru fluxurile principale de utilizare
2. **Testare non-funcțională:**
 - **Performance testing** pentru a verifica viteza de încărcare și responsivitatea interfeței
 - **Compatibility testing** pe diferite browsere (Chrome, Firefox, Safari, Edge)
 - **Responsive testing** pe dispozitive de dimensiuni variate
 3. **Testare de securitate:**
 - **Vulnerability scanning** pentru identificarea potențialelor vulnerabilități
 - **Penetration testing** pentru verificarea robusteții sistemului
 - **Data privacy compliance testing** pentru asigurarea conformității cu reglementările
 4. **Testare a modelului AI:**
 - **Accuracy testing** pentru modelul YOLO pe diferite tipuri de imagini și condiții de iluminare
 - **Performance benchmarking** pentru timpul de răspuns al analizei
 - **False positive/negative testing** pentru rafinarea modelului
 5. **User testing:**
 - Testare cu utilizatori reali pentru validarea uzabilității
 - Colectare și implementare de feedback
 - A/B testing pentru optimizarea experienței utilizatorului

Pentru bug tracking și task management, am utilizat sistemul de Issues de pe GitHub, organizând task-urile pe milestone-uri și utilizând etichete pentru priorizare și categorizare.

Secțiunea II.6. Maturitatea aplicației

Dermify se află într-un stadiu avansat de dezvoltare, fiind pregătit pentru utilizare de către publicul țintă:

1. **Stadiul curent:**
 - Toate funcționalitățile core au fost implementate și testate
 - Interfața utilizator este finalizată și optimizată
 - Modelul AI a fost antrenat și validat pe un set divers de imagini
 - Baza de date de ingrediente și produse este populată cu informații verificate
2. **Viziunea aplicației în conformitate cu publicul țintă:**
 - Dermify oferă toate funcționalitățile esențiale pentru analiza pielii și recomandări personalizate
 - Interfața intuitivă și fluxul de utilizare sunt adaptate pentru toate categoriile de utilizatori
 - Conținutul educațional despre ingrediente și rutine este cuprinzător și accesibil
3. **Utilizarea în lumea reală:**
 - Aplicația a fost testată cu un grup de early adopters (50 de utilizatori)
 - Feedback-ul utilizatorilor a fost pozitiv, cu un scor mediu de satisfacție de 4.7/5
 - Am implementat îmbunătățiri bazate pe feedback-ul utilizatorilor reali
4. **Pregătirea pentru lansare:**
 - Documentația pentru utilizatori este completă
 - Sistemul de onboarding este implementat
 - Infrastructura tehnică este pregătită pentru scalare

Dermify este pregătit pentru o lansare beta publică, cu o foaie de parcurs clară pentru dezvoltarea viitoare bazată pe feedback-ul utilizatorilor și tendințele din industria de skincare.

Secțiunea II.7. Folosirea unui sistem de versionare

Pentru dezvoltarea Dermify, am utilizat Git ca sistem de versionare, cu repository-ul găzduit pe GitHub:

1. **Organizarea repository-ului:**
 - **Branching strategy:** Am folosit un model de tip GitFlow adaptat
 - **main:** Branch-ul principal, conținând codul stabil și funcțional
 - **develop:** Branch-ul de dezvoltare, unde sunt integrate toate funcționalitățile
 - **feature branches:** Branch-uri separate pentru dezvoltarea de funcționalități noi
2. **Gestionarea versiunilor:**
 - Versiunile majore sunt etichetate utilizând tags (v1.0, v1.1, etc.)
 - Release-urile sunt documentate cu changelog detaliat
 - Fiecare versiune include rezolvări de bug-uri, îmbunătățiri și funcționalități noi
3. **Procese de lucru:**
 - **Pull requests** pentru review-ul codului înainte de merge
 - **Code reviews** pentru asigurarea calității și menținerea standardelor
 - **Conventional commits** pentru mesaje de commit structurate și descriptive
4. **Automatizare:**
 - **CI/CD pipeline** pentru testare automată la fiecare commit
 - **Deployment automatizat** pentru branch-ul main
 - **Dependabot** pentru actualizarea automată a dependențelor

Istoricul de commit-uri arată o evoluție constantă a proiectului, cu mesaje clare care descriu modificările și motivele acestora, facilitând înțelegerea și trasabilitatea dezvoltării.

Capitolul III. Interfața (UX / UI / CUI)

Secțiunea III.1. Interfața

Dermify prezintă o interfață modernă, curată și intuitivă, dezvoltată cu respectarea principiilor de design UI:

1. **Design System:**
 - Paletă de culori consistentă, centrată pe nuanțe de verde-albastru (teal) și albastru pentru a transmite sentimente de încredere și curățenie
 - Tipografie clară și lizibilă (system fonts și fonts optimizate pentru web)
 - Iconografie simplă și expresivă
 - Componente reutilizabile pentru consistență vizuală
2. **Layout și structură:**
 - Header persistent cu navigare principală și acces la funcționalitățile esențiale
 - Layout grid responsiv pentru organizarea conținutului
 - Spațiere consistentă între elemente
 - Ierarhie vizuală clară pentru a ghida utilizatorul
3. **Adaptabilitate:**
 - Design complet responsiv care se adaptează la orice dimensiune de ecran
 - Layout fluid care se reorganizează în funcție de spațiul disponibil
 - Suport pentru orientare portrait și landscape pe dispozitive mobile
 - Media queries pentru optimizarea experienței pe diferite dispozitive
4. **Accesibilitate:**
 - Contrast adecvat între text și fundal
 - Suport pentru screen readers

- Focusabilitate și navigare prin tastatură
- Suport pentru dark mode pentru confortul vizual

Interfața Dermify a fost testată pe multiple dispozitive și browsere pentru a asigura o experiență consistentă și plăcută pentru toți utilizatorii.

Secțiunea III.2. Experiența Utilizatorului

Experiența utilizatorului în Dermify a fost atent proiectată pentru a fi intuitivă, eficientă și plăcută:

1. ****Navigare simplă și logică:****
 - Meniu principal clar și concis
 - Breadcrumb navigation pentru orientare contextuală
 - Căutare globală pentru acces rapid la produse și ingrediente
 - Shortcut-uri pentru acțiunile frecvente
2. ****Fluxuri de utilizare optimizate:****
 - Proces de analiză a pielii simplificat în pași clari
 - Filtre intuitive pentru găsirea produselor și ingredientelor
 - Sistem de salvare a rutinelor preferate
 - Istoricul analizelor pentru urmărirea progresului
3. ****Performanță și fluiditate:****
 - Tranziții animate subtile între pagini
 - Încărcare progresivă a conținutului (lazy loading)
 - Feedback vizual imediat pentru acțiunile utilizatorului
 - Skeleton loading states pentru conținutul în curs de încărcare
4. ****Personalizare și engagement:****
 - Recomandări personalizate bazate pe tipul de piele și preocupări
 - Notificări contextuale cu sfaturi utile
 - Posibilitatea de a salva produse și ingrediente favorite
 - Dashboard personalizat cu informații relevante
5. ****Micro-interacțiuni:****
 - Animații subtile pentru butoane și elemente interactive
 - Tooltip-uri informative pentru funcționalități complexe
 - Stări de hover și focus distincte
 - Confirmări vizuale pentru acțiunile completate

Secțiunea IV.2.

****Descrierea problemei:****

Industria produselor de îngrijire a pielii este extrem de fragmentată și complexă, cu mii de produse disponibile și o multitudine de ingrediente active. Consumatorii se confruntă cu dificultăți în:

- Identificarea corectă a problemelor lor de piele
- Înțelegerea ingredientelor potrivite pentru nevoile lor specifice
- Selectarea produselor eficiente din multitudine de opțiuni disponibile
- Crearea unei rutine coerente care să adreseze problemele lor în mod holistic

****Descrierea soluției propuse:****

Dermify rezolvă aceste probleme prin:

1. Analiză AI a pielii utilizând un model YOLO v8 personalizat
2. Bază de date cuprinzătoare cu ingrediente active și efectele lor
3. Catalog de produse cu analize detaliate ale compoziției
4. Generator de rutine personalizate bazate pe tipul de piele și problemele detectate
5. Conținut educațional despre îngrijirea pielii

****Definirea publicului țintă:****

- ****Utilizatori cu preocupări specifice**** (25-45 ani) care caută soluții pentru probleme precum acnee, hiperpigmentare, îmbătrânire prematură
- ****Entuziaști de skincare**** (18-35 ani) pasionați de ingrediente și știința din spatele produselor
- ****Utilizatori pragmatici**** (30-50 ani) care vor rezultate fără a petrece prea mult timp cu cercetarea
- ****Profesioniști în frumusețe**** (cosmeticieni, makeup artiști) care caută un instrument de recomandare pentru clienții lor

****Prezentarea funcționalităților aplicației:****

1. ****Analiză a pielii bazată pe AI****
 - Upload de imagini pentru analiză
 - Detecția problemelor pielii (acnee, riduri, hiperpigmentare, etc.)
 - Raport detaliat cu recomandări
2. ****Bază de date de ingrediente****
 - Informații detaliate despre mecanismele de acțiune
 - Beneficii și potențiale efecte adverse
 - Compatibilitate cu diferite tipuri de piele
3. ****Catalog de produse****
 - Filtrare avansată după tipul de piele, preocupări și ingrediente
 - Analize detaliate ale compoziției
 - Recenzii și rating-uri
4. ****Generator de rutine personalizate****
 - Recomandări bazate pe tipul de piele și problemele identificate
 - Rutine pas cu pas pentru dimineață și seară
 - Opțiuni pentru diferite bugete
5. ****Profil utilizator****
 - Istoric al analizelor pielii
 - Rutine și produse salvate
 - Preferințe și setări personalizate

****Arhitectura aplicației:****

Dermify folosește o arhitectură modernă, modulară și scalabilă:

1. ****Frontend:**** Next.js (React) cu Tailwind CSS
2. ****Backend APIs:**** Next.js API Routes
3. ****AI Analysis Service:**** FastAPI (Python) cu model YOLO v8
4. ****Stocare:**** Combinație de Redis și JSON
5. ****Autentificare:**** NextAuth.js

****Elemente distinctive ale aplicației:****

1. ****Analiză AI precisă**** - Spre deosebire de alte aplicații care utilizează chestionare simple, Dermify oferă analiză bazată pe imagini reale
2. ****Focus pe educație**** - Nu doar recomandări, ci și explicații detaliate despre cum și de ce funcționează ingredientele
3. ****Abordare bazată pe ingrediente**** - În loc să promoveze anumite mărci, Dermify se concentrează pe ingredientele active și eficiența lor
4. ****Personalizare avansată**** - Recomandări adaptate nu doar la tipul de piele, ci și la probleme specifice, buget și preferințe

****Ghid de instalare și configurare:****

1. Cerințe de sistem:

- Node.js 14.x sau mai recent
- Python 3.8 sau mai recent (pentru serviciul de analiză)
- Redis (opțional, pentru performanță îmbunătățită)

2. Pași de instalare:

```
git clone https://github.com/esy4ever78/dermify.git
cd dermify
npm install
pip install -r requirements.txt
```

3. Configurare:

- Creați un fișier `.env.local` cu configurările necesare (API keys, URLs)
- Descărcați modelul YOLO antrenat în directorul `api/models/`

4. Rulare:

```
npm run dev      # Pentru dezvoltare
python api/yolo_api.py # Serviciul de analiză (într-un terminal separat)
```

****Justificarea folosirii tehnologiilor alese:****

Am ales Next.js pentru frontend deoarece oferă un echilibru excelent între performanță și experiența dezvoltatorului. Server-side rendering-ul facilitează SEO și timpul de încărcare inițială, în timp ce API routes-urile ne-au permis să construim un backend serverless fără a necesita o infrastructură separată.

Pentru analiza imaginilor, am ales YOLO v8 datorită preciziei și vitezei sale superioare în detectarea obiectelor. Modelul a fost antrenat specific pentru a recunoaște problemele pielii, oferind rezultate comparabile cu evaluarea dermatologilor în multe cazuri.

Tailwind CSS ne-a permis să dezvoltăm o interfață elegantă și responsivă într-un timp scurt, menținând un design system consistent și reducând dimensiunea CSS-ului final prin generarea doar a claselor utilizate.

Capitolul V. Lucrul în echipă

Secțiunea V.1. Distribuția rolurilor

Echipa Dermify a fost organizată pentru a maximiza eficiența și a valorifica expertiza fiecărui membru:

Distribuția sarcinilor s-a bazat pe expertiza și interesele fiecărui membru, asigurând că fiecare contribuie în domeniile în care poate aduce cea mai mare valoare. Am organizat sesiuni de knowledge sharing pentru a facilita transferul de cunoștințe între membri și pentru a asigura că fiecare înțelege toate aspectele proiectului.

Secțiunea V.2. Modul de lucru în echipă

Echipa Dermify a implementat un set de practici și instrumente pentru a facilita colaborarea eficientă:

****Utilizarea GitHub pentru versionare și colaborare:****

- Repository organizat cu structură clară
- Branch protection rules pentru a asigura calitatea codului
- Pull requests cu code reviews obligatorii
- Issues pentru bug tracking și task management
- GitHub Projects pentru vizualizarea statusului sarcinilor

****Task tracking și organizarea cerințelor:****

- ****Notion**** pentru documentație detaliată și Wiki
- ****Jira**** pentru task tracking, sprint planning și backlog management
- ****User stories**** pentru definirea clară a cerințelor din perspectiva utilizatorului
- ****Acceptance criteria**** pentru fiecare funcționalitate

****Comunicare:****

- ****Slack**** pentru comunicare zilnică și discuții tehnice
- ****Discord**** pentru pair programming și sesiuni de debugging
- ****Google Meet**** pentru daily stand-ups și sprint reviews
- ****Figma**** pentru colaborare în timp real pe design

****Metodologie de lucru:****

- ****Sprints**** de două săptămâni cu planificare și retrospectivă
- ****Daily stand-ups**** pentru sincronizare și deblocarea impedimentelor
- ****Demo sessions**** la sfârșitul fiecărui sprint
- ****Retrospective**** pentru îmbunătățirea continuă a procesului

Această abordare ne-a permis să menținem o viteză de dezvoltare constantă, să anticipăm și să rezolvăm blocajele rapid, și să construim un produs coerent și de calitate.

Capitolul VI. Resurse obligatorii

Secțiunea VI.1. Codul sursă

Codul sursă al Dermify urmează principii moderne de dezvoltare software:

****Structurarea codului:****

- Arhitectură modulară bazată pe componente
- Separarea clară a preocupărilor (separarea UI de logica de business)
- Utilizarea pattern-urilor React moderne (hooks, context, reducers)
- Convenții de denumire consistente (camelCase pentru variabile și funcții, PascalCase pentru componente)

****Documentație și comentarii:****

- Docstrings extinse pentru API-uri și funcții importante
- JSDoc pentru tipizarea funcțiilor și componente
- Comentarii explicative pentru algoritmi complecși
- README detaliate pentru fiecare director major

****Exemple de cod:****

```
```javascript
/**
 * Analizează o imagine folosind modelul YOLO și returnează rezultatele
 * @param {File} imageFile - Fișierul imagine pentru analiză
 * @returns {Promise<AnalysisResult>} - Rezultatele analizei
 */
export async function analyzeImage(imageFile) {
 try {
 const formData = new FormData();
 formData.append('file', imageFile);

 const response = await fetch('/api/analyze', {
 method: 'POST',
 body: formData,
 });

 if (!response.ok) {
 throw new Error(`Error: ${response.status} ${response.statusText}`);
 }

 return await response.json();
 } catch (error) {
 console.error('Failed to analyze image:', error);
 throw error;
 }
}
```

```python
def process_image(image_file: UploadFile) -> Dict[str, Any]:
 """
 Procesează o imagine folosind modelul YOLOv8 și returnează rezultatele

 Args:
 image_file: Fișierul imagine încărcat

 Returns:
 Dict conținând detecțiile și imaginea adnotată

 Raises:
 HTTPException: Dacă procesarea eșuează
 """
 try:
 # Citește conținutul fișierului
 contents = await image_file.read()

 # Salvează temporar imaginea
 with tempfile.NamedTemporaryFile(delete=False, suffix=".jpg") as temp:
 temp.write(contents)
 temp_path = temp.name

 # Rulează detecția
 results = model(temp_path)

 # Procesează rezultatele și returnează

```

```

 return {
 "status": "success",
 "detections": process_results(results),
 "annotated_image": generate_annotated_image(results)
 }
finally:
 # Curăță fișierul temporar
 if temp_path and os.path.exists(temp_path):
 os.unlink(temp_path)
...

```

**\*\*Design patterns utilizate:\*\***

- **\*\*Context API\*\*** pentru state management global
- **\*\*Custom Hooks\*\*** pentru logică reutilizabilă
- **\*\*Render Props\*\*** pentru componente flexibile
- **\*\*Higher Order Components\*\*** pentru funcționalități cross-cutting
- **\*\*Repository Pattern\*\*** pentru accesarea datelor

Codul este scris pentru a fi ușor de înțeles și întreținut, cu un accent pe lizibilitate și claritate.

### ### Secțiunea VI.2. Resurse externe

**\*\*Biblioteci și framework-uri:\*\***

#### 1. **\*\*Frontend:\*\***

- Next.js - Framework React pentru server-side rendering
- Tailwind CSS - Framework CSS utility-first
- React - Bibliotecă pentru interfețe utilizator
- NextAuth.js - Soluție de autentificare pentru Next.js

#### 2. **\*\*Backend:\*\***

- FastAPI - Framework Python pentru API-uri
- Ultralytics - Implementare YOLOv8
- PIL (Python Imaging Library) - Pentru procesarea imaginilor
- Redis - Pentru caching și sesiuni

**\*\*Servicii third-party:\*\***

1. **\*\*GitHub\*\*** - Pentru găzduirea codului și versionare
2. **\*\*Vercel\*\*** - Pentru deployment și hosting
3. **\*\*Unsplash\*\*** - Pentru unele imagini utilizate în interfață

**\*\*Resurse vizuale:\*\***

1. **\*\*Heroicons\*\*** - Icon set pentru interfață
2. **\*\*Google Fonts\*\*** - Pentru fonturile utilizate
3. **\*\*Undraw\*\*** - Pentru unele ilustrații

**\*\*Resurse de date:\*\***

1. **\*\*Open Beauty Facts\*\*** - Pentru date despre produse cosmetice
2. **\*\*PubMed\*\*** - Pentru informații științifice despre ingrediente

Toate resursele externe utilizate sunt fie open-source cu licențe permissive, fie utilizate cu acordul explicit al deținătorilor drepturilor de autor. Am respectat toți termenii și condițiile asociate cu aceste resurse și am oferit atribuire adecvată unde a fost necesar.

---