

MEMÒRIA



Dídac Garcia Dumé
DAW2

ÍNDEX

1. Estudi de viabilitat.

- 1.1. Establiment de l'abast del sistema.
- 1.2. Estudi de la situació actual.
- 1.3. Definició dels requisits del sistema.
- 1.4. Estudi de les alternatives de solució.
- 1.5. Valoració de les alternatives.
- 1.6. Selecció de la solució.

2. Anàlisi del sistema.

- 2.1. Definició del sistema.
- 2.2. Establiment de requisits.
- 2.3. Definició d'interfícies d'usuari.
- 2.4. Especificació del pla de proves.

3. Disseny del sistema

- 3.0. Arquitectura d'Informació, guia d'estils, usabilitat, accessibilitat i disseny de prototipus.
- 3.1. Arquitectura Web.
 - 3.1.1. Definició de nivells d'arquitectura: FrontEnd i Backend.
 - 3.1.2. Especificació d'estàndards, normes de disseny i construcció.
 - 3.1.3. Identificació de subsistemes.
- 3.2. Revisió de casos d'ús.
 - 3.2.1. Revisió dels subsistemes segons els casos d'ús.
 - 3.2.2. Elecció d'alternatives de components i llicències més adequades.
 - 3.2.3. Especificacions de desenvolupament i proves.
 - 3.2.4. Requisits d'implantació.
- 3.3. Anàlisi paradigma estructurat / Orientat a Objectes
- 3.4. Disseny paradigma estructurat / Orientat a Objectes
- 3.5. Persistència de dades: Anàlisi i disseny de bases de dades / ORM

4. Desenvolupament.

- 4.1. Planificació de les activitats de desenvolupament i integració de sistema.
- 4.2. Desenvolupament.
- 4.3. Documentació tècnica del programari.

5. Implantació.

- 5.1. Formació.
- 5.2. Implantació del sistema i proves.
- 5.3. Nivell de serveis.
- 5.4. Acceptació del sistema.

6. Manteniment i versions futures

1. Estudi de viabilitat

1.1. Establiment de l'abast del sistema

El projecte **Match Clothes App** s'ha desenvolupat amb la intenció de transformar la manera com les persones combinen i interactuen amb el seu vestuari. L'aplicació ofereix una experiència digital avançada i intel·ligent per gestionar peces de roba, generar combinacions i provar-les de manera virtual. Està dissenyada per a usuaris que volen guanyar temps, optimitzar el seu estil i experimentar amb noves opcions visuals a través de tecnologies emergents com la Intel·ligència Artificial i la visió per computador.

1.2. Estudi de la situació actual

Tradicionalment, la gestió del vestuari es fa de manera manual, amb limitacions importants: no hi ha una visió global, és difícil recordar totes les peces disponibles i resulta complicat innovar a l'hora de combinar roba. No existeixen aplicacions àmpliament adoptades que permetin fer combinacions intel·ligents amb suport visual i assistit per IA. Aquí és on Match Clothes App marca la diferència.

1.3. Definició dels requisits del sistema

Match Clothes App cobreix tot el procés de combinació de roba de manera guiada i automatitzada. Els requisits que compleix són:

- **Pujar una imatge** d'una peça de roba que l'usuari vulgui combinar.
- **Extracció automàtica de colors** de la peça mitjançant IA.
- **Selecció de tipus de peça complementària** (per exemple, si has pujat una samarreta, tries si vols combinar-la amb pantalons, sabates, etc.).
- **Suggeriments de colors compatibles** amb la peça original, amb base en criteris d'harmonia cromàtica.
- **Visualització de peces de roba reals** (obtingudes d'internet via l'API de SerpApi) segons el color i tipus seleccionats.
- **Prova virtual de la peça escollida** pujant una foto del teu cos, gràcies a la integració amb l'API de Fashn AI.

Tot el sistema funciona mitjançant una interfície intuïtiva, multiplataforma i orientada a una experiència visual de qualitat.

1.4. Estudi de les alternatives de solució

Es van considerar tres enfocaments tecnològics:

- **Aplicació nativa:** rendiment alt i integració directa amb la càmera i funcions del dispositiu, però amb un cost elevat.
- **Aplicació híbrida:** reutilització de codi amb menor cost, però limitacions en experiència d'usuari i funcions avançades.
- **Aplicació web progressiva (PWA):** solució flexible, escalable i fàcilment desplegable, ideal per accedir des de qualsevol dispositiu amb navegador.

1.5. Valoració de les alternatives

Tenint en compte criteris de desenvolupament, mantenibilitat, accés a funcionalitats d'IA i escalabilitat, es va concloure que la millor opció per a una primera versió potent i eficient era la PWA.

- **Nativa:** + Rendiment i accés a hardware | - Cost i temps elevats
- **Híbrida:** + Cost baix | - Limitació en experiències avançades
- **PWA:** + Ràpida, flexible, escalable | + Fàcil integració amb serveis IA | - Dependència del navegador

1.6. Selecció de la solució

Finalment, s'ha optat pel desenvolupament d'una **aplicació web progressiva (PWA)** basada en **Next.js**, **TypeScript** i **Tailwind CSS**. Aquesta arquitectura permet una aplicació moderna, responsiva i fàcilment integrable amb serveis externs com **SerpApi** per la cerca de peces i **Fashn AI** per la prova virtual. El resultat és una eina potent, accessible i intel·ligent que transforma l'experiència de vestir-se en una activitat creativa i tecnològicament avançada.

2. Anàlisi del sistema

2.1. Definició del sistema

El sistema Match Clothes App està format per diferents components interconnectats que permeten una experiència fluida i intel·ligent:

- **Frontend (interfície d'usuari):** desenvolupat amb Next.js, ofereix una interfície moderna, responsiva i intuïtiva.
- **Backend:** integrat també amb Next.js (API routes), gestiona la lògica d'aplicació, la comunicació amb serveis externs.
- **Serveis externs:**
 - **SerpApi:** per obtenir peces reals des d'internet en funció del color i tipus seleccionats.
 - **Fashn AI:** per permetre la prova virtual de peces de roba sobre una imatge del cos de l'usuari.
 - **Mòduls de visió per computador:** per extreure automàticament els colors dominants de les peces pujades.
- **Sistema d'autenticació:** implementat amb tecnologies com NextAuth per garantir la seguretat de l'accés

Aquesta arquitectura modular assegura una separació clara de responsabilitats, facilita el manteniment i permet escalar el sistema de forma eficient.

2.2. Establiment de requisits

Requisits tècnics:

- Utilitzar **Next.js** i **TypeScript** per al desenvolupament frontend i backend.
- **Tailwind CSS** per a l'estilització i coherència visual.
- Integració amb APIs externes (SerpApi, Fashn AI).
- Gestor eficient d'imatges i extracció de colors amb IA.

Requisits de l'usuari:

- Interfície intuïtiva i atractiva.
- Temps de resposta òptim en cada acció.
- Possibilitat de pujar imatges i generar combinacions visuals de manera fàcil.
- Accés multiplataforma sense necessitat d'instal·lacions complexes.

2.3. Definició d'interfícies d'usuari

La interfície d'usuari de Match Clothes App està dissenyada per guiar l'usuari en tot el procés de combinació:

1. **Pujar una imatge** de la seva pròpia roba.
2. El sistema **extreu els colors** dominants de manera automàtica.
3. L'usuari **tria el tipus de peça** que vol combinar-hi.
4. Es mostren **colors suggerits** compatibles.
5. Es llisten peces reals d'internet segons el color i tipus seleccionat.
6. L'usuari pot **provar-se virtualment** la roba pujant una foto del seu cos.

Tot el flux és visual, guiat i responsiu, amb accés ràpid als filtres i a la galeria personal.

2.4. Especificació del pla de proves

Per assegurar la qualitat i estabilitat del sistema, s'ha definit un pla de proves detallat:

- **Proves unitàries:** per validar components individuals del frontend i backend.
- **Proves d'integració:** per garantir la comunicació fluida entre mòduls (per ex., extracció de colors + selecció + SerpApi).
- **Proves d'acceptació:** per comprovar que el flux complet compleix les expectatives de l'usuari final.
- **Proves de rendiment:** per mesurar la ràpida resposta del sistema en dispositius diversos.
- **Proves de seguretat:** centrades en la protecció de dades personals i accés autènticat.

3. Disseny del sistema

3.0. Arquitectura d'informació, guia d'estils, usabilitat, accessibilitat i disseny de prototipus

El disseny de Match Clothes App s'ha orientat a una estructura clara, funcional i visualment atractiva. La navegació està organitzada en pantalles que segueixen el flux real de l'aplicació: pujada de la imatge, selecció del tipus de peça complementària, suggeriment de colors compatibles, visualització de peces que combinen i prova virtual. La guia d'estils utilitza Tailwind CSS, garantint una aparença moderna, coherent i accessible. S'ha prioritzat la usabilitat amb controls grans, textos clars i una navegació intuïtiva. Pel que fa a l'accessibilitat, es mantenen contrastos adequats i suport per a teclat i lectors de pantalla. Els prototipus es van dissenyar amb Figma i es van validar mitjançant proves amb usuaris.

3.1. Arquitectura Web

L'arquitectura del sistema es basa en una aplicació web moderna amb backend integrat, que aprofita la infraestructura de Next.js per combinar servidor i client. Tot i que part de la informació es pot mantenir al navegador, el sistema incorpora crides a serveis externs com SerpApi i Fashn AI, fet que requereix certa lògica backend.

3.1.1. Definició de nivells d'arquitectura: Frontend i Backend

- **Frontend:** Desenvolupat amb React i Next.js, mostra la interfície d'usuari i gestiona les interaccions.
- **Backend (API Routes de Next.js):** Gestiona les peticions a serveis externs com SerpApi (per buscar peces) i Fashn AI (per fer la prova virtual).

3.1.2. Especificació d'estàndards, normes de disseny i construcció

- **Codi:** TypeScript per a robustesa, ESLint i Prettier per mantenir qualitat.
- **Estils:** Tailwind CSS per a coherència visual i mantenibilitat.
- **Components:** Basats en funcions i organitzats modularment.

3.1.3. Identificació de subsistemes

- **Gestor de pujada d'imatges** (extracció de colors amb algoritme).
- **Gestor de selecció i suggeriment de colors.**

- **Subsistema de suggeriment de peces** (SerpApi).
- **Subsistema de prova virtual** (Fashn AI).
- **Gestor de l'estat de l'usuari i la interfície.**

3.2. Revisió de casos d'ús

3.2.1. Revisió dels subsistemes segons els casos d'ús

Cada subsistema resol una part concreta del flux d'usuari:

- **Extracció de colors:** Automatitza la identificació dels tons de la peça.
- **Selector de tipus de peça i suggeriment de color:** Guia l'usuari en la creació de combinacions.
- **SerpApi:** Busca peces reals segons les preferències.
- **Fashn AI:** Permet provar les peces escollides.

3.2.2. Elecció d'alternatives de components i llicències més adequades

- **Tecnologies:** React, Next.js, Tailwind CSS, TypeScript.
- **APIs externes:** SerpApi (comercial), Fashn AI (comercial o amb pla gratuït).
- **Llicències:** Es prioritzen biblioteques de codi obert amb llicències permissives (MIT, Apache 2.0).

3.2.3. Especificacions de desenvolupament i proves

- **Desenvolupament:** Basat en components funcionals amb gestió de l'estat mitjançant hooks.
- **Proves:** Manualment en aquesta versió inicial, amb cobertura de totes les funcionalitats crítiques.

3.2.4. Requisits d'implantació

L'aplicació es pot desplegar fàcilment amb **Vercel**, ja que és compatible amb Next.js. Només cal definir les variables d'entorn per a les API externes.

3.3. Anàlisi paradigma estructurat / Orientat a Objectes

L'aplicació s'ha desenvolupat amb un enfocament **funcional**, propi de React. L'estructura orientada a objectes no ha estat necessària.

3.4. Disseny paradigma estructurat / Orientat a Objectes

El disseny es basa en **components funcionals i modulars**, amb una clara separació de responsabilitats. No s'ha fet servir herència ni polimorfisme.

4. Desenvolupament

4.1. Planificació de les activitats de desenvolupament i integració de sistema

El desenvolupament s'ha estructurat en sprints quinzenals, seguint una metodologia àgil per assegurar una evolució contínua del projecte. Cada sprint inclou la definició d'objectius concrets, el desenvolupament de funcionalitats específiques i la seva validació. S'han realitzat reunions periòdiques per fer seguiment del progrés, resoldre bloquejos i ajustar la planificació si cal. La integració contínua mitjançant eines com GitHub Actions ha permès verificar automàticament els canvis i assegurar l'estabilitat del sistema a cada iteració.

4.2. Desenvolupament

Durant el desenvolupament de Match Clothes App s'han implementat progressivament les funcionalitats clau:

- Extracció de colors d'imatges mitjançant visió per computador.
- Selecció de tipus de peça per combinar i suggeriment de colors compatibles.
- Cerca de peces de roba en línia a través de la SerpApi.
- Integració amb Fashn AI per a la prova virtual.
- Disseny d'una interfície intuïtiva i responsive amb Tailwind CSS.

Cada mòdul s'ha desenvolupat de manera modular, seguint bones pràctiques i revisant el codi de forma iterativa. Les proves manuals han assegurat la qualitat de cada funcionalitat abans de la seva integració.

4.3. Documentació tècnica del programari

La documentació tècnica inclou:

- **Guia d'instal·lació i desplegament** de l'aplicació amb Next.js.
- **Descripció dels components** de React i la seva estructura funcional.
- **Explicació de la integració** amb APIs externes (SerpApi i Fashn AI).
- **Diagrames d'arquitectura** i flux de dades.
- **Manual d'usuari** amb exemples d'ús i captures de pantalla.

Aquesta documentació facilita tant el manteniment com la possible ampliació futura del projecte, assegurant la seva continuïtat i escalabilitat.

6. Manteniment i versions futures

El manteniment del sistema Match Clothes App es planteja com una tasca contínua que inclou:

- **Actualitzacions de seguretat:** revisió periòdica de les dependències i aplicació de pegats per evitar vulnerabilitats.
- **Millores de rendiment:** optimització del codi, reducció del temps de càrrega i millora de la fluïdesa de la navegació.
- **Correcció d'errors:** identificació i solució d'incidències detectades durant l'ús de l'aplicació per part dels usuaris.
- **Suport tècnic:** atenció a consultes i incidències a través d'un sistema de tickets o formulari de contacte integrat.

Versions futures:

De cara al futur, s'han definit diverses línies d'evolució per ampliar la funcionalitat i l'abast de l'aplicació:

- **Integració amb xarxes socials:** permetre compartir combinacions i rebre feedback, així com importar peces vistes a Instagram o Pinterest.
- **Sistema de recomanacions intel·ligent:** implementar algoritmes de machine learning per suggerir combinacions personalitzades basades en l'historial i les preferències de l'usuari.
- **Aplicació mòbil nativa:** desenvolupar versions específiques per a iOS i Android que aprofitin al màxim les capacitats dels dispositius mòbils.
- **Funcionalitats de comunitat:** afegir un espai on els usuaris puguin compartir els seus looks, seguir altres perfils i valorar combinacions.

Aquestes millores s'alineen amb l'objectiu de fer créixer la base d'usuaris, augmentar el compromís i consolidar Match Clothes App com una referència en el sector de la moda digital intel·ligent.