

IoTEAM-Smart-Wardrobe

Document de analiză a cerințelor clientului

Scopul aplicației:

Aplicația permite utilizatorilor gestionarea eficientă a conținutului garderobei și accesul la sugestii personalizate în funcție de vreme și feature-uri.

Obiectivele aplicației:

1. Gestionarea articolelor vestimentare aflate în dressing
2. Recomandarea de outfit-uri în conformitate cu starea meteo, preferințe și particularitățile fizice ale utilizatorilor
3. Acordarea unui rating ținutei recomandate și generarea unei liste de favorite
4. Actualizarea datelor despre garderobă în funcție de update-ul caracteristicilor utilizatorilor
5. Sortarea articolelor în funcție de frecvența purtării

Grupul țintă:

Aplicația este destinată oamenilor care suferă de lipsă de inspirație în alegerea combinațiilor vestimentare, celor care își doresc ajutor în gestionarea garderobei și în ordonarea acesteia.

User Stories:

1. As a person who is interested in fashion, I want to use the recommendation feature in order to create the best outfit for myself.
2. As a person in constant hustle, I want to have a quick solution for putting together an outfit appropriate for the weather.
3. As a person who is in charge with the management of chores, I want to know what needs to be cleaned.
4. As a person who owns many clothes, I want to be able to know which ones I have not worn in a while so that I can rethink their purpose.
5. As a person with a limited wardrobe, I want to receive recommendations for clothes from Zara I could buy that could complement mine.

Colectarea cerințelor:

1. Realizarea unui algoritm de generare a combinațiilor de culori (inițial: Conectarea la API-ul pentru generarea paletelor de culoare pentru asortarea hainelor)
2. Selectarea hainelor după categorie (casual, professional, elegant etc.)
3. Existența unei camere video care să facă scanarea utilizatorului (simulare oglindă)
4. Existența unei camere video care să facă scanarea dulapului
5. Request la caracteristicile utilizatorului (înălțime, culoarea ochilor, greutate etc.)
6. Request în baza de date a articolelor de îmbrăcăminte/încălțăminte
7. Conectare la API-ul de vreme pentru aflarea condițiilor meteorologice
8. Reținerea unui istoric al purtărilor de la ultima spălare până în prezent
9. Sortarea pe cele 3 categorii de spălat (alb, negru, colorat)
10. Request pentru instrucțiuni de spălat (inițial: Conectarea la API-ul pentru instrucțiunile de spălat)
11. Reținerea datei pentru ultima purtare a unui articol
12. Request către magazinul de haine (inițial: Conectare la API-ul magazinului ZARA)
13. Reținerea istoricului outfit-urilor pe ultimele 7 zile

14. *Verificarea articolelor de defecte (găuri, decolorări etc)
15. *Conectare la API-ul de Spotify și să pună muzică în concordanță cu specificul outfit-ului ales
16. *Existența unui senzor de deschidere/închidere a ușii
17. *Recomandarea unor exemple de ținute populare care pot fi adaptate la garderoba proprie
18. *Conectare la API-ul de Instagram pentru a compara ținutele apreciate pe aplicație cu ceea ce deține utilizatorul în dulap și pentru postări din categoria „Outfit of The Day”
19. *Detectarea automată a stării articolului de îmbrăcăminte (dacă chiar ai spălat-o sau nu)

Interpretarea și prioritizarea cerințelor

Stocarea Datelor: 5,6,8,11,13

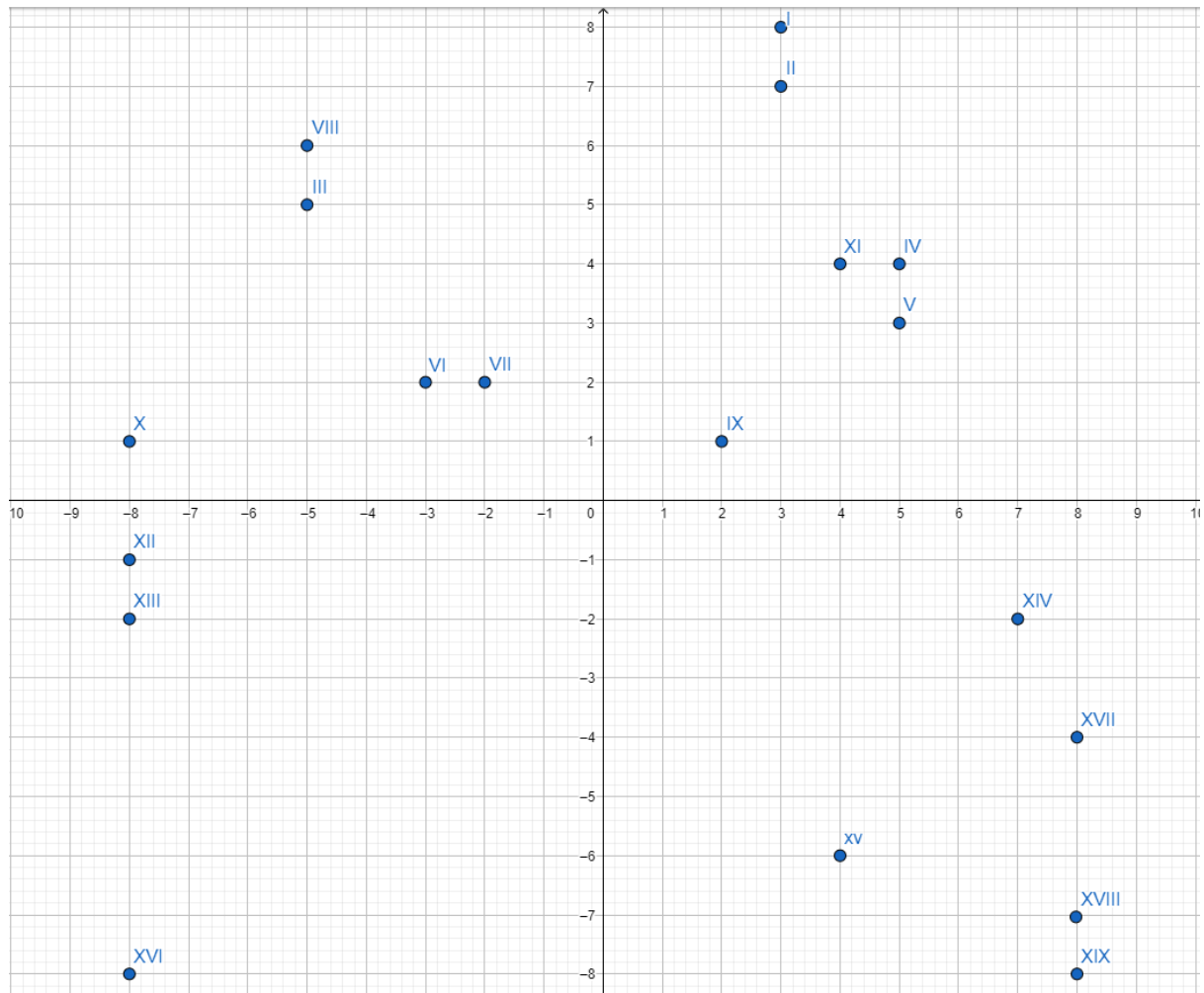
Procesare: 2,9,14,17,19

Comunicare: 1,7,10,12,15,18

Sistem: 3,4,16

Ordine de prioritate inițială

- I. Request la caracteristicile utilizatorului (înălțime, culoarea ochilor, greutate etc.)
- II. Request în baza de date a articolelor de îmbrăcăminte/încălțăminte
- III. Reținerea datei pentru ultima purtare a unui articol
- IV. Reținerea unui istoric al purtărilor de la ultima spălare până în prezent
- V. Reținerea istoricului outfit-urilor pe ultimele 7 zile
- VI. Selectarea hainelor după categorie (casual, profesional, elegant etc.)
- VII. Sortarea pe cele 3 categorii de spălat (alb, negru, colorat)
- VIII. Conectare la API-ul de vreme pentru aflarea condițiilor meteorologice
- IX. Realizarea unui algoritm de generare a combinațiilor de culori (inițial: Conectarea la API-ul pentru generarea paletelor de culoare pentru asortarea hainelor)
- X. Request pentru instrucțiuni de spălat (inițial: Conectarea la API-ul pentru instrucțiunile de spălat)
- XI. Request către magazinul de haine (inițial: Conectare la API-ul magazinului ZARA)
- XII. Existența unei camere video care să facă scanarea utilizatorului (simulare oglindă)
- XIII. Existența unei camere video care să facă scanarea dulapului
- XIV. *Verificarea articolelor de defecte (găuri, decolorări etc)
- XV. *Conectare la API-ul de Spotify și să pună muzică în concordanță cu specificul outfit-ului ales
- XVI. *Existența unui senzor de deschidere/închidere a ușii
- XVII. *Recomandarea unor exemple de ținute populare care pot fi adaptate la garderoba proprie
- XVIII. *Conectare la API-ul de Instagram pentru a compara ținutele apreciate pe aplicație cu ceea ce deține utilizatorul în dulap și pentru postări din categoria „Outfit of The Day”
- XIX. *Detectarea automată a stării articolului de îmbrăcăminte (dacă chiar ai spălat-o sau nu)



DREAPTA SUS – DIFICIL-VALOROS
 STANGA SUS – USOR-VALOROS
 STANGA JOS – USOR-NEVALOROS
 DREAPTA JOS – DIFICIL-NEVALOROS

Reestimare:

1. Realizarea unui algoritm de generare a combinațiilor de culori (inițial: Conectarea la API-ul pentru generarea paletelor de culoare pentru asortarea hainelor)
2. Selectarea hainelor după categorie (casual, professional, elegant etc.)
3. Existența unei camere video care să facă scanarea utilizatorului (simulare oglindă)
4. Existența unei camere video care să facă scanarea dulapului
5. Request la caracteristicile utilizatorului (înălțime, culoarea ochilor, greutate etc.)
6. Request în baza de date a articolelor de îmbrăcăminte/încălțăminte
7. Conectare la API-ul de vreme pentru aflarea condițiilor meteorologice
8. Reținerea unui istoric al purtărilor de la ultima spălare până în prezent
9. Sortarea pe cele 3 categorii de spălat (alb, negru, colorat)

10. Request pentru instrucțiuni de spălat (inițial: Conectarea la API-ul pentru instrucțiunile de spălat)
11. Reținerea datei pentru ultima purtare a unui articol
12. Request către magazinul de haine (inițial: Conectare la API-ul magazinului ZARA)
13. Reținerea istoricului outfit-urilor pe ultimele 7 zile
14. *Verificarea articolelor de defecte (găuri, decolorări etc)
15. *Conectare la API-ul de Spotify și să pună muzică în concordanță cu specificul outfit-ului ales
16. *Existența unui senzor de deschidere/închidere a ușii
17. *Recomandarea unor exemple de ținute populare care pot fi adaptate la garderoba proprie
18. *Conectare la API-ul de Instagram pentru a compara ținutele apreciate pe aplicație cu ceea ce deține utilizatorul în dulap și pentru postări din categoria „Outfit of The Day”
19. *Detectarea automată a stării articolului de îmbrăcăminte (dacă chiar ai spălat-o sau nu)

REUȘIT

MODIFICAT

NEIMPLEMENTAT