

Utilify

creat de Soiu Cristian-Ionuț - clasa a 10-a, Colegiul Național "Gh. M. Murgoci" Brăila

prof. îndrumător Marin Florina

Problema

În cadrul evenimentelor sau a emisiunilor în care este necesară alegerea spontană de melodii, se întâmplă uneori ca persoana care se ocupă de acest lucru, cum ar fi un DJ, un realizator de radio sau un inginer de sunet, să nu aibă la îndemână o listă de melodii din care să aleagă sau să aibă o listă mult prea mare, din care **să nu poată alege foarte rapid**. În aceste cazuri, este nevoie de o aplicație care să permită **alegerea rapidă** a unei melodii, care să fie ușor de utilizat și care să nu necesite mult timp pentru a fi învățată.

Un DJ are în medie în bibliotecă sa undeva la **500 de piese**, iar timpul în care trebuie să aleagă o melodie nouă pe care să o și mixeze este de numai **2 minute**.

Soluția

Utilify **automatizează** procesul de alegere a unei melodii, **reducând timpul** necesar pentru alegerea unei melodii noi, **înlocuind** lista de melodii cu o **interfață grafică** care permite **alegerea rapidă** a unei melodii.

Utilizatorul beneficiază de filtrare bazată pe **energie, dansabilitate și tempo**, căutare pe bază de **nume sau artist**, cât și de **recomandarea** de melodii bazată pe similaritate cu un **sistem inteligent**.

Pe lângă, poate **descărca** melodiile dorite direct din aplicație, obținând și informații despre piese. De asemenea, poate **salva** un playlist pentru a fi folosit mai târziu.

Publicul țintă

Publicul țintă este format din:

- **DJ**
- **realizatori de radio**
- **ingineri de sunet**

Aplicația poate fi folosită în cadrul emisiilor radio sau de televiziune, în cadrul evenimentelor, precum petreceri, nunți, botezuri, etc., sau în cadrul unui studio de înregistrări.

Opinia și utilitatea personală:

Ca DJ în timpul evenimentelor întâmpin mereu problema alesului următoarei melodii. Mă pierd mereu în aproape 500 de melodii din care am de ales una singură. Așa că m-am gândit cum pot să-mi ușurez munca într-un mod inteligent și am venit cu această soluție care, din testele pe care le-am făcut pe propria persoană, reușește să mă uimească. Prin utilizarea aplicației îmi acord mai mult timp mixării melodiilor, putând oferi o experiență îmbunătățită publicului meu.

Arhitectura aplicației

Funcționalitățile aplicației:

- **Căutare** - permite căutarea unei melodii după nume sau artist
- **Filtrare** - permite filtrarea melodiilor după energie, dansabilitate și tempo
- **Recomandare** - recomandă melodii similare cu o melodie dată
- **Descărcare** - permite descărcarea unei melodii
- **Salvare** - permite salvarea și încărcarea unui playlist
- **Informații** - oferă informații despre o melodie

Unelte folosite

- **Python** - limbajul de programare folosit
- **Flet** - framework-ul folosit pentru crearea interfeței grafice
- **Spotipy** - biblioteca folosită pentru a interacționa cu API-ul Spotify
- **SpotDL** - biblioteca folosită pentru a descărca melodii de pe Spotify
- **SciKit-Learn** - biblioteca folosită pentru a crea modelul de recomandare
- **Threading** - biblioteca folosită pentru a crea thread-uri pentru descărcarea melodiilor
- **FastAPI** - framework-ul folosit pentru a crea API-ul
- **Supabase** - baza de date folosită pentru a stoca melodiile

Justificarea folosirii Python

Python este un limbaj de programare foarte popular, care oferă o multitudine de biblioteci și framework-uri pentru a crea aplicații. Este un limbaj de programare foarte ușor de învățat, care permite crearea de aplicații complexe într-un timp foarte scurt. De asemenea, este un limbaj de programare foarte versatil, care poate fi folosit pentru a crea aplicații de orice tip, de la aplicații web, la aplicații de desktop sau aplicații mobile.

Unul dintre cele mai importante avantaje ale Python-ului este că bibliotecile pot funcționa și ca obiecte deci este mult mai ușor de accesat și modificat date.

Exemplu:

```
# Fișierul a.py
valoare = 1
#####

# Fișierul b.py
import a
a.valoare = 3
#####

# Fișierul main.py
import a
import b
print(a.valoare) # Afișează 3
#####

# Se execută comanda în consolă
$ python main.py
```

Justificarea folosirii Flet

Flet este un framework pentru crearea de interfețe grafice în Python. Este un framework foarte ușor de folosit, care permite crearea de interfețe grafice într-un timp foarte scurt. De asemenea, este un framework foarte versatil, care permite crearea de interfețe grafice de orice tip.

Punctul forte al framework-ului este **reducerea drastică** a codului scris pentru a crea interfețe grafice, cât și asemănarea cu framework-ul **Flutter**.

Spre exemplu așa adaugi 3 butoane într-o pagină:

```
import flet as ft

def main(page: ft.Page):
    page.title = "Basic filled buttons"
    page.add(
        ft.FilledButton(text="Filled button"),
        ft.FilledButton("Disabled button", disabled=True),
        ft.FilledButton("Button with icon", icon="add"),
    )

ft.app(target=main)
```

Justificarea folosirii Spotipy

Spotipy este o bibliotecă pentru Python care permite interacționarea cu API-ul Spotify. Este o bibliotecă foarte ușor de folosit, iar API-ul Spotify este foarte bine documentat, ceea ce face foarte ușoară interacționarea cu acesta.

Justificarea folosirii SpotDL

SpotDL este un tool CLI care permite descărcarea melodiilor de pe Spotify. Se execută foarte ușor prin biblioteca `os` din Python, iar melodiile se descarcă la o calitate destul de bună.

Justificarea folosirii SciKit-Learn

SciKit-Learn este o bibliotecă pentru Python care permite crearea de modele de machine learning. Am reușit să creez un mod de calculare a similarității între melodii destul de eficient folosind această bibliotecă.

Justificarea folosirii Threading

Threading este o bibliotecă pentru Python care permite crearea de thread-uri. Am folosit această bibliotecă pentru a crea thread-uri pentru descărcarea melodiilor, astfel încât aplicația să nu se blocheze în timpul descărcării. De asemenea, am folosit această bibliotecă pentru a crea thread-uri pentru calcularea similarității între melodii, astfel încât să evit timpul de așteptare.

Justificarea folosirii FastAPI

FastAPI este un framework pentru crearea de API-uri în Python. Este un framework foarte ușor de folosit, care permite crearea de API-uri într-un timp foarte scurt.

Justificarea folosirii Supabase

Supabase este o bază de date care permite stocarea datelor într-un mod eficient. Este o bază de date foarte ușor de folosit, care permite stocarea datelor într-un mod foarte rapid. Fiind o bază de date bazată pe PostgreSQL, permite stocarea datelor într-un mod foarte sigur și ajută mult la filtrarea melodiilor.

Elementele distinctive ale aplicației în comparație cu competiția

Principalii competitori sunt:

- **Spotify** - aplicația de streaming muzical cu cei mai mulți utilizatori
- **YouTube Music** - aplicația de streaming muzical cu cea mai mare bază de date

Cu ce este diferit?

- **Tehnologia de recomandare**

Aplicația folosește un proces asemănător cu machine learning-ul calculând similaritatea între melodii bazat pe caracteristicile melodiei, după analiza acesteia, față de alți competitori care folosesc un proces bazat pe ascultările utilizatorilor și popularitatea melodiilor

De ce este mai bun?: Cu ajutorul acestui proces utilizatorii pot descoperi melodii noi, care le-ar plăcea, dar pe care nu le-ar fi descoperit altfel.

Exemplul Spotify: Dacă încerci să asculți un playlist amestecat care are o diferență mare de popularitate între melodii, în primele 20 de piese vei asculta mare parte doar melodiile populare, iar cele mai puțin populare vor fi ascultate foarte rar. Astfel, utilizatorii nu vor descoperi melodii noi, care le-ar plăcea, dar care nu sunt populare.

- **Filtrarea specializată**

Aplicația permite filtrarea melodiilor după caracteristici specifice, precum: energie, dansabilitate și tempo

La ce ajută?: În general ca DJ ai nevoie să cauți melodii dintr-o anumită zonă definită de caracteristici, pentru că trecerea de la o melodie la alta să fie cât mai naturală. Cu cât o tranziție este mai naturală, cu atât publicul va fi mai mulțumit.

- **Descărcarea melodiilor**

Aplicația permite descărcarea melodiilor în format MP3, astfel încât să poți folosi melodiile în alte aplicații

Diferența față de competitori: Prin faptul că piesele sunt descărcate în format MP3, utilizatorii pot folosi melodiile în alte aplicații, precum: Virtual DJ, Serato DJ, Rekordbox DJ, etc. În cazul competitorilor nu se poate face acest lucru, deoarece melodiile sunt stocate într-un format proprietar, deci nu pot fi folosite mai departe.

- **Salvarea playlist-urilor local**

Aplicația permite salvarea playlist-urilor local, astfel încât să poți folosi playlist-urile în alte aplicații, să le poți trimite prietenilor sau chiar să le utilizezi pe un alt calculator

Avantaje: Datorită spațiului necesar pentru stocarea playlist-urilor fiind redus, fișierele pot fi trimise foarte ușor prin intermediul oricărei aplicații de comunicare, precum: WhatsApp, Messenger, Discord, etc.

- **Interacțiunea cu publicul**

Aplicația web progresivă permite interacțiunea cu publicul prin intermediul unui playlist unde participanții la eveniment pot adăuga sugestii de melodii astfel, DJ-ul poate vedea ce melodii ar dori publicul să asculte

Avantaje: DJ-ul se poate plia foarte ușor pe dorințele publicului, astfel încât să fie mulțumit de muzica ascultată.

Ghidul de instalare

UtilifyLocal - aplicația entertainer-ului

1. Descarcă arhiva pentru sistemul tău de operare de pe pagina de [Releases](#) a repository-ului de GitHub
2. Dezarhivează folder-ul
3. Deschide folder-ul și execută [Utilify.exe](#)
4. Ai pornit aplicația

UtilifyPWA - aplicația publicului

1. Accesează pagina de [Releases](#) a repository-ului de GitHub
2. Descarcă arhiva [UtilifyPWA.zip](#)
3. Dezarhivează folder-ul
4. Se deschide folder-ul în terminal și se lansează comanda:

```
pip install -r requirements.txt
```

- **Metoda fly.io**

4. Deschide folder-ul și editează fișierul [fly.toml](#) cu un nume la alegere
5. Deschide folderul în terminal și lansează imaginea de Docker pe fly.io:

```
flyctl apps create --name <numele-aplicației-din-fișierul-editat>  
flyctl deploy
```

6. După încărcare partajezi link-ul primit în terminal cu publicul

- **Metoda localtunnel**

4. Se instalează localtunnel folosind comanda:

```
npm install -g localtunnel
```

5. Deschide folderul în terminal și lansează comanda:

```
flet main.py
```

6. Se deschide din nou folderul în terminal și lansează comanda:

```
lt --port 8080
```

7. După încărcare partajezi link-ul primit în terminal cu publicul

Testimoniale

Cristian - DJ Zamfir

O aplicație foarte folositoare mai ales atunci când nu ai idee ce melodie să alegi din atât de multe.

Dan - DJ Ino

Programul este interesant, de abia aștept să văd cum generează playlist-uri, mi-ar ușura munca enorm. Doar vii la petrecere te pui la consolă și poți să te și distrezi, la cât te scapă de muncă.

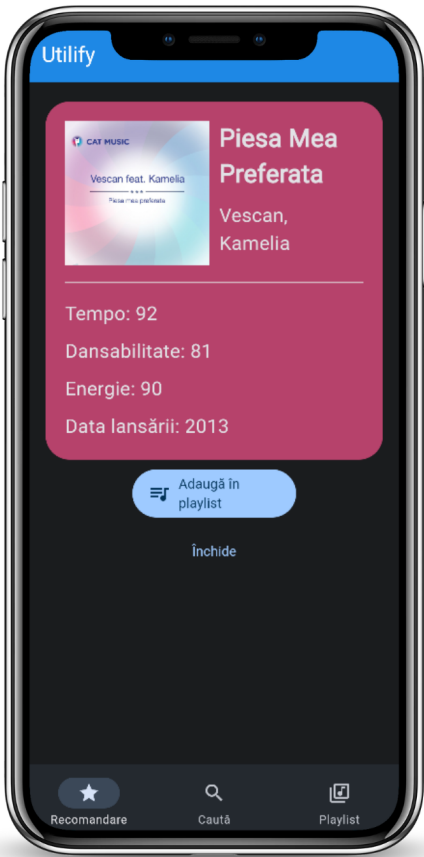
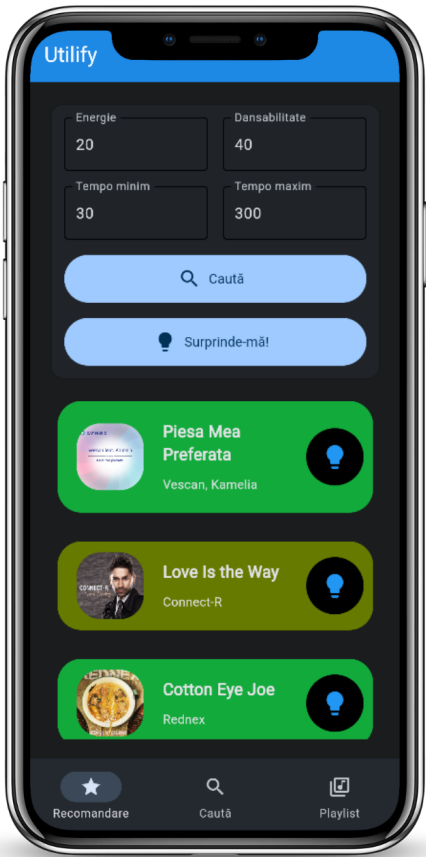
YouDJ - Platformă educațională pentru DJ

Apreciem mult inițiativele noi, iar această idee de aplicație ne-a surprins plăcut. Am testat programul și suntem nerăbdători să începem colaborarea, ar fi de mare ajutor API-ul. Chiar este folositor mai ales pentru începătorii care nu prea au idee ce să aleagă."

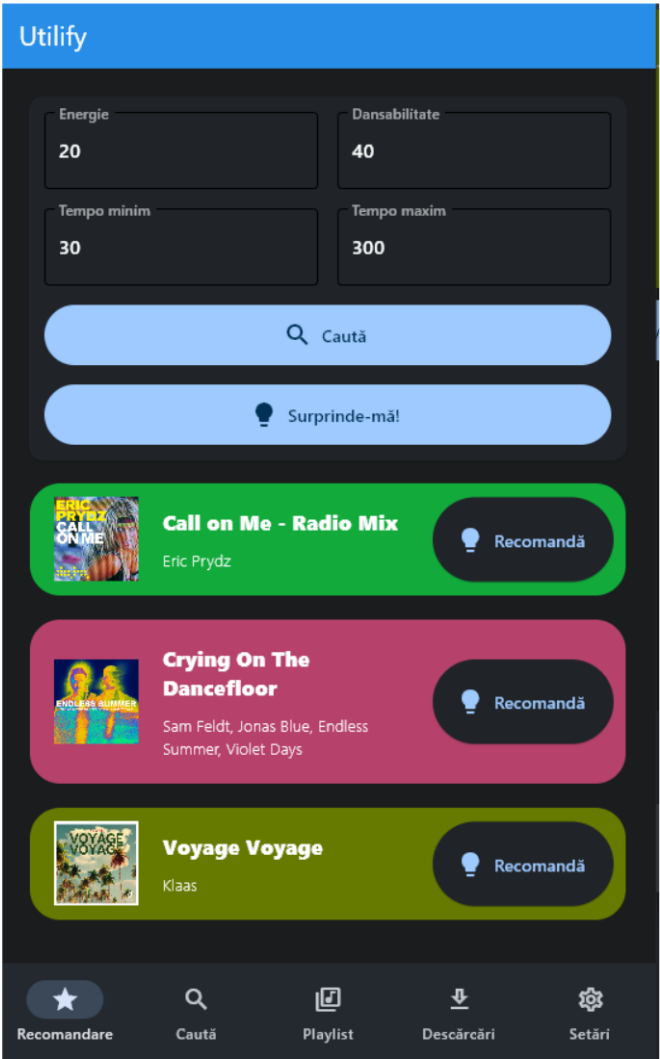
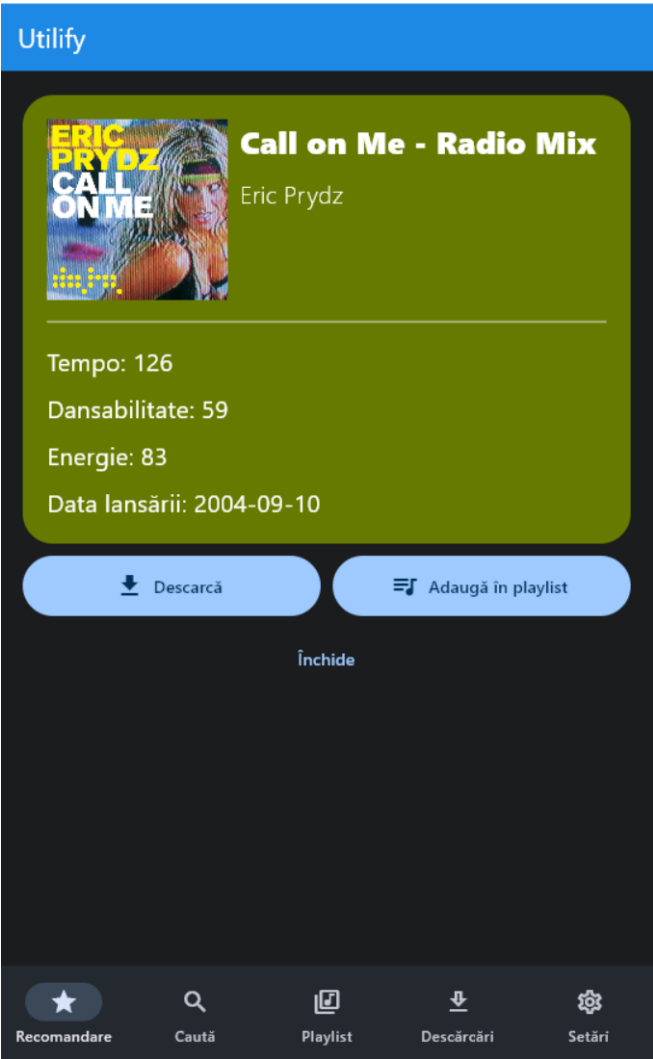
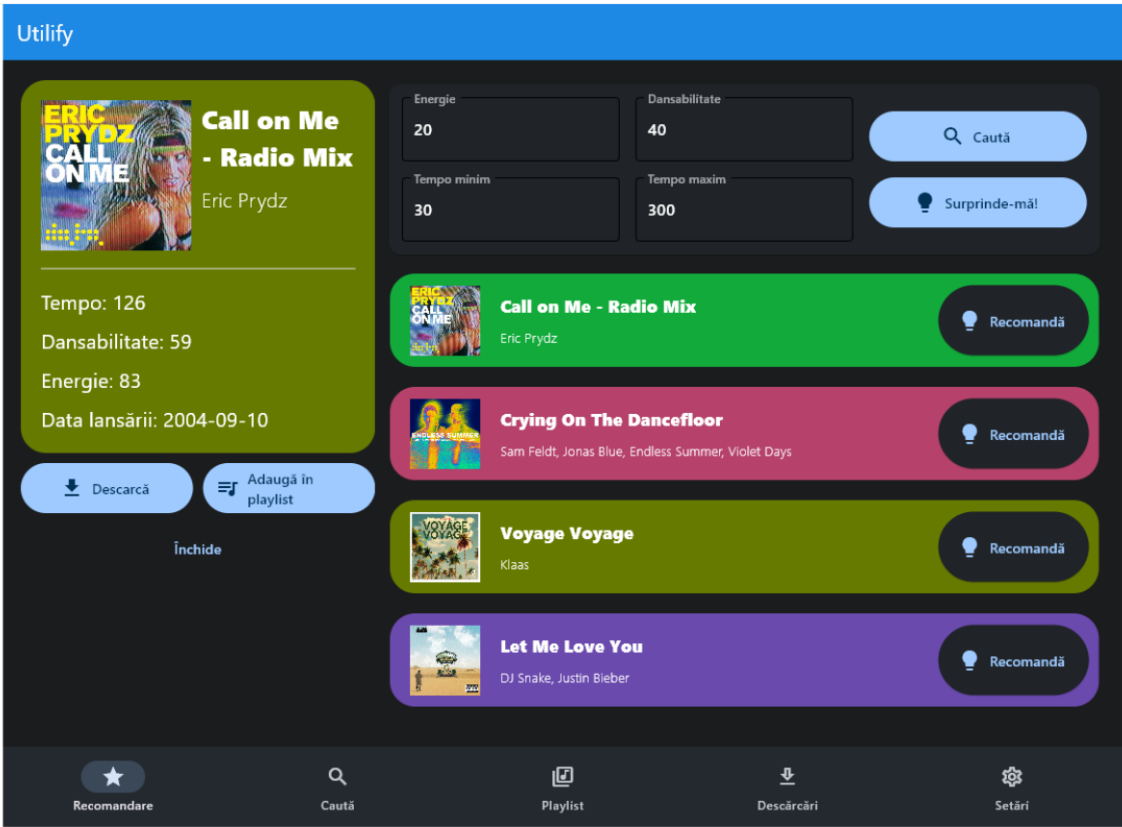
Imagini anexe



Aplicația publicului



Aplicația entertainer-ului



Raport utilizare baza de date

