**Proiect Metode Dezvoltare Software**

Glazov Maria-Liana

Grupa 244

# Cuprins

[Cuprins 1](#_Toc1891998596)

[Descrierea proiectului 2](#_Toc2128794786)

[Motivație 3](#_Toc1048184514)

[User stories 3](#_Toc256172503)

[Diagrama aplicației 4](#_Toc72821230)

[Procesul dezvoltării proiectului 5](#_Toc860240605)

[Aplicația 7](#_Toc938331850)

[Codul sursă 10](#_Toc2367019)

[Folosirea unui tool AI 12](#_Toc1670411030)

# Descrierea proiectului

În cadrul acestui proiect am dezvoltat o aplicație web care oferă statistici și informații în legătură cu activitatea unui utilizator pe platforma de ascultat muzică Spotify. Utilizatorul se poate conecta direct prin intermediul aplicației Spotify, fără a fi nevoie de crearea unui cont, și poate vedea ce piese și artisti ascultă, grafice care reprezintă vizual activitatea acestuia în cadrul platformei Spotify, sau valori care descriu caracteristicile muzicale apreciate de acesta.

Aplicația a fost dezvoltată în framework-ul .NET, folosind patternul Model-View-Controller, în limbajul C#, iar pentru partea de frontend s-au folosit pagini Razor.

# Motivație

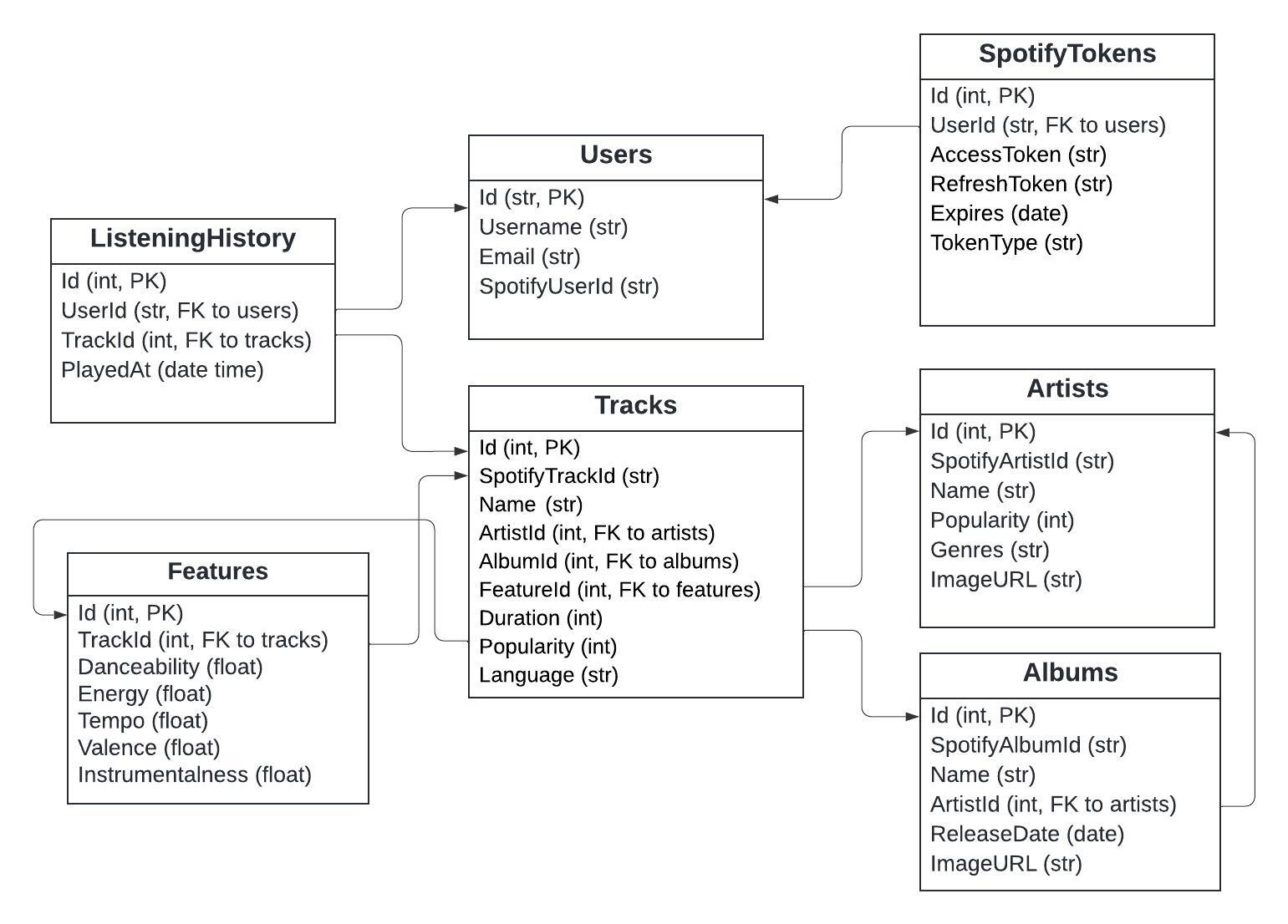
Ideea proiectului a pornit de la dorința de a avea o aplicație care arata statistici pentru Spotify cu funcționalitatea de a vizualiza un sumar al “limbilor cele mai ascultate”, pentru a vedea ce limbi sunt preponderente în melodiile pe care le ascult. Dintre toate site-urile și aplicațiile de acest tip pe care le-am folosit, niciunul nu avea o astfel de funcționalitate, și știind de api-ul de la Spotify care oferă foarte multe informații legate de muzica ascultată de un utilizator, am decis să încerc sa folosesc acest api pentru a calcula aceste statistici.

# User stories

* Ca utilizator îmi doresc să văd ce piese/artiști am ascultat cel mai mult, în cadrul a unei perioade specifice de timp (4 săptămâni, 6 luni sau un an).
* Ca utilizator îmi doresc să văd istoricul ascultărilor mele, ce piese am ascultat și la ce ore din zi.
* Ca utilizator îmi doresc să văd caracteristici detaliate ale pieselor ascultate, pentru a avea o înțelegere profundă asupra preferințelor mele muzicale.
* Ca utilizator doresc să am o privire de ansamblu asupra limbilor prezente în muzica mea preferată, pentru a vedea diversitatea pieselor pe care le ascult.
* Ca utilizator vreau să văd când ascult muzica, în ce ore și zile sunt cel mai activ ascultând.

# 

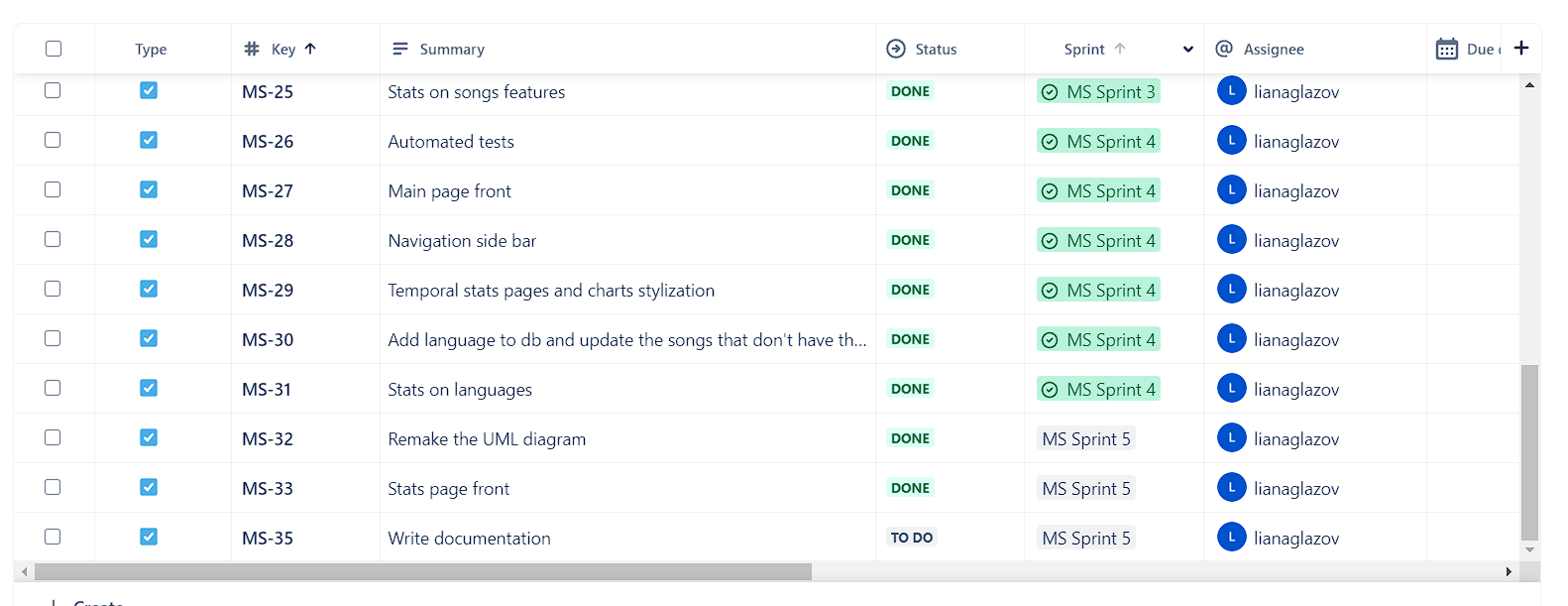
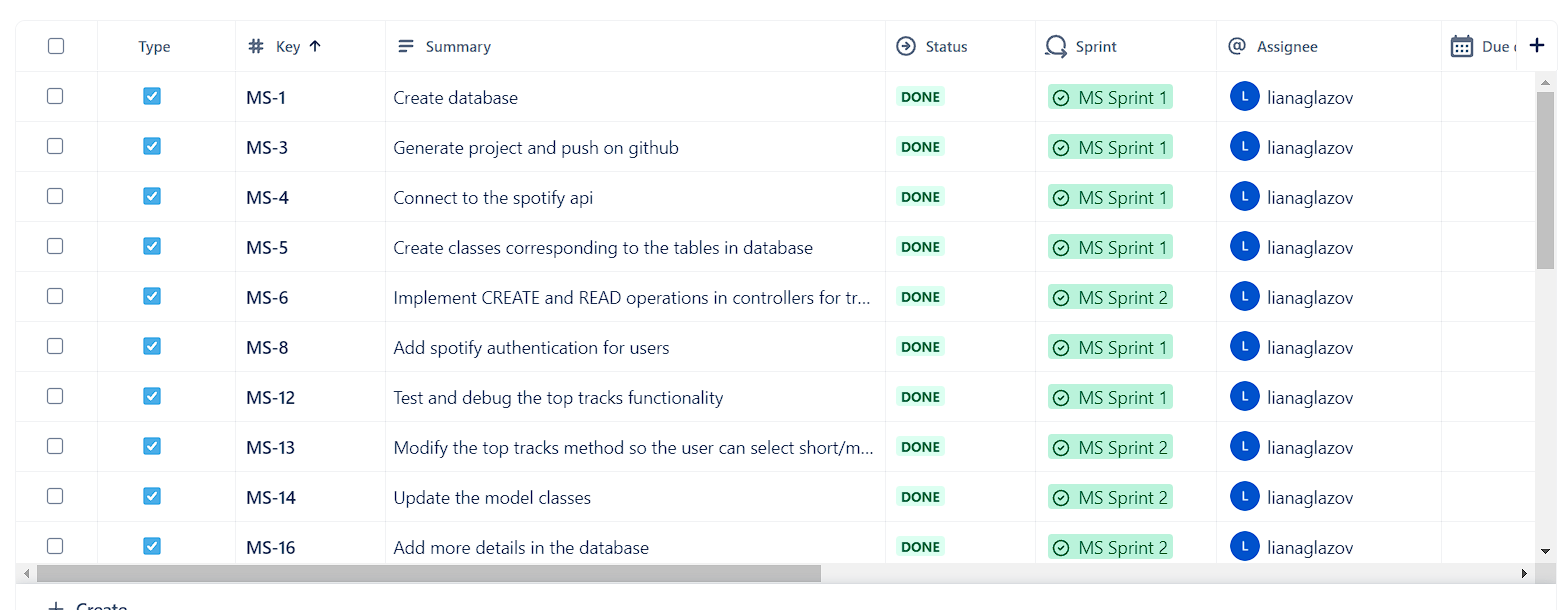
# Diagrama aplicației

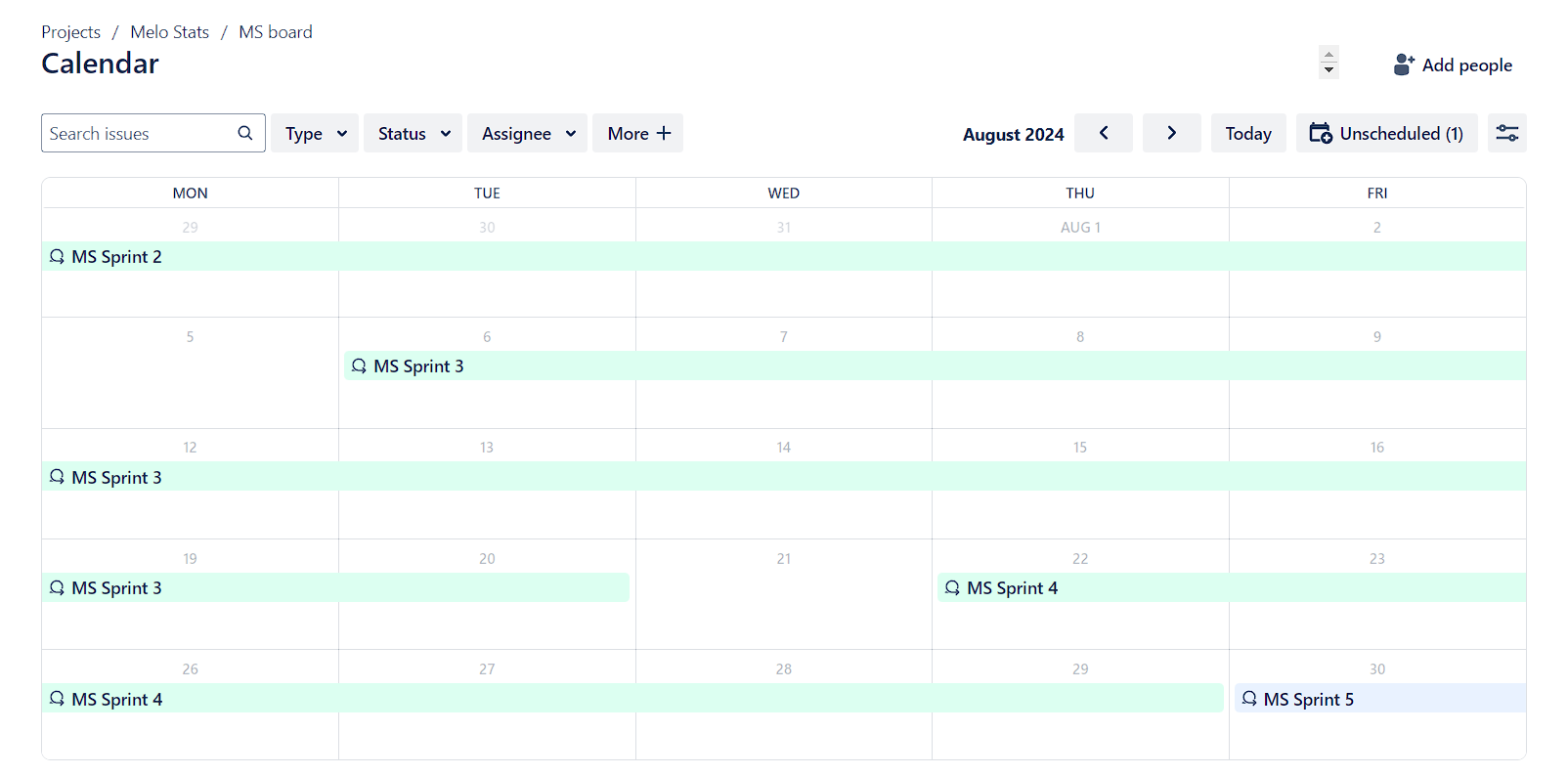


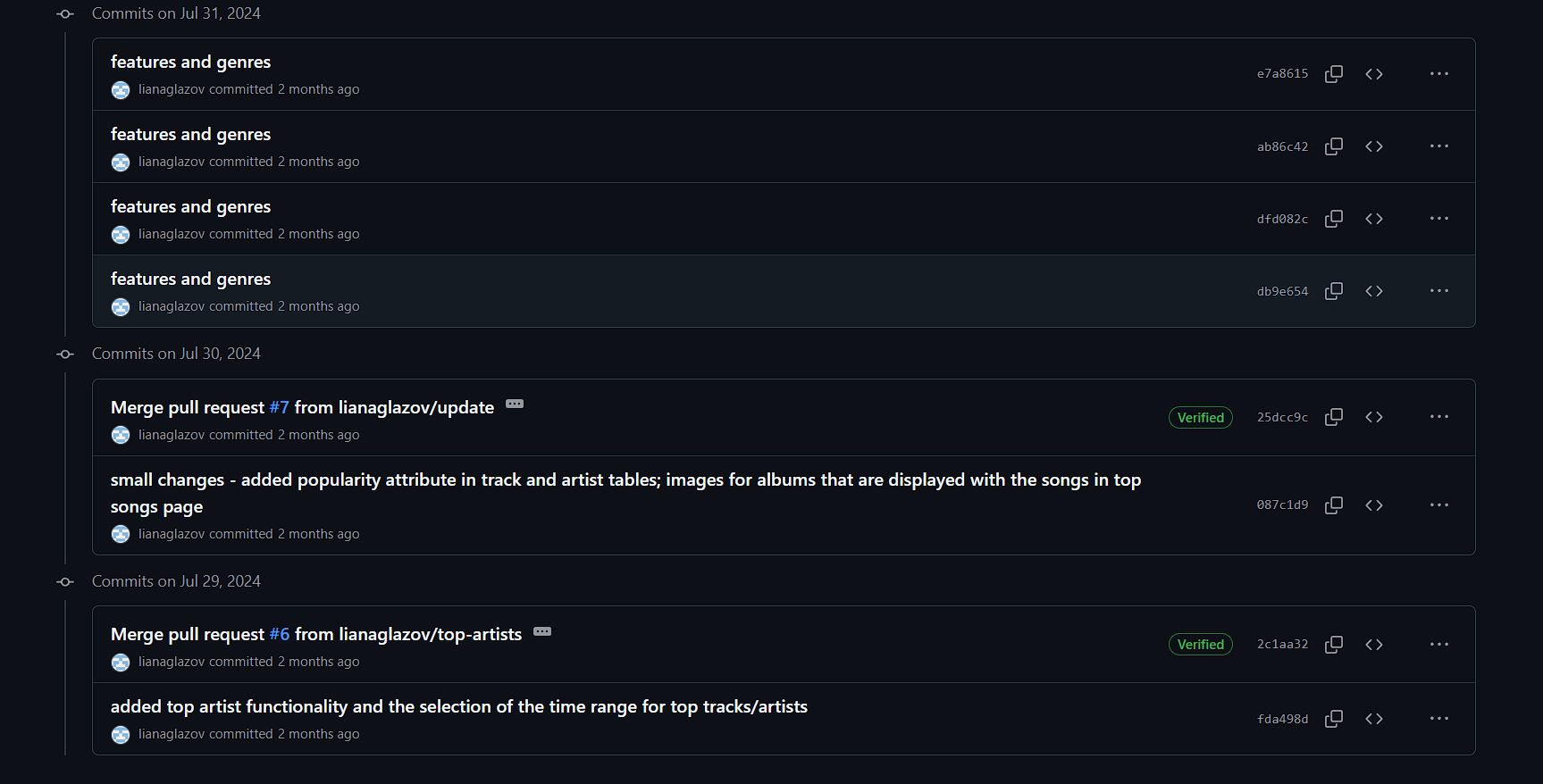
# Procesul dezvoltării proiectului

Chiar dacă am realizat acest proiect singură, mi-am organizat munca și resursele folosind JIRA și GitHub.

Proiectul a fost împărțit în mai multe task-uri, desfășurându-se în 5 sprinturi de câte o săptămână. Fiecare task finalizat a fost apoi încărcat pe GitHub, cu push și pull request.

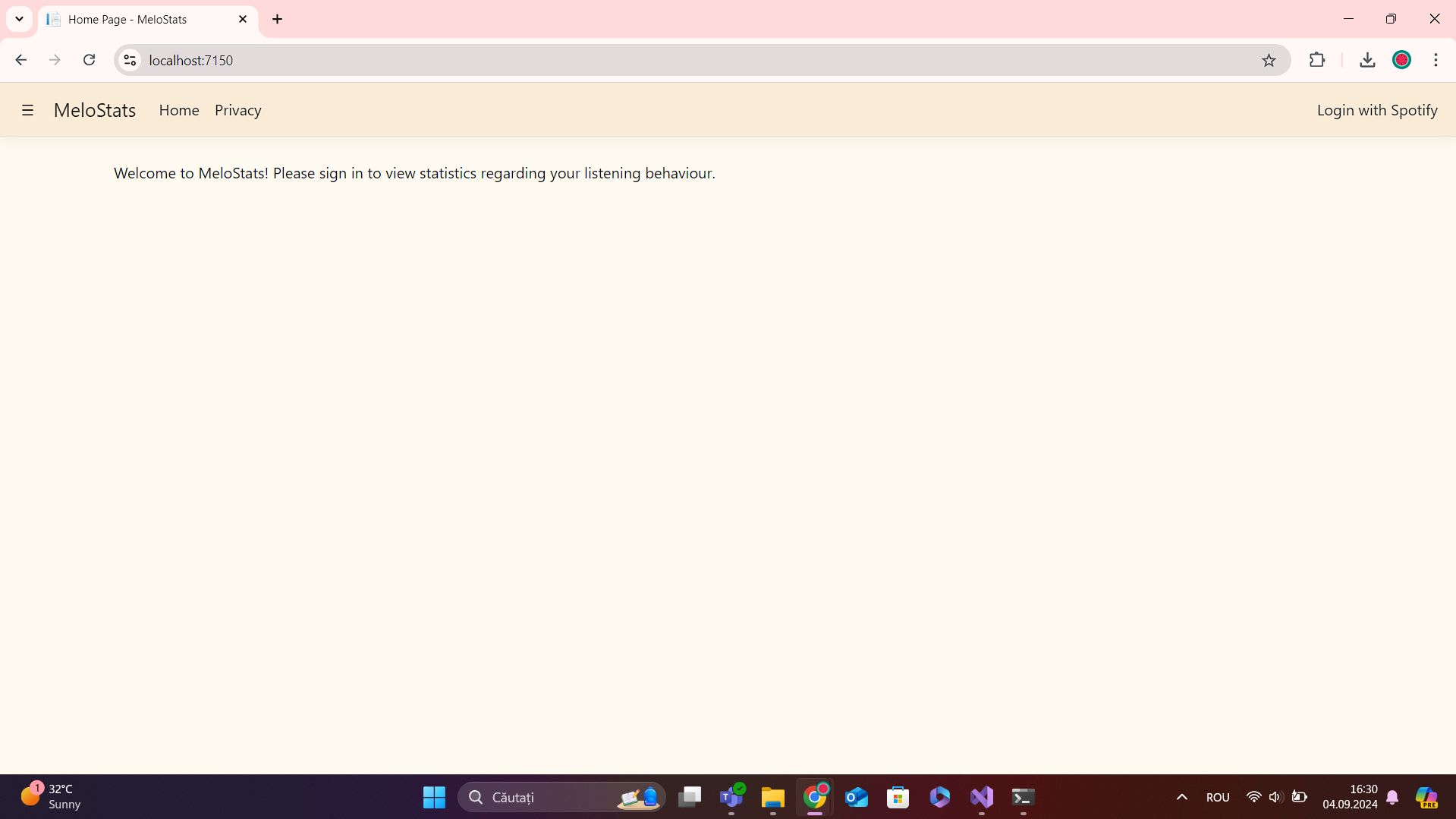




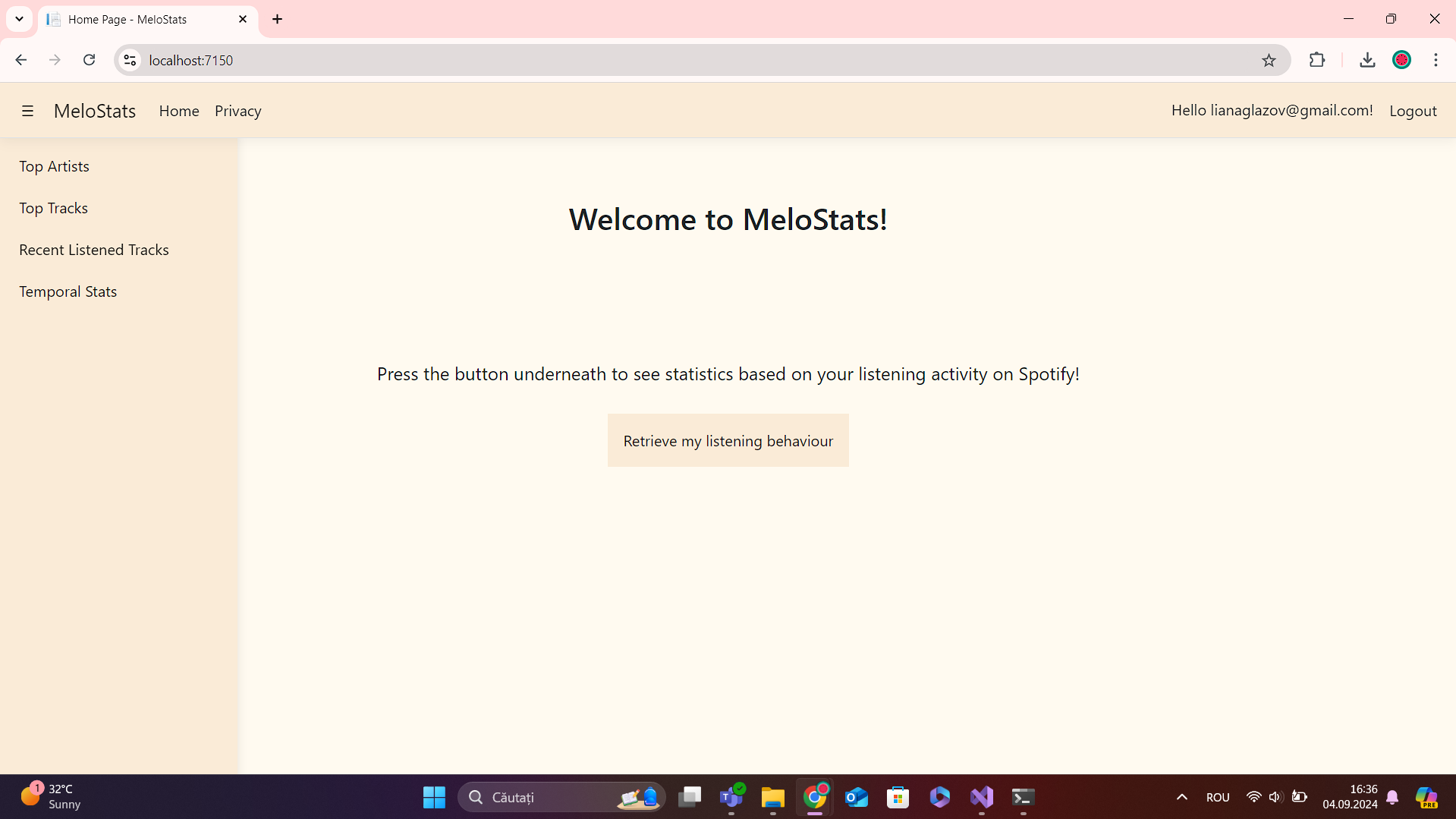


# Aplicația

Pentru a putea folosi funcționalitățile aplicației, utilizatorul trebuie să se conecteze cu Spotify.

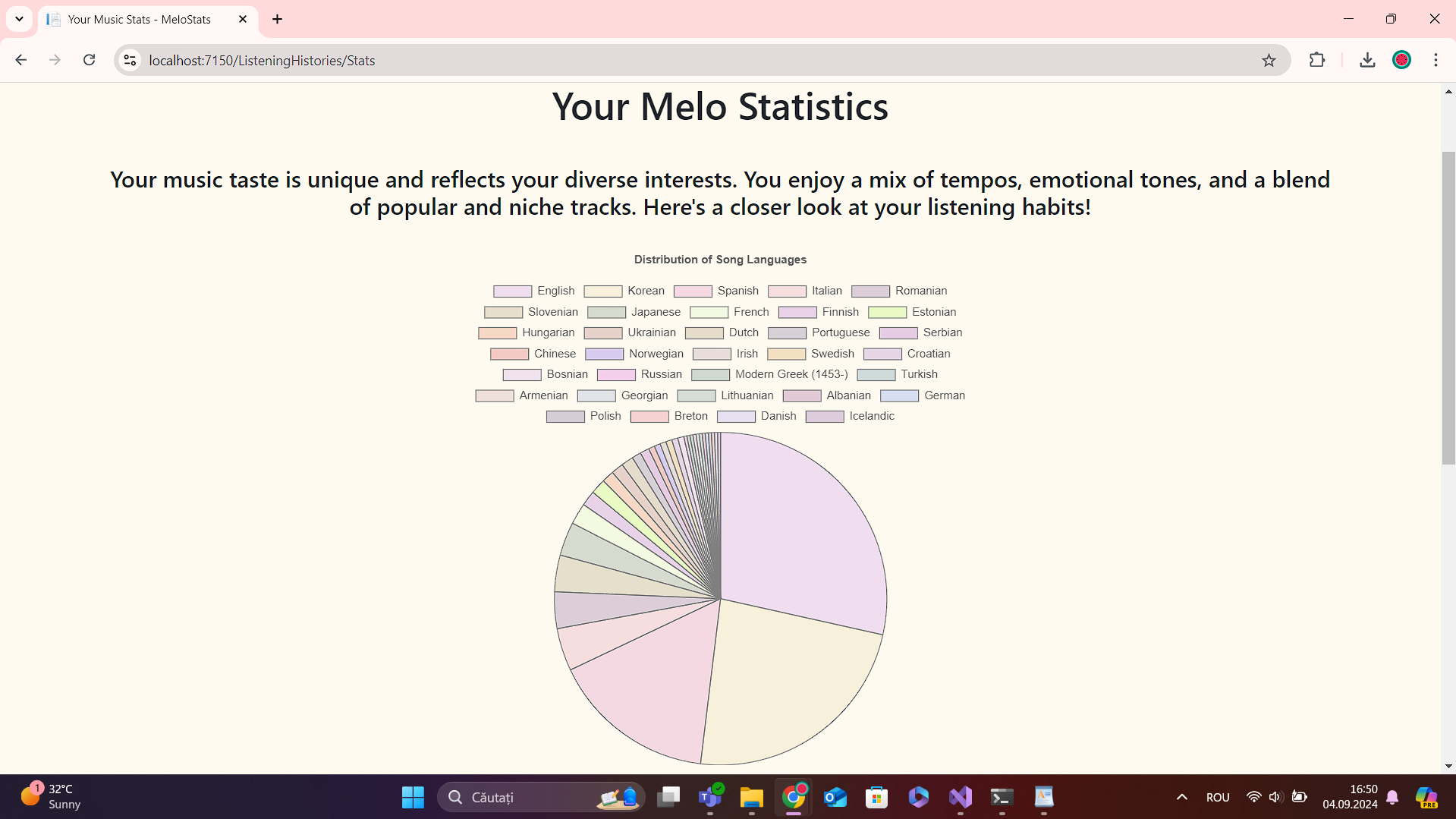


Odată conectat, userul poate accesa informațiile legate de muzica ascultată



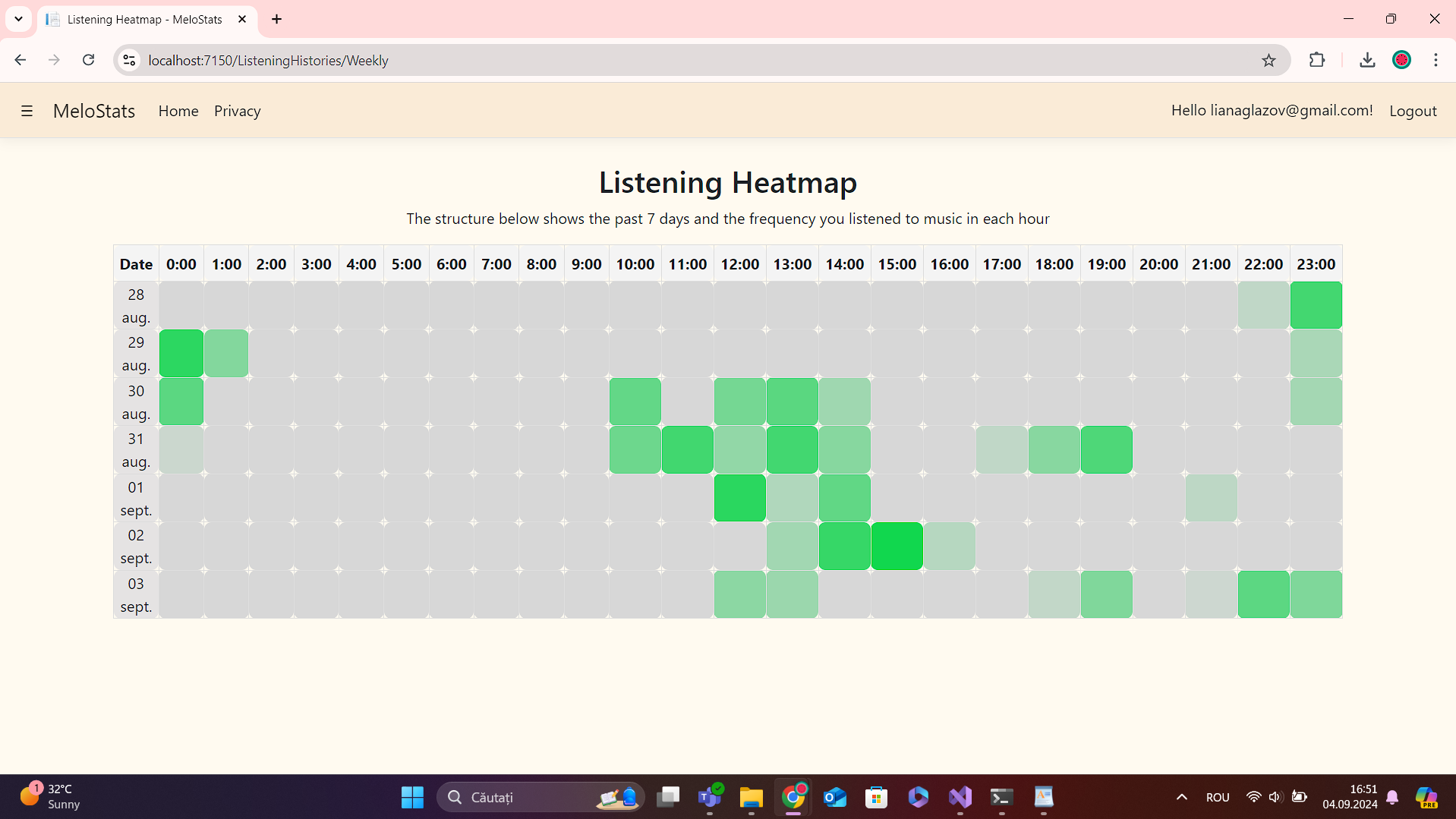
Apăsarea butonului din pagina principala va apela api-ul de la Spotify, obținând astfel piesele ascultate de utilizator, și va calcula mediile valorilor caracteristicilor acelor piese, redirecționând utilizatorul către pagina cu statistici.

Cum api-ul Spotify nu oferă și informații în legătură cu versurile pieselor, am folosit api-ul de la Genius pentru a putea prelua limba unei piese.



De asemenea, aplicația are un meniu lateral cu mai multe funcționalități: vizualizarea celor mai ascultate piese sau celor mai ascultați artiști pe parcursul a diferite perioade de timp (4 săptămâni, 6 luni, un an), istoricul de ascultare al unui user, care cuprinde ultimele 50 de piese ascultate, și pagini care conțin reprezentări vizuale a timpului petrecut ascultând muzică.

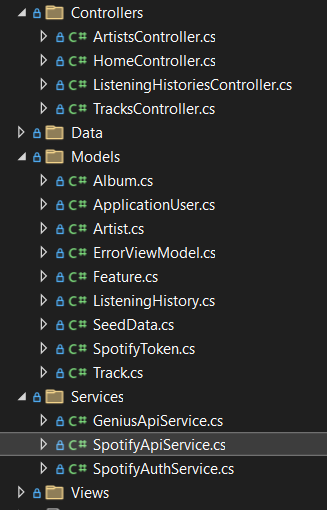




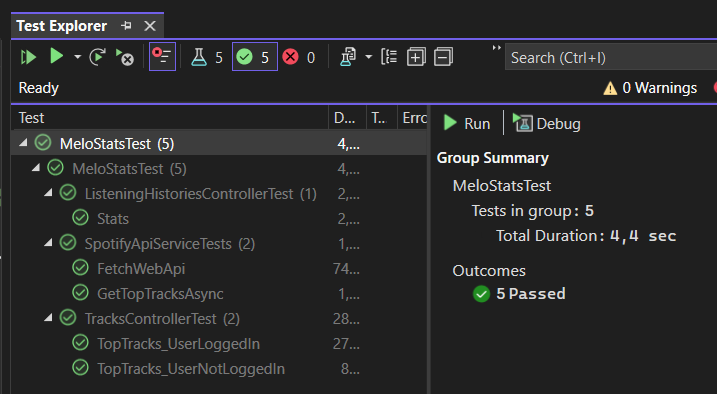
# Codul sursă

Aplicația a fost dezvoltată folosind arhitectura Model-View-Controller, în cadrul framework-ului Microsoft .Net. Partea de back end a fost scrisă în limbajul de programare C#, iar pentru front end s-au folosit pagini Razor cshtml.

Astfel am creat clasele model corespunzătoare diagramei, am folosit controllere pentru a defini acțiunile corespunzătoare funcționalităților aplicației, iar paginile au fost afișate prin intermediul view-urilor.

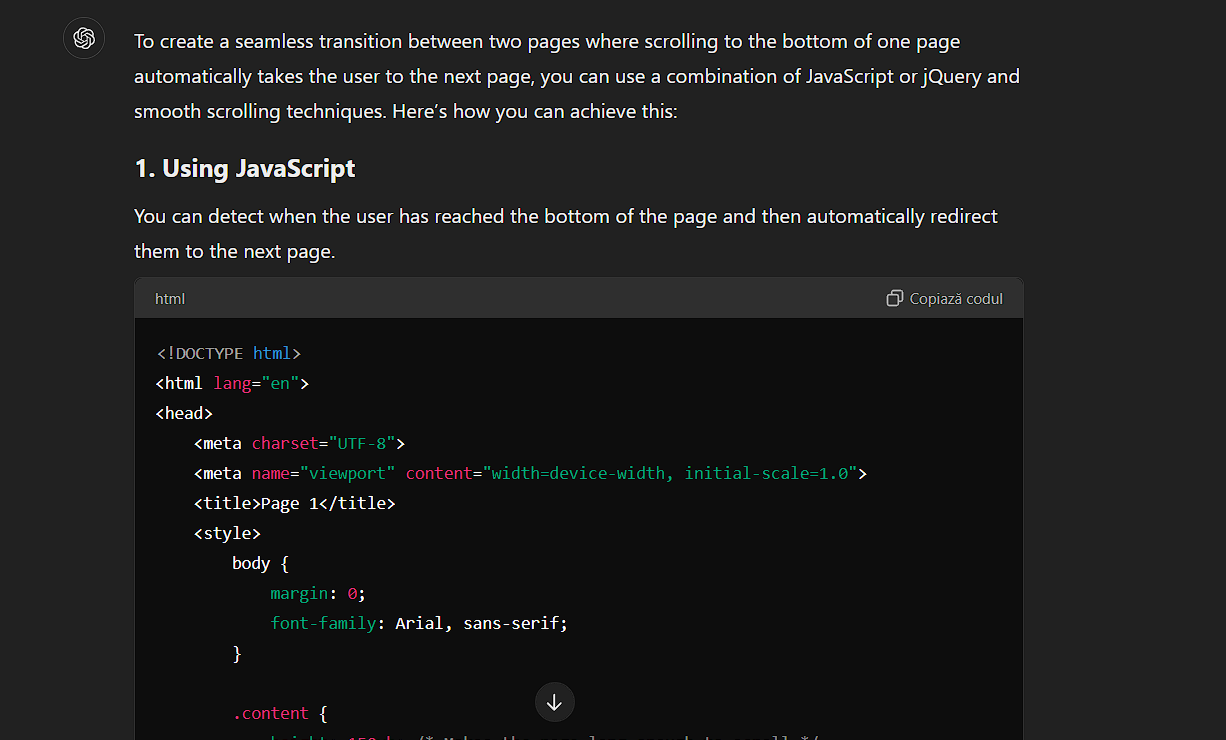


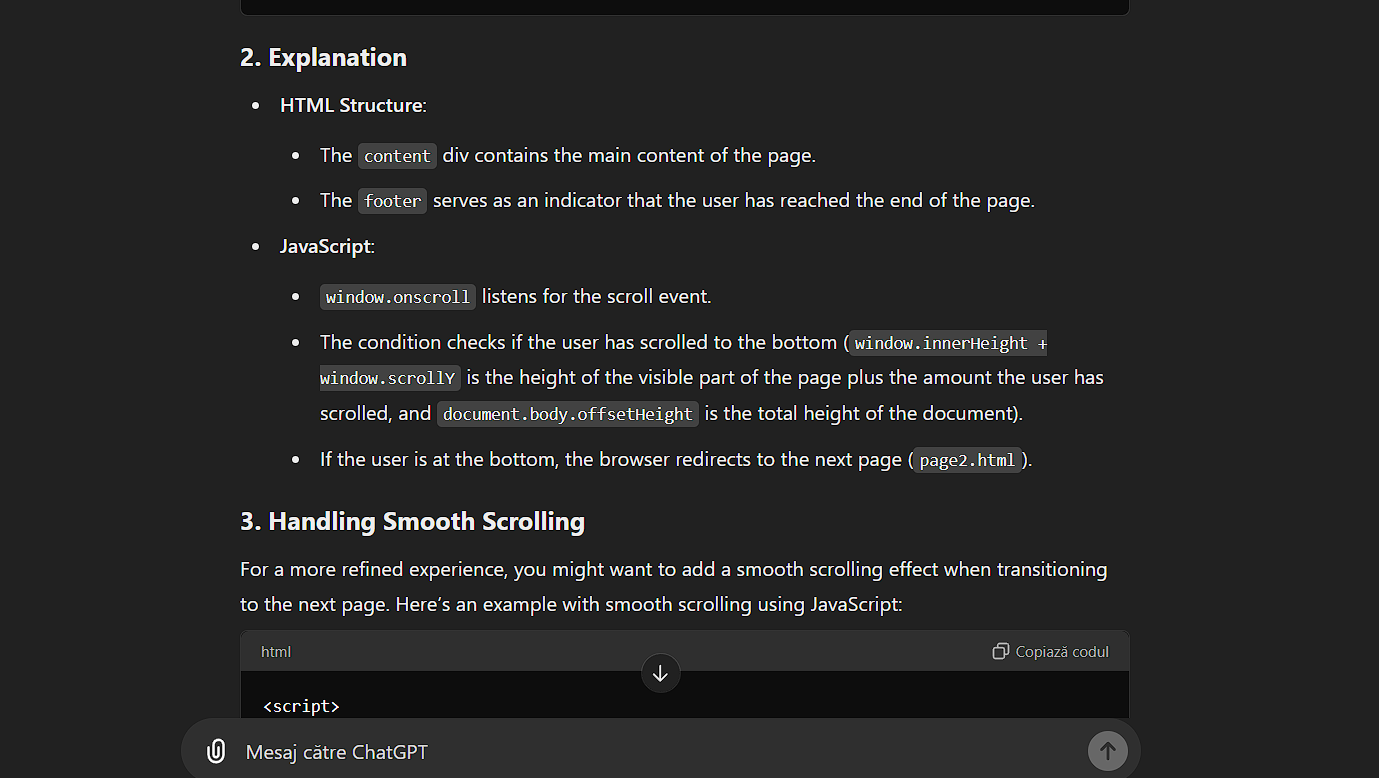
Pentru a apela api-urile corespunzătoare am definit niște clase service care încapsulează logica din interacțiunea cu Spotify API și Genius API.

De asemenea au fost efectuate teste automate pentru o parte din acțiunile aplicației. 

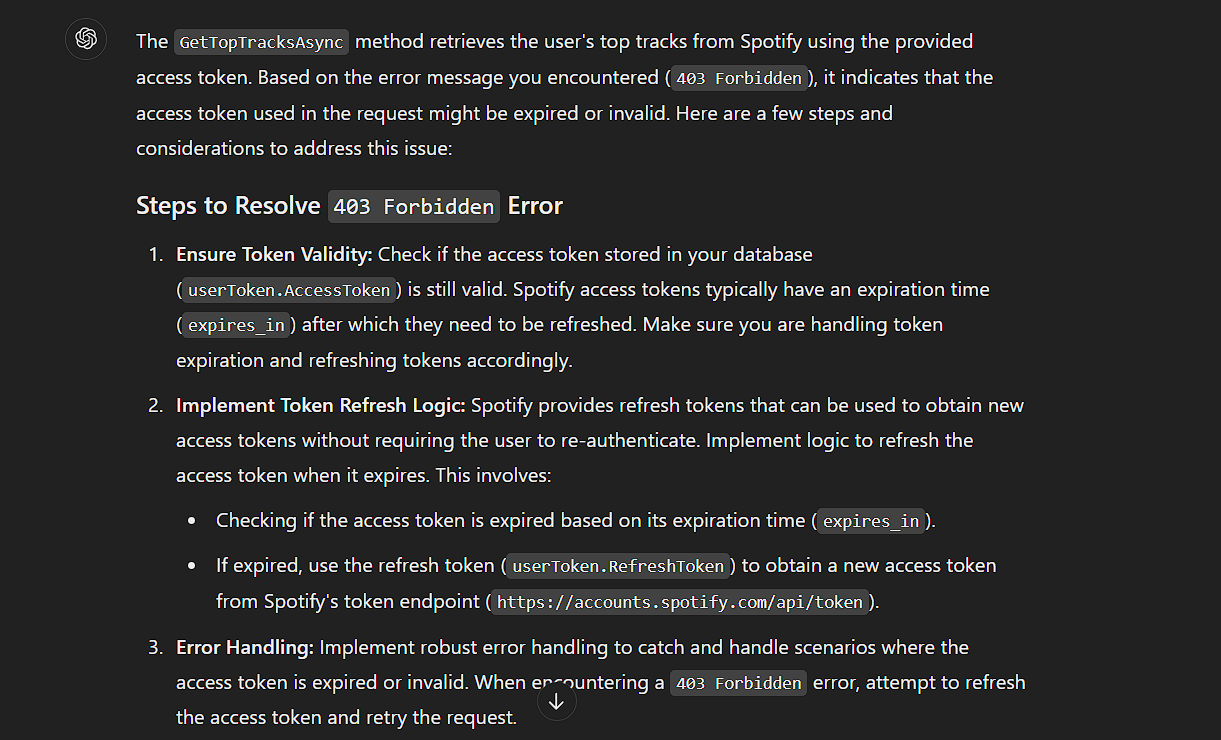
# Folosirea unui tool AI

Pe parcursul dezvoltării proiectului, din dorința de a eficientiza munca, sau din cauza unor probleme întâlnite în cod care păreau că nu au o soluție, am cerut ajutor uneltelor de inteligență artificială.





În conversația de mai sus am cerut ajutor cu implementarea ideii de a avea o tranziție între două pagini, astfel încât acțiunea de scroll să redirecționeze utilizatorul către următoarea pagină.



De asemenea, Chat GPT a reușit să mă scoată de nenumărate ori din impas, în momentele în care întâmpinam erori în dezvoltarea aplicației.