

p-laylist

AI powered music recommendations.

Project Manager: Angelina Tsai, 2026, CS-Physics

Contact: antsai@g.hmc.edu

Project Manager: Korin Aldam-Tajima, 2026, CS-Physics

Contact: kaldamtajima@g.hmc.edu

Project Manager: Tyler Headley, 2026, CS-Math

Contact: theadley@g.hmc.edu

Project Overview

Have you ever been frustrated with Spotify recommending the same songs repeatedly, even when your interests change? Or wished you could expand your music taste and fill in the gaps in your playlists? What about that friend with impeccable music taste whose playlists you can't access because they use Apple Music? Our project tackles these issues by developing an AI-powered web app for collaborative music recommendations.

Our platform will generate daily, personalized playlists by using a hybrid of collaborative filtering and content-based recommendation techniques. The core of our product is a recommendation engine that analyzes both user behavior and song characteristics. In addition, we plan to integrate social media-like features for collaborative music sharing, song reviews, and discussions, enhancing community interaction around music discovery.

On the technical side, our recommendation model will employ an ensemble approach. We'll combine deep learning for music content analysis with traditional recommendation algorithms like matrix factorization or graph-based models for collaborative filtering. By tuning parameterized weights, users can receive diverse recommendations, such as music beyond their typical taste, emerging artists, or popular picks overall. Our ultimate goal is to offer a seamless, engaging experience, connecting users across various streaming platforms, and providing them with dynamic music recommendations based on their playlists and listening habits.

Tech Stack

- Python
- PyTorch

- Google Colab
- Flask
- React
- Docker
- AWS
- PostgreSQL

Dataset

- [Spotify Web API](#)
- [Apple Music API](#)
- [Amazon Music Device API](#)
- [Spotify Research: Million Playlist Dataset](#)

Ethical Considerations

Our database will consider user-specific, personal, data regarding their music taste and demographics. These must remain private unless the user consents to sharing them. We will evaluate the sources of our datasets to ensure compliance with music copyright laws. In recognizing potential biases inherent in streaming platforms, we will remain vigilant in identifying and mitigating any implicit biases in our data analysis. Additionally, our project will aim to promote fairness by promoting lesser-known artists whose music is of high quality but may have fewer streams. Overall, aim to prevent echo chamber, filter bubbles, and polarization effects evident in social media platforms.

Team Structure

- Target Team Size: *5-6 project members*
- Subteam Structure:
 - Creative Development
 - Domain expert
 - SWE Team Roles:
 - Frontend + UI/UX Design (filled)
 - Backend + Database (*looking to recruit for backend specifically deployment*)
 - ML Team Roles:
 - ML Research + Model Building (filled)
 - MLOps + Data Engineering (filled)
- Organizational & Collaboration Frameworks:
 - Slack
 - Github

Timeline

Building off what we developed in the Fall semester our timeline is adjusted as follows:

Spring 2025	SWE Team	ML Team
Week 0	Familiarize with codebase and discuss proper coding practices/code review	Familiarize with codebase and discuss proper coding practices/code review
Week 1	Synchronize development and production environments. Get Nginx server working.	Deploy first hybrid ML recommendation model.
Week 2-4	Backend: Work on reading from the database, and display properly on the web app. Frontend: Fine tune component library and design. Display components properly on web page.	Scale ML models and datasets for performance (speed and accuracy). Research/implement algorithms to assist with much larger datasets (up to 1000 times larger).
Week 5-6	Backend: Implement user interactions and connect with backend APIs. Integrate other music platforms (Spotify already integrated). Frontend: Implement friends page, displaying recommendations, and search page UI.	Review basic deployed models and discuss changes. Implement and deploy changes.
Week 7	Work on exporting music recommendations to respective music apps linked by user.	Model productionizing, continuous online learning if time permits, tweaks and fine tuning.
Week 8-9	Start testing the frontend and backend components together. Address any integration issues. Finalize web app design, begin beta testing.	
Week 10	Fine tune final product. Start preparing final documentation.	Fine tune final product. Start preparing final documentation.

Fall Semester Progress Timeline: (green = completed, yellow = in progress)

Fall 2024	SWE Team	ML Team
Week 0	Familiarize with codebase, learn necessary skills (i.e. React and Flask).	Literature Review (background information, survey papers)
Week 1	Backend: Develop database schema Frontend: Integrate React (+React Router) into existing codebase	Obtain a preliminary dataset from API, create a very simple recommendation model as a placeholder.
Week 2-3	Backend: Begin connecting various streaming service APIs, creating a pipeline to store data in the database. Frontend: Brainstorm using wireframes and figma. Begin to create a site skeleton, coding basic UI components.	Develop matrix factorization optimization algorithm and model for collaborative component of recommender system (libraries will help here).

Week 4	Backend: Develop pipelines to preprocess and write data into the database. Frontend: Implement home page design using React, creating a component library.	Develop a deep learning model for song content analysis and algorithm for measuring song similarity based on output (libraries will help here).
Week 5	Backend: Work on reading from the database, and display properly on the web app. Frontend: Fine tune component library and design. Display components properly on web page.	
Week 6	Backend: Implement user interactions and connect with backend APIs. Frontend: Implement sign in and search page UI.	Deploy first hybrid ML recommendation model.
Week 7	Start testing the frontend and backend components together. Address any integration issues.	Review basic deployed models and discuss changes. Implement and deploy changes.
Week 8-9	Finalize web app design, begin beta testing.	Model productionizing, continuous online learning if time permits, tweaks and fine tuning.
Week 10	Fine tune final product. Start preparing final documentation.	

Project Member Requirements

- Excitement about our project and willingness to learn new skills
- ML Team: Python, experience with ML libraries and/or model deployment preferred
- SWE Team: Python, Docker, and/or production environment experience with some of our tech stack preferred
- Creative Development Team: Music knowledge or graphic design are valuable, even if you don't have any technical experience
- Commit 4-5 hours/week

Minimum Viable Product

- Deployed dynamic webpage on domain name
- Full dataset ML model
- Working pipelines between ML -> backend -> frontend