

AURALY

FEEL THE MUSIC, HEAR THE MOOD



AURALY

THE CURATORS

The minds behind Auraly- curating emotions through data and sound:



NEEMA NALEDI
Data Engineer -
Deployment
Specialist



JUNE HENIA
Data Engineer-
Modelling Specialist



MORGAN AMWAI
Data Analyst-
Reporting Specialist



BRIAN KIMATHI
Presentation Lead-
Communications



MARK MURIITHI
Project Leader

Every Vibe Deserves a Soundtrack

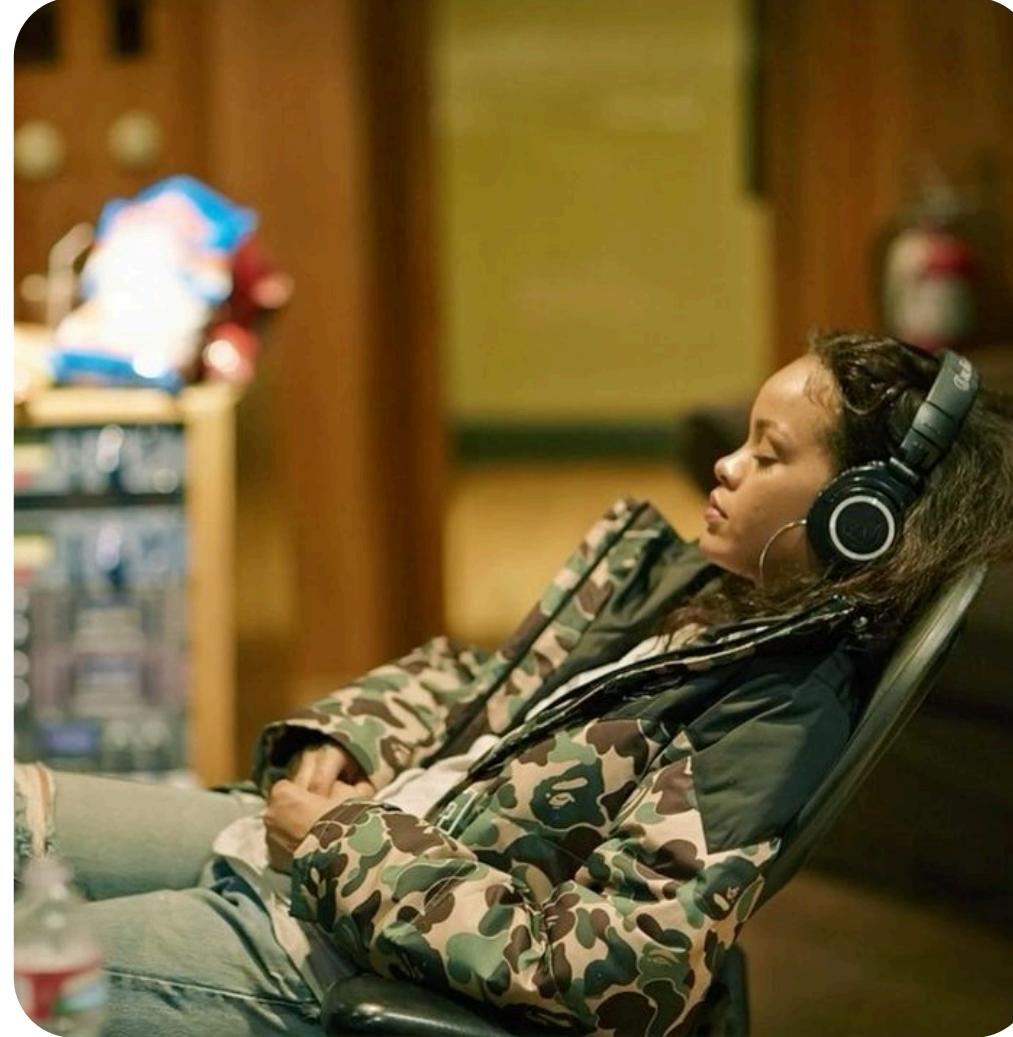
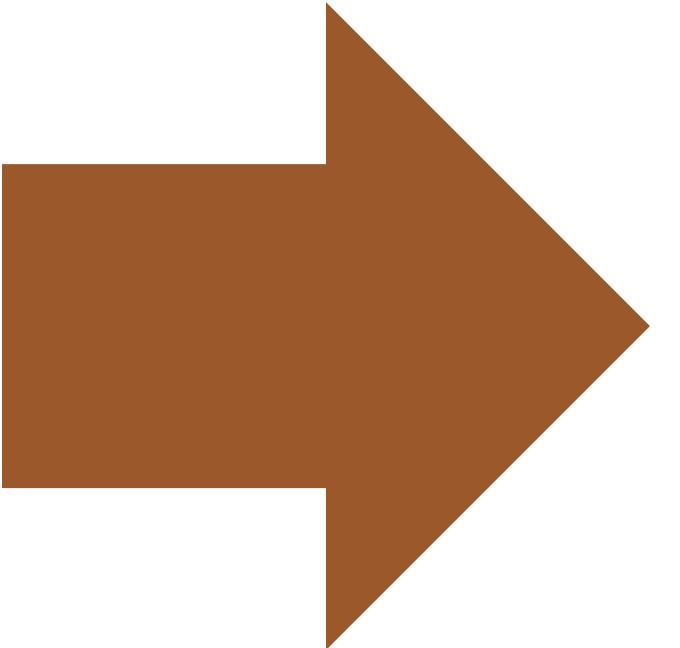
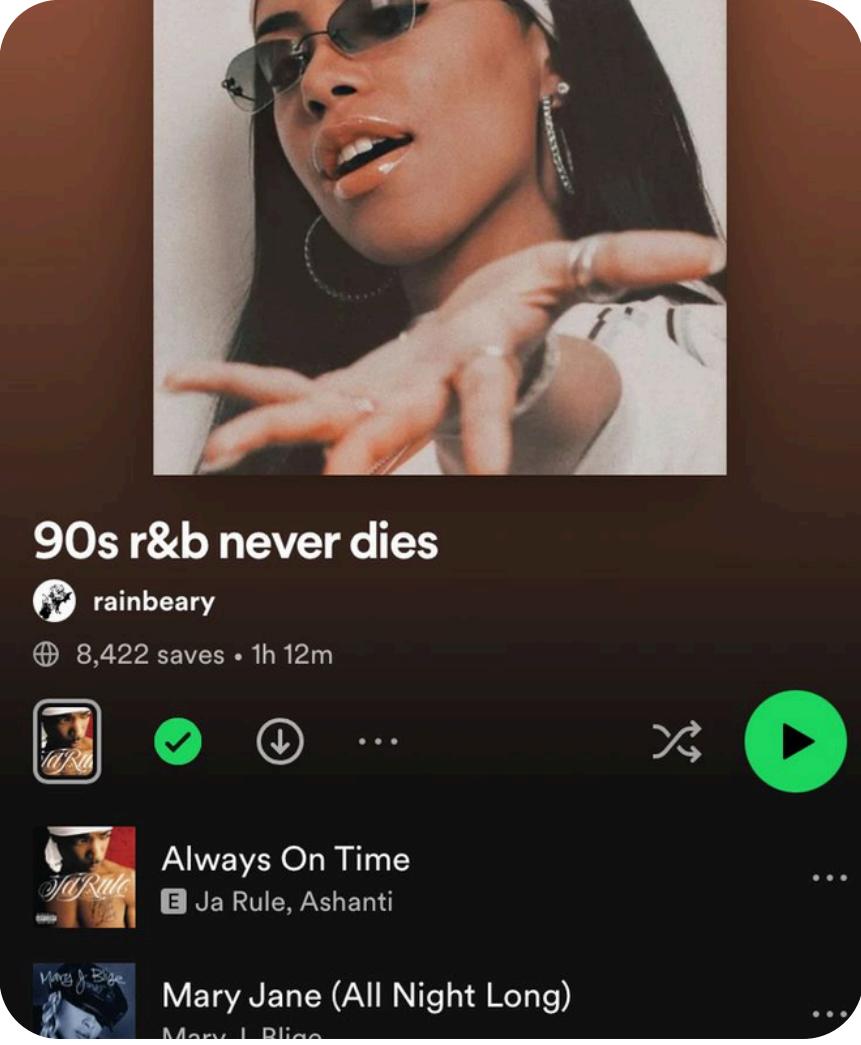


- Auraly is an intelligent mood-based music recommendation system that turns emotions into playlists.
- Auraly is a double entendre : 'Aura' for the vibe and 'Aural' for listening/hearing



EMOTIONAL GAP

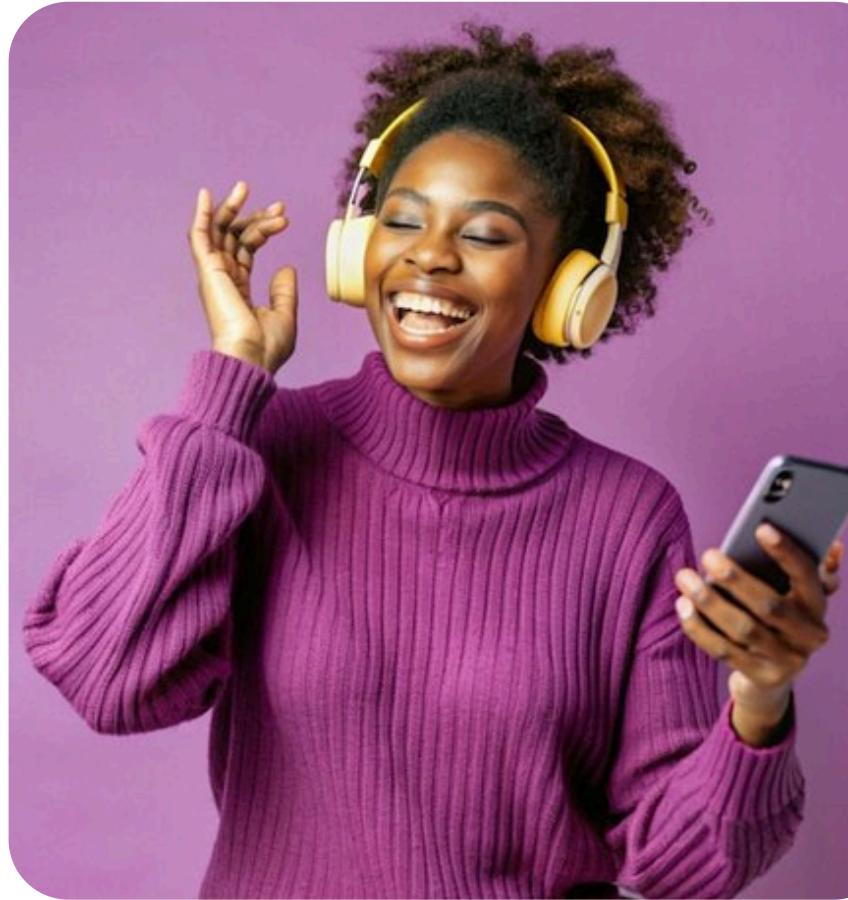
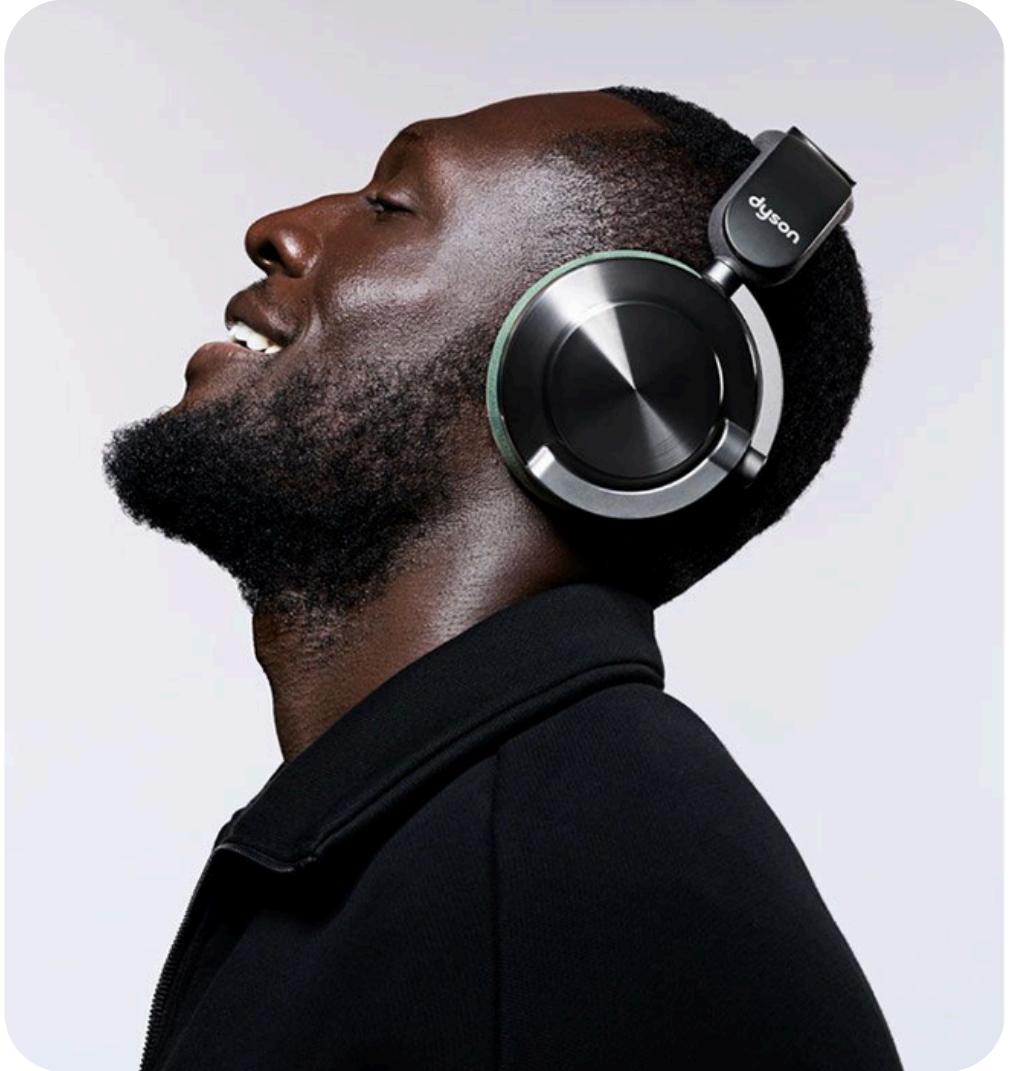
Music apps are aware of the music you listened to yesterday...



But they have no idea how you **feel** right now(your **mood**)

Why Does our Mood Matter?

Listeners choose music based on emotion — not just genre.
Mood drives engagement and retention.



Our Purpose in Harmony

Main Objective

Build an intelligent system that detects mood and curates personalized playlists.

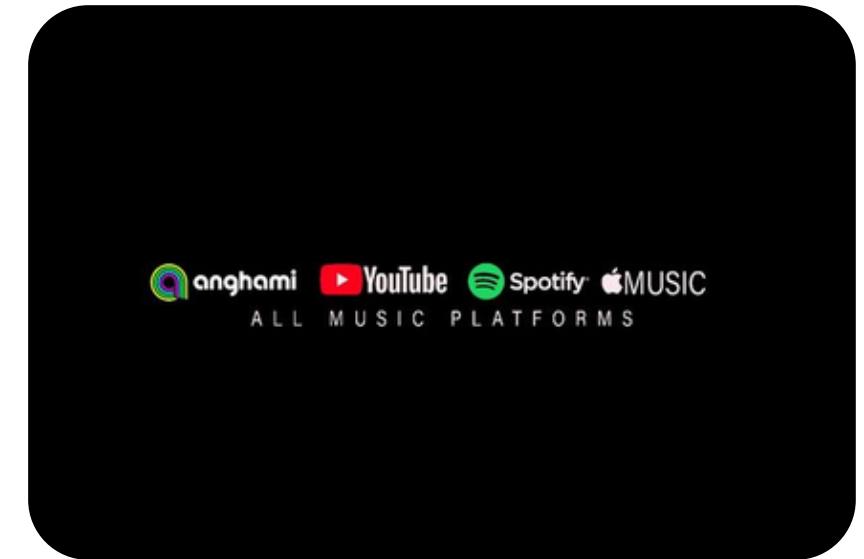
Specific Objectives

1. Automate playlist creation based on emotion.
2. Boost engagement through mood-driven recommendations.
3. Personalize music discovery experiences.



The Auraly Community

1. Listeners: Get playlists that match their mood.
2. Streaming platforms: Boost engagement with personalized music
3. DJs & Curators: Quickly find songs that fit the vibe

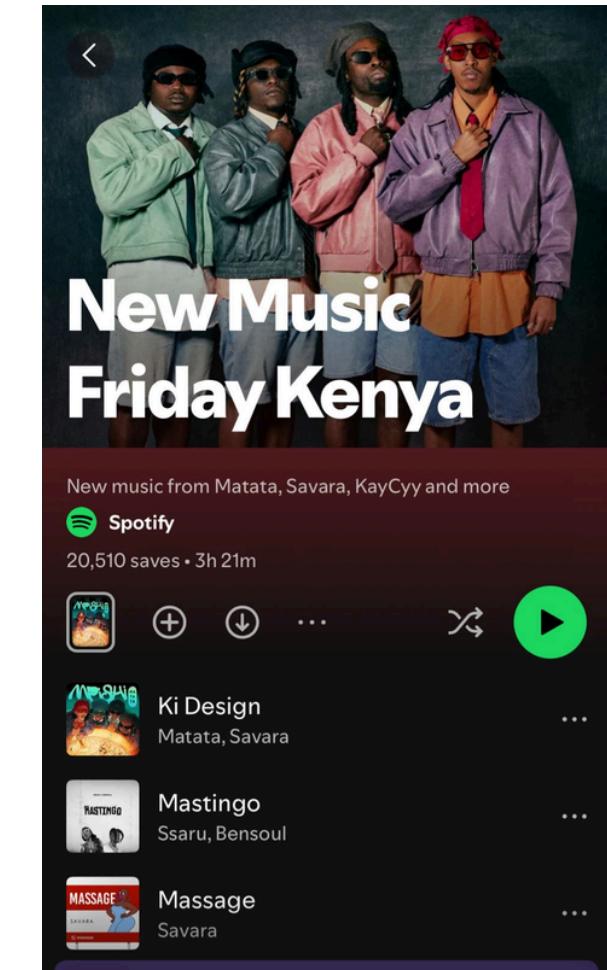
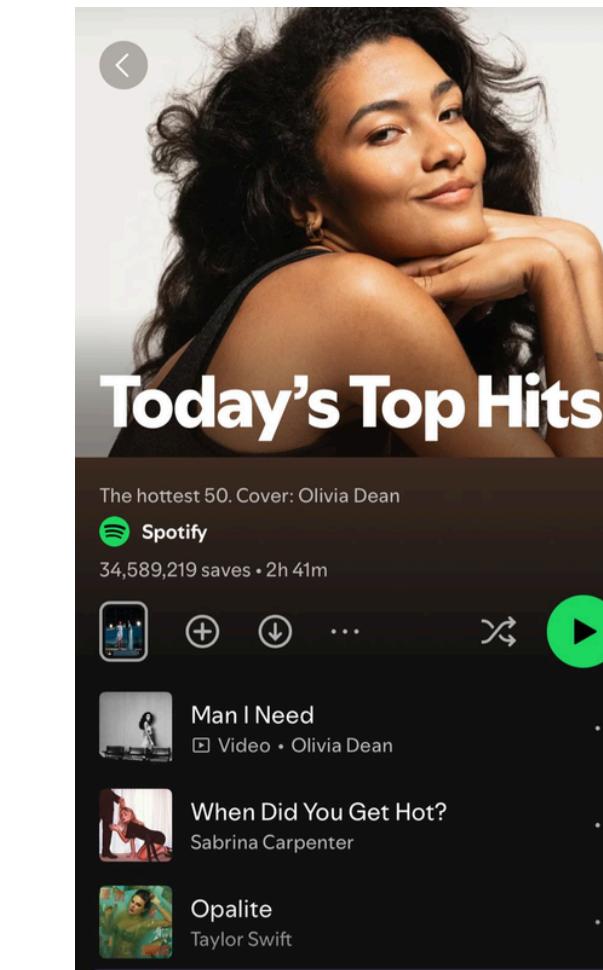
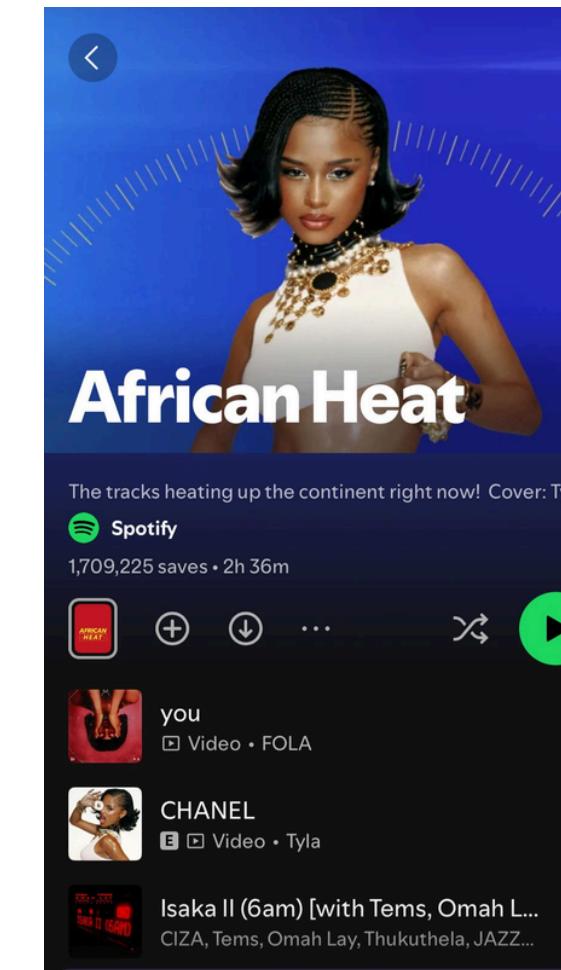
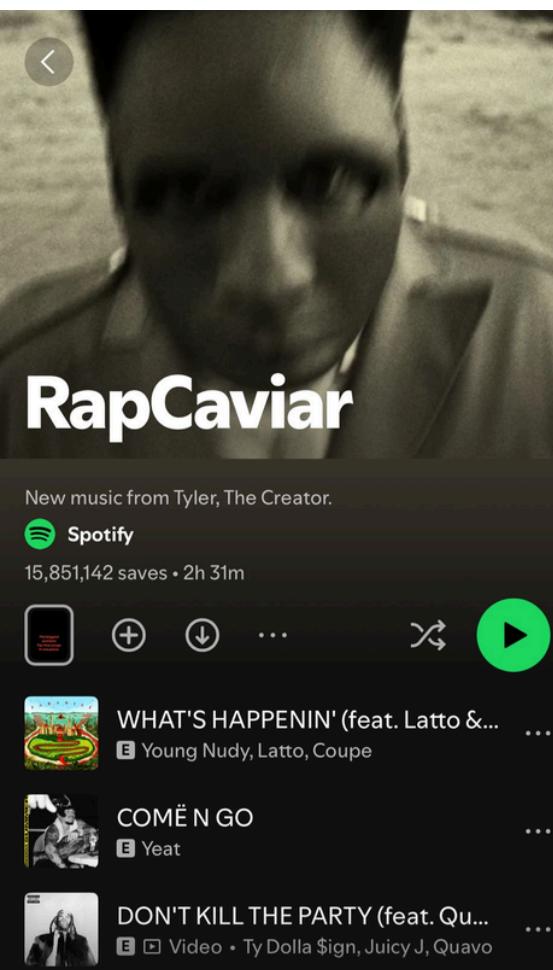
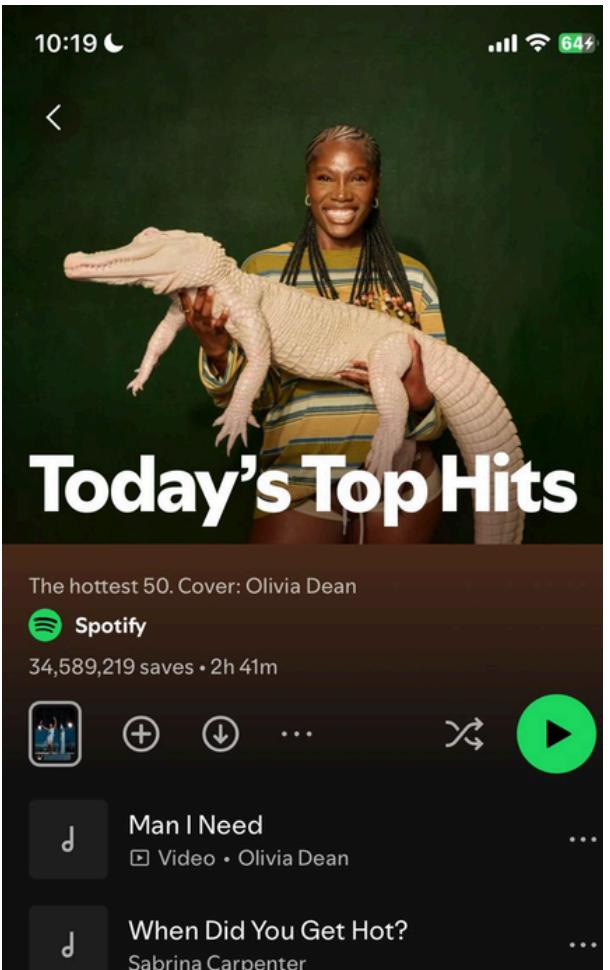


DATA SOURCES

1. Spotify Audio features (**Kaggle**)
2. Mood-labelled phrase Dataset (collected from friends and family)

Data that Feels Real

Kaggle provided reliable audio data and our custom mood phrases added the human touch — making Auraly's predictions more natural and emotionally authentic.



The Brains Behind The Beats ♪



Three notebooks powered Auraly's evolution :

1. Trained the mood prediction model
2. Cleaned Spotify data for playlist generation
3. Processed everyday language into tokens and tones,
giving emotion a data form

MOODS

The Machine Learning model predicts 4 moods:



0=SAD



1=HAPPY



2=ENERGETIC



3=CALM

The moods were predicted using Spotify's audio features:
Energy, Valence, Danceability, Loudness, Acousticness, Tempo, Instrumentalness

MOODS CLASSIFICATION MODELS



1. Logistic Regression Model-Baseline Model
2. Random Forest Tree Model-captured non-linear patterns
3. XGBoost Classifier Model (untuned & tuned models)

Key Feature Importances

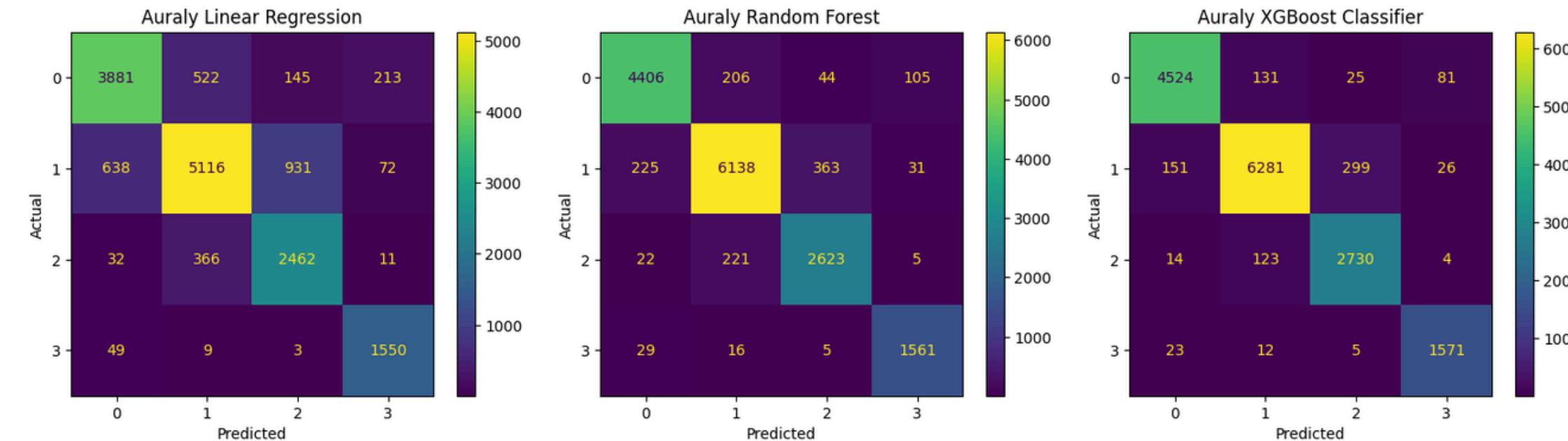
Instrumentalness, Energy, Acousticness and Valence were ranked(in order) as the top predictors



MODEL RESULTS

1. Logistic Regression Model=81.3% Accuracy
2. Random Forest Tree Model=92.1% Accuracy(an improvement)
3. XGBoost Classifier Model: a)**Untuned=94.4%(Highest accuracy)** b)Tuned=93.5%

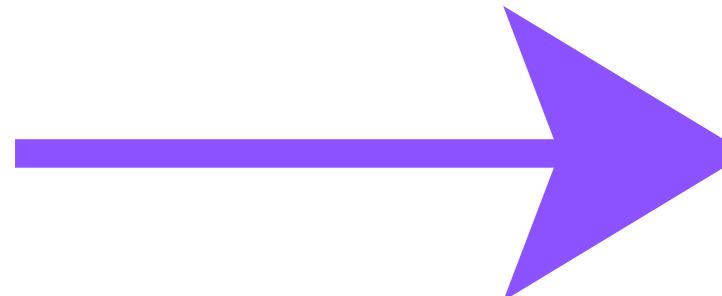
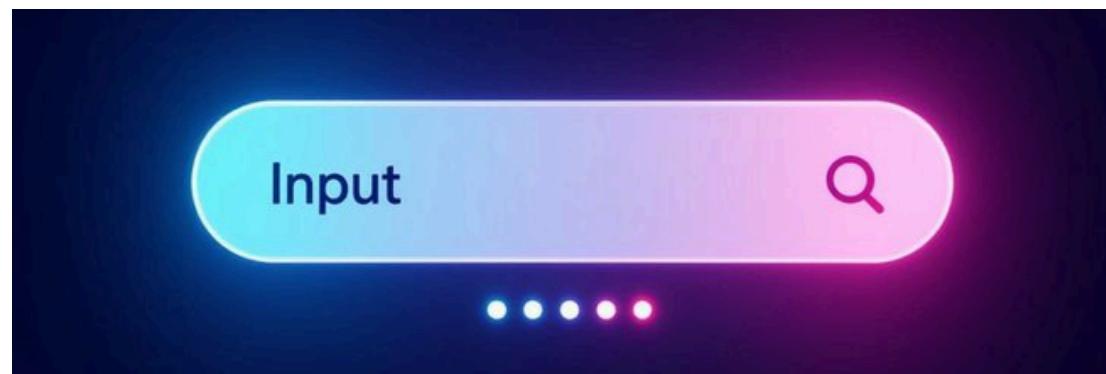
- The diagonal line(top left to bottom right) in the confusion matrix below shows correct predictions — it gets clearer from Logistic Regression to Random Forest, and sharpest with XGBoost, which reached over 94% accuracy. **This lead to XGBoost being selected as Auraly's final model.**



AURALY ENGINE OVERFLOW

The model follows these 4 simple processes:

1. USER INPUT : You type a short Phrase ("I feel good")
2. MOOD DETECTION : Auraly analyzes the phrase and predicts your mood
3. MUSIC SELECTION : The system finds songs that match the mood
4. PLAYLIST GENERATION : You get a personalized playlist instantly



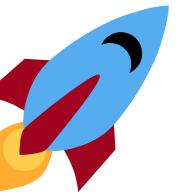
What we Learnt The Hard Way

- 1.Data restrictions made emotion-tagged music hard to collect.
- 2.Balancing mood labels required SMOTE and careful sampling.
- 3.Ambiguous emotions challenged our classifier boundaries.

Lesson: Emotion in Data Science is powerful, but messy



DEPLOYMENT



- Final code was deployed on GitHub
- Connected and launched via Streamlit Cloud
- A working prototype tested on 2,000+ mood-labeled tracks
 - Demonstrating a fully functional 'emotion-aware' playlist generator



Thank You !



Thank you for listening.

Reach out for any questions .

