BETTER SPOTIFY JAM

Trysten Giorgione

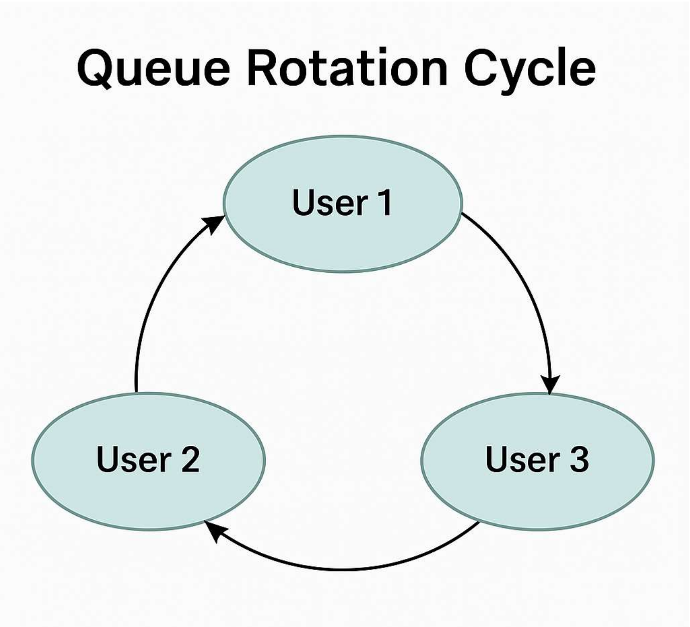
Advisor: Dr. Fred Annexstein

PROJECT GOALS

- Enhance Spotify’s 'Jam' feature to make group listening fair and memorable.
- Rotate songs between users instead of chronological order.
- Allow hosts full control (pause, skip, fallback playlist, etc.).
- Enable playlist saving at the end of sessions.

INTELLECTUAL MERITS

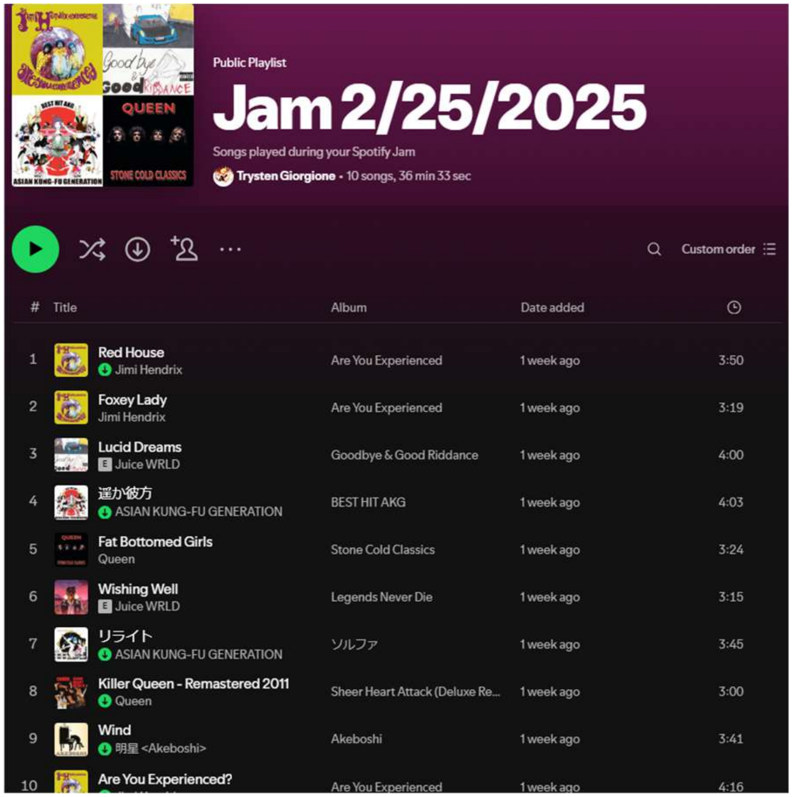
- Fair queueing algorithm for round-robin song playback.
- Integrated Spotify OAuth and Playback API.
- Guest participation without needing a token.
- Optional guest dashboard and semi-real-time views.



Note: Every user gets a single shot at a song being played per rotation. If nobody has a song in queue, a fallback playlist can be played if set

BROADER IMPACTS

- Encourages inclusive and fun music experiences.
- Creates a 'time capsule' by saving playlists.
- Prevents users from dominating the queue.
- Potential for gamification and badges in future.



Public Playlist

Jam 2/25/2025

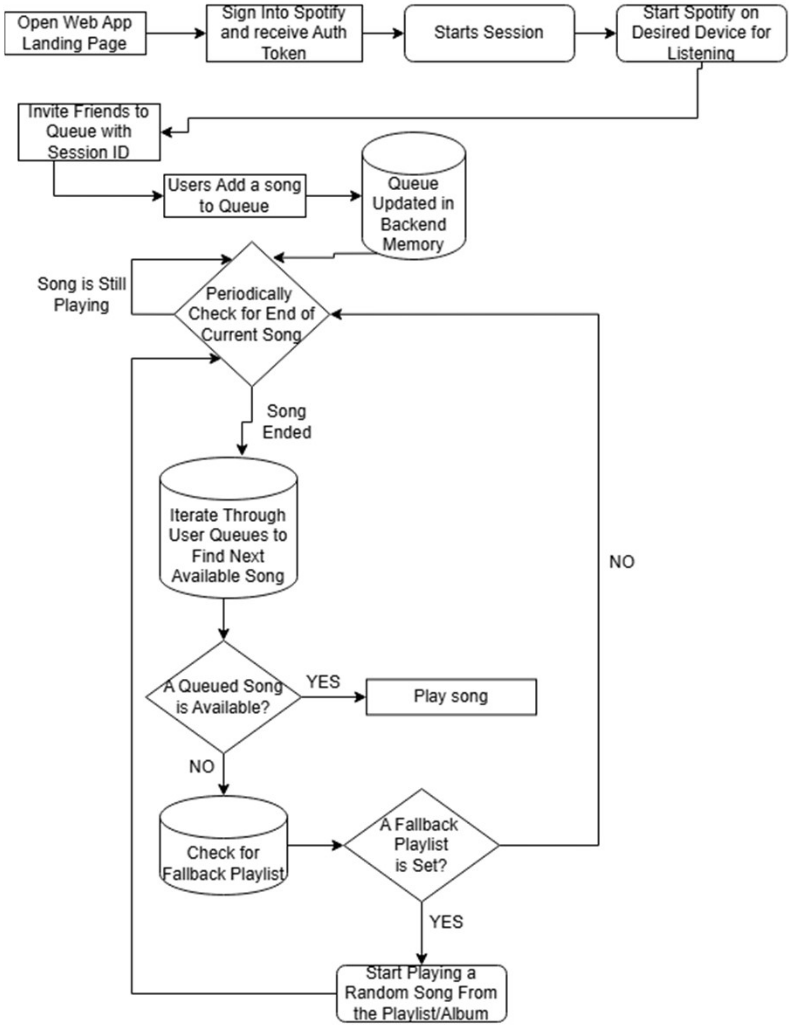
Songs played during your Spotify Jam

Trysten Giorgione • 10 songs, 36 min 33 sec

#	Title	Album	Date added	🕒
1	Red House Jimi Hendrix	Are You Experienced	1 week ago	3:50
2	Foxey Lady Jimi Hendrix	Are You Experienced	1 week ago	3:19
3	Lucid Dreams E Juice WRLD	Goodbye & Good Riddance	1 week ago	4:00
4	遥か彼方 ASIAN KUNG-FU GENERATION	BEST HIT AKG	1 week ago	4:03
5	Fat Bottomed Girls Queen	Stone Cold Classics	1 week ago	3:24
6	Wishing Well E Juice WRLD	Legends Never Die	1 week ago	3:15
7	リライト ASIAN KUNG-FU GENERATION	ソルファ	1 week ago	3:45
8	Killer Queen - Remastered 2011 Queen	Sheer Heart Attack (Deluxe Re...	1 week ago	3:00
9	Wind 明星 <Akeboshi>	Akeboshi	1 week ago	3:41
10	Are You Experienced?	Are You Experienced	1 week ago	4:16

DESIGN SPECIFICATIONS

- Host login, session creation, and queue logic managed by backend.
- Sessions store tokens, guest list, rotated queue, and fallback playlist info.
- Guest access via simple UI without needing Spotify login.
- Guest dashboard shows rotation and history (optional).



Open Web App Landing Page

Sign Into Spotify and receive Auth Token

Starts Session

Start Spotify on Desired Device for Listening

Invite Friends to Queue with Session ID

Users Add a song to Queue

Queue Updated in Backend Memory

Song is Still Playing

Periodically Check for End of Current Song

Song Ended

Iterate Through User Queues to Find Next Available Song

A Queued Song is Available?

YES

Play song

NO

Check for Fallback Playlist

A Fallback Playlist is Set?

NO

YES

Start Playing a Random Song From the Playlist/Album

TECHNOLOGIES USED

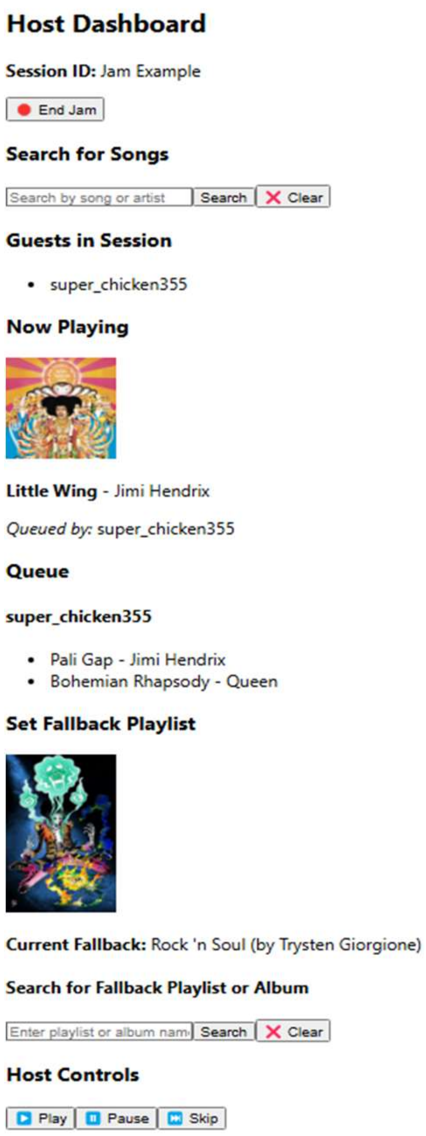
- Backend: Node.js, Express, Spotify Web API.
- Frontend: React.js, Axios, React Router.
- Spotify features: OAuth, search, playback, playlists.
- Optional WebSocket support for real-time updates.

PROJECT MILESTONES

- Fall 2024: Brainstorming, and Ideation of design.
- Winter 2025: Queue algo implementation ideation, UI layout drafting, reasonable achievements goals (fallback playlist, session mgmt., fair queue, usable UI)
- Spring 2025: Frontend/Backend dev, Spotify API integration, Testing.
- Final Week: UI polish, Presentation / Expo prep and testing.

RESULTS AND PROGRESS

- Rotating queue system completed.
- Host authentication and session handling work.
- End Jam saves session playlist.
- Fallback playlists work for albums and playlists.
- Guest dashboard implemented.
- Remaining: polish UI, optional real-time updates.



Host Dashboard

Session ID: Jam Example

Search for Songs

Guests in Session

- super_chicken355

Now Playing

Little Wing - Jimi Hendrix

Queued by: super_chicken355

Queue

super_chicken355

- Pali Gap - Jimi Hendrix
- Bohemian Rhapsody - Queen

Set Fallback Playlist

Current Fallback: Rock 'n Soul (by Trysten Giorgione)

Search for Fallback Playlist or Album

Host Controls

CHALLENGES AND SOLUTIONS

- Spotify Authentication: managed persistent tokens.
- Fair Queueing: custom logic ensured round-robin fairness.
- Guest Access: allowed without login via backend logic.
- Real-time Updates: optional stretch feature handled gracefully.
- Learned full-stack tools and overcame tech unfamiliarity.