

Rubato



Gio Jacuzzi, Parker Odrich, Peter Washington
Assignment 6: Medium-Fi Prototype
Studio: Creation

Value Proposition and Solutions

Value Proposition: Communal playlists in real time.

Problem: Music is a dynamic force that brings communities together, but there is currently no way for all listeners to engage with each other in the creation process in real time--together, right here, right now.

Solution: Rubato allows everyone at an event to contribute in real time to the music. The result is a constantly evolving playlist, curated by the entire community.

Representative Tasks

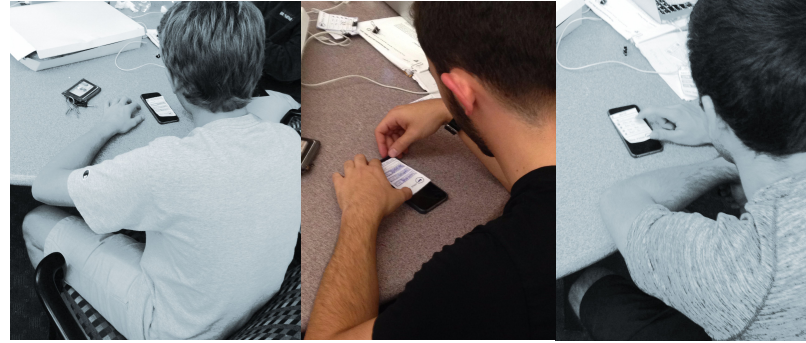
Task 1: Create a communal playlist (easy).

Task 2: Update the playlist in real-time (medium).

Task 3: Play songs from any event attended in the past.

Problems Found in User Testing

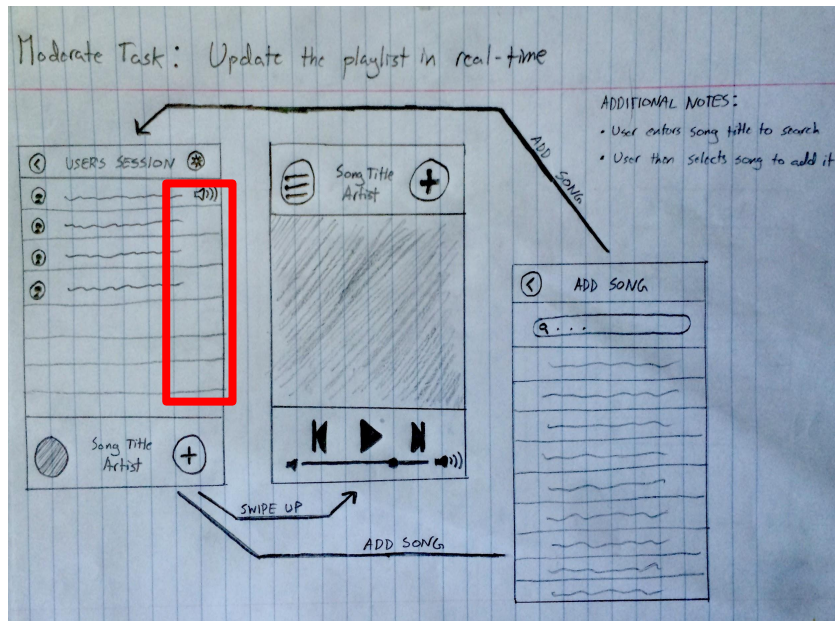
- 1) Users played songs with “malicious” intents.
- 2) Users found the interface to be too confusing due to the over-complexity of the design.
- 3) The meanings of buttons were ambiguous (specifically for settings and adding songs).



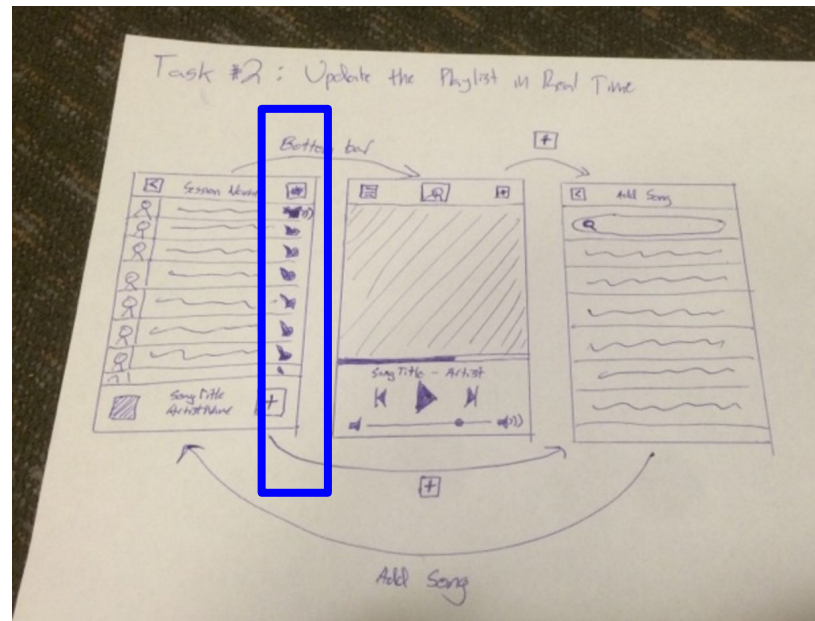
Design Change 1: Add Upvoting To Songs

- Upvoting prevents a single user from hijacking the playlist.
- Adds more user interactivity to the party.
- Doesn't require excessive interaction during the party.

Design Change 1: Add Upvoting To Songs



Before

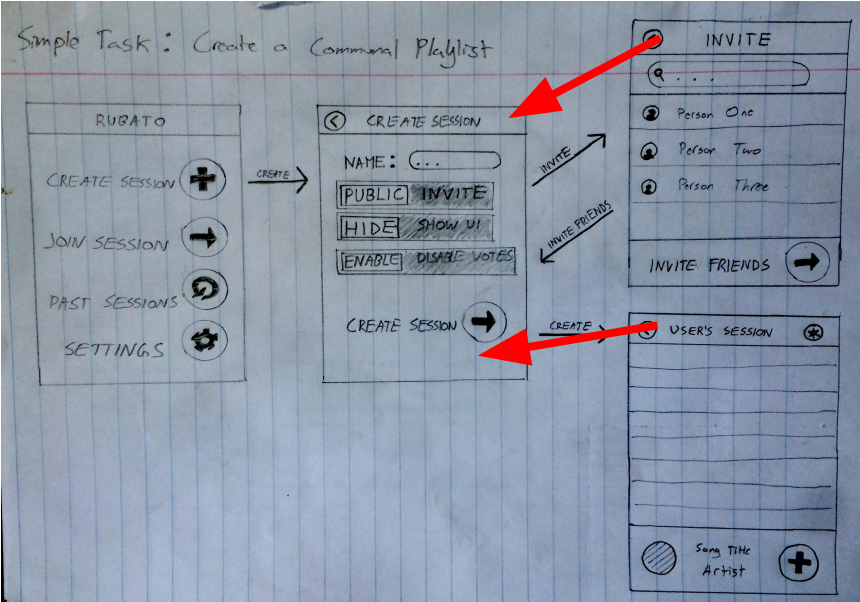


After

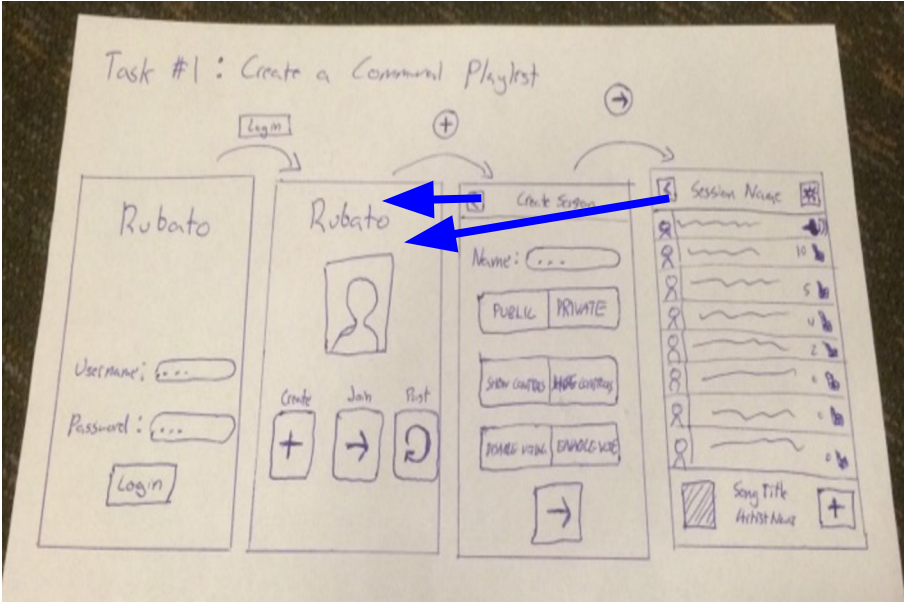
Design Change 2: Centralized Home Screen

- A central screen makes it easier for user to navigate throughout the app.
- Prevents the user from being “stuck” in a certain screen.
- Prevents the user from clicking an unnecessary number of buttons and navigating through many screens.

Design Change 2: Centralized Home Screen



Before

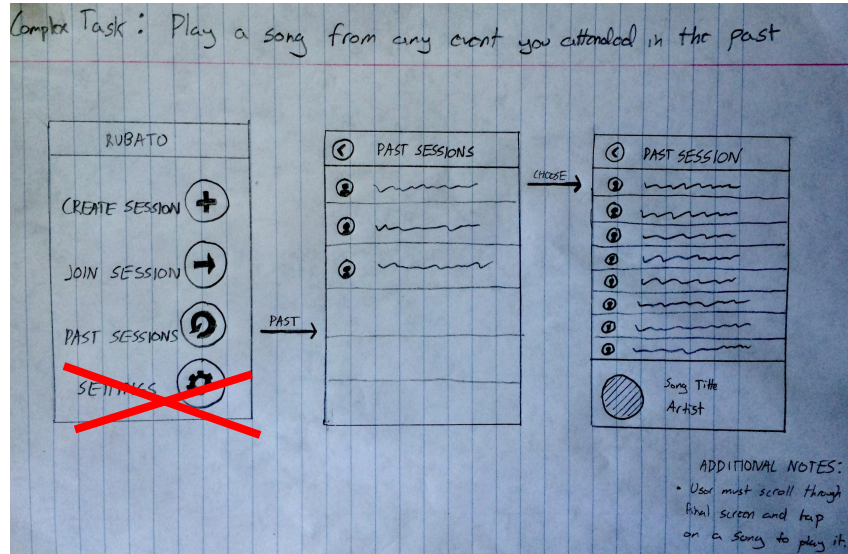


After

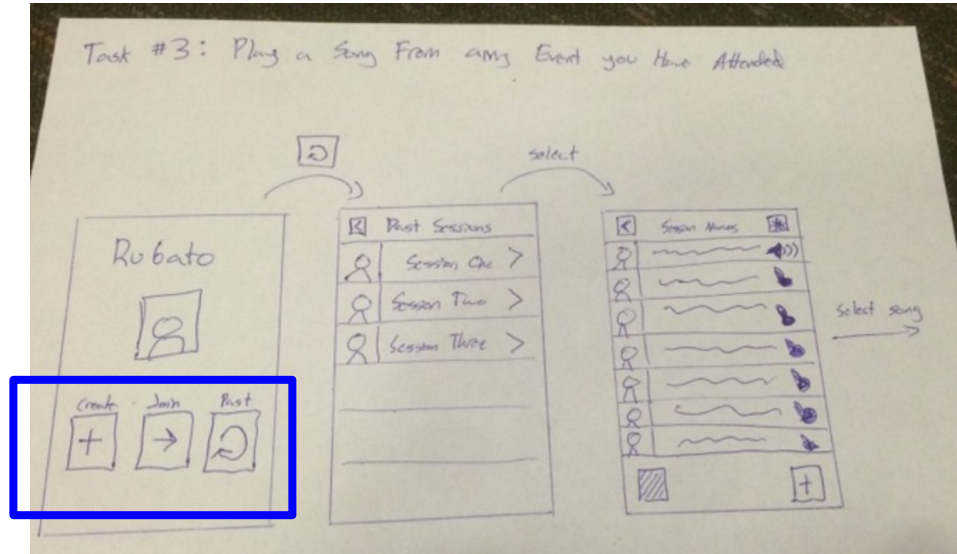
Design Change 3: Remove Global Settings

- Global app settings were largely irrelevant to the app.
- Settings were per-session and could just be done during session create-time.
- Settings button clogged the home screen. Now, have a **much prettier home screen.**

Design Change 3: Remove Global Settings

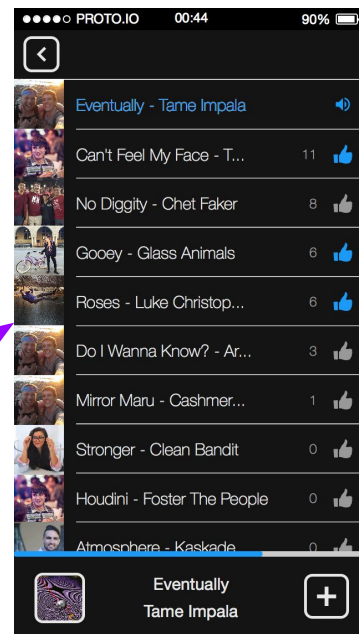
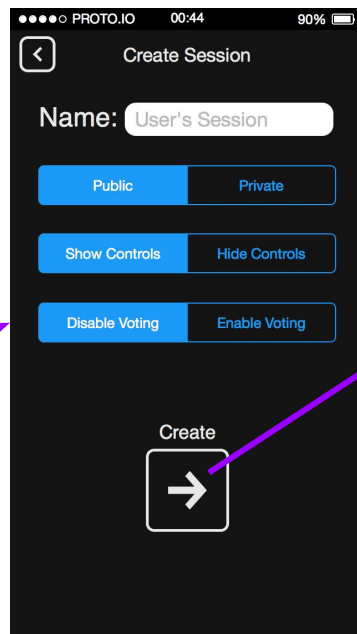
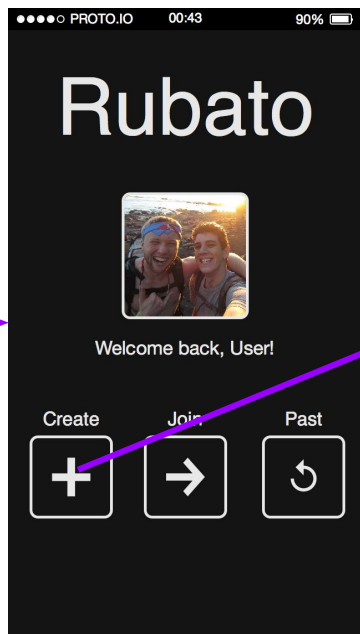
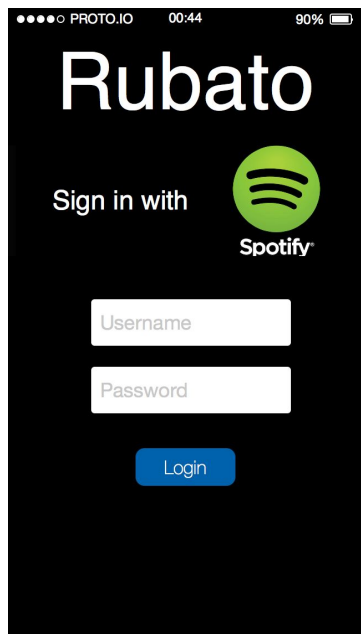


Before

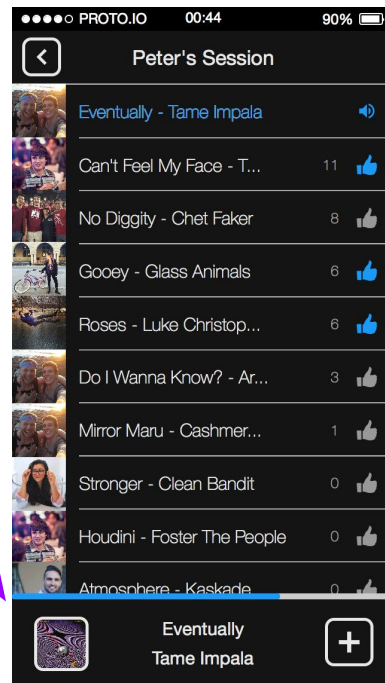
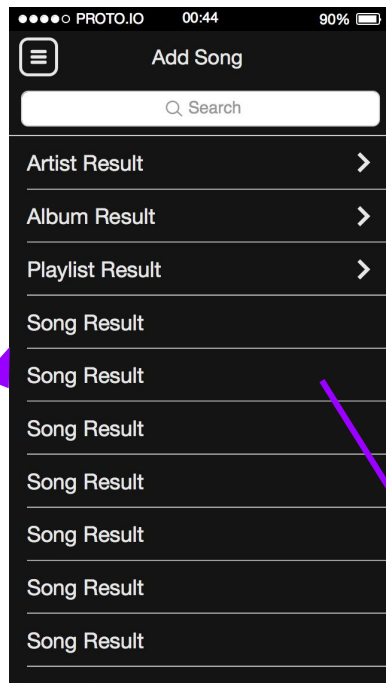
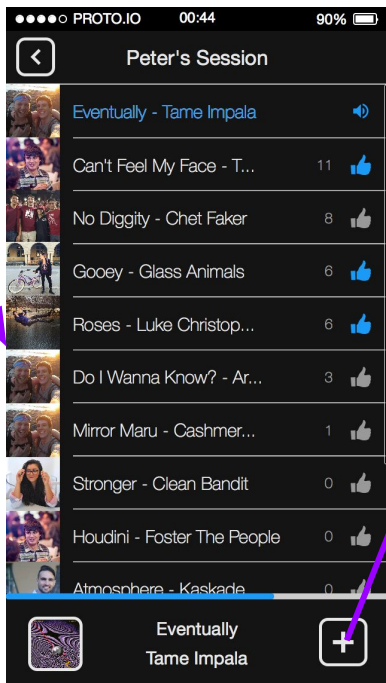
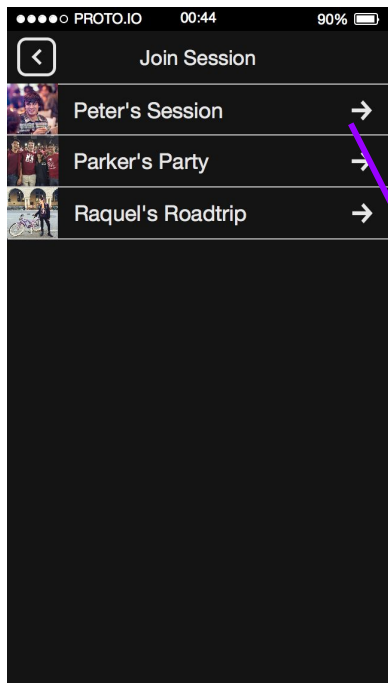


After

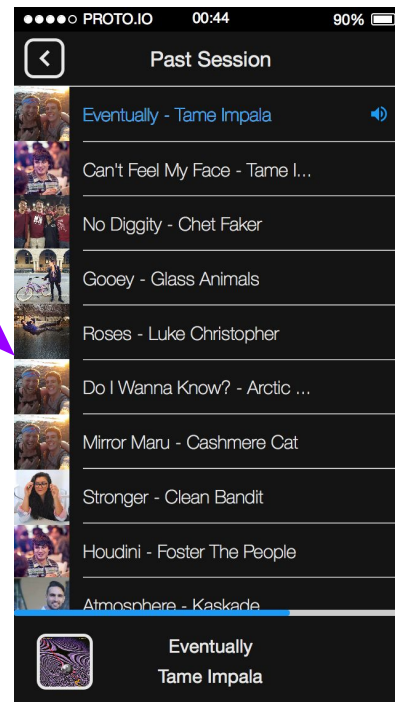
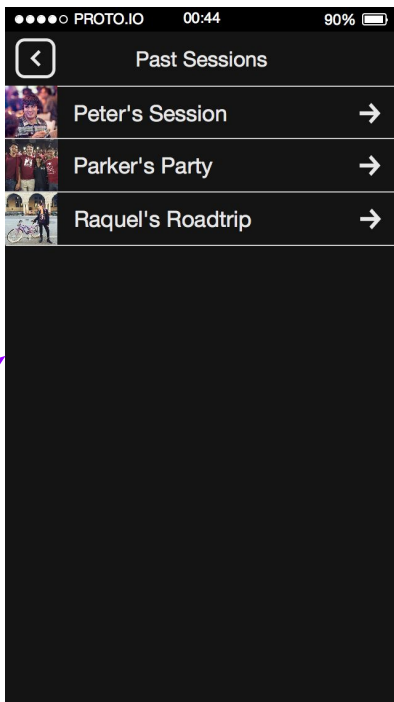
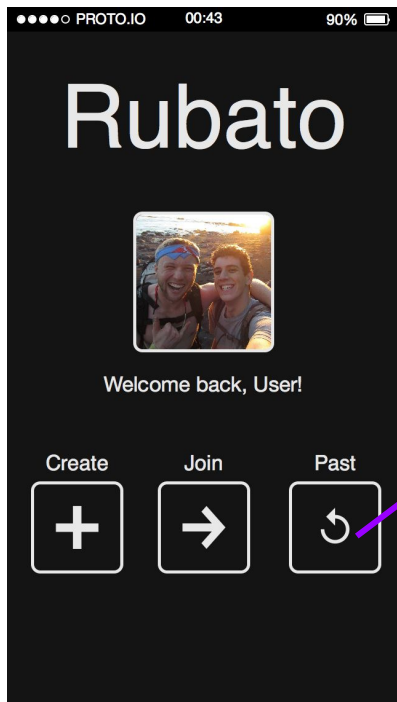
Task Flow #1: Create a Rubato Session



Task Flow #2: Update playlist in real time



Task Flow #3: Relive the playlist



Prototyping Tool

proto.io

- Used Proto.IO
- Makes iOS prototypes that seem real
- Lots of features
- Free trial

Limitations of Proto.IO

- No “conditional” logic prevented many simple prototyping features from being added
 - Selected songs couldn’t actually be added to playlist
 - The results of upvoting a song can’t be seen
- Long learning curve compared to Marvel and InVision

Wizard of Oz Techniques

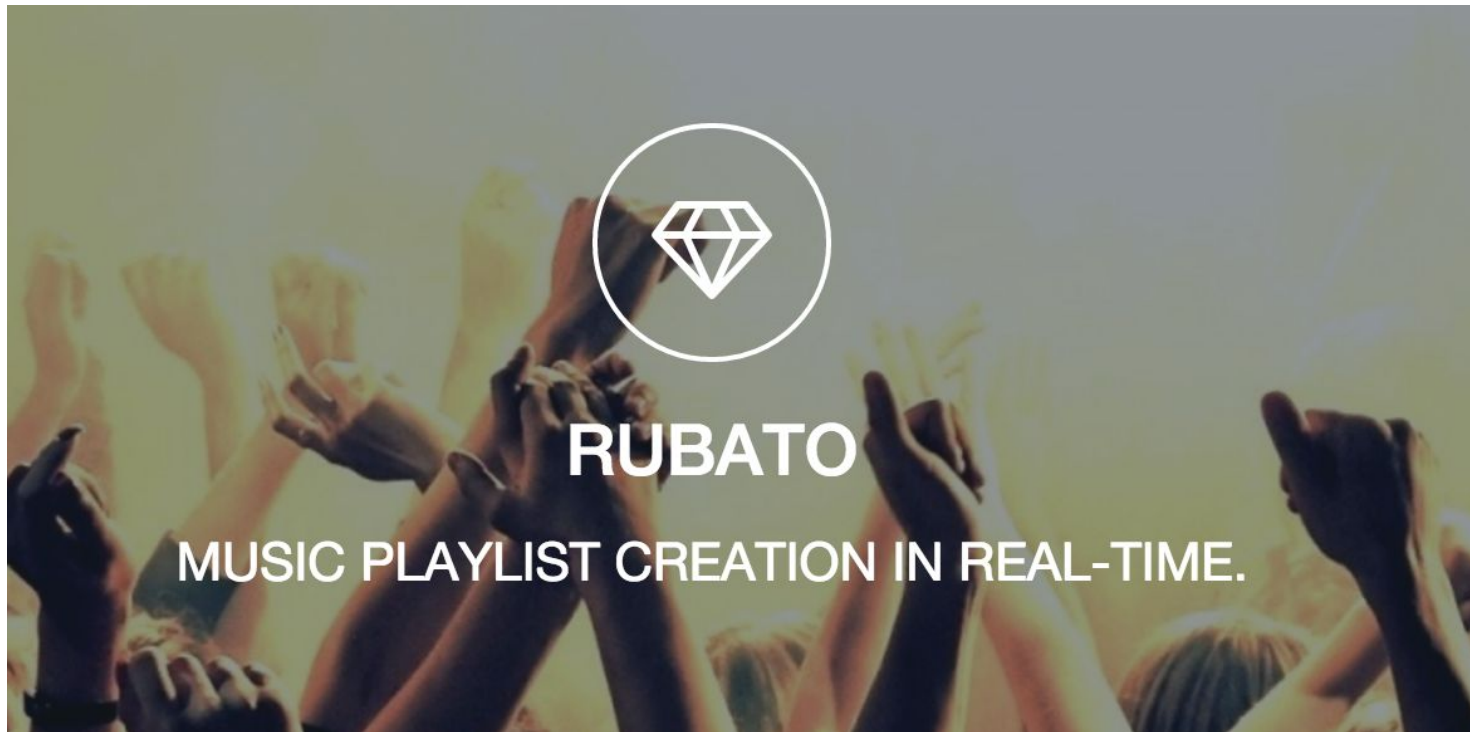
- When a user wants to add a song, we will tell them that the song is added to the playlist.
- When a user wants to play a song, we will play the song ourselves. The prototype cannot play songs.



Hand-Coded Features

- Song names
- Session names
- User profile
- Number of likes per song

Questions?



Website Link

<http://web.stanford.edu/class/cs147/projects/creation/Rubato/index.html>