

DJQ App Design

Sergejs Skusovs

Project overview



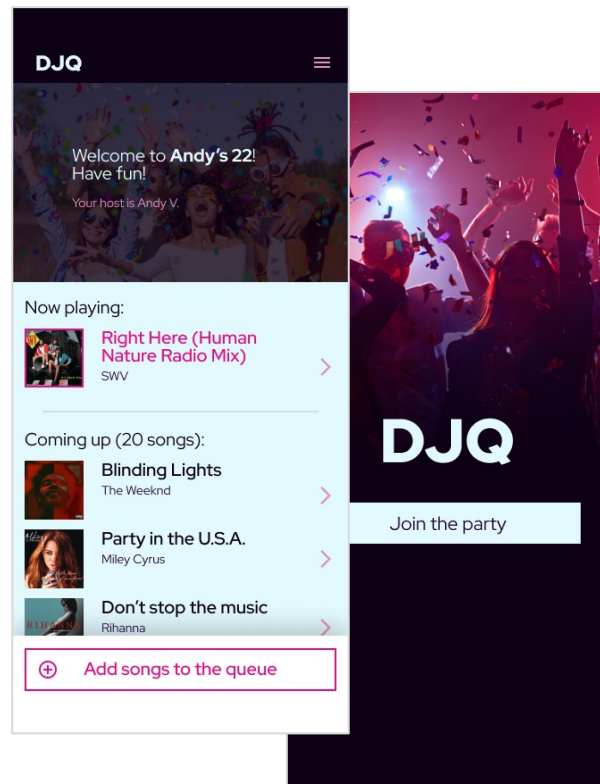
The product:

DJQ is a music queueing app for parties that allows the party goers to manage their party's music playlist: adding songs from their personal libraries, as well as up- and downvoting the songs.



Project duration:

April 2020 to July 2021



Project overview



The problem:

Party goers having no transparent and fair way to affect what music is playing on the parties and no way to affect.



The goal:

Give the party goers a tool to manage good process for making sure their favourite songs are playing and let the host focus on partying more than organising the music

Project overview



My role:

UX designer for the app



Responsibilities:

Conducting interviews, paper and digital wireframing, low and high-fidelity prototyping, conducting usability studies, accounting for accessibility, and iterating on designs.

Understanding the user

- User research
- Personas
- Problem statements
- User journey maps

User research: summary



I conducted interviews and created journey and empathy maps, defining primary user groups as young students, 19-25 years old who go to parties but are frustrated about having to listen to music they don't enjoy without a proper non-conflict way to share their music with others.

This user group confirmed initial assumptions about the target audience, but research also revealed that the problems are not specific to just party goers – the app could also solve some of the host's problems

User research: pain points

1

Ease of use

There are no good ways to crowd source the creation of party playlists while on a party

2

Different services

Users use different services like Apple Music, Spotify, Pandora which makes creating party playlists cumbersome

3

Control

Users don't just want to share their music, they want to be able to control the order of the playlist

4

Share

Users would like to keep the music they hear on parties, to add it to their own libraries

Persona: **Andy**

Problem statement:

Andy is a student party-goer who needs an easy way to make sure he can listen to his favorite music on the party because he likes having fun to his playlist.



Andy

Age: 19

Education: University student

Hometown: Rīga, Latvia

Family: Lives with parents

Occupation: Student

*“I like to do everything to my favourite music.
Even partying”*

Goals

- Listening to his favourite music on the parties
- Make sure that everybody else has their own share of fun, including himself

Frustrations

- “Some parties have just the most awful music ”
- “People keep trying to get their playlists playing when my favourite song isn't finished yet”

Andy is a hard-working student, studying Rocket Science at the University. Music is his second most important passion in life, he does everything listening to his playlist. He doesn't have much free time, but when he does, he likes to go party with his friends, often being frustrated by the lack of a good process for everybody to get their music played during the party for fair share of time.

User journey map

Mapping Andy's user journey revealed how helpful would the DJQ app be to a user with similar needs

Persona: Andy, the party goer

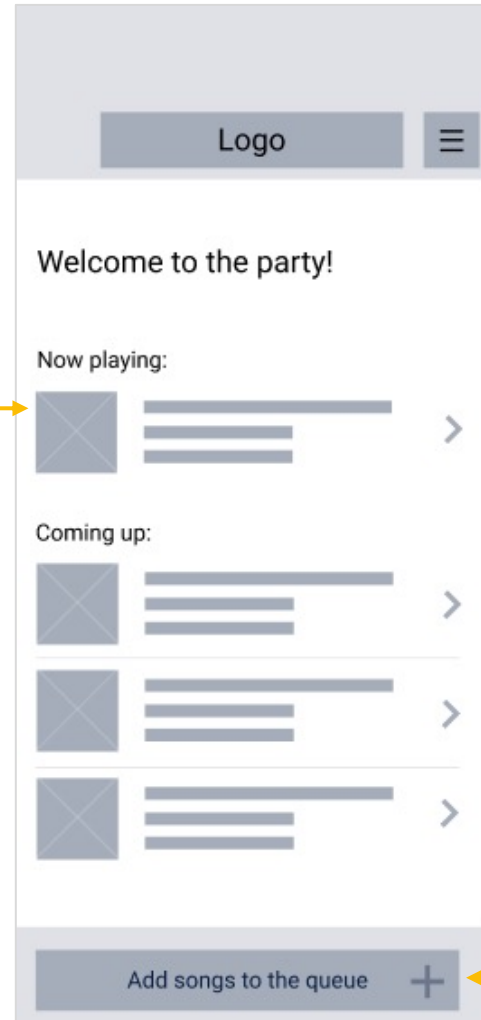
Goal: Making sure his favourite song get played during the party

ACTION	Get App	Review current playlist queue	Submit songs	Vote for songs	Add songs from the party que to a private playlist
TASK LIST	Tasks A. Scan QR code B. Download app C. Connect to the party entering a code	Tasks A. Connect to the party B. Review the list of the songs in the party's queue	Tasks A. Choose an option to add a song B. Select a song to add to the party's queue	Tasks A. Review the list to find songs you like B. Vote the songs you like up	Tasks A. Check the current song playing B. Choose to add it to private playlist
FEELING ADJECTIVE	Excited to Frustrated by making mistakes in entering the code	Surprised by how many songs he likes are already in the playlist	Frustrated about having to add songs one by one	Happy to be able to make his favourite songs play sooner in the queue	Excited to add new songs he likes to his own collection
IMPROVEMENT OPPORTUNITIES	Offer options to connect to the party, using party's unique QR code; Construct code using non-similar characters (excluding "0" and "O", lower case "L" and upper case "I", etc)	Add ability to see not only upcoming songs, but also recently played ones in the list to avoid unintentional duplicates	Add functionality to add multiple songs in one go	Add the functionality to "campaign" songs for party mates by sharing voting suggestion to the	Add ability to find and add the song to popular music streaming services (Apple Music, Spotify, Pandora, etc.)

Digital wireframes

As the initial design phase continued, I made sure to base screen designs on feedback and findings from the user research.

The main interest points are available right on the first screen: the current playlist showing users what's playing and what's coming next

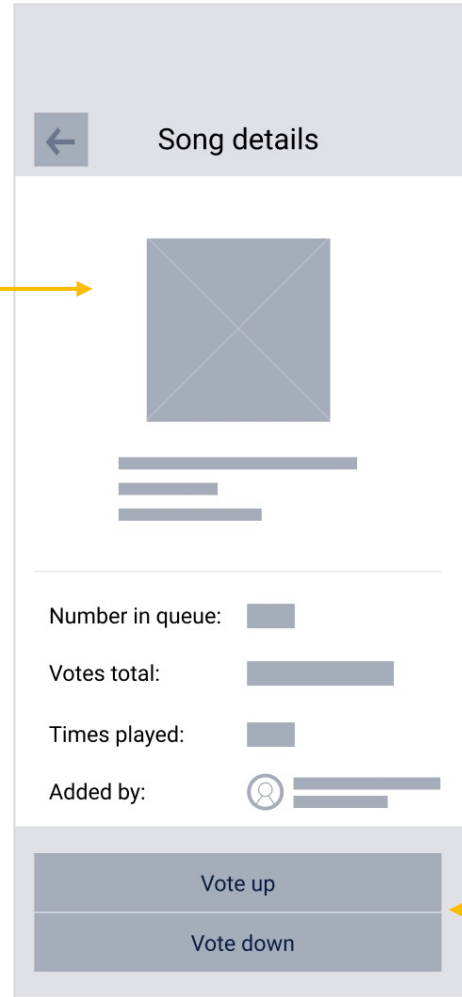


The main call to action button is always visible and ready to be interacted with

Digital wireframes

Continuing the user research it became clear that just being able to add songs to the playlist was not enough – users needed a way to sort the playlist according to their likes and dislikes fairly

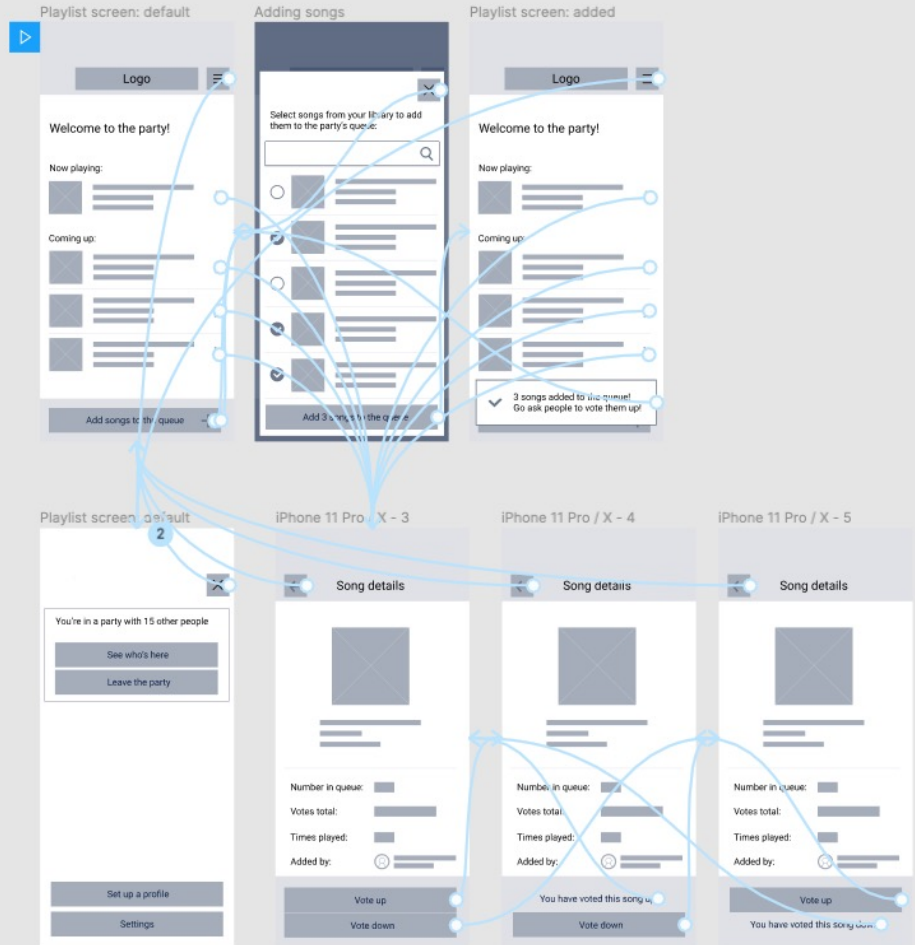
Single song context was added to encompass all the information about the song



Also adding voting buttons to make sure people could affect the playlist order

Low-fidelity prototype

Using the wireframes, I created a [low-fidelity prototype in Figma](#), including the main user-flows: adding songs to the queue and resorting the queue by upvoting and downvoting the songs



Usability study: findings

One round of usability study studies was conducted. Finding helped inform the weaker and confusing points in the design and let me to update it

Findings

- 1 Users need better way to discover what the buttons do
- 2 Users need more context, labels or other cues about available functionality
- 3 Users need more accessible and unified language in the app's labels

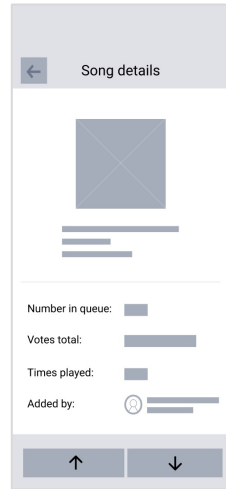
Refining the design

- Mockups
- High-fidelity prototype
- Accessibility

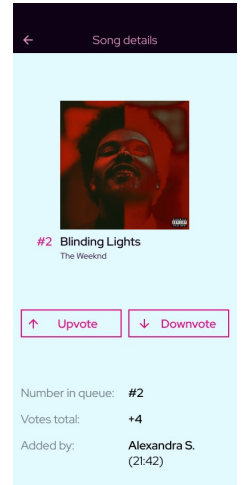
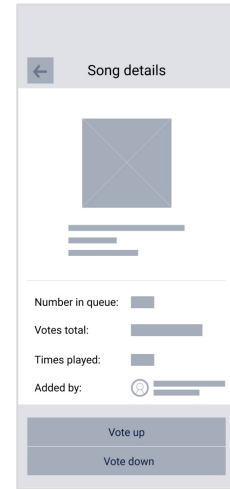
Mockups

Early designs included buttons that lacked comprehensive descriptions on what they did, which was made clear during the usability study. I've also switched the order of the buttons and detailed info blocks to make buttons more accessible

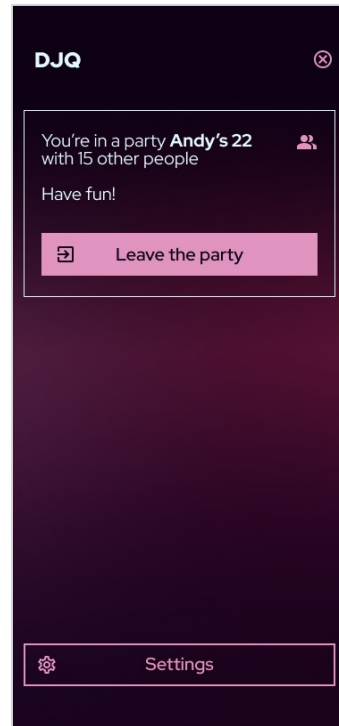
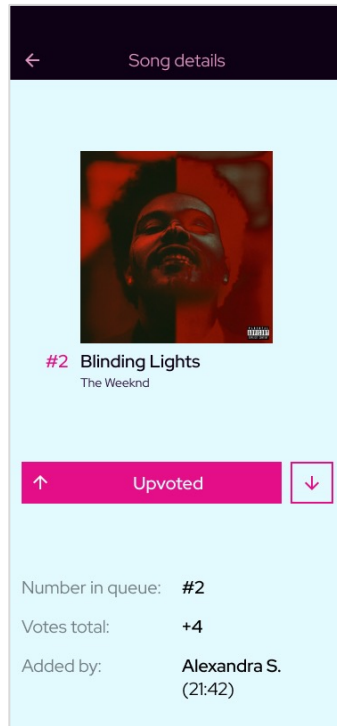
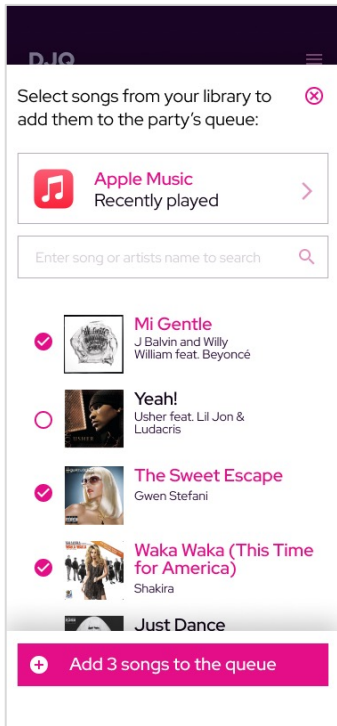
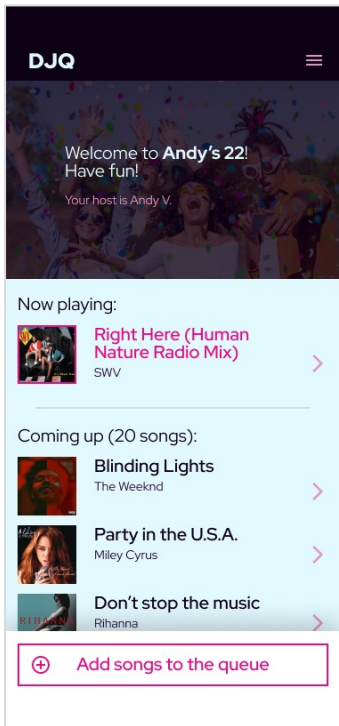
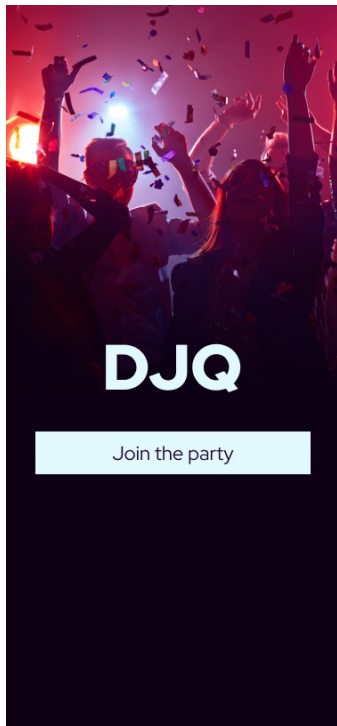
Before usability study



After usability study

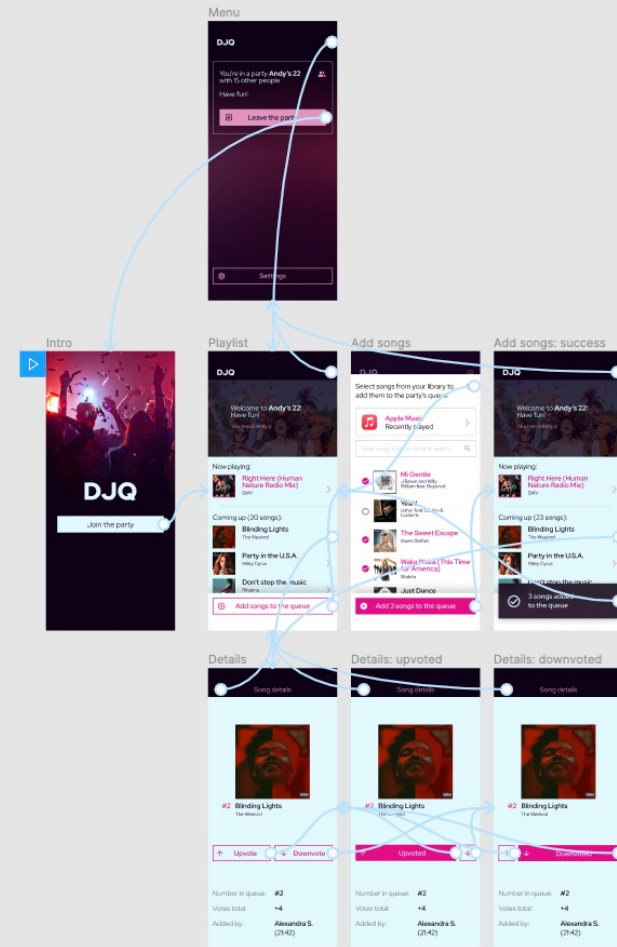


Mockups



High-fidelity prototype

The final [high-fidelity prototype in Figma](#) presented refined user flows with unified styles, colors, typography and iconography



Accessibility considerations

1

Used icons to make it easier to find and use the functionality of the app

2

All color combinations are WCAG 2.0 compliant

3

Used proper album art for songs to make it easier for users to recognize their favorite songs in the queue

Going forward

- Takeaways
- Next steps

Takeaways



Impact:

The app makes users more in control of the party's playlist and helps them achieve their goals



What I learned:

While designing the app I learned that designer's bias could be a real problem for the usability and user studies help inform better designs.

Next steps

1

Perform new rounds of usability studies, this time based on more refined high-fidelity prototypes

2

Conduct more user research to find new ways to improve the app and solve more user problems

3

Conduct more thorough competition audit to see what other areas can the designs be improved

Let's connect!



Thank you for your time reviewing my work on DJQ app!
Special thanks to the participants of the usability studies.
If you'd like to see more or get in touch, my contact information is provided below.

Email: sergejs.skusovs@gmail.com

Website: <https://expeon.lv/>

Thank you!