2012/03/28

Deanna Zhu dwz
Antonio Ono aono
Norbert Chu nrchu

Multi-screen DJ app

Introduction

Our goal is to make a DJing app that can be used simultaneously across multiple devices. In general, there are very currently very few apps that utilize multiple screens to function. However, DJ apps would greatly benefit from being shared across multiple screens. Desktop DJ apps are feature-rich but offer a less intuitive and tactile experience for users. Tablet apps are more ergonomically appropriate for their context but, given constraints in screen space, typically offer a limited subset of features. By modularizing components of a traditional digital or analog DJ setup, we can distribute the interface across a variety of available devices — from handsets to desktops — allowing for both a large number of controls and a tactile and intuitive user experience. Our goal is for the app to be accessible to both novice and experienced users, on multiple devices or only one.

Interface / client

We plan to design and develop a library of modular interface components that can be mixed and matched across one or more devices. Users will be able to arrange their interface based on their needs and the number of available devices. Devices can be assigned to roles that will include a template of related interface components, but users will retain the ability to change the roles and components as needed. We plan to use canvas to implement waveforms and potentially sound visualizations.

While the app can be controlled simultaneously from multiple clients, the audio will only be output from one device. Users can connect devices to each other by either logging in or scanning a QR code unique to their account from a device they are already logged into.

Server-side

Users should be able to import audio from services like Soundcloud via their API on any device, as well as directly upload audio from their laptop or desktop. They can then preserve those songs in playlists, which may also offer the ability to suggest related songs by folksonomy, BPM, or other metrics.

On first run, the server will assign interface components to the user's devices based on the number of devices and their screen sizes. The server will keep track of which devices the user is using at any given time. Sockets will listen for changes and the server will process and apply the changes to the audio output.

Primary Goals

- Log-in and connect devices for a session using QR codes

- Stream music from database that the user loads via desktop/laptop, stream music from SoundCloud
- Have basic control modules (volume, playback rate, play/pause)
- Waveform + sound visualization
- Server calculation of interface layout
- Customization option for modules

Secondary Goals

- Finding songs with similar BPM or other advanced metrics
- More advanced controls (pitch,...)
- Sound visualization
- Recording a set

Schedule

4/2- Final proposal, Complete Competitive Analysis, Storyboard due
4/5- Design planning done, start (if not started already) coding server stuff and front end
4/6- working interaction with client/server sockets, basic module designs + interface setup
4/8- working static server, log-in working with QR code scanning to connect devices, music streaming tested
4/9- Full design done, technology demos, “shippable code”
4/13- Customization of modules/device option fully implemented
4/16- Have a testable product
4/23- User Studies + Iterative Design
4/30- Turn in project. Celebrate with wings.