



BeatBeam FPR SDP15

Duncan Smith-Freedman, EE

Brandon Sprague, CSE

Brian Hickey, CSE

Daniel Bond, CSE

BeatBeam Team



Daniel Bond
(Fearless) Team Leader

Board Interpretation,
Musical Implementation

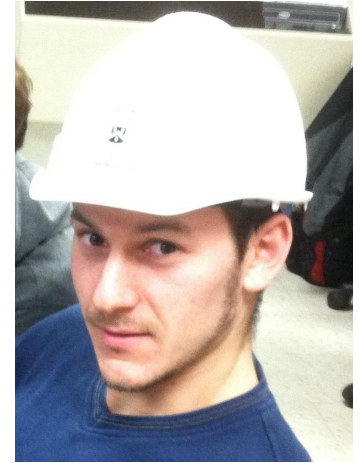


Brian Hickey

Networking, Wireless
Connectivity,
Audio/Acoustics, Power



Duncan Smith-
Freedman
Audio-to-Light
Interpreter, Physical
Design



Brandon Sprague
Web Server, Web
Application
Desktop/Mobile
Interfaces

Overview of the Project

- Provides highly accessible multiplayer music creation to complete novices in the form of an interactive game grid
- Different combinations of cells in an instrument's grid produce many different polyphonic sounds
- Offers a variety of instruments and individual control to customize the timbre of your music

Final Product Specifications

- Users with no prior musical experience will be able to make pleasing music more than 90% of the time
- Groups of 20 people will be able to concurrently create music
- <25 ms delay syncing across clients (this can be verified on our live stats page)

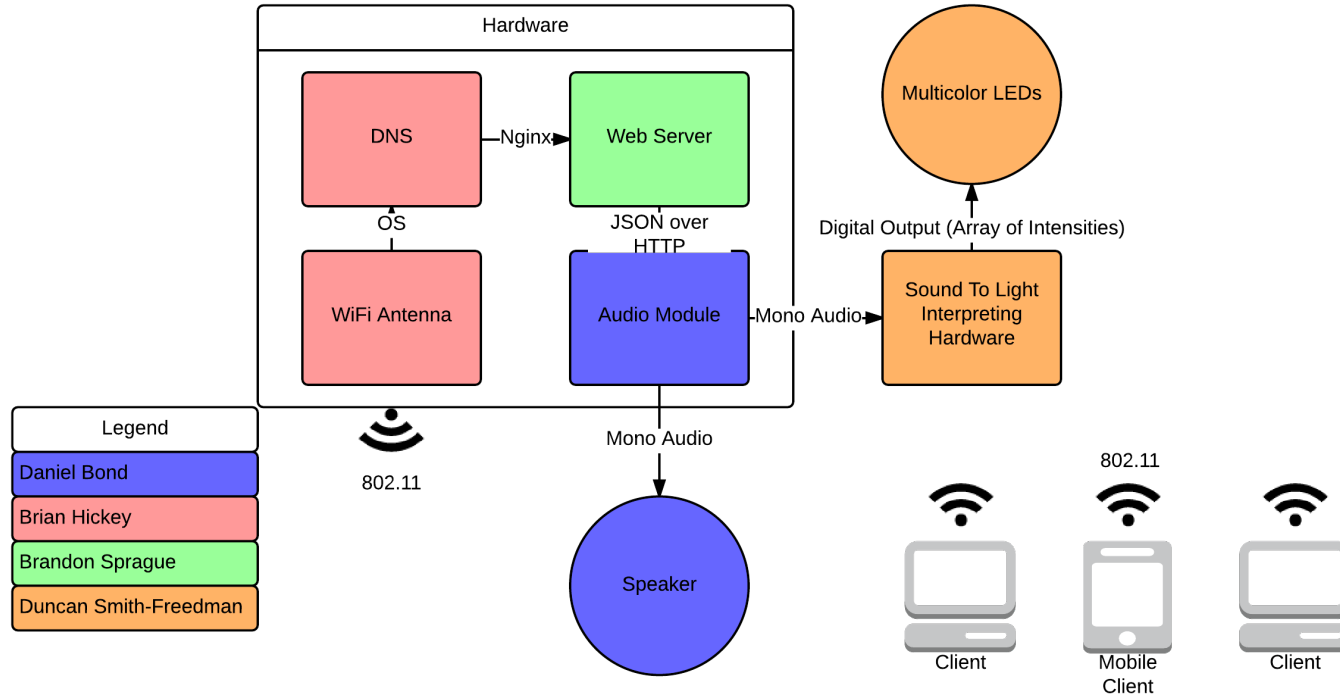
Final Product Functionality

- Speakers, amplifier, audio-to-light interpreter and LEDs housed inside central node
- LEDs respond to music generation by interpreting frequency range as light
- Treble and bass adjustments available on central node, instrument volume controlled on management console
- Live user stat tracking page to diagnose system performance

Final Product Integration

- Music generation module updated to take advantage of existing state space --- leads to more diverse music
- Web server and music module communicate management information over HTTP using JSON
- Speakers and other hardware now housed in 3D-printed central node
- External AC to DC power supply delivers power to all internal components

System Block Diagram (with interfaces defined)



Completed Team Responsibilities from CDR

- Brian & Duncan
 - 3-D design and printing of central node to fit all components
 - LED array responds real-time to music generation with smooth transitions
 - Power system to deliver desired power to internal components
- Brandon & Danny
 - Added a management panel so users can customize their music on the fly
 - Instruments created from Java MIDI library
 - Increased use of grid state space by modulating duration, volume, and pitches of notes selected by music generation algorithm

Demonstration Outline

- BeatBeam's central node (webserver, Teensy) will be booted
- Evaluators and presenters can connect to BeatBeam's SSID with their laptop or mobile device
- Go to Beatbeam.com to begin playing with the different instrument grids
- Modify features such as volume, instruments, and presets in the management console to customize your experience

Questions?



~23 days