Requirements and Specification Document

Sound Stealers

Requirements and Specification Document
<2023-04-28>, version 3.0

Project Abstract

EZBeats is a web-based virtual musical instrument which plays sound/music pieces sampled from different types of sounds across genres in real time. It works as a stand-alone digital musical instrument and a simple demonstrative DAW (Digital Audio Workstation). In EZBeats, users are able to select desired sound/music pieces from the library, play samples at pitches in equal temperament via keyboard/midi input, add their own recorded sound to library, and arrange their compositions in the built-in sequencer with an option of outputting to midi/audio files.

Document Revision History

Rev. 1.0 <2023-02-28>: Initial version

Rev 2.0 <2023-03-28>: Updates to app name, tech stack, UI, security requirements

Rev 3.0 <2023-04-28>: Added navigation flow charts. Clarified use on mobile. Updated customer description. Added use case and importance ratings.

Customer

The target users are DJs and music producers. For DJs, EZBeats provides great efficiency and freedom in live performance; they can play and load sounds from the library instantly, record their created track, and selectively apply sound effects. For music producers, EZBeats allows interaction with midi score and can allow the user to record a sound to be used. Our software would allow DJs or producers to use and create music without having to download dedicated software which might allow for more accessibility and ease of use if they are setting up at a new location.

Competitive Landscape

There is currently a decent amount of competition on the market for digital samplers, these are some of the popular current DAW (Digital Audio Workstation) software's on the market:

Ableton Live

Ableton Live is a Digital Audio Workstation available on macOS and Windows. It is designed to be an instrument for live performances and a tool for recording, mixing, arranging and composing. It was one of the first music applications to automatically beatmatch songs and is also used by DJ's. Ableton currently offers 3 main editions of Live, intro at $99, standard at $449, and suite at $749.

GarageBand

GarageBand is another Digital Audio Workstation that was developed by apple for macOS and iOS devices. The app has a music and podcast creation system which allows users to create multiple tracks with pre-made loops, various instrumental effects, pre-made MIDI keyboards and voice recordings. Even though it is only available for apple users, it is free of cost for them. It also has a new lesson feature where you can download music lessons on the app.

Logic Pro

Logic Pro, like GarageBand is a digital audio workstation for the macOS platform. It provides software instruments, audio effects and recording facilities for music synthesis. Logic Pro can also work with MIDI keyboards and control surfaces for input and processing, and for MIDI output. Logic Pro is even more exclusive than GarageBand, it is only available for apple users and the latest version costs $199. However, if you had bought it in the past it would be a free upgrade.

User Requirements

- Option to select sound samples from a pre-built library modify them
- Anonymous users will only be able to access the pre-built library
- Ability to load in sound samples from a local file system
- Play sound without significant delay
- Play track with sounds selected on beats
- Allow for sound recording from audio input

- Select sound and add to a beat of the track
- Select from a pre-built library of filters to modify the loaded sound

Use Cases

Name	The software presents a welcome page
Actor	Any user
Trigger	Upon first-time run
Events	- Users install and run for the first time - Users see the welcome message - Users will be able to run through a quick tutorial or start without
Exit Condition	- Users click through <# of pages to be decided> pages of starters' tips - Users click "start without tutorial"
Post-Conditions	Main UI shows up letting users know they are good to go
Acceptance Test	Users starts to play around with the buttons
Importance (1-5)	2

Name	Play a sound
Actor	Any user
Trigger	Users enter the main UI
Events	- Users hit their QWERTY keyboard (mapped to midi) - Users hit their midi keyboard
Exit Condition	Users stop hitting keyboard
Post-Conditions	Sound play stop
Acceptance Test	Users hear the sound
Importance (1-5)	5

Name	Choose a source sample
Actor	Any user
Trigger	Users click “browse” button under “sample” tab in the main UI
Events	Users select samples by names in the dropdown menu
Exit Condition	Dropdown menu closes
Post-Conditions	- Return to main UI - State of source sample tab changes
Acceptance Test	Users can check the name of source sample shows up in source sample tab
Importance (1-5)	4

Name	Add a sound
Actor	Any user
Trigger	Users click “add” button under “sample” tab in the main UI
Events	Users select “from local storage” button in pop-up window - Users select sound files in file explorer - User prompted to name the new sound in a pop-up window <Users select “record” button in pop-up window>
Exit Condition	Pop-up windows close
Post-Conditions	- New sound added to the library - New sound appear in the sample tab
Acceptance Test	Users are able to check new sounds in the library and hear it by playing
Importance (1-5)	3

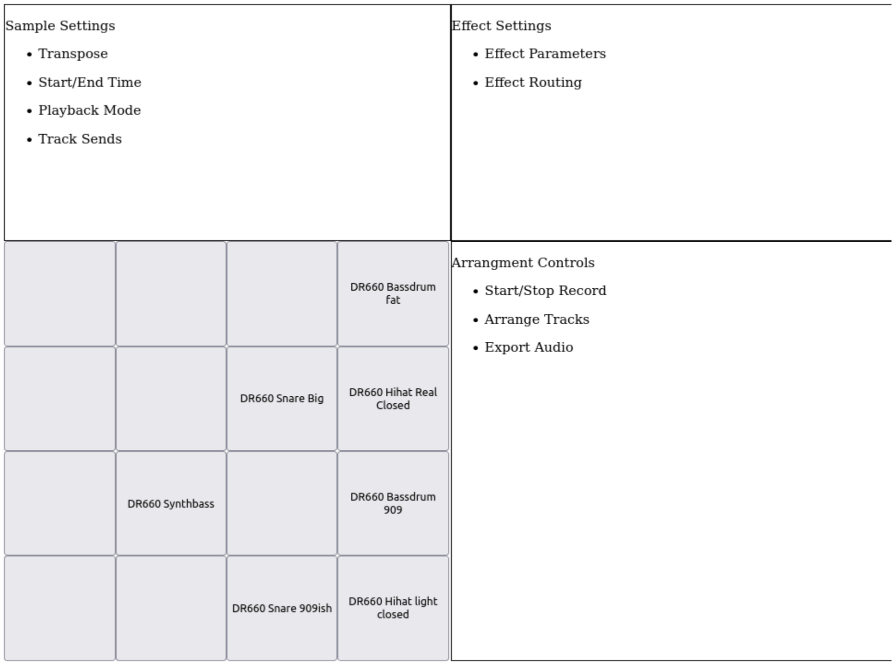
Name	Record

Actor	Any user
Trigger	User click on “Record” button
Events	User - play ready sounds using keyboard or button - or they don’t
Exit Condition	User click on “Record” button
Post-Conditions	Replay recorded sounds
Acceptance Test	User are able to toggle record mode on and off
Importance (1-5)	5

Name	Add Delay Effect
Actor	Any user
Trigger	User turns a knob to increase the delay effect
Events	User presses delay button Sample Info UI turns to delay settings User selects the delay time User selects the delay amount (wet/dry)
Exit Condition	User presses the delay button once more to go back to sample info
Post-Conditions	User can hear the delay effect
Acceptance Test	User can hear the delay effect with the correct delay time
Importance (1-5)	2

User Interface Requirements

Low-fidelity prototype



Sample Settings

- Transpose
- Start/End Time
- Playback Mode
- Track Sends

Effect Settings

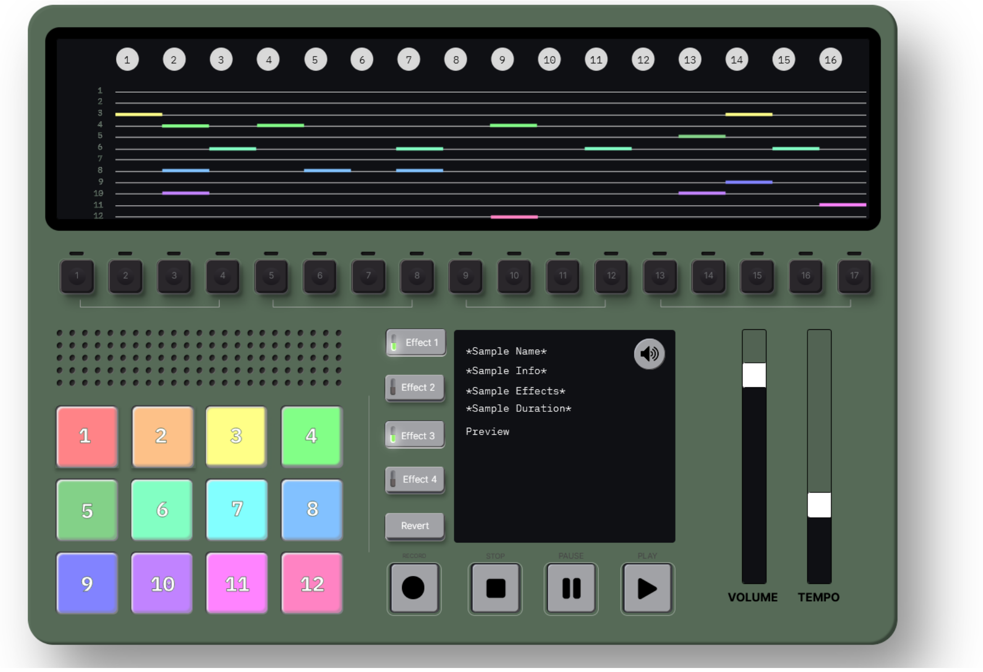
- Effect Parameters
- Effect Routing

			DR660 Bassdrum fat
		DR660 Snare Big	DR660 Hihat Real Closed
	DR660 Synthbass		DR660 Bassdrum 909
		DR660 Snare 909ish	DR660 Hihat light closed

Arrangment Controls

- Start/Stop Record
- Arrange Tracks
- Export Audio

Hi-fidelity prototype



- User interface similar to a sampler such as the MPC 3000.
- Grid of pads for playback.
  - Each pad can be individually assigned a sound to trigger
  - Pads can be triggered via user click, keyboard input, or midi input (through external controller or DAW control.

    - Can drop local file on a pad to load the file into that pad or select a file from a pre built library available
- Sample Settings
    - Displays sample visually
    - Allow for sample chopping
    - Can change pitch of the sample
    - Set playback mode (Play through or stop on release)
- Effect Settings
    - Configure various effects such as delay, reverb, filters, flangers, ect.
    - Create effect chains
- Arrangement Controls
    - Used to compose various tracks
    - Contains recording controls

Security Requirements

<Classified using STRIDE model developed by Microsoft>

- Restrict file formats to audio file types to avoid unnecessary file handling and errors. <Tampering>
- User authentication to be done with standard practice OAuth2. <Elevation of privilege>
- Local storage on the browser is planned so the user does not need to re-log in. If higher security is desired, can use session storage or only cookies to verify every page reload. <Spoofing>
- User input will be allowed in the form of key presses, but no input in the form of typing. The key presses will correlate to pads on the application <Tampering>
- Any queries to the database will be input sanitized and prepared statements will be used. <Tampering and information disclosure>

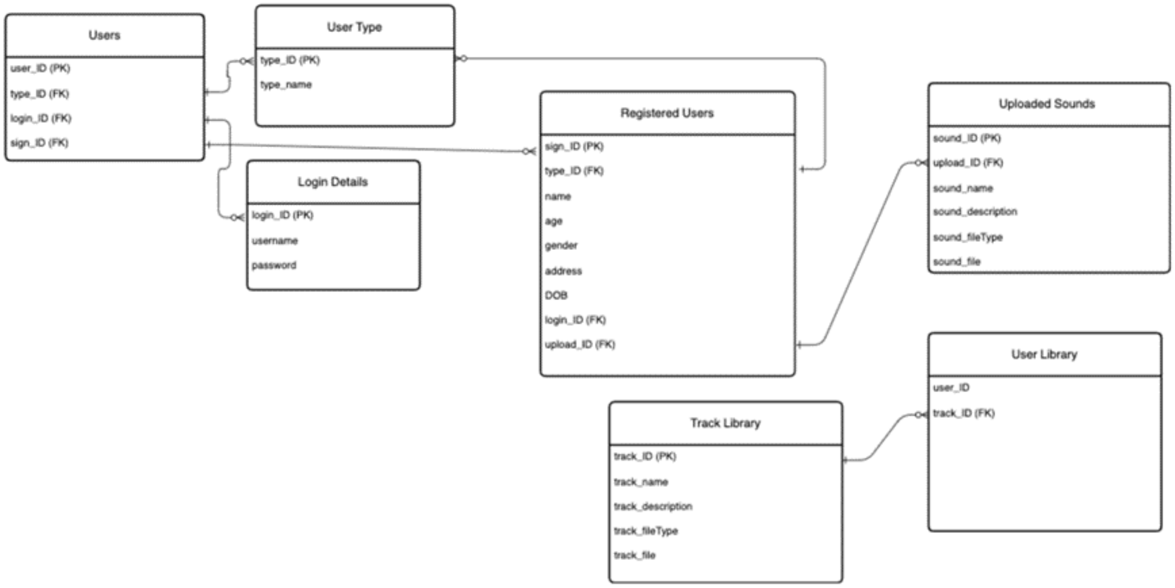
System Requirements

- Computer capable of playing audio
- Application will be accessible on the web and will be using JavaScript + React for a front-end interface for users. This requires users to have an internet browser like Google Chrome or Firefox. EZBeats will not be launched as a mobile app.
- The web server will be ran on a virtual machine within a Docker container. It will use Java Spring which utilizes an embedded Tomcat server that will connect to a MySQL Database.

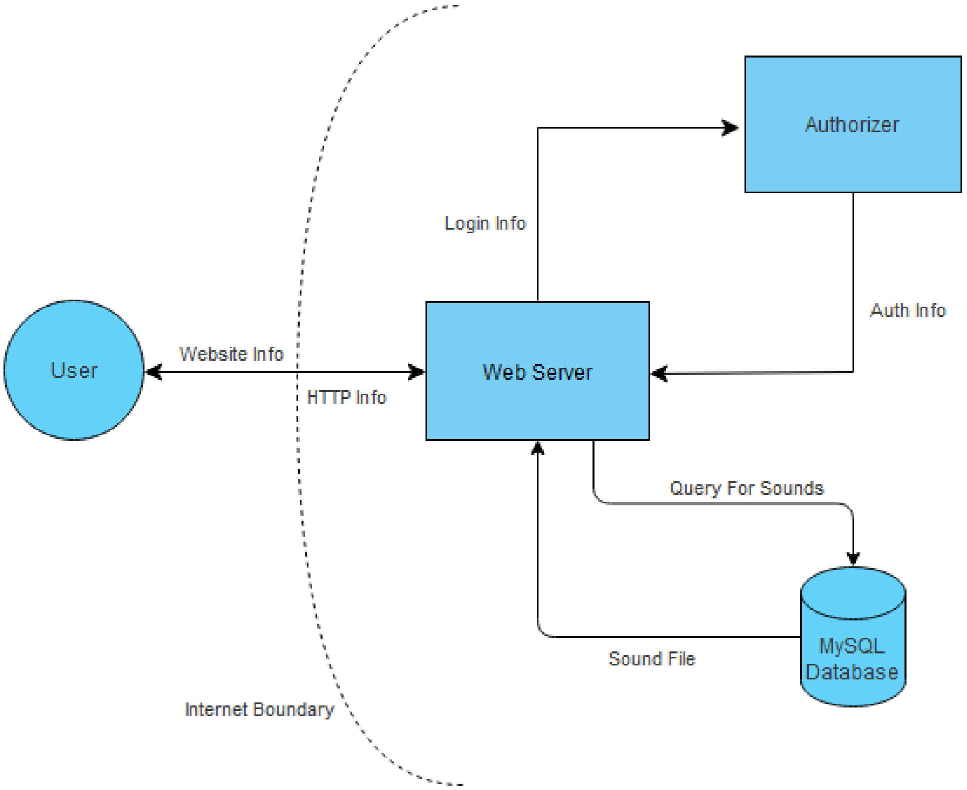
- Specific ports will be open on the virtual machine allowing access. If EZBeats was fully deployed, it would be hosted and accessible on a public domain.
- Internet access is required to access the application on the web. This is not a desktop application.
- The user will not need to have Java installed as Java is only being used to run the server that will relay information between the frontend and the database.

Specification

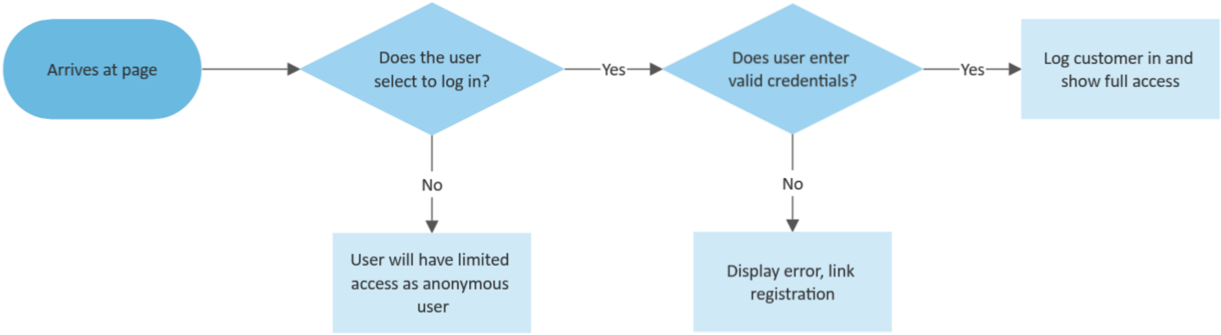
ER Diagram:



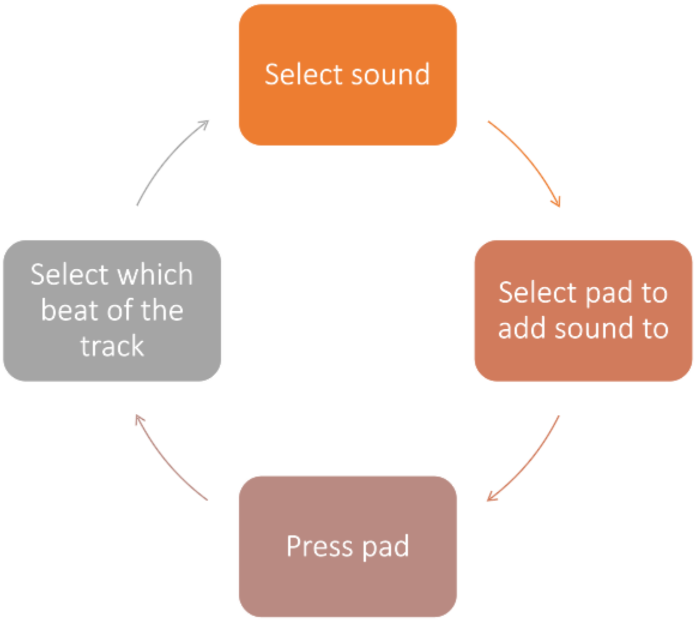
UML Diagram:



Login Process:


Arrives at page
Does the user select to log in?
Yes
Does user enter valid credentials?
Yes
Log customer in and show full access
No
No
User will have limited access as anonymous user
Display error, link registration

Workflow for users:


Select sound
Select pad to add sound to
Press pad
Select which beat of the track