Low Level Design Document (LLD)

REACT DRUM KIT

By: Shubham Singh

Document Version Control

Date Issued	Version	Description	Author
28-01-2023	1.0	Initial LLD	Shubham Singh
30-01-2023	1.1	Final Draft	Shubham Singh

Contents

Abstract

The drum kit is an interactive musical instrument that allows users to play various percussion sounds by clicking on different drum pads. The user interface is designed with a clean and minimalist style, making it easy to use and visually appealing. The responsive design ensures the drum kit works seamlessly on various devices and screen sizes. Overall, this drum kit provides a fun and engaging experience for users of all levels.

1. Introduction

1.1 Why this Low-Level Design Document?

This document provides a detailed description of its architecture and implementation. This document is an essential tool for the success of the project, as it provides a clear and detailed description of the project's technical specifications and helps to ensure that the final product meets the project's goals and requirements.

1.2 Scope

The React Drum Kit application allows users to play different drum sounds by clicking on various buttons or by using keyboard events. The application has a visually appealing design, responsive interface, and user-friendly interface, making it a great tool for music production and for playing around with sound.

1.3 Constraints

The constraints for the React Drum Kit include:

- Browser Compatibility: The application may only work in modern browsers that support the latest JavaScript and HTML standards with support for Tailwindcss.

- Sound Quality: The quality of the drum sounds used in the application may be limited by the sound library used, or by the user's device and browser.

- Customization: The customization options available to users may be limited by the developer's design and implementation choices.

- Keyboard Shortcuts: The keyboard shortcuts used to trigger drum sounds may not work for all keyboard configurations, or may interfere with other keyboard shortcuts used by the user.

- User Experience: The overall user experience may be limited by the application's design, navigation, and functionality.

- Accessibility: The application may not be accessible to users with disabilities, or may not fully support users with accessibility needs.

1.4 Out of Scope

The out of scope items for the React Drum Kit includes:

- Audio Recording: The ability to record and save audio clips is not part of the drum kit's functionality.

- Music Production Features: Features such as synthesizers, samplers, or audio effects are not part of the drum kit.

- Network Collaboration: The ability for multiple users to play and interact with the drum kit in real-time over a network is not part of the drum kit's functionality.

2. Technical Specifications

2.1 JavaScript Libraries

The application I built can be made into a single page application for the user so it is better to use JavaScript libraries React.js for better developer experience. Here is the list of libraries I used along with React.js.

Library	Version
react	18.2
react-router-dom	6.8
react-icons	4.7.1
tailwindcss	3.2.4

2.2 Linters

To maintain good standards across the projects I have used JavaScript linters tokeep out the potential bugs and errors in JavaScript code.

Library	Version
Prettier extension	9.9.0

2.3 Deployment

For hosting the application I used **Netlify** web services because of cost factor and easy to use.

3. Technology Stack

The React Drum Kit is frontend only. Detailed breakdown of frontend technologies used as follows:

3.1 Frontend

React.js	Render application
Tailwind CSS	Styling the application
React Router DOM	Client-side routing

3.2 Backend

No database used

4. Proposed Solution

The solution involved creating a React component for each drum pad in the drum kit. Each component maps a unique audio file to the corresponding pad and triggers the audio file to play on either a click or key press event. A state is set for each pad to keep track of the currently playing sound and dynamically update the display with the pad that is currently playing. This creates an interactive drum kit that can be played with the mouse or keyboard.

5. Work flow as a user

As a user of a React drum kit, the following steps outline the general workflow:

For Home page:

- Load the drum kit interface in the browser
- Observe the various drum pads displayed on the screen, each with its own unique label and sound
- Play a drum pad by clicking on it with the mouse or by pressing the corresponding keyboard key
- Hear the corresponding audio file play
- Observe the label of the currently playing pad change dynamically on the screen
- Repeat the steps to play different drum pads in any order desired
- Enjoy making music with the interactive React drum kit!

For Tutorial page:

- Navigate to the "How to Play" page for the React drum kit
- Read through the instructions to understand how to interact with the drum kit
- Observe the various drum pads displayed on the screen, each with its own unique label and sound
- Enjoy making music with the interactive React drum kit!

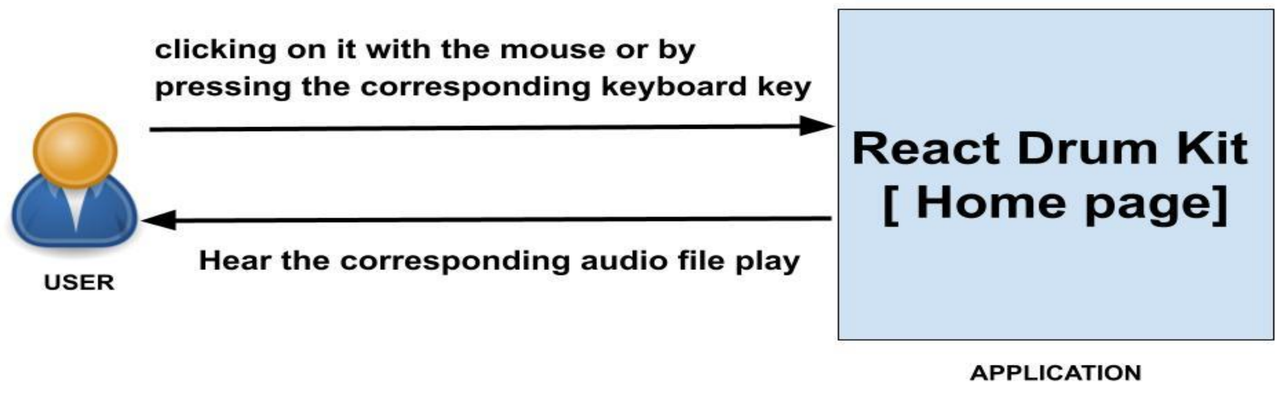
clicking on it with the mouse or by pressing the corresponding keyboard key
Hear the corresponding audio file play
USER
React Drum Kit [Home page]
APPLICATION

Figure 1: Workflow as a User [Home page]

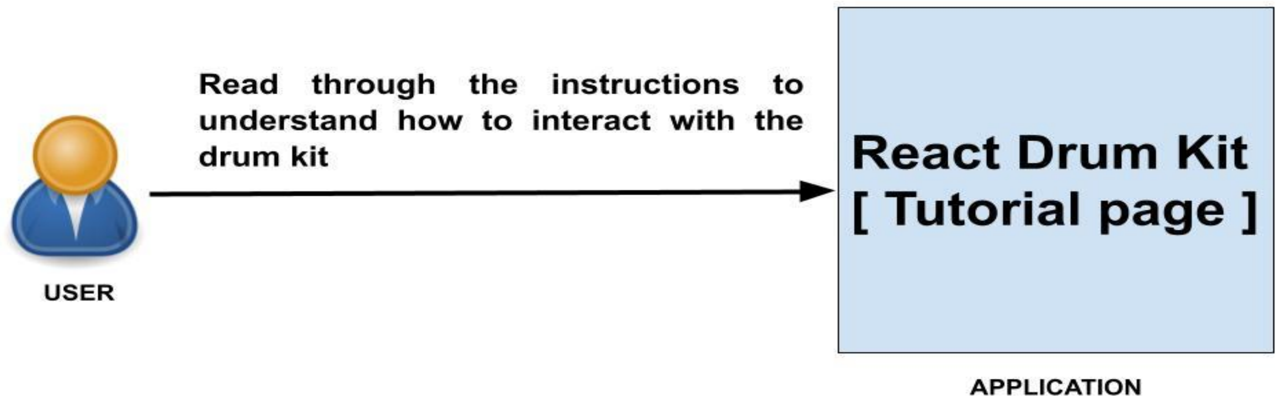
Read through the instructions to understand how to interact with the drum kit
USER
React Drum Kit [Tutorial page]
APPLICATION

Figure 1: Workflow as a User [Tutorial page]