

# CRUD App Documentation

## Introduction

The CRUD app is a web application that allows users to perform Create, Read, Update, and Delete operations on a MySQL database. It provides an interface for managing messages stored in the database.

## Technologies Used

- **Next.js:** A React framework for building full stack user interfaces.
- **MySQL:** A relational database management system.
- **mysql2:** A Node.js package for connecting to MySQL databases.

## Setup and Installation

- Clone the repository from GitHub using <https://github.com/owais-26/next-mysql-crud>
- Install the necessary dependencies using npm or yarn.
- Set up a MySQL server on your local machine if you haven't already.
- Create a new database called "messages\_schema" in MySQL.

## Configuration

To connect the app to the MySQL database, follow these steps:

Open the file named db.js in the app's code.

Import the mysql2 package.

Create a connection object with the database configuration details:

host: The hostname or IP address of the MySQL server.

user: The username for accessing the database.

password: The password for the database user.

database: The name of the database.

Export the connection object.

### Code:

```
import mysql from 'mysql2';
```

```
const db = mysql.createConnection({  
  host: "localhost",  
  user: "root",  
  password: "owaismysql123",  
  database: "messages_schema",  
});
```

*export default db;*

## Functionality

The CRUD app provides the following functionality for managing messages:

- **Create**

Users can create a new message by entering text in an input field and clicking the "Create New Message" button.

The app sends a request to the server to insert the new message into the database.

The message is then displayed in the app's interface.

- **Read**

The app retrieves messages from the database and displays them in a table.

Users can view all messages or filter them by entering a search term.

The app highlights the matching search term in the displayed messages.

- **Update**

Users can edit a message by clicking the "Edit" button next to the message in the table.

The app fetches the selected message's details and populates an input field with the existing text.

After making changes, users can save the updated message by clicking the "Create New Message!" button.

The app sends a request to the server to update the message in the database.

- **Delete**

Users can delete a message by clicking the "Remove" button next to the message in the table.

The app prompts the user to confirm the deletion.

If confirmed, the app sends a request to the server to delete the message from the database.

## Deployment

To deploy the CRUD app to a live server or hosting platform, follow these steps:

- Build the production-ready version of the app using the appropriate command (npm run build or yarn build).
- Upload the build files to the desired server or hosting platform.
- Configure the server to serve the app's files.
- Set up the necessary environment variables, such as the database connection details.
- Start the app on the server or hosting platform.

## Limitations

While the CRUD app provides valuable functionality for managing messages in a MySQL database, there are certain limitations that should be considered:

### **Search Functionality Limitation**

The search functionality in the app requires users to toggle between the pages to find a particular searched term. This means that if the desired search term is not present on the current page, users must navigate through the pagination to find the matching result. Enhancements could be made to improve the search functionality by dynamically updating the displayed messages based on the search term without requiring page navigation.

By considering these limitations, I can further enhance the CRUD app's user experience and address any potential issues that users may encounter.

### **Conclusion**

The CRUD app provides a user-friendly interface for managing messages stored in a MySQL database. By following the setup instructions and understanding the app's functionality, users can easily create, read, update, and delete messages in the database.

### **Screenshots**

