

# Architecture Documentation

## MERN Note App

Siddharth

Rukadikar

Document Version Control

Date Issued	Version	Description	Author
2025-04-07	1.0	Architecture doc	Siddharth Rukadikar

## Contents

Document Version Control .....	2
1 Introduction .....	4
1.1 Purpose of this Low-Level Design Document .....	4
1.2 Scope .....	4
2 System Architecture .....	4
2.1 High Level Diagram .....	4
2.2 Component Breakdown .....	4
3 Technology Stack .....	5
4 Data Flow .....	5
4.1 User authentication .....	5
4.2 Note Management (CRUD) .....	5
5 Deployment Architecture .....	6
5.1 Frontend .....	6
5.2 Backend .....	6
5.3 Database .....	6
6 Security and Future improvements .....	6
7 Component Diagram .....	6
8 Conclusion .....	7

# 1 Introduction

## 1.1 Purpose

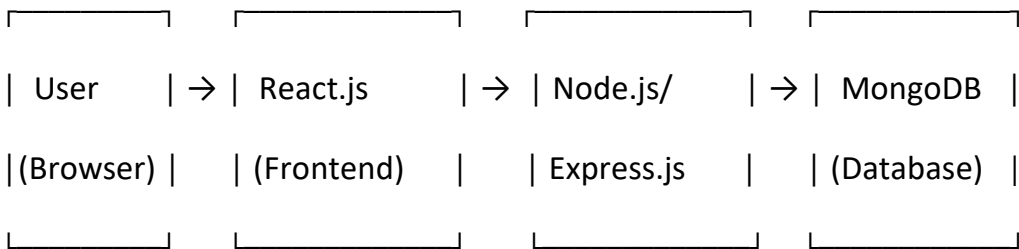
This document provides a technical blueprint of the MERN Note App, detailing its architecture, components, and workflows to guide development and maintenance.

## 1.2 Scope

- System design and technology choices.
- Data flow for authentication and note operations.
- Deployment and security practices.

# 2 System Architecture

## 2.1 High-Level Diagram



## 2.2 Component Breakdown:

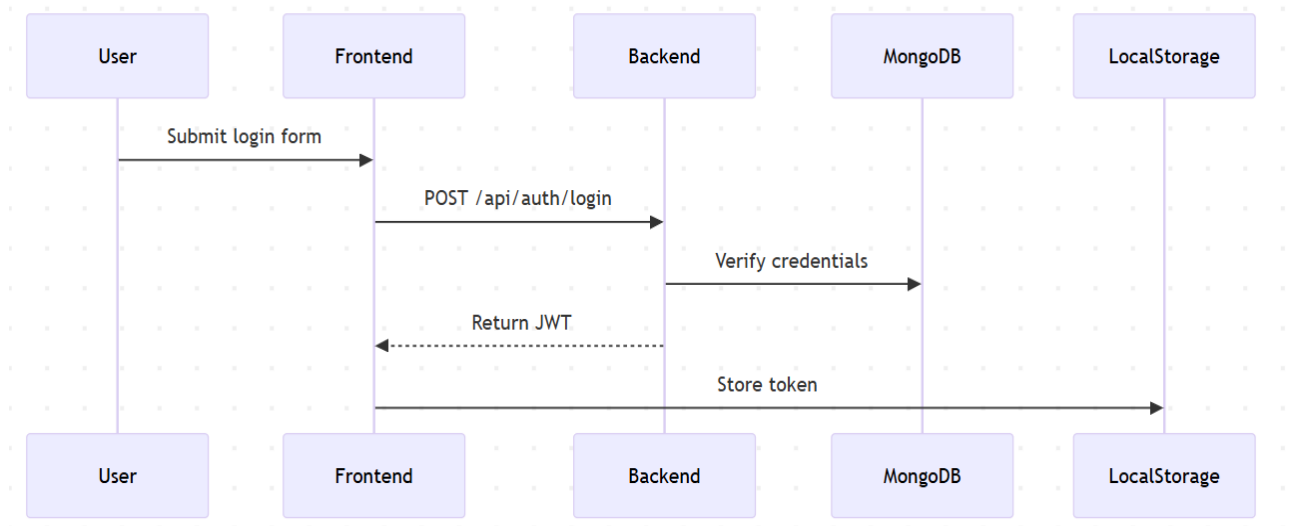
Layer	Key Components	Functionality
Frontend	React, Redux, Axios	Ui rendering, state management API calls
Backend	Express, Mongoose, JWT	API routes, data validation, Auth
Database	MongoDB Atlas	Stores users, notes

### 3 Technology stack

Front End	HTML/CSS/JS/React
Backend	Nodejs, ExpressJs
Database	MongoDB
Deployment	Render

## 4 Data Flow

### 4.1 User Authentication



### 4.2 Note Management

- Create: POST /api/note/add.
- Read: GET /api/note/getNote/:noteId.
- Update/Delete: DELETE /api/note/delete/:noteId

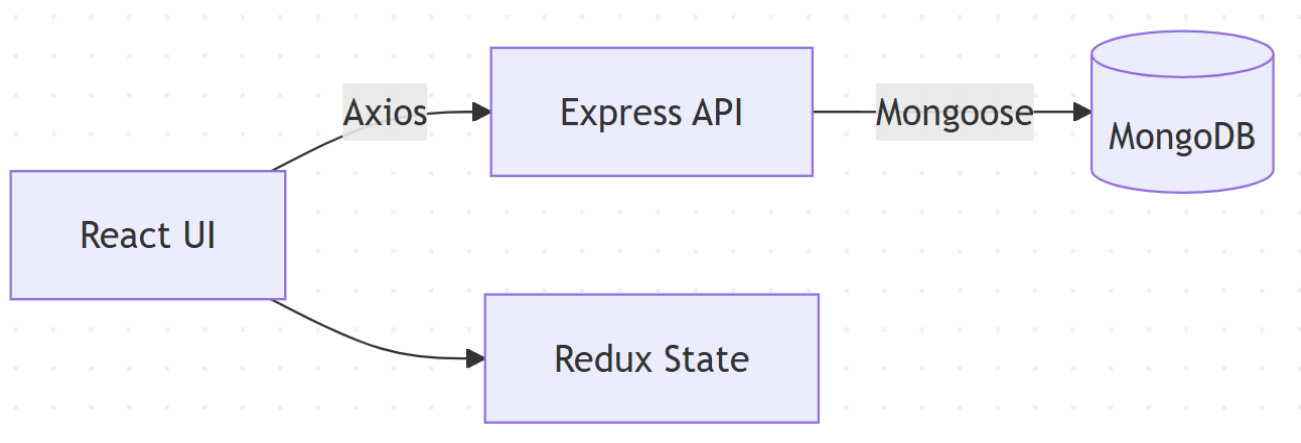
## 5 Deployment Architecture:

Service	Hosting	configuration
Frontend	Render	Environment variables for api endpoints
Backend	Render	JWT secret , MongoDB URI
Database	MongoDB atlas	Automated backups

## 6 Security and Future Improvements:

- **Security:**
  - HTTPS enforced.
  - JWT Expiration.
  - Rate Limit (express rate limit).
- **Future Improvements:**
  - Real-time-sync: socket.io.
  - Offline mode: Service worker+ indexedDb.
  - Analytics: Note usage tracking

## 7 Component Diagram



## 8 Conclusion

The MERN Note App Low-Level Design outlines a modular and scalable architecture for building a user-friendly note management system. By leveraging modern web technologies such as React, Node.js, and MongoDB, along with features like JWT authentication and responsive design, the application ensures both functionality and security. This document will guide developers through implementation and help maintain consistency throughout the project lifecycle.