**Full Stack Development with MERN**

**Project Documentation**

# 1. Introduction

## Project Title:

StockSense - Your Intuitive Stock Trading Platform

## Team Members:

Aditi Babar (Full Stack Developer, UI/UX Designer)

Mrunmayee Phadtare (User Authentication Setup)

Ankit Kumar Singh (AI-powered Stock Predictions)

Sandeepan Chakraborty (Trading Dashboard UI)

# 2. Project Overview

## Purpose:

* Offer traders live market data for better decision-making.
* Provide a centralized platform for stock trading.
* Integrate smart analytics tools to enhance trading efficiency.

## Features:

* Real-time stock tracking dashboard
* Secure trade execution & portfolio management
* Automated alerts for stock movements
* Data-driven investment insights

# 3. Architecture

## Frontend:

* Built using React.js and Tailwind CSS
* API integration for real-time stock updates
* Authentication & user profile management

## Backend:

* Node.js with Express.js
* Secure API endpoints for transactions
* Handles trade execution & portfolio tracking

## Database:

* MongoDB Atlas
* Stores user details & stock transaction history

# 4. Setup Instructions

## Prerequisites:

* Node.js
* MongoDB Atlas account
* Git

## Installation:

* Clone the repository: git clone https://github.com/Aditi16-ctrl/Stocks-Trading-MERN.git
* Navigate to the project directory
* Install client and server dependencies using npm install
* Set up .env files for frontend and backend with proper configurations

# 5. Folder Structure

## Client:

Contains all React components, pages, and services for UI

## Server:

Contains Express routes, controllers, models, and middleware for backend logic

# 6. Running the Application

## Frontend:

cd client && npm start

## Backend:

cd server && npm start

# 7. API Documentation

* POST /register – User registration
* POST /login – User authentication
* GET /stocks – Fetch live stock data
* POST /trade – Execute buy/sell order
* GET /portfolio – Fetch user portfolio

# 8. Authentication

* JWT-based authentication system
* bcrypt used for password hashing
* Secure API endpoints require valid tokens

# 9. User Interface

* Login & Registration Page
* Stock Trading Dashboard
* Portfolio Overview & Stock Predictions

# 10. Testing

* Manual testing with dummy user scenarios
* Tools: Postman for API testing

# 11. Screenshots or Demo

* Repository: https://github.com/Aditi16-ctrl/Stocks-Trading-MERN.git
* Demo Video: Insert Link Here

# 12. Known Issues

* Initial learning curve for new traders
* Dependent on live market data availability
* No offline trading support

# 13. Future Enhancements

* AI-driven predictive analytics for stocks
* Mobile trading app using React Native
* Multi-user support for institutional investors
* Integration with financial news APIs