

📄 backend/README.md

(Fully customized for our project – with Render, Docker, EKS notes etc.)

Copy Trading Backend (MT Project)

This is the backend server for the MT Copy Trading App, built with Node.js, Express, MongoDB, and JWT Auth.

Features

- User Registration & Login (JWT-based)
- Trade Management (Add/Delete/Fetch Trades)
- Secured per-user trade data
- Connected to MongoDB Atlas
- Integrated with Sentry for monitoring
- Ready for CI/CD with Docker, Render, and EKS

Tech Stack

- Node.js + Express
- MongoDB + Mongoose
- JWT Authentication
- Sentry monitoring
- Docker + GitHub Actions for CI/CD

.env File

Make sure to create a `.env` in `backend/``:

`MONGO_URI=your_mongo_connection_string JWT_SECRET=your_secret_key`

Running Locally

1. Install dependencies:

```
cd backend npm install
```

2. Start the server:

```
npm start
```

Server runs on `http://localhost:5000``.

API Endpoints

```
### Auth Routes
- `POST /api/auth/register` – register new user
- `POST /api/auth/login` – login and get JWT token
```

```
### Trade Routes (protected)
- `POST /api/trades` – create trade
- `GET /api/trades` – get user's trades
- `DELETE /api/trades/:id` – delete trade
```

Use JWT in Authorization header:

Authorization: Bearer your_token

```
## CI/CD
- Build & Push images via GitHub Actions (see `.github/workflows/deploy.yml`)
- Deploy to:
  - Render (for simple hosting)
  - EKS (when `EKS_ENABLED=true` in secrets)
```

```
## Monitoring
Sentry is integrated. Provide your DSN in Render environment vars as `SENTRY_DSN`.
```

📄 ****MT Copy Trading Project**** powered by Node.js + MongoDB + React + Docker + EKS.

📄 Files now fully covered in backend:

Category	File	📄 Done
Server Entry	src/server.js	📄 yes
App Setup	src/app.js	📄 yes
Routes Auth	src/routes/authRoutes.js	📄 yes
Routes Trades	src/routes/tradeRoutes.js	📄 yes
Middleware Auth	src/middleware/authMiddleware.js	📄 yes
Models Trade	src/models/Trade.js	📄 yes
Models User	src/models/User.js	📄 yes
README	README.md	📄 yes
Optional RoutesIndex	src/routes/index.js	📄 yes