Alternate You

Here's your first version — you can add more later.

- User Login/Register
- Log an activity (e.g., "Watched YouTube", "Studied", "Slept late")
- Record mood (scale 1–10)
- Simulate an alternate decision (manually for now)
- Timeline view – shows real + alternate path
- Dashboard with insights (optional)

Backend 📁

```
/doppelganger-backend
├── /models
├── /routes
├── /controllers
├── /middleware
└── server.js
```

Frontend 📁

```
/doppelganger-frontend
├── /components
├── /pages
├── /services (API calls)
├── /charts (timeline visual)
└── App.jsx
```

Plan of action 📁

🧠 Frontend: Build Login + Dashboard UI

Create login/register form
On login success, save token in localStorage
Dashboard:

- "Log an activity" input
- "Mood" slider (1–10)
- Button: "Simulate Alternate Day"

Option 1: Use simple rule-based logic

For example:
If mood < 4 after "browsing social media," then future regret = high

If mood > 7 after "studying," then positive simulation

✅ **You can do this 100% with just JavaScript — no TensorFlow needed.**

🗓️ **Day** 🗓️	🎯 **Goal** 🎯	📝 **What You'll Build** 📝
🧠 Day 1	Plan + Setup	- Finalize features 📁 Setup MERN folders 📁 Connect MongoDB
🔐 Day 2	Auth System	- Register/login forms (React) 📁 JWT auth (Express + MongoDB) 📁 Dashboard access after login
📝 Day 3	Logging UI + DB	- Activity + Mood log form 📁 Save to MongoDB 📁 View logs on dashboard
🔄 Day 4	Simulation Logic	- Add rule-based "What if" logic 📁 Store alternate choices 📁 Display suggestion
📊 Day 5	Timeline Visualization	- Use Chart.js or D3.js 📁 Show real vs alternate decision timeline 📁 Mood trends
📈 Day 6	Insights Page	- Top regretful actions 📁 Streaks / mood score history 📁 Summary chart
🎨 Day 7	UI Polish + Error Handling	- Improve UX, styling 📁 Add loading spinners, validation 📁 Empty state messages
🚀 Day 8	Final Touch + Deployment	- Deploy frontend (Netlify/Vercel) 📁 Backend (Render) 📁 Test on mobile & desktop

🔥 **Optional Additions (Add 2–3 Days):**

- TensorFlow.js model to learn patterns
- Voice logging ("I just studied" → auto entry)
- Google login
- Export PDF "monthly report"

💡 **Tip: Log Progress**

Each day, write 3 things:

- What I built

- What I understood
- What I need ChatGPT to help with tomorrow

Project Structure

```
your-project/
├── client/ # React frontend
│ ├── public/
│ │ └── index.html # HTML template
│ └── src/
│ ├── components/ # Reusable UI components
│ │ ├── Navbar.jsx # Navigation bar
│ │ ├── ActivityForm.jsx # Form to log activity
│ │ └── SimulationCard.jsx # Displays simulated result
│ ├── pages/ # Page-level components
│ │ ├── Login.jsx
│ │ ├── Register.jsx
│ │ ├── Dashboard.jsx # User's main screen
│ │ └── Simulation.jsx # View for alternate timeline
│ ├── charts/
│ │ └── TimelineChart.jsx # D3.js or Chart.js timeline visualization
│ ├── services/
│ │ ├── api.js # Axios instance and endpoints
│ ├── utils/
│ │ └── helpers.js # Format timestamps, mood levels, etc.
│ ├── App.js
│ └── index.js
│
├── server/ # Express backend
│ ├── controllers/
│ │ ├── authController.js # Handles login/register
│ │ ├── logController.js # Handles activity logging
│ │ └── simController.js # Handles simulation prediction
│ ├── middleware/
│ │ └── auth.js # JWT auth verification
│ ├── models/
│ │ ├── User.js # Mongoose schema for users
│ │ ├── Log.js # User logs: activity, mood, time
│ │ └── Simulation.js # Alternate activity and predicted mood
│ ├── routes/
│ │ ├── authRoutes.js # /api/auth routes
│ │ ├── logRoutes.js # /api/logs routes
│ │ └── simRoutes.js # /api/simulate routes
│ ├── services/
│ │ └── simulationEngine.js # TensorFlow.js logic or rules for predicting mood
│ ├── utils/
│ │ └── jwtUtils.js # Create/verify tokens
│ ├── config/
│ │ └── db.js # MongoDB connection logic
│ ├── .env # Secrets: JWT, Mongo URI
│ ├── server.js # App entry point
│ └── package.json
│
├── README.md
└── package.json # Root package.json if using concurrently for client+server
```