**AI Assistant Platform - Complete Project Structure**

**📁 Project Structure**

ai-assistant/

├── frontend/ # React frontend

│ ├── public/

│ ├── src/

│ │ ├── components/

│ │ │ ├── Chat/

│ │ │ ├── Integrations/

│ │ │ ├── Dashboard/

│ │ │ └── Auth/

│ │ ├── hooks/

│ │ ├── services/

│ │ ├── utils/

│ │ └── App.js

│ └── package.json

├── backend/ # Node.js/Express API

│ ├── controllers/

│ ├── middleware/

│ ├── models/

│ ├── routes/

│ ├── services/

│ │ ├── integrations/

│ │ ├── ai/

│ │ └── scheduling/

│ ├── utils/

│ └── server.js

├── database/

│ └── migrations/

├── docker/

├── .env.example

├── docker-compose.yml

└── README.md

**🚀 Quick Start Commands**

# Clone and setup

git clone <your-repo>

cd ai-assistant

# Backend setup

cd backend

npm install

cp .env.example .env

npm run dev

# Frontend setup (new terminal)

cd frontend

npm install

npm start

# Database setup

docker-compose up -d mongodb redis

**🔧 Tech Stack**

* **Frontend**: React 18, TypeScript, Tailwind CSS, Socket.io-client
* **Backend**: Node.js, Express, MongoDB, Redis, Socket.io
* **AI**: OpenAI GPT-4, Custom NLP processing
* **Integrations**: Google APIs, Microsoft Graph, Slack SDK
* **Authentication**: JWT, OAuth 2.0
* **Deployment**: Docker, AWS/Vercel

**📦 Key Dependencies**

**Backend package.json**

{

"dependencies": {

"express": "^4.18.2",

"mongoose": "^7.5.0",

"socket.io": "^4.7.2",

"jsonwebtoken": "^9.0.2",

"bcryptjs": "^2.4.3",

"cors": "^2.8.5",

"dotenv": "^16.3.1",

"openai": "^4.0.0",

"googleapis": "^126.0.1",

"@microsoft/microsoft-graph-client": "^3.0.7",

"@slack/web-api": "^6.9.0",

"redis": "^4.6.8",

"node-cron": "^3.0.2",

"helmet": "^7.0.0",

"express-rate-limit": "^6.10.0"

}

}

**Frontend package.json**

{

"dependencies": {

"react": "^18.2.0",

"react-dom": "^18.2.0",

"typescript": "^5.0.0",

"tailwindcss": "^3.3.0",

"socket.io-client": "^4.7.2",

"axios": "^1.5.0",

"react-router-dom": "^6.15.0",

"react-query": "^3.39.3",

"@heroicons/react": "^2.0.18",

"react-hot-toast": "^2.4.1",

"framer-motion": "^10.16.0",

"date-fns": "^2.30.0"

}

}

**🔐 Environment Variables (.env)**

# Server

PORT=5000

NODE\_ENV=development

# Database

MONGODB\_URI=mongodb://localhost:27017/ai-assistant

REDIS\_URL=redis://localhost:6379

# JWT

JWT\_SECRET=your-super-secret-jwt-key

JWT\_EXPIRE=7d

# AI

OPENAI\_API\_KEY=your-openai-api-key

# Google OAuth & APIs

GOOGLE\_CLIENT\_ID=your-google-client-id

GOOGLE\_CLIENT\_SECRET=your-google-client-secret

# Microsoft Graph

MICROSOFT\_CLIENT\_ID=your-microsoft-client-id

MICROSOFT\_CLIENT\_SECRET=your-microsoft-client-secret

# Slack

SLACK\_CLIENT\_ID=your-slack-client-id

SLACK\_CLIENT\_SECRET=your-slack-client-secret

# Email

EMAIL\_SERVICE=gmail

EMAIL\_USER=your-email@gmail.com

EMAIL\_PASS=your-app-password

**🐳 Docker Setup (docker-compose.yml)**

version: '3.8'

services:

mongodb:

image: mongo:latest

ports:

- "27017:27017"

volumes:

- mongodb\_data:/data/db

redis:

image: redis:alpine

ports:

- "6379:6379"

backend:

build: ./backend

ports:

- "5000:5000"

environment:

- NODE\_ENV=development

depends\_on:

- mongodb

- redis

volumes:

- ./backend:/app

frontend:

build: ./frontend

ports:

- "3000:3000"

volumes:

- ./frontend:/app

volumes:

mongodb\_data:

**🎯 Core Features Implementation Plan**

**Phase 1: Foundation (Week 1-2)**

* [ ] Basic chat interface
* [ ] User authentication
* [ ] AI integration (OpenAI)
* [ ] Real-time messaging

**Phase 2: Integrations (Week 3-4)**

* [ ] Google Calendar integration
* [ ] Gmail integration
* [ ] Google Drive integration
* [ ] Basic scheduling

**Phase 3: Advanced Features (Week 5-6)**

* [ ] Smart scheduling
* [ ] Email automation
* [ ] Slack integration
* [ ] Notion integration

**Phase 4: Polish & Scale (Week 7-8)**

* [ ] Advanced AI features
* [ ] Performance optimization
* [ ] Security hardening
* [ ] Deployment setup

**💡 Unique Selling Points for Y Combinator**

1. **Unified AI Experience**: Single interface for all productivity tools
2. **Smart Context Awareness**: AI understands your workflow patterns
3. **Proactive Scheduling**: AI suggests optimal meeting times
4. **Cross-Platform Intelligence**: Data insights across all integrated platforms
5. **Natural Language Processing**: Complex commands in plain English
6. **Privacy-First Architecture**: Local processing where possible

**📊 Market Opportunity**

* **TAM**: $50B+ productivity software market
* **SAM**: $5B AI-powered productivity tools
* **Target**: 100M+ knowledge workers globally

**🚀 Go-to-Market Strategy**

1. **Free Tier**: Basic AI chat + 2 integrations
2. **Pro Tier ($15/mo)**: Unlimited integrations + advanced AI
3. **Enterprise ($50/user/mo)**: Custom integrations + admin controls

**📈 Success Metrics**

* Daily Active Users (DAU)
* Integration connection rate
* Time saved per user
* Revenue per user (ARPU)
* Net Promoter Score (NPS)

Ready to start coding? I'll provide the complete implementation files next!