

AI CHAT INTERFACE

Student: Nikolas Daniel Vincenti

Matriculation: 9211929

Course: DLBCSPJWD01 - Project Java and
Web Development - 19/06/2025

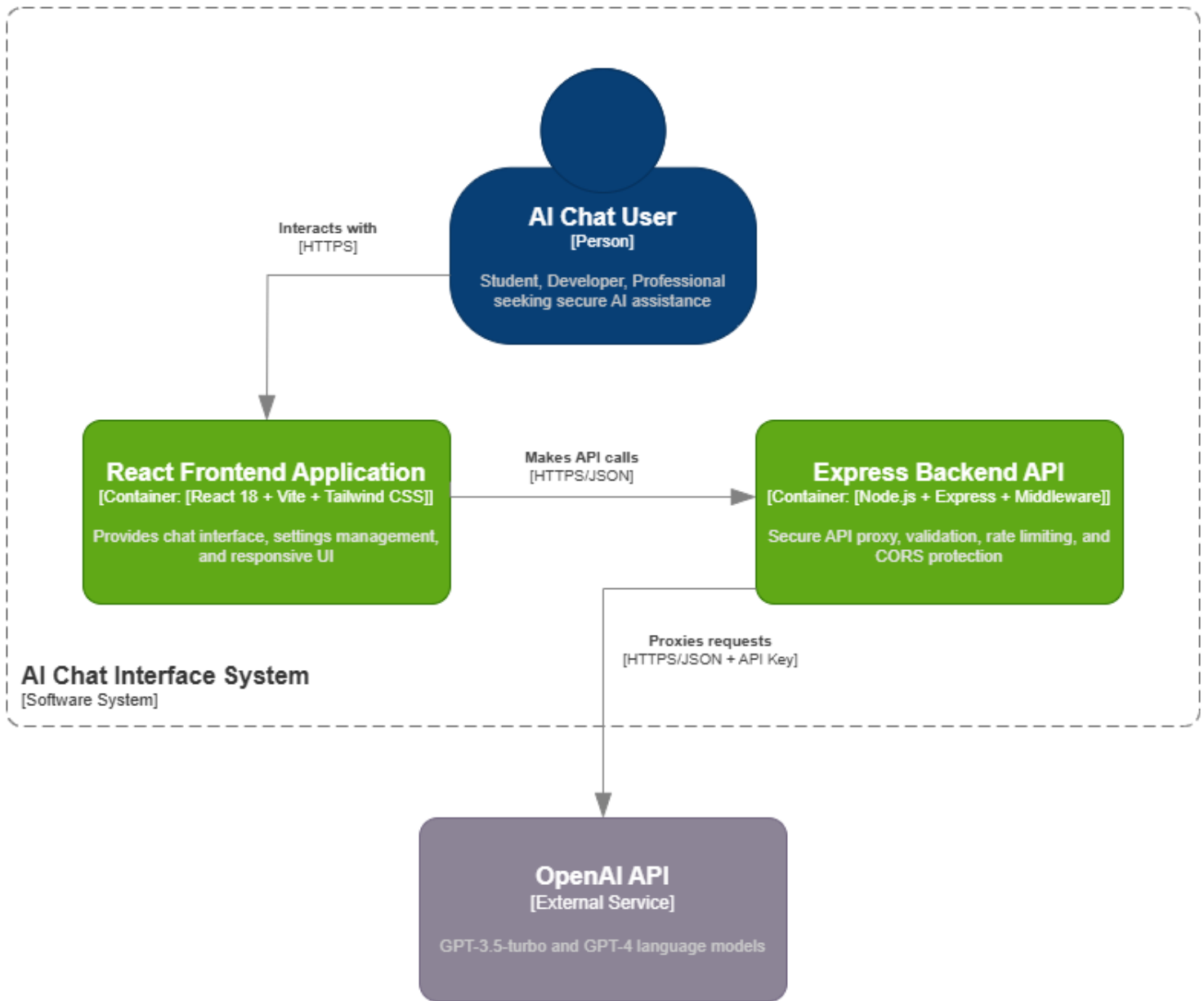
GitHub Repository:

<https://github.com/NikVince/ai-chat-interface>

Live: <https://ai-chat-interface-nine.vercel.app/>

PROJECT PURPOSE

- **Target Users:** Developers, students, and professionals
- **Core Value Proposition:** Privacy-first AI chat interface where users provide their own OpenAI API keys
- **Key Benefits:**
 1. Complete data privacy (no third-party storage)
 2. Cost transparency and control
 3. Direct API usage without intermediaries
 4. Model selection (GPT-3.5-turbo vs GPT-4)
- **Problem Solved:** Secure AI assistance without compromising privacy or losing control over API costs



ARCHITECTURE DIAGRAM

TECH STACK

- **Frontend Technologies:**

- React 18 (modern functional components with hooks)
- Vite (fast development server, replaces deprecated CRA)
- Tailwind CSS (utility-first responsive design)

- **Backend Technologies:**

- Node.js + Express.js (API server)
- OpenAI SDK (official integration)
- Middleware: CORS, rate limiting, input validation

- **Integration:** RESTful API communication, secure proxy pattern

MOBILE FIRST APPROACH

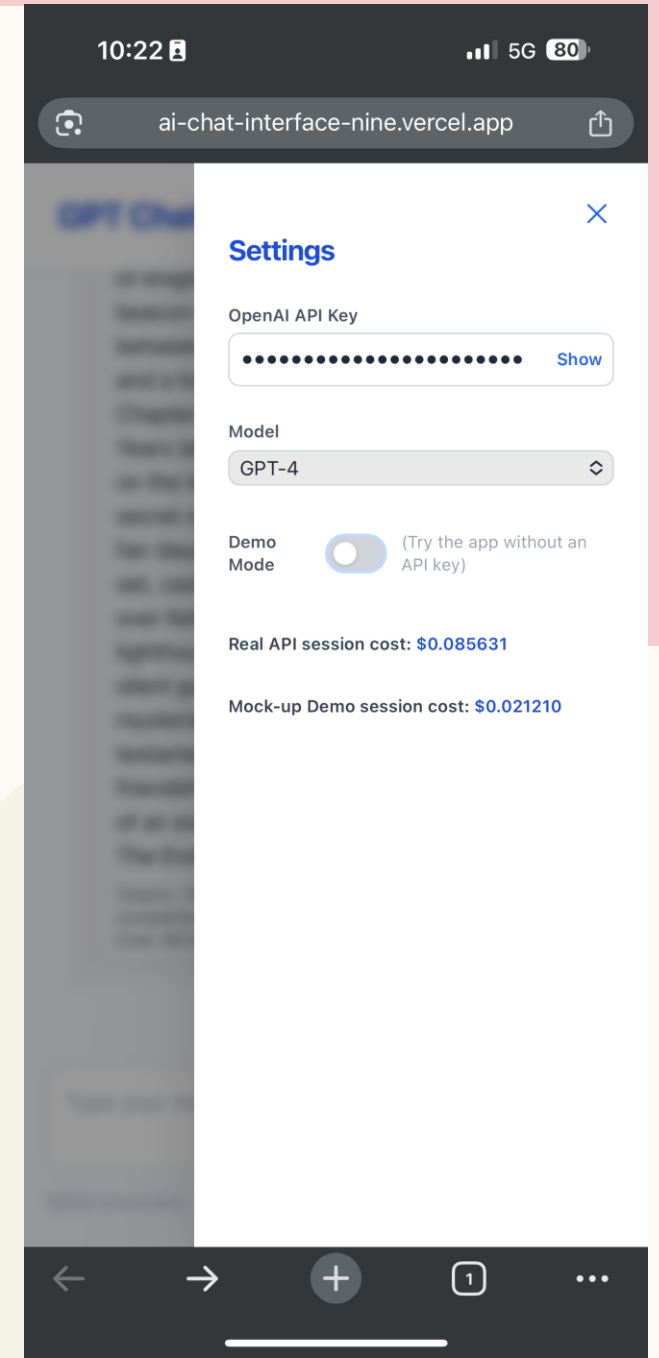
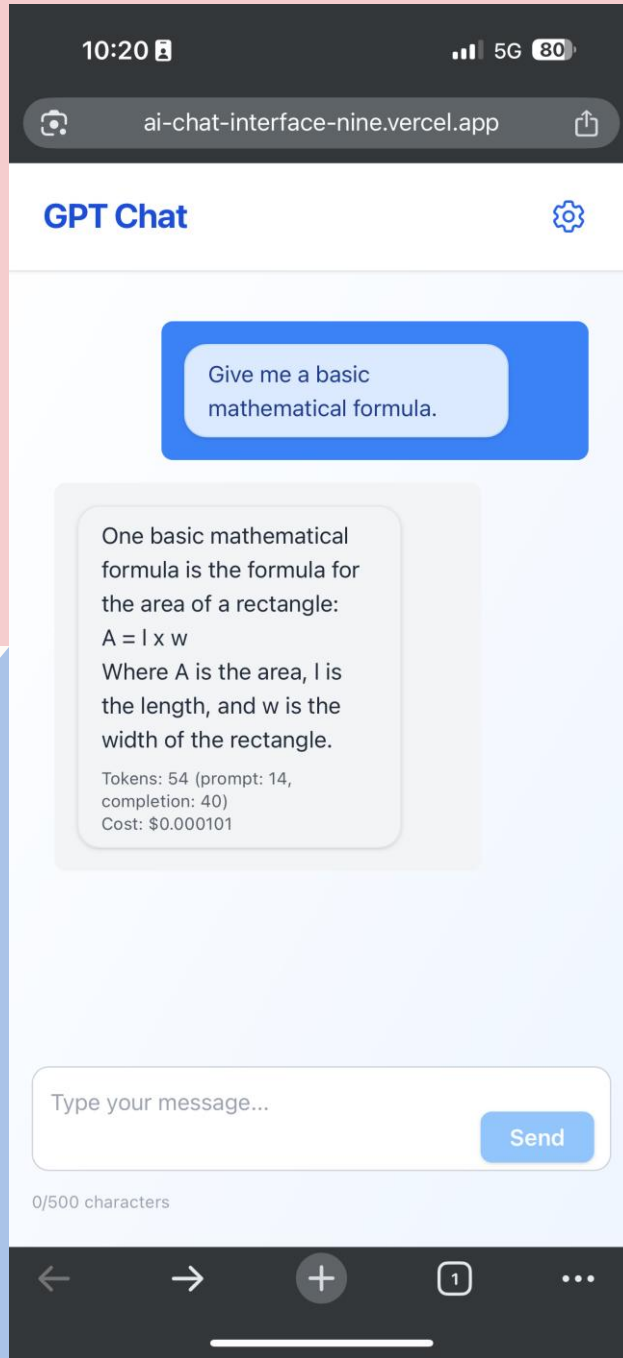
Chat Interface: Real-time messaging, markdown support with fast response.

Settings Panel: API key input, model selection, demo toggle.

Mobile View: Stacked layout, touch-friendly controls.

Demo Mode: Simulated chat for evaluation.

Cost Tracker: For real API cost.



DEVELOPMENT JOURNEY

6

Feature/Decision	Phase 1 Proposal	Phase 2 Implementation	Reason
API Key Handling	Secure proxy pattern - backend-only, never stored permanently	Session-based, backend proxy; API key validated per request, never stored	Improved security, assignment requirement
Demo Mode	Demo mode toggle (Settings Management dynamic aspect)	Fully implemented: toggleable, simple responses to test chat, no API key required	Helpful for evaluating chat functionality
UI Framework	React 18 + Vite + Tailwind CSS	Tailwind CSS only mobile-first, responsive, React functional comps	Assignment requirement, mobile-first focus
Error Handling	Input validation, rate limiting, HTTPS encryption	Comprehensive: input validation, rate limiting, global error middleware, user-friendly frontend errors	Assignment requirement, robust UX
Deployment	Not initially taken into consideration	Vercel deployment, HTTPS-ready, environment config, documented in <code>/docs/vercel_deployment_guide.md</code>	Production-readiness and simplicity during evaluation

NOTE: A more in depth documentation about changes between development phases is available at: /docs/design_choices_and_changes.md inside the repository.

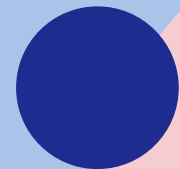
DYNAMIC ASPECTS

Real-Time Chat:

The chat interface enables users to exchange messages instantly with the AI. Features include live message updates, automatic scrolling to the latest message, and visual loading indicators while awaiting responses. This creates a seamless, interactive chat experience.

Chat Export:

Users can instantly export their entire chat history as a downloadable file, enabling easy sharing or record-keeping with a single click.



SECURITY & BEST PRACTICES

- **API keys never stored in frontend:** Entered by user, sent securely to backend per request, never persisted or exposed to third parties.
- **Input validation:** All user input is checked and sanitized on the backend.
- **Rate limiting & CORS:** Prevents abuse and restricts API access to trusted origins.
- **No sensitive errors:** Error messages never reveal private details.
- **HTTPS-ready:** Configured for secure deployment out of the box.

INSTALLATION & USAGE

Live app at: <https://ai-chat-interface-nine.vercel.app/>

Or Run Locally:

- **git clone** <https://github.com/NikVince/ai-chat-interface.git>
- Install dependencies:
 1. **cd backend && npm install**
 2. **cd ../frontend && npm install**
- Run backend:
 1. **npm start**
- Run frontend:
 1. **npm run dev**

Open in browser: <http://localhost:5173>

OpenAI Platform

- DASHBOARD
- Prompts
- Logs
- Traces
- Assistants
- Batches
- Evaluations
- Fine-tuning
- Storage
- Usage**
- API keys

Usage

Default project 06/16/25 - 07/01/25 Export

API capabilities **Spend categories**

Users **Services** API Keys

gpt-4, input

\$0.006 total

\$0.002



gpt-4, output

\$0.175 total

\$0.138



gpt-3_5-turbo-0125, input

\$0 total

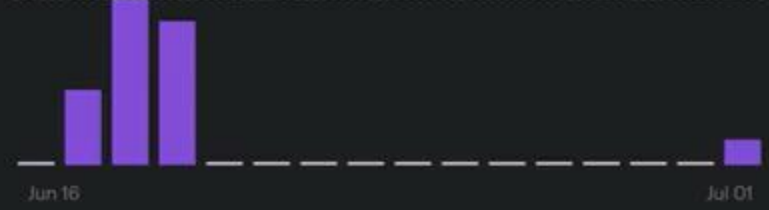
\$0



gpt-3_5-turbo-0125, output

\$0.001 total

\$0



There is no usage data for this period and group.

Claim free tokens
Get free tokens on traffic shared with OpenAI.



THANK YOU

Nikolas Daniel Vincenti

nikolas.vincenti@iu-study.org

nik.vincenti.dev@gmail.com

<https://github.com/NikVince>

CLOSING NOTE:

If the demo mode does not fully meet your evaluation needs, I can provide a temporary OpenAI API key upon request. Please feel free to contact me via one of the emails listed above.