

TechMaa AI Customer Support Chatbot

A lightweight, self-contained, and embeddable customer support chatbot powered by the Google Gemini API. This project provides a floating chat bubble on any website that opens into a clean, modern chat interface, offering instant AI-driven assistance to users.

🌟 Features

- **Floating Bubble UI:** Integrates seamlessly into any website as a floating action button.
- **Dynamic AI Responses:** Leverages the Google Gemini API for intelligent, context-aware conversations.
- **Rich Chat Interface:** Includes timestamps, user/bot avatars, typing indicators, and suggested questions.
- **Session-based Context:** The AI remembers the conversation history within a single session.
- **Easy Customization:** The bot's personality, knowledge, and appearance are easily configurable.
- **Start New Conversation:** A "Clear Chat" button allows users to reset the conversation at any time.
- **Responsive Design:** The chat window is optimized for both desktop and mobile views.

🔧 Tech Stack

- **Frontend:** HTML5, JavaScript (ES6)
- **Styling:** Tailwind CSS (via CDN for simplicity)
- **AI Backend:** Google Gemini API (gemini-1.5-flash-latest model)
- **Fonts:** Google Fonts (Inter)

🚀 Getting Started

Follow these instructions to set up and run the project on your own website or local environment.

1. Prerequisites

- A modern web browser.
- A text editor (like VS Code).

- A Google Gemini API Key. You can get one from [Google AI Studio](#).

2. Installation & Configuration

1. **Download the Code:** Clone or download the repository containing the index.html file.
2. **Add Your API Key:** Open the HTML file and find the following line in the <script> section:

```
const apiKey = "YOUR_GEMINI_API_KEY"; // IMPORTANT: Use your actual API key here!
```

Replace "YOUR_GEMINI_API_KEY" with your actual Google Gemini API key.

4. **Customize the Bot's Persona (Optional):** Modify the systemPrompt constant to define the chatbot's personality, instructions, and domain knowledge.

```
const systemPrompt = `You are a friendly, helpful, and expert AI customer support assistant for "Your Company Name Here"...`;
```

3. Running the Project

Since this is a self-contained HTML file with no build step, you can simply **open the index.html file directly in your web browser**.

For the best experience during development, use a live server extension (like "Live Server" in VS Code) to automatically reload the page when you make changes.

How It Works

The entire logic is contained within the <script> tag in the HTML file.

1. **initializeChat():** This function runs on page load and when the "Clear Chat" button is clicked. It resets the chat history and displays the initial welcome message and suggestion buttons.
2. **addMessage(sender, message):** This function dynamically creates and appends new message bubbles (either from the 'user' or 'bot') to the chat window.
3. **getAIResponse(prompt):** This is the core function. It displays a typing indicator, constructs a payload with the current conversation history and system prompt, sends a POST request to the Gemini API, and processes the returned response.