Telegram AI Chatbot - Project Documentation

1. Project Overview

The **Telegram AI Chatbot** is a smart assistant that integrates AI capabilities to enhance user interactions on Telegram. The bot can process text, analyze files, and provide intelligent responses using AI models. It is designed to improve user engagement by automating responses and providing AI-driven insights.

2. Features

- AI-powered chatbot responses
- File analysis capability
- Web search integration
- Chat history storage
- User metadata handling
- Configurable settings
- **Bonus Innovation:** Added emoji support 😊 🎉 and included comments in the code.

3. Technology Stack

- **Programming Language:** JavaScript (Node.js)
- **AI Model:** Gemini AI
- **Database:** MongoDB (if applicable)
- **Frameworks & Libraries:**
  - Telegram Bot API
  - Express.js
  - dotenv (for configuration management)
  - `mongoose`
  - `Nodemon`

4. Installation & Setup

Prerequisites

  - Node.js installed
  - A Telegram bot token from BotFather

Steps to Set Up

Step 1: Clone the Repository

Download the project from GitHub by running: **Command:**

```
git clone https://github.com/Saini-Yogesh/Telegram-AI-Chatbot.git
```

Step 2: Navigate to the Project Directory

Move into the cloned project folder: **Command:**

```
cd Telegram-AI-Chatbot
```

Step 3: Install Dependencies

Run the following command to install required dependencies: **Command:**

```
npm install
```

Step 4: Configure Environment Variables

Create a `.env` file and add the necessary environment configurations.

Step 5: Start the Server

Run the bot using: **Command:**

```
node server.mjs

      Or

npm start
```

5. Usage Guide

- Start the Telegram bot by running the server.
- Send messages in Telegram to interact with the AI bot.
- Upload files for analysis and AI-based insights.
- The bot will analyze message sentiment and respond accordingly.

6. File Structure Explanation

```
Telegram-AI-Chatbot/
│   ├── bot/
│   ├── bot.js         # Main bot logic
│   ├── fileAnalysices.js # File analysis module
│   ├── gemini.js       # AI model integration
│   ├── saveChat.js      # Chat storage module
│   ├── webSearch.js     # Web search module
│   ├── config/
│   ├── config.js       # Configuration settings
│   ├── models/
│   ├── chatHistory.js   # Manages chat logs
│   ├── fileMetadata.js  # Stores file-related data
│   ├── user.js         # Manages user information
│── server.mjs         # Entry point for the bot
│── .env             # Environment variables
│── .gitignore         # Files to ignore in Git
│── package.json       # Project dependencies
│── README.md         # Project documentation
```

7. AI Usage in Development

To enhance development efficiency, AI tools like GitHub Copilot and ChatGPT were used:

- **Boilerplate Code Generation:** Used AI to generate base structure for files, reducing setup time.
- **Code Commenting:** AI-assisted commenting for better readability and maintainability.
- **Feature Suggestions:** Leveraged AI to explore additional features like sentiment analysis.

8. Contributors & Contact Info

- **Developer:** Yogesh Saini
- **GitHub Repository:** Telegram-AI-Chatbot
- **Contact Email:** yogesh.saini4002@gmail.com
- **Contact Number:** +91 8852983794
- Loom Video: Loom video here