"Car? Sure!" Telegram Bot Documentation

1. Overview

"Car? Sure!" is a Telegram bot designed to assist users in obtaining car insurance policies.
It guides users through document submission, communicates with AI for data extraction and verification, and generates a PDF insurance policy.

---

2. Features

- User-friendly interaction via Telegram
- Step-by-step guidance for insurance policy application
- Integration with OCR (Mindee) and OpenAI APIs
- PDF generation with embedded fonts
- Secure and session-based user tracking

---

3. Technologies Used

- C# (.NET 8.0)
- Telegram Bot API
- Mindee OCR API
- OpenAI GPT
- PDFSharp
- Docker

---

4. How to Run the Bot

1. Clone the repository:
   https://github.com/Sparc0ctavo/CarSureBotDotNet
2. Set environment variable CARSURE_BOT_TOKEN
3. Build and run using .NET CLI:
   dotnet build && dotnet run
4. (Optional) Use Docker for deployment

---

5. Class Reference

CarSureBot.cs

Method	Description
StartAsync	Starts the bot, subscribes to events, initializes emoji statuses and key prompts. The main loop runs while _isRunning == true.
StopAsync	Stops the bot by cancelling the cancellation token _cts.
OnUpdate	Handles incoming Telegram updates. Determines the update type and delegates processing to the appropriate handler.
PingChat	Sends a ping action to check if chat still active.
UpdateTextMessageHandler	Processes incoming text messages. Determines the next step in the flow, sends replies, and manages navigation between scenario steps.
UpdatePhotoMessageHandler	Processes photo messages using the Mindee OCR service to extract text, stores results in session, and sends them to OpenAI for further processing.
InitStep	Sends the user a prompt message based on the current scenario step.
InitStepBack	Navigates the user back to the previous scenario step based on their current position (keyStepOrder).
ExtractPhoto	Downloads a photo from Telegram and returns it as a byte array.
OpenAiResponseAsync	Sends a prompt to the OpenAI API and returns the model's response.
SetKeyPrompts	Initializes a collection of textual prompts for each stage of the user interaction flow.

Program.cs

Method	Description
Main	Entry point. Loads environment, sets font resolver, and starts the bot.

UserSession.cs

Property / Field	Description
ChatId	Telegram user's chat ID
keyStepOrder	Current step/order of document processing
openAiApi	Instance of OpenAI service
userDocumentData	Holds extracted document data per chat session
Method	**Description**
ToStringUserDocumentData	Returns all user document data in string format

MindeeService.cs

Method	Description
GetTextData	Sends the document to Mindee OCR API and parses returned fields.

CustomFontResolver.cs

Method	Description
GetFont	Loads embedded font resource for PDF generation.
ResolveTypeface	Maps requested font family to embedded font definition.
LoadFontData	Retrieves the binary font data from embedded resources.

OpenAiService.cs

Method	Description
GetResponseAsync	Sends a chat prompt array to the OpenAI API using _chatClient. Returns the model's response as a string. Also logs and stores the assistant's reply.
SetBehaviorPattern	Iterates through predefined behavior rules (_behaviorRules), adds them as system messages, and invokes GetResponseAsync() to apply the behavior.

6. Environment Variables

Name	Description
CARSURE_BOT_TOKEN	Telegram bot API token.
OPENAI_TOKEN	OpenAI API token.
MINDEE_TOKEN	Mindee API token.