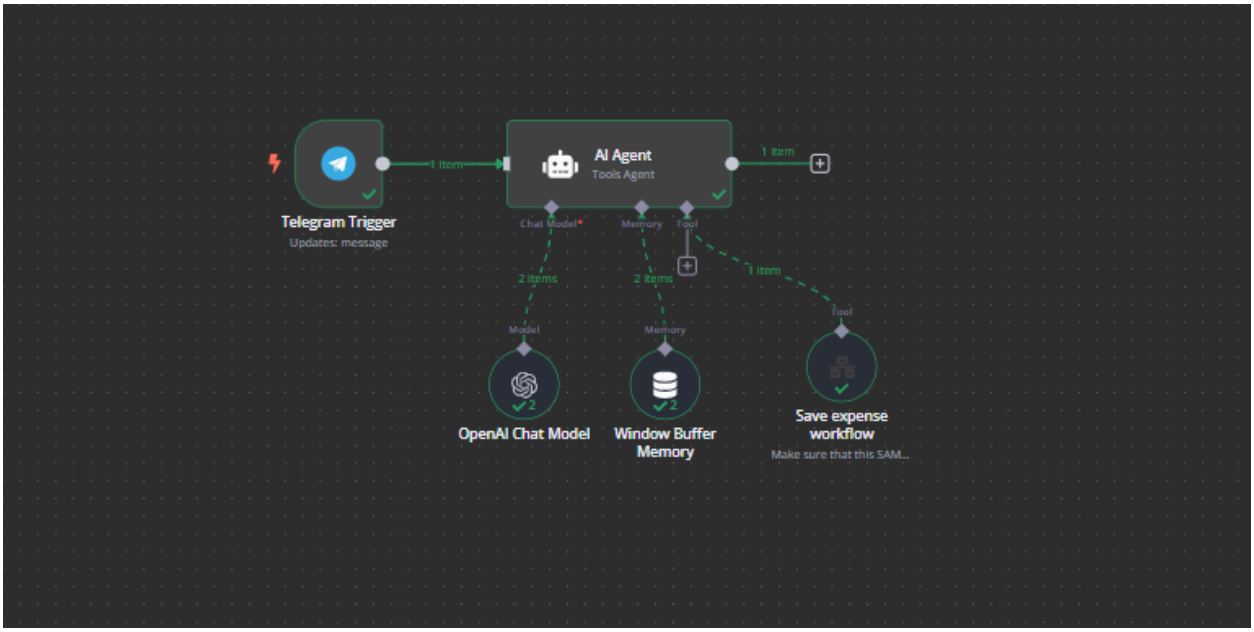
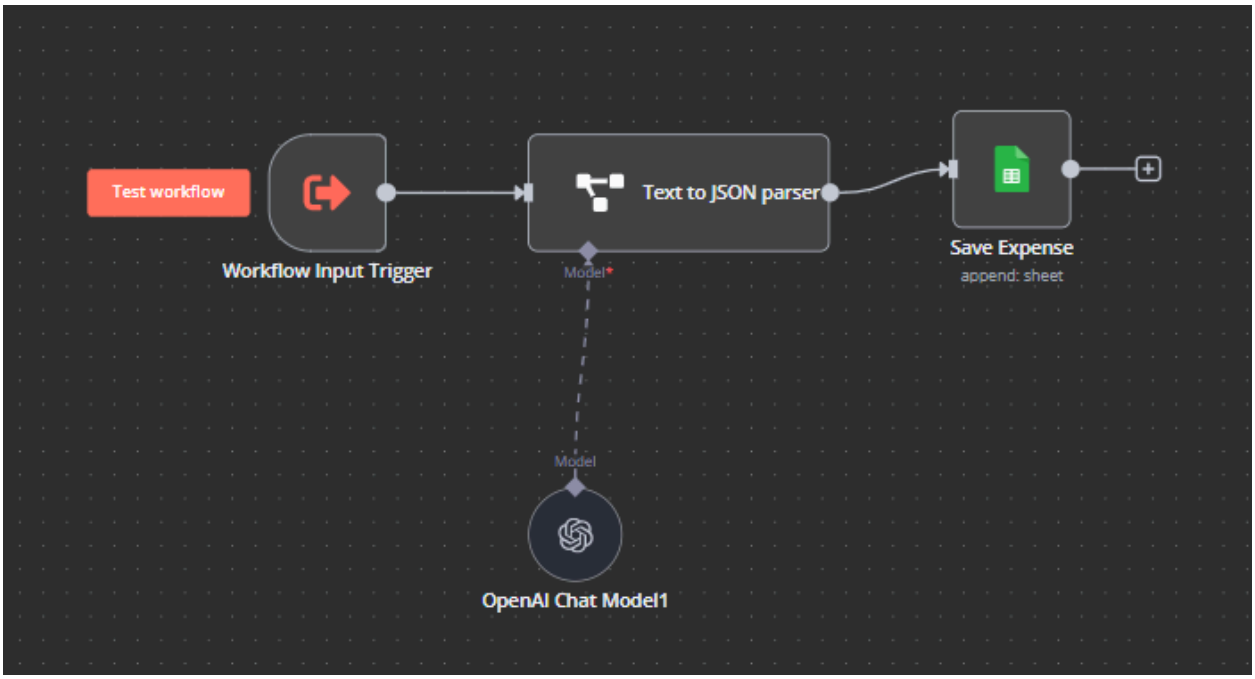
Expense Tracking Agent

This workflow automates **expense tracking** via Telegram. Users can send expense messages (e.g., **"Dinner at Italian Bistro; 42.75 USD; 10 Feb 2024"**) through **Telegram**, and the system will:

1. **Extract structured data** from the message.
2. **Save the expense into a Google Sheet database**.





How It Works

1. User Sends a Message in Telegram

- The **Telegram Trigger** (`Telegram Trigger` node) captures the message.

2. AI Processes the Message

- The **AI Agent** (`AI Agent` node) converts the message into structured JSON format.
- It then **calls a sub-workflow** (`Save Expense Workflow`) to store the data.

3. Data Parsing & Storage

- The **Text to JSON Parser** extracts:
  - **Cost**
  - **Description**
  - **Date** (or defaults to **current date** if missing)
- The **Save Expense** node logs this data into a **Google Sheet**.

Setup Instructions

Prerequisites

1. **Google Sheets Account** – Clone the Expense Tracker Sheet and link it to n8n.

2, **Telegram Bot** – Set up a bot and connect it to n8n via API.

3. **OpenAI API Key** – For AI-powered expense parsing.

Configuration Steps

This setup consists of two separate workflows that need to be imported separately.

1. Connect Google Sheets

- Open the **"Save Expense"** node.
- Enter your **Google Sheets credentials**.
- Select the copied **Expense Tracker Sheet**.

2. Configure Telegram Bot

- Open **"Telegram Trigger"** and add your **Telegram Bot Token**.
- Ensure the bot is added to your chat/group.

3. Fix Sub-Workflow Execution

- Open the **"Save Expense Workflow"** node.
- Ensure it is set to **run the correct sub-workflow (**Save_expense_workflow) that processes expenses.

4. Test the Agent

Send a message like:

`Lunch at Café de Paris; 25.50 USD; 12 March 2024`

1. The system should extract the expense details and save them.

2. You should receive a **confirmation response** (if enabled).

3. The entry should appear in **Google Sheets**.