

Kunde anlegen per Sprachnachricht

Kunden einfach einsprechen — über Telegram, WhatsApp oder Threema.
Die KI hört zu, Lexware legt den Kontakt an und meldet die Kundennummer
an E-Mail und alle drei Messenger zurück.

Version	2.0
Stand	16.08.2026
Status	Einsatzbereit
System	n8n-Workflow · Telegram · WhatsApp · Threema · KI (Whisper) · Lexware Office · E-Mail
Herausgeber	FlowTrix · flowtrix.de

Inhaltsverzeichnis

1 · Was ist das System?	3
2 · Voraussetzungen	3
3 · Erste Schritte	4
4 · Benutzer und Berechtigungen	4
5 · Hauptfunktionen	5
6 · Typischer Arbeitsablauf	6
7 · Die Ansicht	6
8 · Administration & Setup	7
9 · Datenschutz und Sicherheit	8
10 · Fehlerbehebung	8
11 · Häufige Fragen & Einschränkungen	9
12 · Support und Versionshistorie	9

Was ist das System?

Ein n8n-Workflow, der eine **Sprachnachricht** über **drei Messenger — Telegram, WhatsApp und Threema** — entgegennimmt, per KI transkribiert und in Kundendaten übersetzt, den Kontakt in **Lexware Office** anlegt und mit der **Kundennummer** bestätigt. Der Nutzer spricht den neuen Kunden einfach ein — der Rest passiert automatisch, egal über welchen Messenger.

KI-Variante von A02. Gleiche Aktion wie A02 (Kontakt anlegen), nur gesprochen statt aus Tabelle oder Visitenkarte. Wer A02 kennt, findet sich sofort zurecht.

Status: Einsatzbereit. Sprach-Eingang über drei Messenger, KI-Transkription und -Extraktion, Anlage in Lexware mit Kundennummer und Rückmeldung an E-Mail plus alle drei Messenger sind fertig gebaut. Vor dem ersten Lauf nur die eigenen Zugänge und Ziele in der Config eintragen (Kapitel 3 und 8).

Voraussetzungen

- Laufende **n8n-Instanz**
- Mindestens einer der drei Messenger: **Telegram-Bot, WhatsApp Cloud API, Threema** (es müssen nicht alle drei sein)
- **OpenAI-kompatibler KI-Zugang** mit `/audio/transcriptions` und `/chat/completions`
- **Lexware Office API** und der Workflow „Lexware – API-Call (Baustein)“
- Workflow „Messenger – Nachricht senden (Baustein)“ und ein **SMTP-Zugang** für die E-Mail

Erste Schritte

1. Workflow-Datei in n8n importieren.
2. Für jeden genutzten Eingangskanal das Credential zuweisen: **Telegram** (Trigger), **WhatsApp** (Trigger + Graph-API-Calls), **Threema** (Webhook-URL beim Gateway hinterlegen).
3. Header-Auth-Credential „**KI API (OpenAI)**“ in „**Transkribieren**“ und „**Daten extrahieren**“ zuweisen.
4. In beiden „**Baustein: ...**“-Lexware-Nodes den Ziel-Workflow „**Lexware – API-Call (Baustein)**“ im Dropdown neu wählen.
5. In „**Baustein: Nachricht senden**“ den **Messenger-Baustein** wählen (Mode „each“).
6. In „**Mail: Ausgabe**“ SMTP-Zugang und Absender setzen.
7. In **Config** die Ziele setzen: `aiBaseUrl`, `aiModel`, `chefEmail`, `absenderEmail`, `telegramChatId`, `whatsappZiel`, `threemaZiel`.
8. **Workflow aktiv schalten** und eine Sprachnachricht an einen Kanal senden.

Einzelne Kanäle: Ein leeres Ziel in der Config schaltet den jeweiligen Messenger ab — der Workflow läuft auch mit nur einem oder zwei Messengern. Telegram-Chat-ID: dem Bot `@userinfobot` schreiben.

Benutzer und Berechtigungen

Rolle	Aufgabe	Braucht
Sprecher	Kunden per Sprachnachricht einsprechen	Telegram / WhatsApp / Threema
Administrator	richtet Kanäle, KI, Bausteine, SMTP und Config ein	n8n-Zugang

Hauptfunktionen

5.1 Sprach-Eingang über drei Messenger

Telegram: der Trigger lädt die Voice-Message direkt als Audio-Binär in das Feld `data`. **WhatsApp:** der Trigger meldet die Audio-Nachricht, die Graph-API (`/mediaId`) liefert die Medien-URL, ein zweiter Call holt die Audiodatei. **Threema:** Eingang per Webhook. Jeder Kanal wird auf ein gemeinsames Format (`kanal`, `antwortZiel`, Audio-Binär) normalisiert — dahinter läuft ein einziger Ablauf.

5.2 KI-Transkription & Datenextraktion

Ein HTTP-Node sendet die Aufnahme an `/audio/transcriptions` (Modell `whisper-1`, `language=de`). Ein zweiter ruft `/chat/completions` mit `response_format=json_object` und liefert `{vorname, nachname, firma, strasse, plz, ort, email, telefon}`.

5.3 JSON parsen & Lexware-Body bauen

Ein Code-Node parst robust (try/catch, Fallback). Ein weiterer baut den Lexware-Body: Firma → Firmenkontakt (Person als Ansprechpartner), sonst Personenkontakt. Adresse, E-Mail, Telefon werden ergänzt, Land fest `DE`.

5.4 Anlegen, Kundennummer und Rückmeldung

Ohne erkennbaren Namen oder Firma fragt der Bot nach. Sonst legt der Baustein den Kontakt an (`POST /contacts`), ein zweiter Call liest die Kundennummer (`GET /contacts/{id}`). Jeder der drei Fälle — **angelegt** / **Fehler** / **unverständlich** — geht per **E-Mail** und über den **Messenger-Baustein** an Telegram, WhatsApp und Threema. Ein leeres Ziel wird übersprungen.

Einsatzbereit. Alle Funktionen sind gebaut; vor dem Produktivbetrieb nur die eigenen Zugänge und Ziele eintragen.

Typischer Arbeitsablauf

1. Kunden per Sprachnachricht einsprechen — über Telegram, WhatsApp oder Threema (z. B. „Leg mir den Kunden Müller an, Hauptstraße 12 in Rosenheim“).
2. Der Kanal wird normalisiert, die KI transkribiert und extrahiert die Felder.
3. n8n baut den Lexware-Body und legt den Kontakt an (`POST /contacts`).
4. Ein zweiter Call (`GET /contacts/{id}`) holt die Kundennummer.
5. Die Bestätigung „Kontakt angelegt: Müller — Kundennummer 10248“ geht an E-Mail und alle drei Messenger.

Die Ansicht

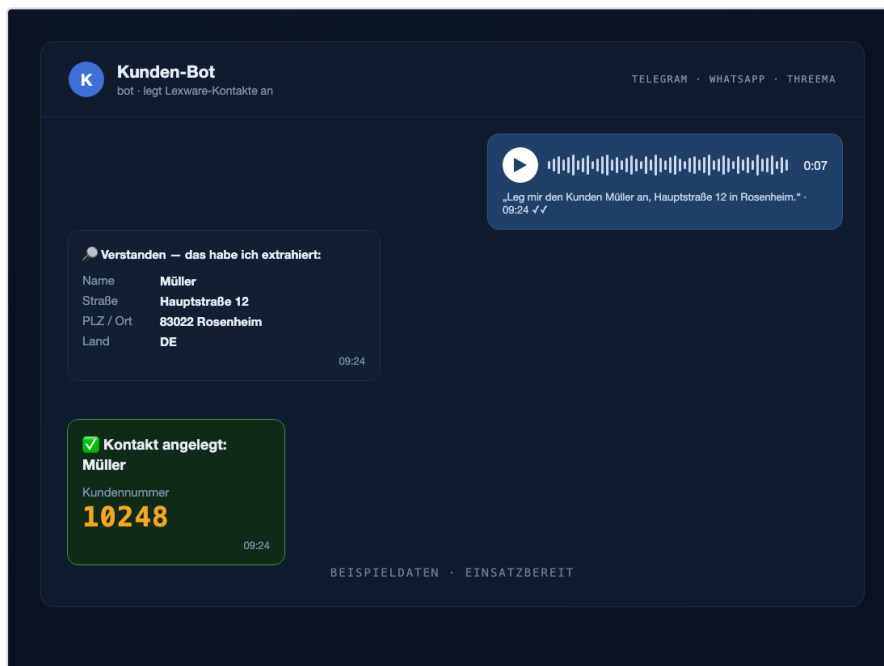


Abbildung 1: Sprachnachricht → extrahierte Daten → Bestätigung mit Kundennummer. Beispieldaten, keine echten n8n-Screenshots.

Administration & Setup

Die Grundeinstellungen liegen im Node **Config**.

Feld	Bedeutung	Standard
<code>aiBaseUrl</code>	OpenAI-kompatibler Endpunkt (ohne Slash am Ende)	<code>...</code> <code>openai.com/v1</code>
<code>aiModel</code>	Chat-Modell für die Extraktion	<code>gpt-4o-mini</code>
Transkriptionsmodell	fest im HTTP-Node „Transkribieren“	<code>whisper-1</code>
<code>chefEmail</code> / <code>absenderEmail</code>	Empfänger / Absender der E-Mail-Bestätigung	eigene Adresse
<code>telegramChatId</code>	Telegram-Ziel (leer = Kanal aus)	—
<code>whatsappZiel</code>	WhatsApp-Ziel (leer = Kanal aus)	—
<code>threemaZiel</code>	Threema-Ziel (leer = Kanal aus)	—

KI-Zugang, Messenger-Baustein und E-Mail

Beide KI-HTTP-Nodes teilen sich ein Header-Auth-Credential „**KI API (OpenAI)**“ mit `Authorization = Bearer <API-SCHLÜSSEL>`. Der Versand an die Messenger läuft über „**Baustein: Nachricht senden**“ (Mode „each“) — er sendet je aufgefächertem Ziel. Die E-Mail geht direkt über „**Mail: Ausgabe**“ (SMTP).

Warum zwei Lexware-Calls

`POST /contacts` liefert nur `id`, `resourceUri`, `version` — **nicht** die Kundennummer. Sie steht erst im vollen Kontakt unter `roles.customer.number`, daher der zweite `GET /contacts/{id}`.

Nur Platzhalter — echte Werte liegen ausschließlich im n8n-Credential-Store (`REPLACE_ME`, 16x).

Datenschutz und Sicherheit

- Verarbeitet werden die **Sprachaufnahme** und die daraus extrahierten **Kundendaten** (Name, Adresse, E-Mail, Telefon) — personenbezogen.
- Empfänger: die eigene n8n-Instanz, die genutzten Messenger, der gewählte KI-Dienst, der SMTP-Server und Lexware Office.
- **Zu bedenken:** Aufnahme und Kundendaten gehen an einen KI-Dienst. Bei Cloud-KI gehört das in die Auftragsverarbeitung; lokale Transkription (Whisper) reduziert den Datenabfluss.
- Zugangsdaten liegen ausschließlich im n8n-Credential-Store. Die Eingänge lassen sich auf bekannte Absender beschränken.

Fehlerbehebung

Problem	Mögliche Ursache	Lösung
„binary not found“ beim Transkribieren	Trigger ohne Audio / falsches Binärfeld	Telegram <code>download: true</code> ; WhatsApp Graph-Calls prüfen; Feld heißt <code>data</code>
Whisper lehnt die Datei ab	Dateiendung im <code>file</code> - Part fehlt	Dateinamen mit <code>.oga</code> / <code>.ogg</code> / <code>.mp3</code> setzen
Extraktion liefert kein JSON	Modell ignoriert <code>json_object</code>	anderes Modell wählen; sonst greift der Fallback
Kontakt wird nicht angelegt	weder Firma noch Nachname erkannt	deutlicher einsprechen — Bot fragt nach
Baustein-Fehler	„Baustein: ...“-Nodes zeigen auf <code>REPLACE_ME</code>	Ziel-Workflow im Dropdown neu wählen
Ein Messenger antwortet nicht	Ziel in Config leer/Platzhalter	Ziel setzen (<code>telegramChatId</code> u. a.)
Keine E-Mail	SMTP/Absender fehlt	„Mail: Ausgabe“ prüfen

Häufige Fragen & Einschränkungen

Muss der Sprecher etwas installieren?

Nein. Er schickt nur eine Sprachnachricht per Telegram, WhatsApp oder Threema.

Muss ich alle drei Messenger nutzen?

Nein. Nicht gesetzte Ziele werden übersprungen — ein Kanal genügt, zwei oder drei gehen genauso.

Welche KI kann ich nutzen?

Einen OpenAI-kompatiblen Dienst (OpenAI, Azure, OpenRouter oder lokal) mit

`/audio/transcriptions` und `/chat/completions`.

Braucht das Lexware?

Ja. Der Kontakt wird in Lexware Office angelegt — über den API-Baustein.

Bekannte Einschränkungen

- Adressen aus frei gesprochenem Text sind fehleranfällig (PLZ, Hausnummer) — Feinschliff ggf. in Lexware.
- Land wird fest als `DE` gesetzt.
- Keine Dublettenprüfung — dieselbe Person mehrfach eingesprochen wird mehrfach angelegt.

Support und Versionshistorie

FlowTrix · Kontakt: <https://flowtrix.de/kontakt/> · Projektübersicht: <https://flowtrix.de/projekte/>

Version	Datum	Änderung
2.0	16.08.2026	Mehrkanal-Ausbau: Sprach-Eingang über Telegram, WhatsApp und Threema; Ausgabe an E-Mail plus alle drei Messenger über den Messenger-Baustein (leeres Ziel übersprungen). Einsatzbereit.
1.0	22.07.2026	Erstausgabe. Sprachnachricht-Empfang (Telegram), KI-Transkription und -Extraktion, Lexware-Anlage über Baustein, Bestätigung mit Kundennummer.