

Foto-zu-Rechnung aus dem Angebot

Fertige Arbeit fotografieren, Angebotsnummer in die Bildunterschrift —
Lexware bekommt die Rechnung aus dem passenden Angebot, das Foto liegt als Nachweis
in der Cloud.

Version	2.0
Stand	16.08.2026
Karte	C02
Status	Einsatzbereit
System	n8n · Telegram/WhatsApp/Threema · Cloud · KI · Lexware
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	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 Monteur fotografiert die fertige Arbeit und schreibt in die Bildunterschrift die **Angebotsnummer oder den Kundennamen** — per Telegram, WhatsApp oder Threema. Das Foto wird als **Nachweis** in der Cloud abgelegt (Google Drive, Nextcloud oder HiDrive). Eine KI liest die Referenz aus der Bildunterschrift, der Workflow findet das passende **offene Angebot** und baut daraus die **Rechnung** — Positionen und Preise wie im Angebot vereinbart, als **Entwurf**. Die Meldung geht an E-Mail und Messenger.

Rechnung aus dem Angebot. Positionen und Preise stammen aus dem gefundenen Angebot — nichts wird erfunden. Der Beleg entsteht als Entwurf; das Büro prüft und gibt frei.

Folgebeleg sauber nachgebaut. Lexware Office kennt über die API keinen „Angebot-in-Rechnung-umwandeln“-Befehl. Der Workflow übernimmt Positionen und Kontakt aus dem Angebot und legt daraus über `POST /invoices` die Rechnung an — eine bewusste, kontrollierte Eigenschaft.

Foto als Nachweis, nicht im PDF. Ein Bild lässt sich per Lexware-API nicht in das Rechnungs-PDF einbetten. Das Foto liegt deshalb als Nachweis in der Cloud, und die Rechnung verweist im Vermerk auf den Ablageort.

Voraussetzungen

- Laufende **n8n-Instanz**
- Mindestens ein **Messenger**: Telegram-Bot · WhatsApp Business Cloud · Threema-Gateway
- **Cloud-Speicher** für den Nachweis: Google Drive · Nextcloud · HiDrive
- **KI-Credential** (OpenAI-kompatibel), Default `gpt-4o-mini`
- **Lexware Office API** über den Baustein `Lexware #1 – API-Call (Baustein)`
- **Messenger-Baustein** `Messenger – Nachricht senden (Baustein)` und ein **SMTP-Konto**

Erste Schritte

1. Workflow-Datei in n8n importieren.
2. **Eingang wählen:** Am genutzten Messenger das Credential setzen — Telegram-Trigger (Option „**Download Image/File**“ aktivieren), WhatsApp-Trigger + Graph-Token, oder die Threema-Webhook-URL im Gateway hinterlegen.
3. Node „**Referenz erkennen**“ → Credential „**KI API (OpenAI)**“ (Header-Auth: `Authorization = Bearer <Key>`) zuweisen; in Config `aiBaseUrl` / `aiModel` prüfen.
4. Die drei Nodes „**Baustein: Angebote laden**“ / „**Baustein: Angebot laden**“ / „**Baustein: Rechnung anlegen**“ → den Lexware-Baustein im Dropdown neu wählen.
5. Node „**Baustein: Nachricht senden**“ → den Messenger-Baustein wählen. Node „**Mail: Ausgabe**“ → SMTP-Credential setzen.
6. In **Config**: `cloud` (googledrive/nextcloud/hidrive) + Ordner/Pfad, Ausgabe-Ziele (`chefEmail`, `telegramChatId`, `whatsappZiel`, `threemaZiel`), `absenderEmail`.
7. **Workflow aktiv schalten.**
8. Ein Foto mit Angebotsnummer/Kunde in der Bildunterschrift schicken und den Rechnungsentwurf in Lexware prüfen.

Standard: Entwurf. Die Rechnung entsteht als Entwurf und wird im Büro geprüft. Soll sie direkt finalisiert werden, `rechnungFinalisieren = true` setzen (siehe Kapitel 8) — erst nach bewusster Prüfung.

Benutzer und Berechtigungen

Rolle	Aufgabe	Braucht
Monteur (Absender)	Foto per Messenger, Angebotsnummer/Kunde in die Bildunterschrift	Messenger
Büro	Rechnungsentwurf prüfen und freigeben	Lexware
Administrator	richtet Messenger, Cloud, KI, Bausteine, SMTP, Config ein	n8n-Zugang

Hauptfunktionen

5.1 Foto-Empfang über drei Messenger

Telegram, WhatsApp und Threema nehmen Foto und Bildunterschrift entgegen. Jeder Kanal wird auf eine einheitliche Form normalisiert (Absender, Bildunterschrift, Antwortziel, Foto-Binär). Bei WhatsApp holt die Graph-API das Bild über die Medien-ID.

5.2 Nachweis-Ablage in der Cloud

Aus Absender, Bildunterschrift und Datum entstehen Ablagepfad und Dateiname. Das Foto wird in die gewählte Cloud geladen (Google Drive, Nextcloud oder HiDrive, per `cloud`-Feld umschaltbar). Der Ablageort wird als Link bzw. Vermerk aufbereitet und in die Rechnung übernommen.

5.3 Referenz erkennen (KI)

Der Node „Referenz erkennen“ ruft `POST {aiBaseUrl}/chat/completions` mit `temperature: 0` und `response_format: json_object` auf und liest aus der Bildunterschrift striktes JSON `{angebotsnummer, kundennamen}`. Es werden keine Werte erfunden.

5.4 Angebot finden und laden

Der Workflow lädt die offenen Angebote (`/voucherlist`, `voucherType: quotation`, `voucherStatus: open`) und sucht den Treffer nach Nummer oder Kundennamen (neuestes zuerst). Das gefundene Angebot wird über `GET /quotations/{id}` mit Positionen und Kontakt geladen.

5.5 Rechnung bauen und melden

„req: Rechnung bauen“ übernimmt die Positionen (inkl. Preise) und den Kontakt aus dem Angebot und legt über `POST /invoices` die Rechnung als **Entwurf** an; der Nachweis-Ablageort steht im Vermerk. Erfolg, „kein Angebot gefunden“ und Fehler werden an E-Mail und die gesetzten Messenger-Ziele gemeldet.

Einsatzbereit. Alle Funktionen sind gebaut; vor dem ersten Lauf werden nur die eigenen Zugangsdaten und Ziele gesetzt.

Typischer Arbeitsablauf

1. Monteur fotografiert die fertige Arbeit, schreibt Angebotsnummer/Kunde in die Bildunterschrift und schickt das Foto per Messenger.
2. Das Foto wird als Nachweis in die Cloud geladen.
3. Die KI liest Angebotsnummer/Kundenname aus der Bildunterschrift.
4. Der Workflow findet das passende offene Angebot und lädt dessen Positionen.
5. Aus den Angebots-Positionen entsteht der Rechnungsentwurf; der Nachweis-Ablageort steht im Vermerk.
6. Der Absender erhält die Meldung per E-Mail und Messenger; das Büro prüft den Entwurf in Lexware.

Die Ansicht

FLOWTRIX SHOWCASE · FOTO-ZU-RECHNUNG

Vom Foto zum Rechnungsentwurf

Foto per Messenger mit Angebotsnummer in der Bildunterschrift → Foto als Nachweis in der Cloud → der Workflow findet das offene Angebot und baut daraus die Rechnung. Beispieldaten.

Foto-zu-Rechnung-Bot
Telegram · Kevin Bauer



FOTO · FERTIGES BAD

AG-2026-0087 Familie Weber
Bildunterschrift = Angebotsnummer / Kunde

Nachweis abgelegt: Google Drive · Kevin-Bauer / 2026-08-16

→
ANGEBOT
GEFUNDEN

Rechnung ENTWURF

Aus Angebot: **AG-2026-0087**
Kunde: **Familie Weber** · Badsanierung

Position aus dem Angebot		Betrag
Wandfliesen verlegen	24 m²	1.080,00 €
Bodenfliesen verlegen	8 m²	416,00 €
Silikont fugen erneuern	12 l/m	102,00 €
Fliesenkleber	6 Sack	108,00 €
Netto (aus dem Angebot)		1.706,00 €

Entwurf — im Büro prüfen und freigeben
Positionen und Preise aus dem Angebot übernommen. Nachweisfoto in der Cloud abgelegt, im Vermerk verlinkt. Beispieldaten.

FlowTrix · Automatisierung · Telegram/WhatsApp/Threema → Lexware

Abbildung 1: Foto + Bildunterschrift → gefundenes Angebot → Rechnungsentwurf (Positionen und Preise aus dem Angebot, Nachweis-Link). Beispieldaten.

Administration

Alle Einstellungen liegen im Node **Config**.

Feld	Bedeutung	Standard
<code>aiBaseUrl</code>	Basis-URL der KI (OpenAI-kompatibel)	<code>...openai.com/v1</code>
<code>aiModel</code>	Modell zum Auslesen der Bildunterschrift	<code>gpt-4o-mini</code>
<code>binaerProperty</code>	Name des Foto-Binärfeldes	<code>data</code>
<code>cloud</code>	Ablageort des Nachweises	<code>googledrive</code>
<code>googleDriveOrdnerId</code>	Ziel-Ordner in Google Drive	<i>Ordner-ID</i>
<code>nextcloudBasisPfad</code>	Basispfad in Nextcloud	<code>/Rechnungsnachweise</code>
<code>hidriveBaseUrl</code>	WebDAV-Basis-URL bei HiDrive	<i>WebDAV-URL</i>
<code>rechnungFinalisieren</code>	<code>false</code> = Entwurf, <code>true</code> = finalisiert	<code>false</code>
<code>telegramChatId</code> / <code>whatsappZiel</code> / <code>threemaZiel</code>	Ausgabe-Ziele je Messenger (leer = übersprungen)	<i>je Ziel</i>
<code>chefEmail</code> / <code>absenderEmail</code>	E-Mail-Empfänger und Absender (SMTP)	<i>E-Mail</i>

Rechnung finalisieren — optional

Standard ist der **Entwurf** — das Büro prüft und gibt frei. Soll die Rechnung direkt finalisiert werden, `rechnungFinalisieren = true` setzen; der Workflow ruft dann `POST /invoices` mit `finalize=true` auf. Erst nach bewusster Prüfung einsetzen — eine finalisierte Rechnung ist buchhalterisch bindend.

Zugangsdaten

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

Dienst	Credential
Messenger	Telegram-Bot / WhatsApp Graph-Token / Threema-Gateway
Cloud	Google Drive OAuth / Nextcloud / HiDrive WebDAV (Basic Auth)
KI	„KI API (OpenAI)“ · Header-Auth <code>Bearer <Key></code>
Lexware	über den Baustein-Workflow
E-Mail	SMTP-Konto

Datenschutz und Sicherheit

- Verarbeitet werden **Fotos und Metadaten** (Absender, Bildunterschrift, Antwortziel). Das Foto geht als Nachweis in die Cloud, die Bildunterschrift (Text) zur Referenz-Erkennung an den KI-Anbieter.
- Empfänger: eigene n8n-Instanz, Messenger-Dienst, Cloud-Speicher, KI-Anbieter, Lexware Office (über den Baustein), SMTP-Server.
- Positionen, Preise und Kontakt stammen aus dem in Lexware vorhandenen Angebot.
- Zugangsdaten liegen ausschließlich im n8n-Credential-Store. Der Messenger-Eingang kann auf bekannte Absender beschränkt werden.
- Die Rechnung entsteht standardmäßig als Entwurf — die Finalisierung ist ein bewusster, per Config schaltbarer Schritt.

Fehlerbehebung

Problem	Mögliche Ursache	Lösung
Kein Foto / „binary not found“	Telegram: „Download Image/File“ nicht aktiv oder falscher Feldname	Option aktivieren; <code>binaerProperty</code> prüfen (oft <code>data</code>)
Trigger reagiert nicht	Workflow nicht aktiv	Workflow aktiv schalten
„Kein Angebot gefunden“	keine/unklare Angebotsnummer oder Kunde in der Bildunterschrift	Angebotsnummer oder eindeutigen Kundennamen angeben; Angebot muss offen sein
Foto liegt nicht in der Cloud	falsche <code>c l o u d</code> -Wahl oder Cloud-Credential fehlt	<code>c l o u d</code> -Feld und Cloud-Credential prüfen
Rechnung wird nicht angelegt	Lexware lehnt Body ab / Baustein nicht gewählt	Fehlertext prüfen; Baustein im Dropdown wählen
Referenz-Fehler / 401	KI-Credential fehlt	„KI API (OpenAI)“ setzen
Keine Messenger-Meldung	Zielfeld leer oder <code>REPLACE_ME</code>	Ausgabe-Ziel in Config setzen

Häufige Fragen & Einschränkungen

Erfindet die KI Preise?

Nein. Positionen und Preise stammen aus dem gefundenen Angebot. Die KI liest nur die Referenz (Angebotsnummer/Kunde) aus der Bildunterschrift.

Wird eine echte Rechnung gebucht?

Standardmäßig entsteht ein Entwurf. Die Finalisierung ist ein bewusster, per Config schaltbarer Schritt (`rechnungFinalisieren`).

Warum baut der Workflow die Rechnung nach?

Lexware Office kennt über die API keinen „Angebot-in-Rechnung-umwandeln“-Befehl. Der Workflow übernimmt Positionen und Kontakt aus dem Angebot und legt daraus sauber die Rechnung an.

Ist das Foto in der Rechnung?

Nein. Ein Bild lässt sich per API nicht ins Rechnungs-PDF einbetten — das Foto liegt als Nachweis in der Cloud, die Rechnung verweist darauf.

Bekannte Einschränkungen

- Es muss zu Nummer/Kunde ein **offenes Angebot** geben — sonst gibt es nur die „kein Angebot“-Meldung (das Foto ist dann bereits abgelegt).
- Die Angebotsnummer oder ein eindeutiger Kundenname in der Bildunterschrift führt am sichersten zum richtigen Angebot.
- Das Nachweisfoto liegt in der Cloud, nicht im Rechnungs-PDF (API-Grenze von Lexware).

Support und Versionshistorie

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

Version	Datum	Änderung
2.0	16.08.2026	Neu gedacht: Rechnung aus dem offenen Angebot (Positionen + Preise wie vereinbart, <code>POST /invoices</code> als Entwurf), Foto als Nachweis in der Cloud (Google Drive/Nextcloud/HiDrive, umschaltbar), Eingang und Ausgabe über drei Messenger (Telegram/WhatsApp/Threema) plus E-Mail. Folgebeleg nachgebaut.
1.0	22.07.2026	Erstausgabe (Foto → frischer Angebotsentwurf per KI-Vision).