# Inhaltsverzeichnis

# Verfahrensdokumentation – Agynamix Invoicer

**Software:** Agynamix Invoicer
**Zweck:** Erstellung, Verwaltung und revisionssichere Archivierung von Verkaufsbelegen und Zeiterfassungen gemäß GoBD
**Geltungsbereich:** Finanzbuchhaltung / Einnahmen-Überschuss-Rechnung / Rechnungsstellung
**Stand:** 01.12.2025

---

## 1. Systembeschreibung

### 1.1 Einsatzzweck

Agynamix Invoicer ist eine Desktop-Anwendung zur Erstellung und Verwaltung von: - Angeboten, Verkaufsdokumenten (Rechnungen), Korrekturdokumenten und Stornobelegen - Zeiterfassungen (Timesheets) und deren Abrechnung

Die Anwendung unterstützt dich dabei, deine Beleg- und Buchführungsprozesse GoBD-konform zu dokumentieren und die Daten revisionssicher zu archivieren.

### 1.2 Systemumgebung

- **Betriebssysteme:** Windows, macOS, Linux (Desktop)
- **Applikation:** Lokale Desktop-App, keine zentrale Server-Komponente
- **Datenbank:** Lokale SQLite-Datenbank
  - Windows: `%APPDATA%\agynamix-invoicer`,
  - MacOS: `~/Library/Application Support/agynamix-invoicer`,
  - Linux: `~/.config/agynamix-invoicer/invoicer_db.db`)
- **Dateiablage:** Exportierte Dokumente und WORM-Archiv im Dateisystem (konfigurierbares Verzeichnis)

Backups der Datenbank und des WORM-Archivs sind organisatorische Pflicht des Nutzers.

---

## 2. Geschäftsprozesse

### 2.1 Stammdaten

- Anlage und Pflege von Mandanten (Tenants)
- Anlage und Pflege von Kunden
- Pflege von Nummernkreisen für Verkaufsdokumente

Stammdatenänderungen werden im Prüfprotokoll (Audit Log) nachvollziehbar protokolliert.

### 2.2 Belegprozesse

**Erstellung von Verkaufsdokumenten**

- Belege werden zunächst als **Entwurf** angelegt.
- Beim Veröffentlichen erhält der Beleg eine **fortlaufende, Mandanten-spezifische Nummer** aus dem Nummernkreis.
- Nach der Veröffentlichung gelten Belege als **unveränderbar**. Inhaltliche Änderungen erfolgen ausschließlich über Korrektur- oder Stornobelege.

**Zahlungen**

- Zahlungen werden Belegen zugeordnet (teilweise oder vollständig bezahlt).
- Zahlungen können nicht durch Löschung des Belegs verdeckt werden (FK-Constraints mit `ON DELETE RESTRICT`).

**Korrekturen und Stornos**

- Fehler in veröffentlichten Belegen werden über **Korrekturdokumente** bzw. Stornobelege korrigiert.
- Der ursprüngliche Beleg bleibt erhalten, alle Korrekturen werden im Prüfprotokoll dokumentiert und über Transaktions-IDs verknüpft.

### 2.3 Zeiterfassung

- Zeiteinträge werden pro Kunde/Projekt erfasst.
- Aus Zeiteinträgen können abrechenbare Belege (Rechnungen) erzeugt werden.

- Relevante Statusänderungen werden im Prüfprotokoll protokolliert.

---

## 3. GoBD-Anforderungen und Umsetzung

### 3.1 Ordnungsmäßigkeit

**Anforderung:** Das System muss nachvollziehbar dokumentiert und organisiert sein.

**Umsetzung:**

- Diese Verfahrensdokumentation beschreibt System, Prozesse und Datenflüsse.
- Die Benutzeranleitung (Kapitel „GoBD-Compliance“) erklärt dir die tägliche Nutzung.
- Technische Details zu Prüfprotokoll, Hashing, Export und WORM-Speicher sind in einer separaten technischen Dokumentation beschrieben (`doc/GoBD/GOBD_COMPLIANCE.md`).

### 3.2 Vollständigkeit

**Anforderung:** Alle relevanten Geschäftsvorfälle müssen vollständig erfasst werden.

**Umsetzung:**

- Alle wesentlichen Ereignisse (Erstellung, Veröffentlichung, Korrektur, Zahlung, Export, WORM-Speicherung) werden im **Prüfprotokoll** als Audit-Events aufgezeichnet.
- Belege erhalten eine eindeutige, sequenziell vergebene Dokumentnummer je Mandant.
- Löschverbote und Fremdschlüssel mit `ON DELETE RESTRICT` verhindern das „Verschwinden“ von Belegen, Nummernkreisen, Zahlungen und Timesheets.

### 3.3 Richtigkeit

**Anforderung:** Aufzeichnungen müssen sachlich richtig sein.

**Umsetzung:**

- Belege können vor der Veröffentlichung als Entwurf geprüft und korrigiert werden.

- Nach Veröffentlichung sind nur noch Korrektur- bzw. Stornobelege erlaubt.

- Fehlerhafte Belege bleiben erhalten und werden durch gegenläufige Korrekturen neutralisiert, sodass der Verlauf nachvollziehbar bleibt.

### 3.4 Zeitgerechte Erfassung

**Anforderung:** Geschäftsvorfälle müssen zeitnah erfasst werden.

**Umsetzung:**

- Belege enthalten Erfassungs- und Buchungsdaten (z. B. Rechnungsdatum).

- Änderungen und Statuswechsel werden mit Zeitstempel im Prüfprotokoll festgehalten.

- Die Anwendung unterstützt dich mit Workflows für zeitnahe Erstellung und Export.

### 3.5 Unveränderbarkeit

**Anforderung:** Einmal erfasste Daten dürfen nicht unbemerkt verändert oder gelöscht werden.

**Umsetzung (Anwendungslogik):**

- Veröffentliche Belege sind in der Datenbank **logisch unveränderbar**:
  - `update`-Operationen auf veröffentlichte, bezahlte oder stornierte Dokumente werden durch Validierung und Fehlercodes blockiert.

  - Änderungen erfolgen ausschließlich durch neue Korrektur-/Stornobelege.

- Fremdschlüssel mit `ON DELETE RESTRICT` verhindern das Löschen relevanter Daten (Belege, Zahlungen, Timesheets, Nummernkreise).

**Umsetzung (Prüfprotokoll & Hash-Kette):**

- Alle Audit-Log-Einträge enthalten eine **kryptographische Verkettung** über `sequence_number`, `previous_entry_hash` und `current_entry_hash`.

- Änderungen an bestehenden Einträgen oder das Entfernen von Einträgen würden die Hash-Kette brechen und bei einer Verifikation auffallen.

**Umsetzung (WORM-Archiv):**

- Exportierte PDF/XML-Dokumente können in ein **WORM-Archiv** verschoben werden.

- Je nach Betriebssystem werden Dateiattribute und Berechtigungen so gesetzt, dass normale Benutzer Dateien nicht mehr verändern oder löschen können (z. B. ACLs, `uchg`-Flag, POSIX-Rechte).

- Die Unveränderbarkeit wird zusätzlich durch gespeicherte **SHA-256-Hashes** der Dateien abgesichert; jede spätere Änderung lässt sich durch Vergleich erkennen.

### 3.6 Nachvollziehbarkeit

**Anforderung:** Geschäftsvorfälle müssen vom Ursprung bis zum Abschluss nachvollziehbar sein.

**Umsetzung:**

- Das Prüfprotokoll protokolliert alle wesentlichen Ereignisse mit:
  - Mandant, Kunde, Beleg-ID, Dokumentnummer
  - Ereignistyp (z. B. `DOCUMENT_PUBLISHED`, `DOCUMENT_CORRECTED`, `PAYMENT_ADDED`, `DOCUMENT_EXPORTED_PDF`, `DOCUMENT_STORED_WORM`)
  - Zeitstempel und Benutzerkontext (soweit verfügbar)
  - Transaktions-IDs zur Verknüpfung zusammengehöriger Ereignisse
- Aus dem Prüfprotokoll lässt sich der komplette Lebenszyklus eines Belegs (von Erstellung über Veröffentlichung, Zahlungen, Korrekturen bis zur Archivierung) lückenlos nachvollziehen.

### 3.7 Nachprüfbarkeit

**Anforderung:** Ein Betriebsprüfer muss die Ordnungsmäßigkeit der Daten überprüfen können.

**Umsetzung:**

- Die Anwendung bietet einen **GoBD-Export** (Z3-Datenträgerüberlassung):

  - ZIP-Archiv mit CSV-Datei (Audit-Log), Verifikations-JSON und den referenzierten Dokumenten (PDF/XML).

  - Die JSON-Datei enthält Prüfsummen und Metadaten zur Verifikation der Hash-Kette und der Dokumentdateien.

- In-App-Tools ermöglichen:
  - Verifikation der Audit-Log-Kette (Hash- und Sequenzprüfung).

  - Verifikation der Export-Archive und Dokument-Hashes.

- Der Prüfer kann darüber hinaus das CSV und die JSON-Datei mit eigenen Werkzeugen auswerten und Stichproben durchführen.

---

## 4. Technische Funktionsbeschreibung (Kurzfassung)

### 4.1 Prüfprotokoll (Audit Log)

- Alle relevanten Ereignisse werden in der Tabelle `audit_log` gespeichert.

- Wichtige Felder:
  - `tenant_id`, `entity_type`, `entity_id` (Bezug auf Belege, Zahlungen, Timesheets)

  - `sequence_number` (mandantenspezifische, fortlaufende Nummer)

  - `event_type` und strukturierte Ereignisdaten

  - `previous_entry_hash`, `current_entry_hash` (Hash-Verkettung)
- Die `current_entry_hash`-Werte werden aus dem Inhalt des Eintrags und dem `previous_entry_hash` berechnet.

- Eine Manipulation einzelner Einträge oder das Entfernen von Einträgen führt zu einer erkennbar gebrochenen Hash-Kette.

### 4.2 Dateihashes und Export-Metadaten

- Beim Export von PDF- oder XML-Dokumenten werden **SHA-256-Hashes** der Dateien berechnet und:
  - Im Audit Log als Ereignisdaten hinterlegt.

  - In Dateinamen und/oder Sidecar-Metadateien (`.meta.txt`) abgelegt.

- Export-Archive enthalten eine Verifikations-JSON, die alle relevanten Hashes und Prüfsummen zusammenfasst.

### 4.3 WORM-Speicher

- Für das WORM-Archiv wird ein Verzeichnis im Dateisystem genutzt, dessen Pfad du konfigurierst.

- Beim Verschieben von Dateien in dieses Archiv sorgt der WORM-Dienst je nach Plattform für:
  - Schreibschutz und Löschschutz auf Datei- und Ordnerebene (z. B. ACLs, `uchg`, POSIX-Rechte).

  - Optionale Sperrung übergeordneter Ordner, um das Anlegen/Löschen von Dateien zu verhindern.

- Die tatsächliche GoBD-Konformität des Speichermediums (z. B. Backup, physische Sicherheit, Admin-Rechte) liegt in deiner organisatorischen Verantwortung.

---

## 5. Organisatorische Maßnahmen

### 5.1 Verantwortlichkeiten

- Du (bzw. dein Unternehmen) bist verantwortlich für:
  - Einrichtung und Konfiguration der Anwendung (Mandanten, Nummernkreise, WORM-Archiv).

  - Regelmäßige Backups von Datenbank und WORM-Archiv.

  - Nutzung der bereitgestellten Export- und Verifikationsfunktionen.

  - Schulung der Mitarbeitenden im Umgang mit GoBD-relevanten Funktionen.

### 5.2 Arbeitsabläufe

Empfohlene Standardprozesse: - **Monatliche Exporte:** Für jeden Monat ein GoBD-Export-Archiv erstellen und sicher ablegen (z. B. im WORM-Archiv oder in einem revisionssicheren Cloud-Speicher).

- **Regelmäßige Verifikation:** In regelmäßigen Abständen (z. B. quartalsweise) das Prüfprotokoll und ausgewählte Export-Archive verifizieren.
- **Korrekturen:** Fehlerhafte Rechnungen ausschließlich über Korrekturdokumente bzw. Stornos berichtigen, nicht durch manuelle Datenbankeingriffe oder direkte Dateimanipulation.
- **Keine manuellen DB-Änderungen:** Direkte Änderungen an der Datenbank oder am WORM-Archiv sind unzulässig und gefährden die GoBD-Compliance.

### 5.3 Vorbereitungen für Betriebsprüfungen

Für eine Betriebsprüfung solltest du: - Diese Verfahrensdokumentation und die Benutzeranleitung bereithalten.
- Für die geforderten Zeiträume GoBD-Exporte erstellen und dem Prüfer als ZIP bereitstellen (Z3).
- Auf Wunsch des Prüfers die Verifikationsfunktionen der Anwendung demonstrieren (Prüfprotokoll- und Export-Verifikation).
- Bei Z1/Z2-Zugriff dem Prüfer an einem Arbeitsplatz Einsicht in die Anwendung geben.

---

## 6. Datenzugriff der Finanzverwaltung (Z1, Z2, Z3)

- **Z1 (Unmittelbarer Zugriff):** Der Prüfer arbeitet direkt an deinem System mit der Anwendung. Du stellst Benutzerzugang und betreute Einsicht in Belege, Prüfprotokoll und Auswertungen bereit.

- **Z2 (Mittelbarer Zugriff):** Du führst auf Anweisung des Prüfers Auswertungen, Filterungen oder Exporte in der Anwendung aus und stellst dem Prüfer die Ergebnisse zur Verfügung.

- **Z3 (Datenträgerüberlassung):** Du erstellst mit der Exportfunktion GoBD-konforme ZIP-Archive (CSV, JSON, Dokumente) für die angeforderten Zeiträume und übergibst sie dem Prüfer auf Datenträger oder über ein anderes vereinbartes Medium.

---

## 7. Versionierung und Pflege der Dokumentation

- Diese Verfahrensdokumentation wird bei relevanten Änderungen an der Software oder an GoBD-relevanten Funktionen aktualisiert.

- Release-Notes der Anwendung dokumentieren funktionale Änderungen; technische Details sind in `doc/GoBD/GOBD_COMPLIANCE.md`

beschrieben.

- Es wird empfohlen, diese Dokumentation zusammen mit den GoBD-Exporten und der Benutzeranleitung aufzubewahren, damit sie im Fall einer Betriebsprüfung jederzeit verfügbar ist.

---

## 8. Bestätigung des Steuerpflichtigen

Ich bestätige, dass ich diese Verfahrensdokumentation gelesen habe und dass die hierin beschriebenen organisatorischen Maßnahmen und Arbeitsabläufe meinem tatsächlichen Einsatz der Software entsprechen.

Ort, Datum: _____

Name (in Druckbuchstaben): _____

Unterschrift: _____