

Pflichtenheft

Desktop-Fakturierungsanwendung — Gruppe C: Verwaltung von Kunden

Team 1 – Gruppe C

Inhaltsverzeichnis

Dokumentenhistorie	2
1. Einleitung	3
1.1 Zweck des Dokuments	3
1.2 Ziel	3
1.3 Geltungsbereich	3
1.4 Definitionen und Abkürzungen	4
1.5 Referenzen	4
2. Systemüberblick	4
2.1 Kurzbeschreibung	4
2.2 Abgrenzung (Was gehört dazu / was nicht)	4
2.3 Grobe Systemfunktionen	5
3. Stakeholder und Kontext	5
4. Funktionale Anforderungen	6
4.1 Kunde anlegen (aus BA-01)	6
4.2 Kundendaten ändern (aus BA-02)	7
4.3 Kunde löschen (aus BA-03, GR-04)	7
4.4 Kunden suchen und auflisten (aus BA-04)	7
4.5 Übergreifende Regeln und Dienste	8
5. Nicht-funktionale Anforderungen	8
6. Daten und Schnittstellen	9
6.1 Datenobjekte und Datentypen	9
6.2 Schnittstellen	10
7. Systemarchitektur (logisch, grob)	12
7.1 Klassendiagramm	12
7.2 Sequenzdiagramm	13
8. Testbare Abnahmekriterien	13
9. Traceability LH ↔ PH	15
10. Modultestplan	16
11. Anhänge	18
11.1 Abkürzungen	18
11.2 Glossar	18
11.3 Referenzen	18

Autor	Prüfer	Freigebender
Ahadyar, Mahsuna Kilic, Kübra Weidmann, Mara Gruppe C (Kunden) 10.06.2026	Prof. Dr. Marmitt, Gerd Modulverantwortlicher 10.06.2026	Prof. Dr. Marmitt, Gerd Modulverantwortlicher 10.06.2026

Freigabevermerk: Dieses Dokument ist nach Prüfung und Freigabe durch den Modulverantwortlichen verbindliche Spezifikationsgrundlage für die Implementierung und den Modultest der Komponente *Verwaltung von Kunden*.

Dokumentenhistorie

Version	Datum	Autor	Grund der Änderung
1.0	10.06.2026	Mahsuna Ahadyar, Kübra Kilic, Mara Weidmann	Initiale Erstellung
1.1	25.06.2026	Mahsuna Ahadyar, Kübra Kilic, Mara Weidmann	Einarbeitung Feedback

1. Einleitung

1.1 Zweck des Dokuments

Dieses Pflichtenheft (System Requirements Specification, SRS) beschreibt aus Sicht des Auftragnehmers, **wie** die Komponente *Verwaltung von Kunden* der Desktop-Fakturierungsanwendung die Anforderungen des Lastenhefts (v1.3) erfüllt. Es konkretisiert die fachlichen Anforderungen in testbare Systemanforderungen und dient als direkte Grundlage für Design, Implementierung sowie den Komponenten- bzw. Modultestplan (Kapitel 10).

1.2 Ziel

Ziel dieses Pflichtenhefts ist die vollständige und testbare Spezifikation der Verwaltung von Kundenstammdaten: Anlegen, Ändern, Löschen (mit referenzieller Integrität gemäß GR-04) sowie Suchen und Auflisten von Kunden, einschließlich der Vergabe eindeutiger Kundennummern, der Validierung der Eingaben und der Bereitstellung der Kundendaten für den Dokumentenzyklus (Gruppe A).

1.3 Geltungsbereich

Dieses Dokument gilt für die Komponente **Gruppe C — Verwaltung von Kunden**. Die Gesamtanwendung wird arbeitsteilig in vier Komponenten entwickelt; jede Untergruppe pflegt ein eigenes Pflichtenheft:

Gruppe	Komponente	Eigenes Pflichtenheft
A	Prozess / Dokumentenzyklus	separat
B	Verwaltung von Produkten	separat
C	Verwaltung von Kunden	dieses Dokument
D	Programmoberfläche	separat

Die Komponente C **stellt** Kundenstammdaten lesend für den Dokumentenzyklus (Gruppe A) über eine definierte interne Schnittstelle bereit (Kapitel 6.2) und wird über die Programmoberfläche (Gruppe D) bedient. Die Erzeugung von Belegen und die Übernahme

der Kundendaten in Dokumente (Gruppe A) sowie die Produktverwaltung (Gruppe B) sind **nicht** Gegenstand dieses Dokuments.

1.4 Definitionen und Abkürzungen

Fachbegriffe (Kunde, Dokumentenzyklus, GoBD, DSGVO, CRUD, ...) sind im Glossar des Lastenhefts (§ 8.1) definiert und gelten unverändert. Dokumentspezifische Abkürzungen siehe Kapitel 11.

1.5 Referenzen

- Lastenheft „Desktop-Fakturierungsanwendung“, Team 1, Version 1.3, 09.06.2026
- Project Charter, Team 1, Version 1.3, 14.05.2026
- Pflichtenheft Gruppe A „Prozess / Dokumentenzyklus“, Version 1.0, 09.06.2026
- DSGVO — EU-Verordnung 2016/679 (personenbezogene Kundendaten)
- GoBD — Grundsätze zur ordnungsmäßigen Führung und Aufbewahrung von Büchern

2. Systemüberblick

2.1 Kurzbeschreibung

Die Anwendung ist eine **Einzelplatz-Stand-Alone-Desktop-Anwendung** mit **lokaler Datenhaltung** (keine Cloud, kein Server). Die Bedienung erfolgt über eine **minimale grafische Benutzeroberfläche** (Gruppe D), über die die Funktionalität der Kundenverwaltung zugänglich gemacht wird.

Die Komponente *Verwaltung von Kunden* stellt die Stammdatenpflege der Kunden bereit: Anlegen, Ändern und Löschen von Kunden, Vergabe eindeutiger Kundennummern, Validierung der Eingaben, referenzielle Integrität gegenüber dem Dokumentenbestand (GR-04), Suche und sortierte Auflistung sowie der lesende Zugriff für die Dokumenterstellung (Gruppe A) und der Export der Kundenstammdaten in einem offenen Format.

2.2 Abgrenzung (Was gehört dazu / was nicht)

Im Umfang dieser Komponente:

- Anlegen von Kunden mit Pflicht- und optionalen Feldern
- Vergabe eindeutiger, vom System generierter Kundennummern
- Ändern von Kundendaten (Name, Anschrift, Kontaktdaten, USt-IdNr.)
- Löschen von Kunden inkl. Löschsperre bei verknüpften Dokumenten (GR-04)
- Suchen und sortiertes Auflisten von Kunden (Volltextsuche Name/Kundennummer)
- Bereitstellung des lesenden Zugriffs für Gruppe A (KundenService)
- Export der Kundenstammdaten in einem offenen, dokumentierten Format (Anteil an Q-08)

Nicht im Umfang dieser Komponente:

- Mahnwesen, Buchhaltung, Kundenportale (LH-Nichtziele)

2.3 Grobe Systemfunktionen

Erfassen eines Kunden → Validieren der Eingaben → Vergeben der Kundennummer → Persistieren → Suchen/Auflisten → Ändern → Löschen (mit Referenzprüfung, GR-04) → Export.

3. Stakeholder und Kontext

Stakeholder und Systemkontext sind im Lastenheft (§ 2, § 3) beschrieben und gelten unverändert. Für diese Komponente ist der maßgebliche Akteur:

- **Anwender:in** — natürliche Person (Selbstständige:r, Freiberufler:in, Kleinstunternehmer:in), die ihre Kundenstammdaten eigenverantwortlich pflegt.

Angrenzende Systeme/Komponenten: lokales Dateisystem (Persistenz, Datenexport) sowie intern die Komponenten Dokumentenzyklus (A, lesender Konsument der Kundendaten) und Programmoberfläche (D). Da Kundendaten **personenbezogene Daten** im Sinne der DSGVO sind, gilt die lokale Datenhaltung (Q-06) für diese Komponente in besonderem Maße.

4. Funktionale Anforderungen

Die Anforderungen sind nach CRUD-Operationen gruppiert und mit den Satzschablonen des Foliensatzes formuliert. Jede Anforderung ist eindeutig, vollständig, widerspruchsfrei und verifizierbar.

Hinweis (Erläuterung, keine eigenständige Anforderung) — Kundennummern: Dieser Kasten erläutert das gemeinsame Verständnis der Kundennummern; die **bindenden Anforderungen** sind **F-02** (Vergabe einer eindeutigen, vom System generierten, fortlaufenden Nummer mit Präfix K - und führenden Nullen, z. B. K-000017) sowie **F-07** (Unveränderlichkeit nach der Vergabe). Die Nummer wird auf Basis der höchsten bisher vergebenen Nummer ermittelt; anders als bei Rechnungsnummern (GR-01, Gruppe A) besteht **keine** Lückenlosigkeits-Pflicht. Zur Begründung der Führung als String (statt int, wegen Präfix und führender Nullen) siehe die Designgrundsätze in Kapitel 6.1.

4.1 Kunde anlegen (aus BA-01)

F-01: Das System MUSS es der Anwender:in ERMÖGLICHEN, einen neuen Kunden mit den Pflichtfeldern Name und Anschrift (Straße, PLZ, Ort) sowie den optionalen Feldern E-Mail, Telefon und USt-IdNr. anzulegen.

F-02: WENN ein Kunde gespeichert wird, DANN MUSS das System eine eindeutige Kundennummer (Präfix K-, fortlaufend, führende Nullen) vergeben und anzeigen.

F-03: WENN ein Pflichtfeld fehlt (kein Name, keine vollständige Anschrift), DANN MUSS das System das Speichern ablehnen und das fehlende Pflichtfeld benennen (Q-09).

F-04: WENN eine E-Mail-Adresse angegeben wird, DANN MUSS das System deren Format prüfen (mindestens ein @ mit Zeichen davor und dahinter) und bei ungültigem Format das Speichern ablehnen.

4.2 Kundendaten ändern (aus BA-02)

F-05: Das System MUSS es der Anwender:in ERMÖGLICHEN, die Felder Name, Anschrift, E-Mail, Telefon und USt-IdNr. eines bestehenden Kunden zu ändern und persistent zu speichern; die Pflichtfeldprüfung (F-03) gilt unverändert.

F-06: WENN ein Kunde geändert wird, DANN MUSS das System sicherstellen, dass bereits versendete Dokumente unverändert bleiben; dies ist gewährleistet, weil Gruppe A die Kundendaten zum Erstellzeitpunkt in den Beleg übernimmt (GR-02). Die Komponente C speichert ausschließlich den jeweils aktuellen Stand.

F-07: Das System MUSS die Kundennummer nach der Vergabe vor jeder Änderung schützen; ein Änderungsversuch an der Kundennummer MUSS abgelehnt werden.

4.3 Kunde löschen (aus BA-03, GR-04)

F-08: Das System MUSS es der Anwender:in ERMÖGLICHEN, einen Kunden ohne verknüpfte Dokumente nach einer Bestätigungsabfrage dauerhaft zu löschen.

F-09 (GR-04): WENN der zu löschende Kunde aktive oder archivierte Dokumente referenziert, DANN MUSS das System den Löschvorgang ablehnen und einen Hinweis mit der **Anzahl der verknüpften Dokumente** anzeigen.

F-10: WENN ein Löschvorgang ausgelöst wird, DANN MUSS das System vor dem Löschen über die Schnittstelle KundenReferenzPruefung (Gruppe A, Kapitel 6.2) die Anzahl der Dokumente ermitteln, die den Kunden referenzieren.

4.4 Kunden suchen und auflisten (aus BA-04)

F-11: Das System MUSS es der Anwender:in ERMÖGLICHEN, alle Kunden in einer nach Name sortierten Liste anzuzeigen.

F-12: Das System MUSS es der Anwender:in ERMÖGLICHEN, Kunden über eine Volltextsuche nach Name oder Kundennummer zu filtern; die Suche MUSS Teilzeichenketten finden und Groß-/Kleinschreibung ignorieren.

F-13: WENN eine Suche ausgeführt wird, DANN MUSS das System das gefilterte Ergeb-

nis innerhalb der Vorgabe aus Q-02 anzeigen (siehe NF-PERF-01).

4.5 Übergreifende Regeln und Dienste

F-14 (Bereitstellung für Gruppe A): Das System MUSS eine lesende Schnittstelle KundennService bereitstellen, die es der Komponente Dokumentenzyklus (Gruppe A) ERMÖGLICHT, einen Kunden anhand seiner Kundennummer abzurufen; existiert kein Kunde zur Nummer, MUSS null zurückgegeben werden.

F-15 (Datenexport, Anteil an Q-08): Das System MUSS es der Anwender:in ERMÖGLICHEN, alle Kundenstammdaten vollständig in ein offenes, dokumentiertes Format (CSV, UTF-8, Semikolon-getrennt, mit Kopfzeile) in das lokale Dateisystem zu exportieren.

5. Nicht-funktionale Anforderungen

NF-PERF-01 (aus Q-01/Q-02): Das System MUSS Such- und Auflistungsergebnisse der Kundenverwaltung INNERHALB VON 1 SEKUNDE anzeigen, bei einem Datenbestand von bis zu 5.000 Kunden (Q-01) auf einem typischen Endanwender-PC.

NF-EXP-01 (aus Q-08, anteilig): Das System MUSS den vollständigen Export der Kundenstammdaten (F-15) INNERHALB VON 30 SEKUNDEN abschließen, bei einem Datenbestand gemäß Q-01.

NF-USE-01 (aus Q-09): Das System MUSS fehlende Pflichtangaben im Formular „Kunde anlegen/ändern“ so markieren und benennen, dass mindestens 80 % der Testpersonen die fehlende Eingabe ohne externe Hilfe im ersten Korrekturversuch ergänzen können (Nachweis durch Usability-Test mit mind. 5 Testpersonen).

NF-SEC-01 (aus Q-06, anteilig / DSGVO): Das System MUSS 100 % der personenbezogenen Kundendaten ausschließlich lokal auf dem Anwender-PC ablegen; eine Übertragung an externe Dienste findet NICHT statt (Nachweis durch Netzwerk-Monitoring während eines repräsentativen Nutzungslaufs).

6. Daten und Schnittstellen

Dieses Kapitel ist direkter Input für den Modultestplan (Kapitel 10). Datentypen werden bereits als Java-Typen angegeben.

6.1 Datenobjekte und Datentypen

Designgrundsätze (konsistent zu Gruppe A):

- **Kundennummern** werden als `String` geführt (festes Format mit Präfix und führenden Nullen, z. B. "K-000017") — **nicht** als `int`.
- **Postleitzahlen** werden als `String` geführt — **nicht** als `int`, weil führende Nullen erhalten bleiben müssen (z. B. "01067" Dresden).
- Optionale Felder sind als `null` zulässig; Pflichtfelder dürfen weder `null` noch leer sein.

Klasse Kunde

Attribut	Java-Typ	Beschreibung
kundennummer	<code>String</code>	eindeutig, vom System generiert, unveränderlich (F-02, F-07)
name	<code>String</code>	Pflichtfeld, nicht leer (Firmen- oder Personennamen)
strasse	<code>String</code>	Pflichtfeld (Anschrift)
plz	<code>String</code>	Pflichtfeld (Anschrift, führende Nullen)
ort	<code>String</code>	Pflichtfeld (Anschrift)
eMail	<code>String (optional, null)</code>	Format gemäß F-04
telefon	<code>String (optional, null)</code>	Freitext
ustIdNr	<code>String (optional, null)</code>	Umsatzsteuer-Identifikationsnummer

6.2 Schnittstellen

Externe Schnittstellen:

ID	Schnittstelle	Zweck
IF-01	Lokales Dateisystem	Persistenz der Kundenstammdaten
IF-04	Datenexport (CSV)	Export der Kundenstammdaten als CSV (F-15, Q-08)

Anforderungen an die externen Schnittstellen

IF-01: Das System MUSS eine Schnittstelle zum lokalen Dateisystem bereitstellen, die es ERMÖGLICHT, die Kundenstammdaten dauerhaft zu speichern und wieder zu laden.

IF-04: Das System MUSS eine Export-Schnittstelle bereitstellen, die es der Anwender:in ERMÖGLICHT, alle Kundenstammdaten in einem offenen Format (CSV, UTF-8, Semikolon-getrennt, mit Kopfzeile) in das lokale Dateisystem zu exportieren (F-15, Q-08).

Interne Schnittstellen: Die Schnittstellen werden hier **fachlich** beschrieben (Zweck, ausgetauschte Daten, Richtung); konkrete Methodensignaturen und Datentypen sind dem Komponentenentwurf bzw. dem Modultestplan (Kapitel 10) vorbehalten.

Genutzte Schnittstellen (Komponente C ruft auf):

Schnittstelle	Partner	Richtung	Fachlicher Zweck
Referenzprüfung Kunden	Gruppe A	$A \rightarrow C$	Anzahl aktiver und archivierter Belege, die einen Kunden referenzieren (Löschsperre GR-04, F-09/F-10)

Bereitgestellte Schnittstellen (Komponente C wird aufgerufen):

Schnittstelle	Partner	Richtung	Fachlicher Zweck
Kundenzugriff (lesend)	Gruppe A, D	$C \rightarrow A, C \rightarrow D$	Kunde per Kundennummer abrufen (F-14); Kundensuche über Name oder Kundennummer (F-12)

Liefert die Kundenabfrage keinen Treffer, wird dies dem Aufrufer eindeutig signalisiert (die Abfrage per Kundennummer liefert „kein Treffer“).

Komponenteninterne Dienste (rein C-intern, fachlich beschrieben):

- **Kundennummernvergabe (F-02):** liefert die nächste fortlaufende Kundennummer im festen Format mit Präfix und führenden Nullen (z. B. K-000017) auf Basis der höchsten bisher vergebenen Nummer; ohne Lückenlosigkeits-Pflicht. (Realisierung: EinfacherKundennummernGenerator)
- **Kundenpersistenz (IF-01):** speichert Kunden im lokalen Dateisystem und liefert einen Kunden zur Kundennummer, alle Kunden sortiert nach Name sowie Suchergebnisse (Name/Nr.). (Realisierung: JsonKundenRepository, JSON-Ablage)
- **Datenexport (IF-04):** exportiert alle Kundenstammdaten als CSV in das lokale Dateisystem (F-15). (Realisierung: KundenCsvExport)

- **Ereignisbenachrichtigung (Observer, Paket gemeinsam):** meldet Datenänderungen am Kundenbestand (`melde(DatenBereich.KUNDEN)`) an die abonnierten Modulansichten (Gruppe D).

7. Systemarchitektur (logisch, grob)

Die Komponente folgt einer einfachen Schichtung: die GUI (Gruppe D) ruft den KundenverwaltungService auf, der die Fachlogik (Validierung, Nummernvergabe, Löschsperre GR-04) kapselt und die Dienste KundennummernGenerator, KundenRepository und KundenReferenzPruefung (Gruppe A) nutzt. Gegenüber Gruppe A implementiert die Klasse KundenVerwaltungsService das Interface KundenService. Kunden werden über das KundenRepository (konkrete Implementierung JsonKundenRepository) im lokalen Dateisystem persistiert (JSON-Ablage); die fortlaufenden Kundennummern erzeugt der EinfacherKundennummernGenerator (Implementierung von KundennummernGenerator). Nach jeder schreibenden Operation (Anlegen, Ändern, Löschen) meldet der KundenVerwaltungsService die Änderung über einen **EreignisBus** (Observer-Muster, Paket gemeinsam; `melde(DatenBereich.KUNDEN)`), den die Kundenansicht der Gruppe D abonniert und sich daraufhin automatisch aktualisiert.

7.1 Klassendiagramm

Beschreibung zu Abbildung 1: Das Klassendiagramm zeigt die Entitätsklasse Kunde mit ihren Attributen (Kapitel 6.1). Die Klasse KundenVerwaltungsService orchestriert Anlegen, Ändern, Löschen und Suche und **realisiert** das Interface KundenService (lesender Zugriff für Gruppe A und D, F-14). Die komponenteninternen Interfaces besitzen jeweils eine **konkrete Implementierung**: JsonKundenRepository realisiert KundenRepository (Persistenz, IF-01) und EinfacherKundennummernGenerator realisiert KundennummernGenerator (Vergabe eindeutiger Kundennummern, F-02). Die Schnittstelle KundenReferenzPruefung (Löschsperre GR-04, F-09/F-10) wird **von Gruppe A bereitgestellt** und ist daher als extern dargestellt (ihre Implementierung liegt in Gruppe A). Der KundenVerwaltungsService **nutzt** diese Dienste sowie den EreignisBus (Observer-Muster) und meldet Datenänderungen an die abonnierte Kundenansicht der Gruppe D. Der Zugriff auf Ku

nde ist eine **Nutzungs-/Abhängigkeitsbeziehung** — das KundenRepository liefert und speichert Kunde-Objekte —, **keine** 1:n-Aggregation oder -Komposition; ein Interface hält somit keinen „Container“ von Entitäten. Dokumente (Gruppe A) referenzieren einen Kunde ausschließlich über die Kundennummer (lose Kopplung).

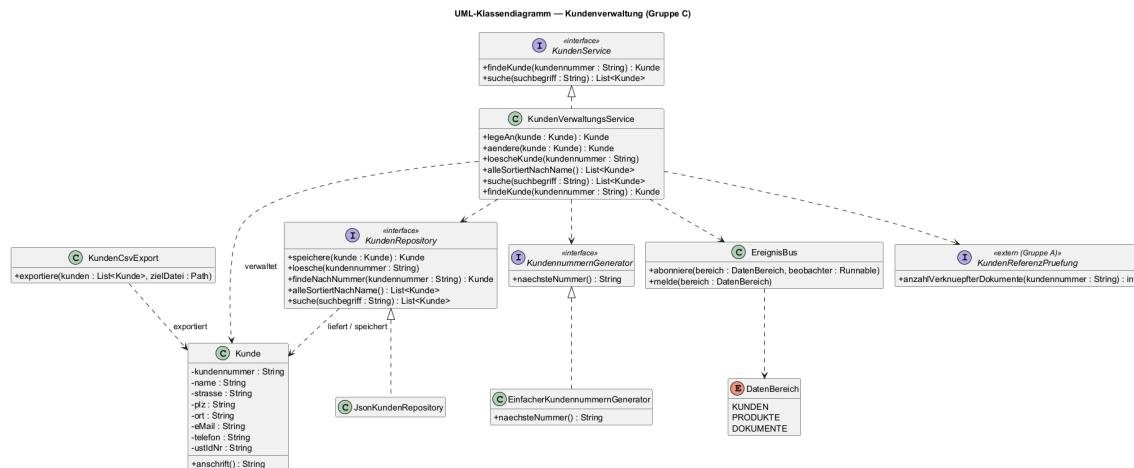


Abbildung 1: UML-Klassendiagramm Kundenverwaltung

7.2 Sequenzdiagramm

Beschreibung zu Abbildung 2: Das Sequenzdiagramm stellt den Ablauf *Kunde löschen* dar. Die Anwender:in löst über die GUI (Gruppe D) `loescheKunde(kundennummer)` am `KundenVerwaltungsService` aus. Dieser ermittelt zuerst über `KundenReferenzPruefung.anzahlVerknuepfterDokumente(kundennummer)` (Gruppe A) die Anzahl der Dokumente, die den Kunden referenzieren. Ist die Anzahl größer als 0, wird der Löschvorgang abgelehnt und ein Hinweis mit der Anzahl der verknüpften Dokumente an die GUI zurückgegeben (F-09, GR-04). Ist die Anzahl 0, löscht das System den Kunden nach Bestätigung der Anwender:in (F-08) über `KundenRepository.loesche(kundennummer)` dauerhaft aus dem lokalen Datenbestand und meldet die Änderung über den EreignisBus (`melde(DatenBereich.KUNDEN)`) an die abonnierte Kundenansicht der Gruppe D.

8. Testbare Abnahmekriterien

AC-C-01 (zu F-01–F-03, NF-PERF-01) — *Kunde anlegen und auffinden* Vorbedingung:
Anwendung gestartet, Modul Kundenverwaltung geöffnet; höchste vergebene Kunden-

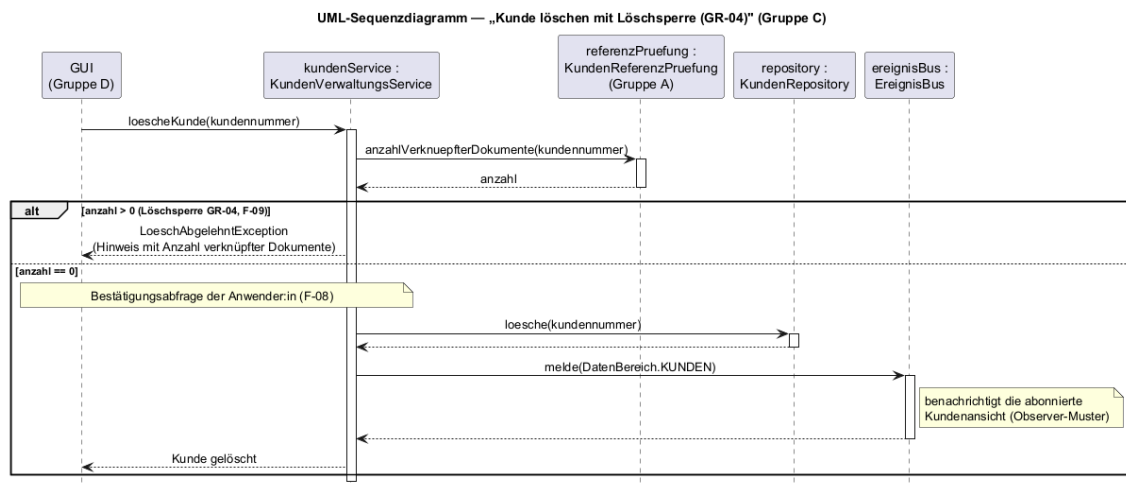


Abbildung 2: UML-Sequenzdiagramm „Kunde löschen mit Löschsperre (GR-04)“

nummer = K-000016. Aktion: Anwender:in erfasst einen neuen Kunden mit Pflichtfeldern (Name, Straße, PLZ, Ort) und speichert. Erwartet: Das System vergibt die Kundennummer K-000017 und zeigt sie an; der Kunde erscheint in der Suchergebnisliste innerhalb von ≤ 1 Sekunde (Q-02).

AC-C-02 (zu F-03, F-04, NF-USE-01) — *Pflichtfeld- und Formatprüfung* Vorbedingung: Formular „Kunde anlegen“ geöffnet. Aktion: Anwender:in lässt den Ort leer und versucht zu speichern; anschließend trägt sie eine ungültige E-Mail-Adresse („max.mustermann“) ein und versucht erneut zu speichern. Erwartet: Beide Speicherversuche werden abgelehnt; das fehlende Pflichtfeld „Ort“ bzw. das ungültige E-Mail-Format wird benannt.

AC-C-03 (zu F-05–F-07) — *Kundendaten ändern* Vorbedingung: Ein Kunde mit mindestens einer verknüpften, versendeten Rechnung existiert. Aktion: Anwender:in ändert einen Adressbestandteil und speichert. Erwartet: Die Änderung ist persistent gespeichert; die bereits versendete Rechnung (Gruppe A) zeigt weiterhin die ursprüngliche Anschrift; die Kundennummer ist unverändert.

AC-C-04 (zu F-08–F-10, GR-04) — *Löschsperre für verknüpfte Kunden* Vorbedingung: Kunde K-000010 referenziert 3 Dokumente; Kunde K-000011 ist unverknüpft. Aktion: Anwender:in versucht, K-000010 zu löschen; anschließend löscht sie K-000011 nach Bestätigung. Erwartet: Das Löschen von K-000010 wird abgelehnt, der Hinweis nennt die Anzahl „3“ verknüpfter Dokumente; K-000011 ist dauerhaft entfernt und erscheint nicht

mehr in der Liste.

AC-C-05 (zu F-11–F-13, NF-PERF-01) — *Kunden suchen und auflisten* Vorbedingung: Mindestens 100 Kunden sind im System. Aktion: Anwender:in sucht einen Kunden anhand eines Teils des Namens und anschließend anhand der Kundennummer. Erwartet: Beide Trefferlisten erscheinen in ≤ 1 Sekunde (Q-02), sind nach Name sortiert und enthalten den gesuchten Kunden (auch bei abweichender Groß-/Kleinschreibung).

AC-C-06 (zu F-15, NF-EXP-01, NF-SEC-01) — *Kundenstammdaten exportieren* Vorbedingung: Mindestens 100 Kunden sind im System. Aktion: Anwender:in exportiert die Kundenstammdaten; während des Nutzungslaufs läuft ein Netzwerk-Monitoring. Erwartet: Eine CSV-Datei (UTF-8, Semikolon-getrennt, mit Kopfzeile) mit allen Kunden und allen Attributen liegt im gewählten Zielordner; der Export dauert ≤ 30 Sekunden; das Monitoring zeigt keine Datenübertragung an externe Dienste.

9. Traceability LH ↔ PH

Jede für Gruppe C relevante Lastenheft-Anforderung ist mindestens einer Pflichtenheft-Anforderung zugeordnet.

LH-Anforderung	Beschreibung (LH)	PH-Anforderung(en)
BA-01	Kunden anlegen	F-01, F-02, F-03, F-04
BA-02	Kundendaten ändern	F-05, F-06, F-07
BA-03	Kunden löschen	F-08, F-09, F-10
BA-04	Kunden suchen und auflisten	F-11, F-12, F-13
GR-02	Unveränderlichkeit versendeter Dokumente	F-06 (Abgrenzung)
GR-04	Referenzielle Integrität Kunden	F-09, F-10
Q-01	Datenbestand 5.000 Kunden	NF-PERF-01
Q-02	Suche/Auflistung ≤ 1 s	NF-PERF-01, F-13
Q-06	Lokale Speicherung (DSGVO)	NF-SEC-01
Q-08	Datenexport ≤ 30 s	F-15, NF-EXP-01
Q-09	Pflichtfeldhinweise ≥ 80 %	NF-USE-01, F-03

Hinweis: Die Übernahme der Kundendaten in Belege (Snapshot zum Erstell-

zeitpunkt) liegt in der Verantwortung von Gruppe A; Komponente C liefert lediglich den jeweils aktuellen Datenstand über KundenService. PZ-01 (CRUD-Verwaltung der Kundenstammdaten) wird durch BA-01–BA-04 vollständig abgedeckt.

10. Modultestplan

Die folgenden Testfälle sind deterministisch (feste Ein-/Ausgaben) und mit JUnit 5 umsetzbar. Die Schnittstelle KundenReferenzPruefung (Gruppe A) wird im Modultest durch einen Stub/Mock ersetzt.

Abgedeckte				
TC	PH-Anf.	Vorbedingung	Eingabe	Erwartetes Ergebnis
TC-01	F-01, F-02	Höchste Kundennummer K-00 0016	Kunde („Muster GmbH“, „Hauptstr. 1“, „68163“, „Mannheim“) speichern	Kunde persistiert; Kundennummer = K-000 017
TC-02	F-02 (Format)	Zähler = 7	naechsteNummer()	liefert K-000007 (führende Nullen, String)
TC-03	F-03, NF- USE-01	Kunde ohne Ort	speichere()	Speichern abgelehnt; Validierungsfehler benennt „Ort“
TC-04	F-03	Kunde mit leerem Namen (" ")	speichere()	Speichern abgelehnt; Validierungsfehler benennt „Name“
TC-05	F-04	Kunde mit E-Mail "m ax.mustermann"	speichere()	Speichern abgelehnt (ungültiges E-Mail-Format)
TC-06	F-04	Kunde mit E-Mail "m ax@beispiel.de"	speichere()	Kunde gespeichert (gültiges Format)

Abgedeckte				
TC	PH-Anf.	Vorbedingung	Eingabe	Erwartetes Ergebnis
TC-07	F-05	Kunde K-000017 mit Ort „Mannheim“	Ort auf „Heidelberg“ ändern, speichern	gespeicherter Kunde hat ort = „Heidelberg“
TC-08	F-07	Kunde K-000017	Änderungsversuch der Kundennummer auf K-999999	wirft IllegalArgumentException / Änderung abgelehnt
TC-09	F-08	Stub: anzahlVerknu epfterDokumente → 0	loescheKunde("K-000011") mit Bestätigung	Kunde entfernt; nicht mehr in alleSortiertNachName()
TC-10	F-09, F-10, GR-04	Stub: anzahlVerknu epfterDokumente("K-000010") → 3	loescheKunde("K-000010")	Löschen abgelehnt; Kunde weiterhin vorhanden; Hinweis enthält Anzahl 3
TC-11	F-11	Kunden „Zimmer“, „Albrecht“, „Maier“	alleSortiertNachName()	Reihenfolge: „Albrecht“, „Maier“, „Zimmer“
TC-12	F-12	Kunde „Muster GmbH“	suche("MUSTER")	Trefferliste enthält „Muster GmbH“ (case-insensitive, Teilstring)
TC-13	F-12, F-14	Kunde K-000017 vorhanden; K-999999 nicht	suche("K-000017"); findeKunde("K-999999")	Treffer enthält K-000017; findeKunde liefert null
TC-14	F-15	3 Kunden im Bestand	exportiereCsv(ziel)	CSV-Datei mit Kopfzeile + 3 Datenzeilen, Semikolon-getrennt, UTF-8

Damit sind 14 Testfälle (> 10) spezifiziert, die alle funktionalen Kernregeln (F-02, F-03, F-04, F-07, F-09, F-12, F-14, F-15) sowie die zentrale Geschäftsregel GR-04 und die Qualitätsvorgaben (Q-02, Q-08, Q-09) abdecken.

11. Anhänge

11.1 Abkürzungen

Abkürzung	Bedeutung
F	Funktionale Anforderung (Pflichtenheft)
NF	Nicht-funktionale Anforderung (Pflichtenheft)
IF	Schnittstelle (Interface)
AC	Abnahmekriterium
TC	Testfall (Test Case)
BA	Benutzeranforderung (Lastenheft)
GR	Geschäftsregel (Lastenheft)
Q	Qualitätsanforderung (Lastenheft)
CSV	Comma-Separated Values (offenes Exportformat)
USt-IdNr.	Umsatzsteuer-Identifikationsnummer
SRS	System Requirements Specification (Pflichtenheft)

11.2 Glossar

Es gilt das Glossar des Lastenhefts (§ 8.1) unverändert.

11.3 Referenzen

Siehe Kapitel 1.5.