

Software Engineering 1

Project Charter

Team 1

Inhaltsverzeichnis

1	Dokumentenhistorie	2
2	Projektübersicht	2
2.1	Projektzweck	2
2.2	Projekthintergrund	2
3	Projektziele	2
3.1	Ziele	2
3.2	Nicht-Ziele	3
4	Business Case	3
5	Stakeholder	3
5.1	Auftraggeber (extern/intern)	3
5.2	Regulatorisch	4
5.3	Qualitätsmanagement	4
6	Projekt-Team und Rollen	5
7	Zeitplan / Meilensteine	5
8	Risikomanagement	6
9	Budget und Ressourcen	6
10	Kommunikations- und Entscheidungswege	7
11	Genehmigung / Unterschriften	7

Autor	Prüfer	Freigebenden
Name, Vorname	Name, Vorname	Name, Vorname
Abteilung/Funktion	Abteilung/Funktion	Abteilung/Funktion
Datum, Unterschr.	Datum, Unterschr.	Datum, Unterschr.

1 Dokumentenhistorie

Version	Datum	Autor	Grund der Änderung
1.0	10.04.2026	Lucas Strubel	Initiale Erstellung

2 Projektübersicht

2.1 Projektzweck

Im Rahmen des Moduls Software Engineering wird eine Desktop-Fakturierungsanwendung für Kleinunternehmen, Freiberufler und Selbstständige entwickelt. Die Anwendung bildet den vollständigen kaufmännischen Dokumentenzyklus ab – von der Angebotserstellung über Auftragsbestätigung und Lieferschein bis zur finalen Rechnung. Ziel ist eine schlanke, lokal betriebene Alternative zu kostenpflichtigen SaaS-Lösungen, die dem Nutzer vollständige Datensouveränität bietet. Als Referenzsystem dient die Open-Source-Software Fakturama.

2.2 Projekthintergrund

Die fortschreitende Digitalisierung des Rechnungswesens sowie die gesetzliche E-Rechnungspflicht im B2B-Bereich (ab 01.01.2025) stellen insbesondere Kleinunternehmen vor erhebliche Herausforderungen. Dieses Projekt entwickelt eine vereinfachte lehrveranstaltungsbegleitende Fakturierungslösung, die die Kernprozesse eines Fakturierungsprogramms abdeckt und gleichzeitig als praxisnahes Lehrprojekt im Bereich Software Engineering dient.

3 Projektziele

3.1 Ziele

Nr.	Ziel	Erfolgskriterien
Z-01	<i>Digitale Verwaltung von Produkten und Kunden</i>	<i>CRUD-Operationen für beide Module vollständig implementiert und funktionsfähig</i>
Z-02	<i>Abbildung des vollständigen Dokumentenzyklus</i>	<i>Alle 4 Dokumenttypen (Angebot, Auftragsbestätigung, Lieferschein, Rechnung) erstellbar und untereinander verknüpfbar</i>
Z-03	<i>Funktionsfähige und bedienbare Programmoberfläche</i>	<i>Anwendung kann ohne technische Vorkenntnisse zur Erstellung eines Dokuments genutzt werden</i>

3.2 Nicht-Ziele

Die folgenden Punkte sind **explizit nicht** Teil dieses Projekts:

- Mehrbenutzer- oder Netzwerkfähigkeit (gleichzeitiger Zugriff mehrerer Nutzer)
- Vollständiges Buchhaltungsmodul (keine Bilanzierung)
- Webshop-Anbindung (z. B. WooCommerce, Gambio Connectoren)
- Mobile Clients oder Web-Applikation
- Unterstützung von E-Rechnungsformaten (ZUGFeRD / XRechnung)
- Mahnwesen und automatisiertes Forderungsmanagement
- Garantierter kommerzieller Support oder Service Level Agreements (SLAs)

4 Business Case

Kommerzielle Fakturierungssoftware ist für Kleinunternehmen und Freiberufler häufig mit monatlichen Lizenzkosten verbunden. Bestehende Open-Source-Alternativen (z. B. Fakturama) sind funktional umfangreich, jedoch technisch anspruchsvoll in Installation und Wartung. Das vorliegende Projekt schafft eine schlanke, wartungsarme Lösung.

Nutzen: Kosteneinsparung gegenüber SaaS-Abonnements, vollständige lokale Datenhaltung ohne Cloud-Zwang, geringer Einrichtungsaufwand.

5 Stakeholder

5.1 Auftraggeber (extern/intern)

Rolle	Name	Verantwortlichkeit
Auftraggeber	<i>Prof. Dr. Gerd Marmitt</i>	Anforderungen, Abnahme, Bewertung

5.2 Regulatorisch

Folgende regulatorische Rahmenbedingungen sind für Fakturierungssoftware in Deutschland relevant und werden im Projekt als bekannte Anforderungen geführt, auch wenn ihre vollständige Umsetzung explizit kein Projektziel ist (siehe Nicht-Ziele):

- **GoBD** (Grundsätze zur ordnungsmäßigen Führung und Aufbewahrung von Büchern): Erstellte Rechnungen dürfen nach Versand nicht mehr verändert werden; alle Geschäftsvorfälle müssen lückenlos erfasst werden.
- **E-Rechnungspflicht ab 01.01.2025**: Im B2B-Bereich sind strukturierte elektronische Rechnungsformate (ZUGFeRD, XRechnung) gesetzlich vorgeschrieben. Die Unterstützung dieser Formate ist für dieses Projekt **kein Ziel**, wird aber als bekannte Anforderung dokumentiert.
- **DSGVO**: Kundendaten werden ausschließlich lokal gespeichert; es erfolgt keine Übertragung an Dritte.

5.3 Qualitätsmanagement

Ein Feature gilt als **fertiggestellt**, wenn:

- Der Code ist implementiert und funktioniert lokal
- Unit Tests sind geschrieben und bestehen
- Der Code wurde von mindestens einem anderen Teammitglied reviewed (Pull Request)
- Die Änderungen sind in den main-Branch gemergt
- Die relevante Dokumentation wurde aktualisiert
- Das Feature wurde manuell getestet

Abnahmekriterien – Das Projekt gilt als erfolgreich abgeschlossen, wenn:

1. Alle **Muss**-Anforderungen (Priorität: Hoch) vollständig implementiert sind
2. Alle Unit- und Integrationstests bestehen
3. Die Anwendung lauffähig demonstriert werden kann
4. Die technische Dokumentation vollständig vorliegt
5. Das Projekt erfolgreich präsentiert wurde

6 Projekt-Team und Rollen

Bezeichnung	Details
Entwicklung 1	<i>Lucas Strubel</i> (Matrikelnummer: 3023626) – Schwerpunkt:
Entwicklung 2	<i>Luca Kaiser</i> (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 3	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 4	[] (Matrikel: []) – Schwerpunkt:
Entwicklung 5	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 6	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 7	[] (Matrikel: []) – Schwerpunkt:
Entwicklung 8	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 9	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 10	[] (Matrikel: []) – Schwerpunkt:
Entwicklung 11	[] (Matrikel: [Nr.]) – Schwerpunkt:
Entwicklung 12	[] (Matrikel: [Nr.]) – Schwerpunkt:

7 Zeitplan / Meilensteine

Projektphasen (V-Modell):

Phase	Bezeichnung	Kernaufgaben
1	Anforderungsanalyse	Project Charter, Stakeholder-Analyse, Lastenheft
2	Systementwurf	Systemarchitektur, Technologiewahl
3	Komponentenentwurf	UI/UX Mockups, Datenbankdesign, Pflichtenheft
4	Implementierung	Produkt-, Kundenverwaltung, Dokumentenzzyklus, UI
5	Integrationstest	Schnittstellentests, Modulübergreifende Tests
6	Systemtest	Integrationstests, Systemvalidierung

Phase	Bezeichnung	Kernaufgaben
7	Abnahmetest	Abnahme durch Betreuer, Abschlusspräsentation

Jede Entwicklungsphase korrespondiert mit ihrer jeweiligen Testphase im Rahmen des V-Modells.

Meilensteinplan:

Meilenstein	Beschreibung	Datum	Status
M-01	Project Charter abgeschlossen	15.04.2026	Abgeschlossen
M-02	Lastenheft & Anforderungsanalyse	[Datum]	In Bearbeitung
M-03	Systementwurf & Architektur abgeschlossen	[Datum]	Offen
M-04	Pflichtenheft & Komponentenentwurf	[Datum]	Offen
M-05	Implementierung abgeschlossen (Feature-Complete)	[Datum]	Offen
M-06	Integrations- & Systemtests abgeschlossen	[Datum]	Offen
M-07	Abnahmetest & Präsentation	[Datum]	Offen

8 Risikomanagement

ID	Risiko	W/A	Gegenmaßnahme
R-01	Ausfall eines Teammitglieds	M/H	Wissensteilung, Pair Programming
R-02	Technische Komplexität unterschätzt	M/H	Frühzeitige Spikes, Scope-Reduktion
R-03	Anforderungsänderungen	N/M	Klare Change-Request-Prozesse
R-04	Integrationsprobleme	M/M	Frühzeitige Integrationstests

9 Budget und Ressourcen

- **Teamgröße:** 12 Personen
- **Verfügbare Zeit pro Person:** ca. 2 Stunden/Woche
- **Gesamtprojektlaufzeit:** [Startdatum] – [Enddatum]
- **Budget:** kein monetäres Budget (studentisches Projekt)

- **Infrastruktur:** *Gitea, Lokale Entwicklung*

Rahmenbedingungen (Constraints):

- Die Abgabe erfolgt bis :
- Der Technologie-Stack muss mit der Lehrveranstaltung kompatibel sein
- Alle Teammitglieder müssen gleichmäßig zum Projekt beitragen (erkennbar in Git-History)

Technologie-Stack:

Bereich	Technologie / Tool
Frontend	<i>[z. B. React, Vue, HTML/CSS]</i> – Begründung: <i>[...]</i>
Backend	<i>[z. B. Spring Boot, Node.js, Django]</i> – Begründung: <i>[...]</i>
Datenbank	<i>[z. B. PostgreSQL, MySQL, MongoDB]</i> – Begründung: <i>[...]</i>
Versionskontrolle	<i>Gitea (Standard, Kollaboration)</i>
Projektmanagement	<i>[z. B. GitHub Projects, Trello]</i> – Begründung: <i>[...]</i>
CI/CD	<i>[z. B. GitHub Actions]</i> – Begründung: <i>[...]</i>

10 Kommunikations- und Entscheidungswege

Kanal	Zweck	Frequenz
<i>Discord</i>	Team-Kommunikation	täglich
<i>Wöchentliches Meeting</i>	Fortschrittsbesprechung	wöchentlich
<i>E-Mail</i>	Kommunikation mit Professor	bei Bedarf

11 Genehmigung / Unterschriften

Mit ihrer Unterschrift bestätigen alle Beteiligten, dass sie den Inhalt dieser Project Charta gelesen haben und damit einverstanden sind.

Betreuer/in: _____ Datum: _____

Teammitglied: _____ Datum: _____

Teammitglied: _____ Datum: _____

Teammitglied: _____ Datum: _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____

Teammitglied: _____ **Datum:** _____