

# Team 2

## Lastenheft

### Inhaltsverzeichnis

Project Charter – Fakturierungssystem .....	2
1. Freigabeübersicht .....	2
2. Dokumentenhistorie.....	2
3. Projektübersicht.....	2
3.1 Projektzweck .....	2
3.2 Projekthintergrund .....	3
4. Projektziele.....	3
4.1 Nicht-Ziele.....	3
5. Business Case .....	3
6. Stakeholder und Teamstruktur .....	4
6.1 Stakeholder .....	4
6.2 Teamstruktur, Repositories und Gruppenleiter .....	4
7. Vorgehensmodell.....	4
8. Zeitplan / Meilensteine (V-Modell-orientiert) .....	5
9. Risikomanagement .....	5
10. Ressourcen und Rahmenbedingungen .....	6
11. Kommunikationswege .....	6
12. Definition of Done.....	6
13. Abnahmekriterien.....	7
14. Genehmigung .....	7

# Project Charter – Fakturierungssystem

**Modul:** Software Engineering 1

**Team:** SE1 Team 2 – Hochschule Mannheim

**Version:** 1.1

**Stand:** 05.05.2026

## 1. Freigabeübersicht

Ersteller	Prüfer	Freigebender
Oleg Akimenko	Prof. Dr. Gerd Marmitt	SE1 Team 2 (Gruppenleiter)
SE1 Team 2	Hochschule Mannheim	SE1 Team 2
15.04.2026	15.04.2026	30.06.2026

## 2. Dokumentenhistorie

Version	Datum	Autor	Änderung
1.0	15.04.2026	Oleg Akimenko	Initiale Erstellung
1.1	05.05.2026	Christopher Lampert	Überarbeitung nach Review (Kapitelnummerierung, Anforderungen ins Lastenheft verschoben, Zeitplan konkretisiert, Technologie-Stack in separates Dokument ausgelagert, Gruppenleiter dokumentiert)

## 3. Projektübersicht

### 3.1 Projektzweck

Das Ziel des Projekts ist die konzeptionelle und praktische Entwicklung eines modularen Fakturierungssystems im Rahmen des Moduls Software Engineering 1.

Das System bildet einen vollständigen Geschäftsprozess von der Angebotserstellung über die Auftragsbestätigung und den Lieferschein bis hin zur Rechnungserstellung ab. Dabei steht nicht nur die Implementierung im Vordergrund, sondern insbesondere die Anwendung strukturierter Softwareentwicklungsprozesse und die Umsetzung eines klassischen Vorgehensmodells.

## 3.2 Projekthintergrund

Die Entwicklung moderner Softwaresysteme erfordert strukturierte Vorgehensmodelle, klare Anforderungen und eine saubere Trennung von Entwicklungs- und Testphasen.

Im Rahmen des Moduls Software Engineering 1 wird ein praxisnahes Projekt durchgeführt, das die Anwendung klassischer Entwicklungsprozesse im Team ermöglicht. Das Fakturierungssystem dient als realistisches Szenario, um zentrale Konzepte der Softwareentwicklung wie Anforderungsanalyse, Architekturdesign, Implementierung, Integration und Test praktisch umzusetzen.

Das Projekt orientiert sich am V-Modell als strukturiertem Vorgehensmodell.

## 4. Projektziele

Nr.	Ziel	Erfolgskriterium
Z1	Produktverwaltung	Produkte können erstellt, bearbeitet und gelöscht werden
Z2	Kundenverwaltung	Kundendaten sind vollständig verwaltbar
Z3	Dokumentenworkflow	Angebot → Auftragsbestätigung → Lieferschein → Rechnung
Z4	GUI	Benutzerfreundliche und funktionale Oberfläche

### 4.1 Nicht-Ziele

- Mobile Anwendung
- Cloud-System
- Mehrbenutzer-Online-System
- Buchhaltungssystem
- E-Rechnung

## 5. Business Case

- **Zielgruppe:** kleine Unternehmen und Lernprojekt
- **Nutzen:** Automatisierung von Fakturierungsprozessen

- **Problem:** manuelle Rechnungsprozesse sind fehleranfällig und ineffizient

## 6. Stakeholder und Teamstruktur

### 6.1 Stakeholder

Rolle	Beschreibung
Auftraggeber	Prof. Dr. Gerd Marmitt
Entwicklungsteam	SE1 Team 2
Endnutzer	spätere Anwender des Systems

### 6.2 Teamstruktur, Repositories und Gruppenleiter

Pro Untergruppe ist ein **Gruppenleiter** benannt, der die Untergruppe gegenüber dem Gesamtteam und dem Auftraggeber vertritt und das Projekt-Charter unterzeichnet.

Gruppe	Repository	Mitglieder	Gruppenleiter	Verantwortungsbereich
Gruppe E	SE1_Gruppe_E	Hadil Jondi [3030438], Nicolas Seelinger [3027710]	[zu bestimmen]	Programmoberfläche
Gruppe F	SE1_Gruppe_F	Andreas Ivanovic [3028874], Armin Omanovic [3028711], Alexander Teller [3028801]	[zu bestimmen]	Dokumentenprozess
Gruppe G	SE1_Gruppe_G	Rahaf Alhosny [3026969], Fatemeh Mohammadi [3029148], Lulia Silk [3030489]	[zu bestimmen]	Produktverwaltung
Gruppe H	SE1_Gruppe_H	Oleg Akimenko [3028868], Christopher Lampert [3027248], Kenan Pekarovic [3027541]	[zu bestimmen]	Kundenverwaltung

## 7. Vorgehensmodell

Das Projekt orientiert sich am **V-Modell**.

- Anforderungen
- System- und Softwaredesign
- Implementierung
- Integration und Test
- Abnahme

## 8. Zeitplan / Meilensteine (V-Modell-orientiert)

Das Projekt orientiert sich am V-Modell mit Fokus auf Verifikation und Validierung. Jeder Entwicklungsphase ist eine entsprechende Testphase zugeordnet. Das Project Charter ist ein lebendes Dokument – bei Anpassungen wird eine neue Version erstellt.

Nr.	Phase	Inhalt	Datum
M1	Anforderungen	Erhebung und Dokumentation der System- und Softwareanforderungen (Lastenheft)	15.04.2026 – 15.05.2026
M2	Architektur	Systemarchitektur und Schnittstellendesign	22.05.2026
M3	Detailentwurf	Moduldesign (Produkt-, Kundenverwaltung, UI, Prozess)	29.05.2026
M4	Implementierung	Umsetzung aller Module im Code	12.06.2026
M5	Integrationstest	Zusammenführung und Schnittstellentests	19.06.2026
M6	Systemtest	Prüfung gegen Anforderungen	26.06.2026
M7	Abnahme	Präsentation und finale Abgabe	30.06.2026

**Hinweis:** Der Technologie-Stack ist in einem separaten Dokument (`Technologiestack.md`) dokumentiert und wird perspektivisch im Architekturdokument (vgl. SE2) fortgeschrieben.

## 9. Risikomanagement

Risiko	Wahrscheinlichkeit / Impact	Gegenmaßnahme
Ausfall von Teammitgliedern	Mittel / Hoch	Wissensaustausch
Merge-Konflikte	Mittel / Mittel	Code Reviews

Risiko	Wahrscheinlichkeit / Impact	Gegenmaßnahme
Integrationsprobleme	Mittel / Mittel	frühe Tests
Zeitverzug	Hoch / Mittel	MVP-Fokus

## 10. Ressourcen und Rahmenbedingungen

- **Teamgröße:** 11 Personen
- **Zeit pro Person:** 2–3 Stunden pro Woche
- **Projektlaufzeit:** 15.04.2026 – 30.06.2026
- **Budget:** kein Budget
- **Infrastruktur:** Gitty, Discord, lokale Entwicklung

### Rahmenbedingungen:

- Umsetzung aller Pflichtmodule
- saubere Repository-Struktur
- teamübergreifende Integration
- dokumentierter Entwicklungsprozess

## 11. Kommunikationswege

Kanal	Zweck	Frequenz
Discord / WhatsApp	Kommunikation	täglich
Gitty	Codeverwaltung	kontinuierlich
Meetings	Planung	wöchentlich
E-Mail	Betreuerkontakt	bei Bedarf

## 12. Definition of Done

Ein Feature gilt als abgeschlossen, wenn:

- implementiert und funktionsfähig
- getestet
- Code Review durchgeführt
- dokumentiert
- integriert

## 13. Abnahmekriterien

- alle Pflichtmodule implementiert
- vollständiger Dokumentenprozess vorhanden
- GUI funktionsfähig
- Tests erfolgreich
- Präsentation bestanden

## 14. Genehmigung

Je Untergruppe unterzeichnet ausschließlich der jeweilige **Gruppenleiter** (siehe Kapitel 6.2).

<b>Rolle</b>	<b>Name</b>	<b>Unterschrift</b>	<b>Datum</b>
Betreuer	Prof. Dr. Gerd Marmitt		
Gruppenleiter Gruppe E	<i>[zu bestimmen]</i>		
Gruppenleiter Gruppe F	<i>[zu bestimmen]</i>		
Gruppenleiter Gruppe G	<i>[zu bestimmen]</i>		
Gruppenleiter Gruppe H	<i>[zu bestimmen]</i>		