**Vision:**

Oftmals wenn Unternehmen ihre Produkte nicht verkaufen, geht ihnen viel Platz weg und Produkte, die einfach nur rumstehen, kosten viel Unterhalt. Wir sind der Ansicht, dass wir diese Problematik lösen können, indem wir individuelle, funktionale Software-Online-Auktionsplattformen entwickeln. Dem Kunden steht eine hohe Anzahl von Erweiterungen und Spezifikationen zu Verfügung (siehe Schema). Die domänenspezifische Software kann sich somit mit ihrem eigenen, einzigartigen Charakteristikum auf dem Markt behaupten.

Bildbeschreibung:

- 1 - Kernedomäne
- 2 - generisch
- 3 - spezifisch

Funktionale Anforderungen:

- ❑ Waren können in einer privaten Auktion (geschlossener Markt) oder öffentlich verkauft werden (offener Markt)
- ❑ Produkt kann als Versteigerung (auch Bieten) oder als Fix-Preis (statisch) publiziert werden
- ❑ Kunde kann sich jederzeit über den Status seiner Verkäufe und Käufe informieren
- ❑ Kunde kann bei zu vielen nicht bezahlten Rechnungen, Zahlungsaufforderungen gesperrt werden (wird durch Kundendienst individuell entschieden)
- ❑ Verkäufe an Mitarbeiter werden über die Lohnabrechnung abgebucht (private Auktionen)
- ❑ ~~Lieferart wird nach einem Verkauf bestimmt~~ Lieferart wird vor dem Verkauf bestimmt (auch mehrere möglich)
- ❑ Zahlungsmöglichkeiten sind Bar, Rechnung, Kreditkarte, ~~PayPal, Twint~~ (wurde für PayPal ersetzt, da weniger Gebühren fällig werden)
- ❑ Schnittstelle zum Buchhaltungssystem
- ❑ Lagerverwaltung
- ❑ Versandsystem
- ❑ externe Zahlungsplattform
- ❑ Authentifizierungs- und Autorisierungs-System (Single One Sign)
- ❑ Waren können in beliebigen Kategorien unterteilt werden
- ❑ Käufer und Verkäufer erhalten Notifikationen bei Verkauf / Kauf
- ❑ Verkäufer können bewertet werden (nur bei öffentlichen Auktionen)
- ❑ Waren können ins System importiert werden (z.B. als CSV)
- ❑ Es kann zwischen verschiedene Auktionsarten ausgewählt werden
- ❑ läuft OS-unabhängig (auf Smartphone, Tablet, Laptop und PC)
- ❑ automatisches Aktualisieren, z.B. eines Gebotspreis, ohne Seite zu refreshen

Bemerkung: Orange-geschriebene Anforderungen, sind Ergänzungen unsererseits.

- ❑ Verkäufer kann Bilder uploaden und eine textuelle Beschreibung hinzufügen.
- ❑ Der Zustand des Produktes muss angegeben werden (fabrikneu, fast neu, leichte Gebrauchsspuren, starke Gebrauchsspuren, defekt)
- ❑ Kunde kann mittels sehr benutzerfreundlichen Filters seine Suche einschränken
- ❑ Käufer- und Verkäuferschutz für z.B. Vorauszahlung ohne Lieferung oder nicht bezahlte Rechnung

Nicht-funktionale Anforderungen:

- ❑ benutzerfreundliches GUI
- ❑ spezifisches Firmen-Layout
- ❑ Software ist in ein Frontend- und Backend-Teil geteilt
- ❑ sollte modernen Sicherheitsstandards entsprechen
- ❑ termingerechter und hoch- qualitativer erster Release, dafür Kompromiss in Funktionsumfang

Domänenspezifische Anforderungen:

- ❑ als On-Premises oder Software-as-a-Service erhältlich
- ❑ 50% mehr Umsatz mit Lagerrestbeständen
- ❑ 80% aller Lagerrestbestände sollen innerhalb von zwei Wochen verkauft werden
- ❑ On-Premises werden beim Kunde eingerichtet und angepasst werden
- ❑ als Software-Produktlinie (mit individuellen Variationen) entwickelt werden
- ❑ der Software-Benutzer kann Gebühren für Verkäufe erheben (prozentual oder fix)
- ❑ Da Produkt internationalisiert werden soll, muss eine angemessene Systemdokumentation vorhanden sein

Ähnliche Produkte (Konkurrenz):

Plattform	Link	Bemerkungen
Ricardo	https://www.ricardo.ch	B2C, C2C, besonders für Schweizer Marktplatz, Grossbetrieb; Mobil- & Browserversion zeigen Unterschiede
Ebay	https://www.ebay.ch/	B2C, C2C, internationale Reichweite, globales Unternehmen
Kunsthaus	https://www.kunst-haus.ch/	B2C, Kunst- und Antiquitäten-Verkauf, Ankauf möglich, Schweizerisches KMU

Faktoren des Zühlke-Software-Prozessmodell:

- ❖ **Grösse** (Anzahl beteiligte Personen)
 - **agil:** ca. 3 Personen (direkte Kommunikation, mehrere Rollen pro Person)
 - **plangetrieben:** ca. 300 Personen (Führung durch ein Projektmanagement, formalisierte Projektinformationen)
- ❖ **Kritisch** (wie gross kann der Schaden auf eine einzelne Person auswirken)
 - **agil:** Komfortverlust (kaum Dokumentation, Testing innerhalb des Teams)
 - **plangetrieben:** Lebensverlust (sehr formelles Akzeptanz-Prozedere, fixe Anforderungen)
- ❖ **Teamfähigkeiten** (Verhältnis von den Entwicklern, die potentiell im Prozess tätig sein können und von den Entwicklern, die potentiell die Prozesse gestalten können)
 - **agil:** Änderungen am Prozess werden kontrolliert, Prozesse werden selten optimiert
 - **plangetrieben:** externe Entwicklern engagieren, welche sich um die Prozesse kümmern
- ❖ **Änderung:** Anzahl Anforderungen, welche zu Beginn aufgenommen werden und die bis zum Ende des Projektes gleich bleiben
 - **agil:** ca. 10% aller Anforderungen;
 - **plangetrieben:** ca. 90% aller Anforderungen
- ❖ **Kultur:** inwiefern wird Agilität im Umfeld des Projektes akzeptiert
 - **agil:** ca. 90% aller involvierten Arbeiter (Charakteristik: flache Hierarchien, kaum dokumentieren, kaum Zeremonien);
 - **plangetrieben:** ca. 10% aller involvierten Arbeiter (Charakteristik: steile Hierarchien, formelles Dokumentieren, viele Zeremonien)

Das Zühlke-Modell ist sicherlich ein geeignetes Tool, um herauszufinden, welcher Vorgehensprozess für das jeweilige Projekt am besten geeignet ist. Wir stehen aber zum Faktor «Change» sehr kritisch gegenüber, denn nur weil viele Anforderungen sich nicht ändern, ist eine tendenziell plangetriebene Entwicklungsumgebung nicht zwingend am besten geeignet. Beim Faktor «Culture» kann das Resultat insofern verfälscht werden, wenn viele Entwickler die agile Vorgehensweise gar nicht kennen und somit nicht beurteilen können. Was den Faktor «Size» betrifft, halten wir ihn für relativ. Denn es gibt auch die Möglichkeit, ein agiles Projekt, an dem viele Teilnehmer involviert sind, mit Hilfe von Frameworks wie SAFe (Scaled Agile Framework) abzuwickeln.

Software-Prozessmodell-Analyse projiziert auf das Projekt:

Size	Criticality	Team Skills	Change	Culture
ca. 10 Personen	Essential money	L1: 30% / L2&3: 70%	20%	ca. 50 - 70%
Geschäftsleiter, der Kund/Grosshändler, Vertretern des Grosshändler, Produktmanager, Chef-Softwarearchitektur, Development-Team (3 Personen)	Da das Produkt ein grosses Potenzial hat, von vielen Kunden benutzt zu werden, dürfen die Auswirkungen eines Fehlers keinen grossen Schaden am Produkt und an den Benutzer verursachen. Eine gute Dokumentation würde insofern helfen, um das Risiko eines Auftretens von einem Fehler vermindert werden zu können.	Weil das Development-Team klein ist, ist es einfacher Änderungen am Prozess zu kontrollieren. Der Projektleiter ist zugleich der Requirement Engineer.	Der Kunde erwartet als ein erstes Release ein termingerechtes MMP mit hoher Qualität.	Das Develop-Team kennt sich schon aus früheren Online-Shop-Projekten, weshalb sie miteinander schon Erfahrung sammeln konnten. Produkt sollte erweiterbar und individuell angepasst werden.

Dieses Projekt weist Merkmale auf, die dem agilen Vorgehen entsprechen. Eine passendes Software-Prozessmodell wäre zum Beispiel Scrum. Der einzige Faktor «Criticality» entspricht eher dem plangetriebenen Paradigma.

Artefakte, die vor einer Wahl einer Vorgehensweise helfen können:

- ❖ **SWOT Analyse**
SWOT steht für Strengths Weaknesses Opportunities Threats (deutsch: Stärken, Schwächen, Möglichkeiten, Gefahren). Der Begriff hat ökonomischen Ursprung und dient dazu, ein Unternehmen (oder auch eine Privatperson) zu analysieren und zu charakterisieren. Dadurch kann ein besseres Bild von sich gemacht werden und allenfalls können somit Verbesserungen vorgenommen werden. Kritisiert kann SWOT für die Beschränkung mit den vier Eigenschaften, denn kein Unternehmen (oder Person) lässt sich nur auf diese Eigenschaften beschreiben.
- ❖ **Marktforschung**
Für ein Unternehmen ist es immer wichtig, bei einer Entwicklung eines Produkts einen Blick auf den Markt zu werfen. Denn es ist wichtig, frühzeitig zu erkennen, wenn sich die Bedürfnisse auf den Markt verändern, entsprechend reagieren zu können. Im Grunde gibt es zwei Super-Kategorien:
 - ❖ **Primärmarktforschung:** diese wird angewendet, wenn kaum Daten vorhanden sind. Mittels Befragungen und Tests können Daten gesammelt und ausgewertet werden.
 - ❖ **Sekundärmarktforschung:** hier greift man auf innerbetriebliche Datenquellen oder holt diese über externe Quellen (z.B. Bundesamt für Statistik, Marktforschungsinstitut).

LE03 - Fallstudie

Ivo Gurtner & Sabrina Westhauser

Alle relevanten Stakeholder: Annahme: in der Fallstudie finden sich kaum echte Stakeholder, weshalb angenommen wird, dass z.B. der Produktmanager auch ein Stakeholder ist.

Wer?	Bemerkungen	potenzielle Stakeholder
Start-up Unternehmen "Cool Software Solution"	im E-Commerce-Softwarelösungen tätig	Nein, nur bedingt in einzelnen Personen und nicht als ganzes Unternehmen.
Geschäftsleitung von «Cool Software Solutions»	unbekannte Anzahl Personen	Ja, da GL hauptverantwortlich für Unternehmen ist = hauptverantwortlich für Projekt.
Projektleiter bzw. designierten Produktmanager von «Cool Software Solutions»	Hauptverantwortlich für Anforderung und Schnittstelle zum Kund	Ja, grosse Verantwortung als PL und PM.
Chef-Softwarearchitekt von «Cool Software Solutions»	verantwortlich für Gesamtarchitektur der E-Commerce-Suit	Ja, hauptverantwortlich für Architektur.
«Big Sales», der Grosshändler	Kunde der Online-Shop-Software	Ja, er hat grosses Interesse am Produkt.
Vertreter von Grosshändler «Big Sales»: Verkaufschef, IT-Verantwortlicher	Mitentwickler	Ja, Interesse und Motivation ist gross, aber vielleicht zu grossen Einfluss aufgrund Unternehmensgrösse und Finanzstärke.
Development-Team	3 Entwickler	Nein, sie haben keine Einfluss auf die Anforderungen des Produkts.

Ziele, mit Stakeholdern verbinden (allfällige Zielkonflikte markieren)

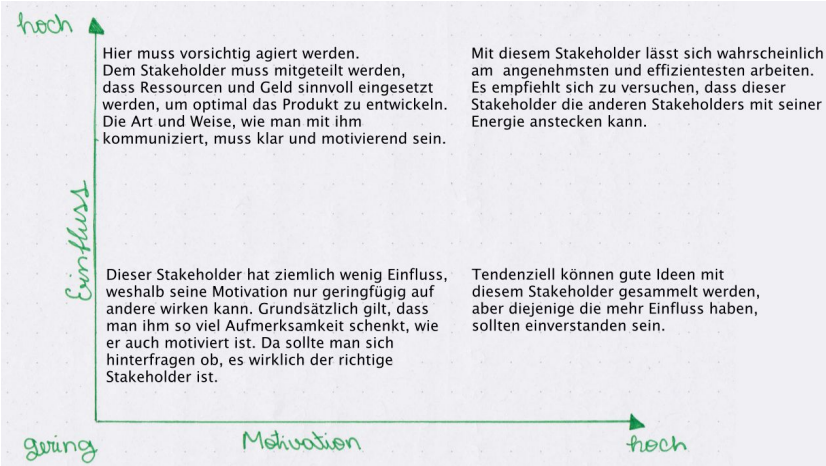
Was, bis wann, und wie? SMART?

- 1) 80% der Lagerrestbestände sollen innerhalb von 2 Wochen nach Artikelveröffentlichung verkauft werden. **Lukrative Preise (z.B. Startpreise ab Fr. 1.-) sollen helfen, dass möglichst viel Aufmerksamkeit erzeugt werden kann).**
- 2) 50% mehr Umsatz soll mit dem Verkauf der Lagerrestbestände erreicht werden
- 3) Akzeptierte Zahlungsmittel sind Bar, Rechnung, Kreditkarte und Twint. **Rechnungen müssen innerhalb von 30 Tagen beglichen werden.**
- 4) Mittels einmaliges Anmelden soll der Client identifiziert und ihm die entsprechenden Berechtigungen erteilt werden.
- 5) Die Online-Auktionsplattform soll als Software-Produktlinie entwickelt werden. Mit individuelle Ausprägungen auf Basis einer gemeinsamen Plattform.
- 6) On-Premises werden beim Kunde eingerichtet und angepasst.

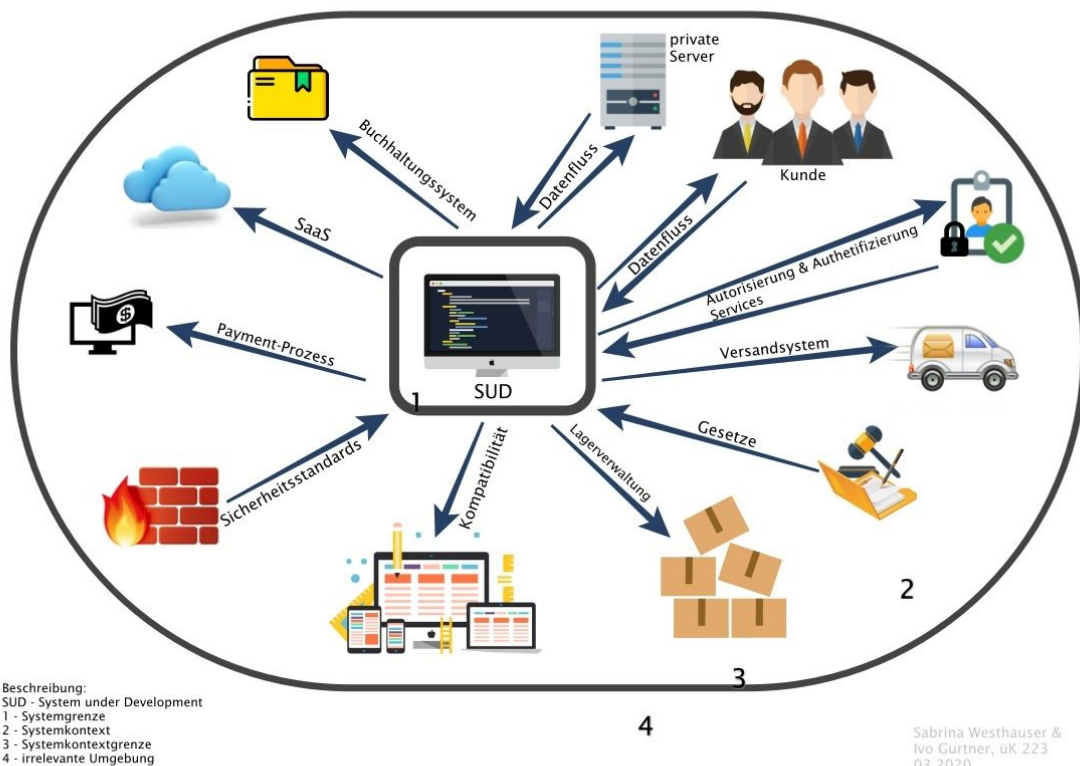
Bemerkung: **Orange-geschriebene Sätze, sind Ergänzungen unsererseits.**

Ziel Nr.	Stakeholder	S	M	A	R	T
1	Vertreter von Grosshändler und Projektleiter seitens Start-Up	Ja	Ist relativ.	Ja	Nein, gerade am Anfang stellt sich die Frage, ob innerhalb von 2 Wochen so viele mögliche Kunden gelotst werden können.	Nur bedingt, kein fixes Datum, nur Periode.
2	Vertreter von Grosshändler und Projektleiter seitens Start-Up	Ja	Ist relativ.	Ja	Ja	Ist nicht gegeben.
3	Vertreter von Grosshändler und Projektleiter seitens Start-Up	Nur bedingt, Kreditkarten-marke unbekannt.	Ja	Ja	Ja	Ist relativ.
4	Vertreter von Grosshändler und Projektleiter seitens Start-Up	Ja	Ja	Ja	Ja	Ist nicht gegeben.
5	Start-up Geschäftsleitung, Produktmanager, Chef-Softwarearchitekt	Ja	Ja	Ja	Ja	Ist relativ.
6	Start-up Geschäftsleitung, Produktmanager, Chef-Softwarearchitekt	Zu oberflächlich.	Kundenzufried-enheitindikator.	Ja	Zu wenig spezifisch, mehr Details benötigt. Was ist, wenn wir 50 Kunden gleichzeitig On-Premises liefern müssen → Ressourcen sind nicht vorhanden.	Ist nicht gegeben.

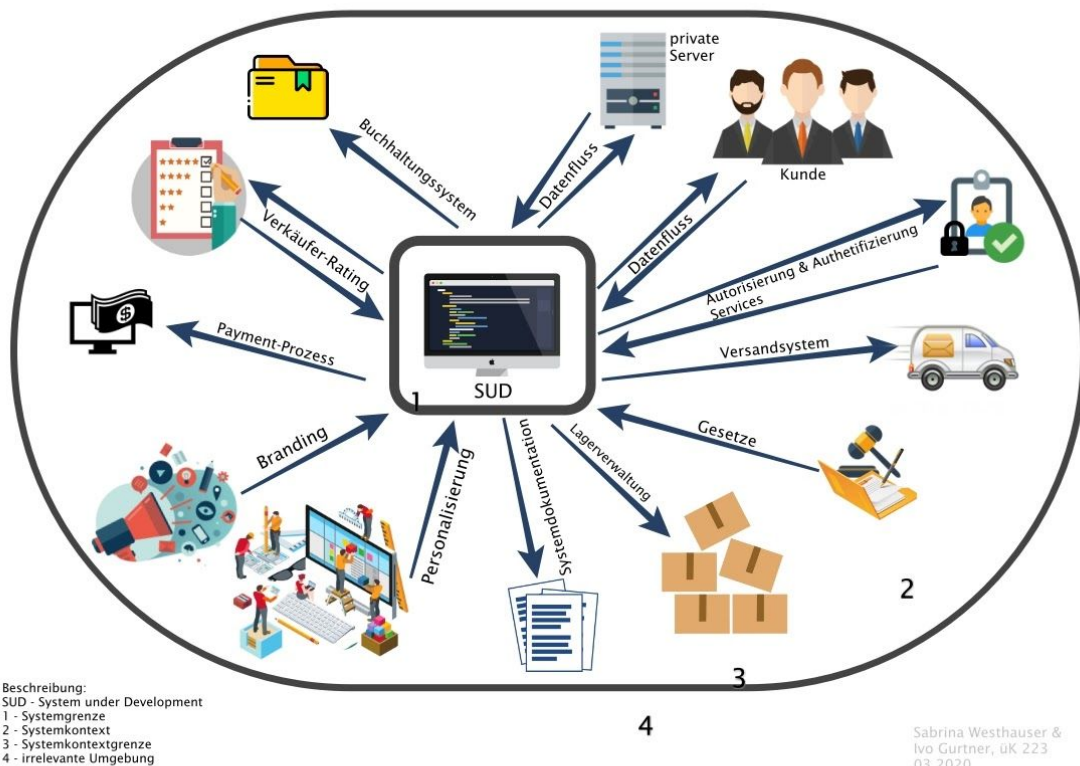
Mit SRM alle 4 Quadranten beschreiben, wie man Stakeholders ermutigen kann.



System- und Kontextgrenze des Kernsystems



System- und Kontextgrenze des erweiterten Systems



Welche Rolle nahm der Dozent ein?

Der Dozent war in der Rolle als Moderator. Er stellte den Stakeholders die Frage («Welche Eigenschaften muss ein frischer Informatik-Ingenieur haben, damit ihr ihn einstellt?») in den Raum. Er war sich bewusst, dass bei einer solch offener Frage unterschiedliche Meinungen aufkommen, weshalb er zuerst alle Inputs der Stakeholders aufnahm. Dabei fragte er manchmal nach, ob er die Stakeholders richtig verstanden hatte. Falls ihm der Input zu wenig genau war, machte er Vorschläge oder suchte auch Kompromisse. Der nächste Schritt bestand darin, dass er die Inputs gemeinsam mit den Stakeholders kategorisierte. Hier entstand viel Diskussionsbedarf. Der Dozent blieb aber konzentriert, handelte lösungsorientiert und suchte aktiv nach Kompromisse. Daraufhin wurden die Eigenschaften priorisiert. Jeder Stakeholder konnte beliebig 2 Punkte vergeben. Als Output kamen 3 Eigenschaften heraus: Teamfähigkeit, Technisches Wissen und Motivation.

Wie hat er sich gegenüber den Stakeholdern verhalten?

- neutral → er behandelte jeden Stakeholder gleich, er wertete die aufgenommenen Inputs nicht
- lösungsorientiert → bei Unstimmigkeiten nahm er das Zepter in die Hand und suchte nach Kompromissen
- empathisch → er zeigte ein gutes Einfühlvermögen in die Stakeholders und bei Kompromissen fragte er jedes Mal nach, ob alle damit einverstanden sind
- führend → ohne sich von den Stakeholdern abzuheben oder besser darzustellen, führte er die Diskussion an. Er hielt stets den Überblick und achtete darauf, dass alle Schritte artgerecht ausgeführt wurden
- präzise → war ihm ein Input zu vage, schlug er Gegenvorschläge vor
- selbstbewusst → er war sich seinem Handeln ganz gewiss und zeigte dies auch mit seiner Ruhe und Abgeklärtheit
- nachhakend → falls er sich unsicher bei der Aufnahme eines Inputs fühlte, fragte er nach, um seine Annahme zu bestätigen oder zu ersetzen

Welche Techniken wurden angewandt, um das Interview aktiv zu beeinflussen?

- Einleitung (Frage erklären, Interesse wecken)
- Input-Aufnahme (normalerweise mit Dashboard oder Pinnboard, heute aber mit Stickies auf MacBook)
- Kategorisieren der Inputs
- Priorisierung der Inputs
- Abschluss (Präsentieren der Ergebnisse)

LE06 - Fallstudie

Sabrina Westhauser & Ivo Gurtner

Quellenangabe: Anforderungen wurde von der Fallstudie-Dokumentation und von der LE01 übernommen.

Funktionale Anforderungen:

Ursprüngliche Anforderung	Bemerkungen / Mängel	Überarbeitende Anforderung
Waren können in einer privaten Auktion (geschlossener Markt) oder öffentlich verkauft werden (offener Markt).	Keine.	Waren können in einer privaten Auktion (geschlossener Markt) oder öffentlich verkauft werden (offener Markt).
Die Online-Auktionsplattform soll den aktuellen Preis einer Auktion allen aktiven Benutzern ohne Browser-Refresh anzeigen.	Inwiefern soll das realisiert werden?	Das Produkt sollte den aktuellen Preis einer Auktion automatisch aktualisieren, ohne dass die Seite neu geladen werden muss. Dies kann mit einer Single-Page-Application erreicht werden.
Lieferart wird vor dem Verkauf bestimmt (auch mehrere möglich)	Welche Lieferarten sind möglich?	Lieferart wird vor dem Verkauf bestimmt. Lieferarten sind Abholung, Versand, Spediteur, Lieferung von Verkäufer. Der Ort (=Adresse) und die Zahlungsarten können angepasst werden.

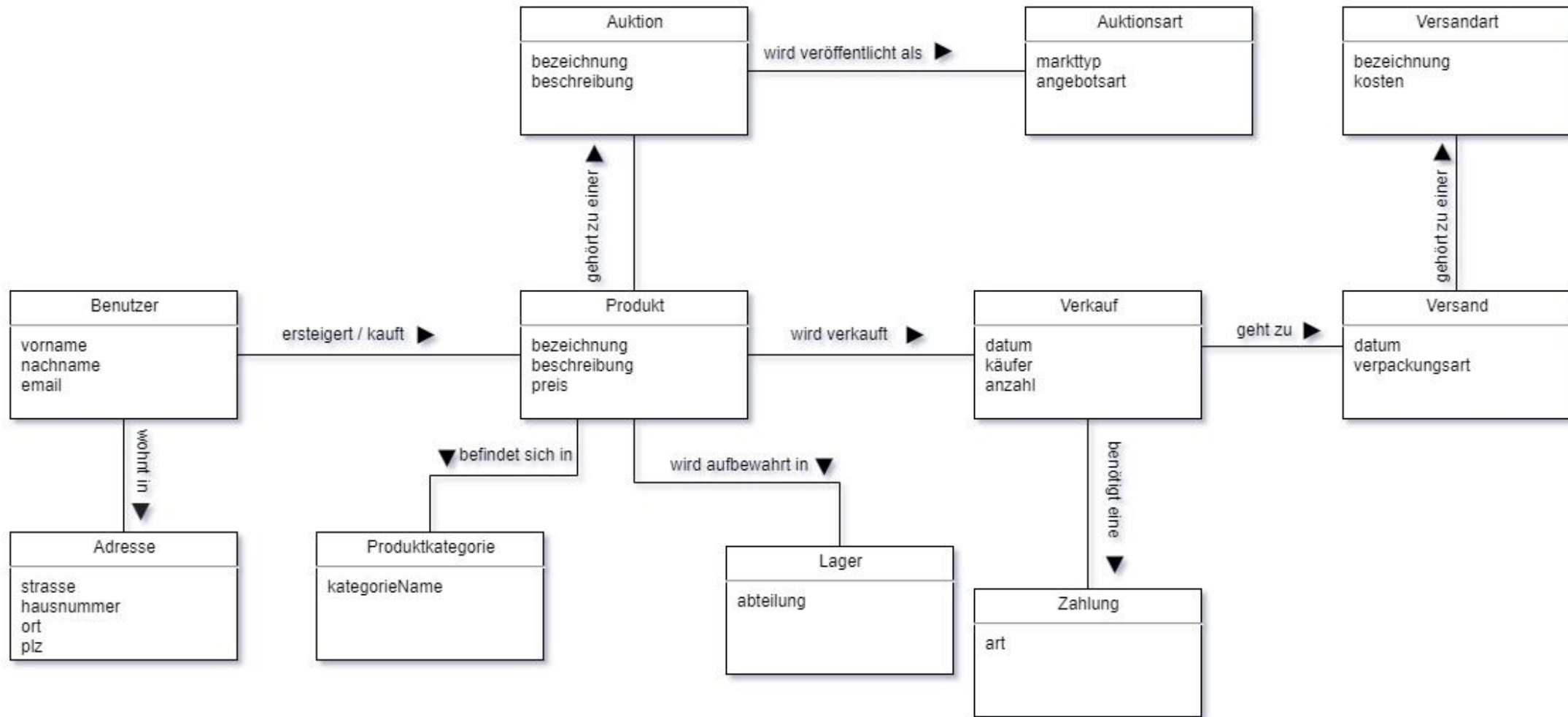
Nicht-funktionale Anforderungen (Qualitätsanforderungen):

Ursprüngliche Anforderung	Bemerkungen / Mängel	Überarbeitende Anforderung
Die Online-Auktionsplattform soll in einen Frontend- und einen Backend-Teil aufgesplittet werden.	Zu oberflächlich, gibt es irgendwelche Schnittstellen?	Die Software soll in einer 3-Tier-Architektur (Presentation-Layer (Frontend), Business Layer und Data Layer (beide Backend)) integriert werden.
Branding der Webseite	Zu vage.	Das Firmenlayout kann ins kundenspezifische Software-Produkt eingebettet werden. Dafür müssen Kundengespräche zusammen mit dem Designer-Head-Developer und dem Projektleiter stattfinden.
Externe Payment-Prozess via Datatrans oder Saferpay für Autorisierung und Settlement	Kosten? Sicherheit?	Damit die Zahlungsprozess sicher und erfolgreich abläuft, sollte das System eine externe Zahlungsplattform inkludieren. Eine Option wäre Datatrans, da sie hochskalierbare Infrastruktur für einen guten Preis offerieren.

Nicht-funktionale Anforderungen (Randbedingungen):

Ursprüngliche Anforderung	Bemerkungen / Mängel	Überarbeitende Anforderung
Das neue E-Commerce-Modul soll so ausgelegt werden, dass Sie als On-Premises oder SaaS bei Kunden betrieben werden kann.	Widersprüchliche Anforderung → entweder als On-Premises oder SaaS	Das E-Commerce-Modul wird als Saas (= Software as a service) angeboten und wird auf einer Cloud platziert.
Da Produkt internationalisiert werden soll, muss eine angemessene Systemdokumentation vorhanden sein.	Zu wenige Details, welche Sprachen?	Da das Produkt expandiert werden soll, muss eine angemessene Systemdokumentation in den Sprachen Englisch, Deutsch, Französisch, Spanisch, Portugiesisch und Chinesisch erfasst werden.
Termingerechter und hochqualitativer erster Release, dafür Kompromiss in Funktionsumfang.	Wann? Genau Qualitätsangaben müssen deklariert werden.	Das erste MMP, welches released werden soll, muss nach der ersten Iteration (ca. 4 Wochen) implementiert werden und muss den Software-Qualitätsmerkmalen des Modells ISO 25010 entsprechen (Functional Suitability, Performance Efficiency, Compatibility, Usability, Reliability, Security, Maintainability, Portability).

Sabrina Westhauser & Ivo Gurtner

**Domänenmodell / Konzeptionelles
Klassendiagramm
Fallstudie**

Ursprüngliche Anforderung: Verkäufer kann Bilder uploaden und eine textuelle Beschreibung hinzufügen.
Bemerkung: Anforderung stammt von LE01 und wurde von uns selbst hinzugefügt.
Diese Anforderung wird nun in die drei verschiedenen Use-Case-Spezifikationen transformiert.

Brief (= sehr kurzgefasste Anforderung, meistens 1 Satz, implementationsunabhängig [Analyse]):
Verkäufer kann Bilder und Beschreibung für seine Artikel hinzufügen.

Casual (ausführlichere, textuelle Beschreibung, mehrere Sätze, beinhaltet auch Fehler-Behandlung, oft implementationsunabhängig [Analyse]):

Der Verkäufer kann bis zu 10 Bilder hochladen und beliebig mit einem Tool bearbeiten. Dazu kann er die Beschreibung und Spezifikationen des Produkts schreiben. Ihm steht für die «Artikelbeschreibung» ebenso ein Tool zur Verfügung. Falls die Bilder nicht hochgeladen werden können, erscheint ein kleines aufpoppendes Fensterchen mit Fehlermeldung.

Fully-dressed (tabellarische, ausführlichere Erläuterung der Anforderung, oft implementationsabhängig [Design]):

Abschnitt	Erläuterung
Use Case Name	Bilder hochladen + textuelle Beschreibung hinzufügen
Akteure	Client (Verkäufer)
Vorbedingung	<ul style="list-style-type: none">- Ein User muss eingeloggt und dementsprechend authentifiziert sein.- Upload von Bilder muss funktionieren sowie Validierung der Texteingaben.
Nachbedingung	Nach erfolgreichem Erstellen eines Artikels kommt der Client zurück zur Verkaufsübersicht. Im Hintergrund werden Daten in die Datenbank gespeichert.
Ergebnis	Artikel erstellt
Hauptszenario	<ol style="list-style-type: none">1. Der Client kann bis zu 10 Bilder hochladen.2. Als Hilfe kann er ein Tool nutzen um Bild zuschneiden, Kontrast ändern und Bild beliebig drehen.3. Der Client kann Textuelle Beschreibung hinzufügen.<ol style="list-style-type: none">a. Zustand (Auswahlbox; fabrikneu, fast neu, leichte Gebrauchsspuren, Gebrauchsspuren, starke Gebrauchsspuren und defekt)b. Titel (Textfeld)c. Untertitel (Textfeld; optional)d. Artikelbeschreibung (Textfeld; dieses wird mit einem kleinen Tool ergänzt, mit welchem man den Text formatieren kann [Schriftgrösse, Textausrichtung, Auflistung (numerisch oder förmig), Schriftfarbe, Texthervorhebungstyp])e. Artikelanzahl4. Eine Validation wird gebraucht, um herauszufinden, ob der Client alle relevante Felder ausgefüllt hat und ob die Einschränkungen befolgt wurden.5. Mittels eines Buttons kann der Client den Artikel erstellen..
Alternativszenarien	<ol style="list-style-type: none">1. Bilder können nicht hochgeladen werden<ol style="list-style-type: none">a. System schickt Fehlermeldungen (z.B. Bilder zu gross)2. Die Validierung wird nicht befolgt vom Client<ol style="list-style-type: none">a. Ihm wird angezeigt, wie er es eingeben muss
Ausnahmeszenarien	Artikel kann aufgrund keiner Verbindung zum Backend nicht erstellt werden.
Qualitäten	Benutzerfreundlichkeit, Einfachheit, Übersicht.