

Projektarbeit

Anforderungsanalyse

**Design eines Kundenverwaltungssystems für die Firma
PackZeugs AG**

Datum: 05.05.2013

Autoren: Pascal Kern
David Marmy

Klasse: TSI1209I

Inhaltsverzeichnis

1. Aufgabe: Anforderungsanalyse (Vision und Rahmenbedingungen)	
1.1. Die Vision	1
1.2. Sechs mögliche Stakeholder.....	1
Indirekte Stakeholder:	1
1.3. Spezifische Ziele der Stakeholder	2
Indirekte Stakeholder:	2
1.4. Ergänzung der Ausgangslage	2
Funktionelle Anforderungen	2
Nicht-Funktionelle Anforderungen	3
1.5. Vier mögliche Risiken.....	3
Weiter Risiken:	3
1.6. Systemkontextdiagramm.....	4
2. Aufgabe: Anforderungsanalyse (Anforderungen beschreiben)	
2.1. Anwendungsfalldiagramm.....	5
Anwendungsfall „Kunde Erstellen“ und fünf weitere	5
2.2. Priorisieren Anwendungsfälle und nicht-funktionale Anforderungen	5
Erläuterung Kriterien	5
Anwendungsfälle	6
Nicht-funktionale Anforderungen	6
2.3. Beschreiben Anwendungsfälle	6
„Kunde erstellen“	6
2.4. Beschreibung weiterer Anwendungsfall (Priorität hoch).....	8
Weiterer Anwendungsfallbeschreibung (zusätzliche Übung).....	8
2.5. Aktivitätsdiagramm „Kunde erstellen“ Normal- und Sonderfall	10
2.6. Iterationen im Entwicklungsprozess	11
Anwendungsfälle	11
3. Aufgabe: Objektorientierte Analyse (OOA) (Analysemodell/Fachklassenmodell erstellen)	
3.1. Fachklassenmodell zu Anwendungsfälle	13
Anwendungsfall „Kunde Erstellen“ und „Kunde anzeigen“	13
3.2. Sequenzdiagramm für Normalablauf	13
Anwendungsfall „Kunde Erstellen“	13
3.3. Zustandsdiagramm für die Abstraktion „Kunde“	14
3.4. Glossar zur Fachliche Verantwortlichkeit	14
4. Aufgabe: Objektorientiertes Design & Programmieren (OOA/D/P) (Entwurfsmodell erstellen und implementieren)	
4.1. Glossar ergänzen englische Namen der Klassen	15
4.2. Fachklassendiagramm 1. Iteration in Entwurfsmodell Fachklassenmodell überführen.....	16
4.3. Fachklassenmodell in 3-Schichtenarchitektur integrieren.....	17
4.4. Entwurfsmodell mit Sequenzdiagramm überprüfen	18
4.5. Entwurfsmodell in Prototyp implementieren	18
4.6. Prototyp testen	19
4.7. Quelltext per Javadoc als HTML-Dokument erstellen.....	19
5. Aufgabe: Objektorientierte Analyse, Design & Programmieren (OOA/D/P) (Analyse- und Entwurfsmodell anpassen, Entwurfsmodell implementieren)	

Inhaltsverzeichnis

5.1. Anpassungen (Fach-) Klassendiagramm - Analysemodell.....	20
5.2. Angepasstes Klassendiagramm - (Entwurf-) Designmodell.....	21
5.3. Erweiterung Design- Entwurfsmodell in "Business" und "Persistence" Schicht um abstrakten Klassen und Schnittstellen	22
5.4. Klassen- und Komponentendiagramm mit "Singleton" und "Factory" (GoF) Design Pattern auf Business und Persistence Schicht (3-Schichten Modell)	22
5.5. Sequenzdiagramm zur Überprüfung Normalfall "Kunde erstellen" - Korrekturen ableiten.....	23
5.6. Glossar ergänzen um Klassen und Schnittstellen aus Design- Entwurfsmodell [5.3]	23
Layer Presentation	23
Layer BusinessLogic	23
Layer Persistence.....	24
5.7. Angepasstes Design- Entwurfsmodell implementieren [5.3]	25
5.8. JUnit Test auf "Business" und "Persistenz" Schicht für Normalfall "Kunde erstellen"	25
5.9. Normalfall "Kunde erstellen" und "Kunde anzeigen" testen.....	25
5.10. JavaDoc generieren	25
6. Anhang	
6.1. Aufgabenstellungen.....	26
Aufgabenstellung 1:	26
Aufgabenstellung 2:	28
Aufgabenstellung 3:	31
6.2. Aufgabenstellung 4:	33
Bewertung Aufgabe 4 eMail vom 27.11.13.....	34
Aufgabenstellung 5:	35
Bewertungskriterien Aufgabe 5:	36
Bewertung Aufgabe 5 (Teil 1 und 2) eMail vom	36
Bewertung Aufgabe 5 eMail vom	37
6.3. Abgaberichtlinien.....	38
6.4. Zusatzinfos zu den Aufgaben	39
Aufgabe 4.....	39
Aufgabe 5.....	41

1. Aufgabe: Anforderungsanalyse (Vision und Rahmenbedingungen)

1.1. Die Vision

Ein Kundenverwaltungssystem ...
 ... zur Pflege von Kundeninformationen an allen Firmenstandorten
 ... um Kundeninformationen zwecks Kundenbindung ~~allen Mitarbeitern~~ zur Verfügung zu stellen.

1.1. Sechs mögliche Stakeholder

Stakeholder	Begründung
Sachbearbeiter (Auftragswesen)	<ul style="list-style-type: none"> Sie haben täglich mit Kundendaten zu tun.
Standortleiter	<ul style="list-style-type: none"> Sind interessiert an sauberen und vollständigen Kundendaten für ihren Standort, ihre Region.
CEO / GL	<ul style="list-style-type: none"> Sind interessiert an sauberen und vollständigen Kundendaten für das Unternehmen. Die Mitarbeiter sollen effizient und einfach ihre täglichen Arbeiten verrichten können. Interesse an Statistiken und Auswertungen der Umsätze und Verkauf im Bezug auf die Kunden(-daten). (Denn besten und persönlichen Auftritt für die Kundschaft.)
Verkäufer/Vertrieb/Aussendienst	<ul style="list-style-type: none"> Müssen/wollen sich auf die Kundendaten im System verlassen können. Müssen die Ansprechpersonen und Adressen ihres Gebiets kennen und einfach abfragen können.
Marketing	<ul style="list-style-type: none"> Brauchen korrekte und komplette Kundendaten für ein gezieltes Marketing.
In-House IT	<ul style="list-style-type: none"> Schulung der Mitarbeiter. Wartung und Verantwortung für das System und dessen Verfügbarkeit.

Indirekte Stakeholder:

Stakeholder	Begründung
Logistik	<ul style="list-style-type: none"> Bei Lieferungen keine Fehler in den Adressen welche zu Zusatzaufwänden führen.
Gesetz/Behörden	<ul style="list-style-type: none"> Die Firma handelt im Wirtschaftsraum und muss somit Bedingungen erfüllen und Regeln einhalten.

Pascal Kern Schule Apr 6, 7:15 pm

Anpassungen nach Besprechung:
 An der Grenze der Ausführlichkeit/Details. Keine Redundanz!

=> Nicht zu allgemein erfassen, in diesem Falle so OK!

Pascal Kern Schule Apr 6, 7:27 pm

Ergänzend.

Partner (Geschäftspartner/ Produktion)	<ul style="list-style-type: none"> • Stabiles Geschäftsfeld des Partners (Abnehmers) der Produkte.
---	---

1.2. Spezifische Ziele der Stakeholder

Stakeholder	Ziele
Sachbearbeiter (Auftragswesen)	<ul style="list-style-type: none"> • Einfacher zugriff auf alle Daten eines Kunden. • Einfache Anpassung der Kundendaten mit Schutz vor Fehler (Doppelte, falsche Einträge). • Möglichst intuitive Arbeitsweise und einfache Vernetzung mit anderen Systemen im Betrieb (Auftragsbearbeitung, Rechnungswesen)
Standortleiter	<ul style="list-style-type: none"> • Übersicht auf Kundendaten seines Standortes. • Statistiken zu seinem Standort.
CEO / GL	<ul style="list-style-type: none"> • Statistiken zum Gesamtgeschäft (var. Kriterien.) • Saubere Kundendaten aller Kunden (keine Doppelerfassungen, Falscheinträge).
Verkäufer/Vertrieb/Aussendienst	<ul style="list-style-type: none"> • Kundenliste(n) der jeweiligen Segmente (Kundenart, Region). • Korrekte Adressen (Lieferung und Rechnung) sowie Ansprechpersonen.
Marketing	<ul style="list-style-type: none"> • Statistiken zu Kundenart und Verkäufen.
In-House IT	<ul style="list-style-type: none"> • Einfache Wartung und Ausbaubarkeit des Systems. • Stabilität und Datensicherung des zentralen Systems. • Einfache Schulung der Mitarbeiter. • (Möglichst geringe Systemvielfalt.)

Indirekte Stakeholder:

Stakeholder	Ziele
Logistik (indirekter Stakeholder)	<ul style="list-style-type: none"> • Korrekte Lieferadressen.
Gesetz/Behörden	<ul style="list-style-type: none"> • Die Firma soll die Gesetze (Datenschutz, etc) einhalten und Nachweise erbringen können.
Partner (Geschäftspartner/ Produktion)	<ul style="list-style-type: none"> • Konstante Nachfrage und zuverlässige Partnerschaft erwünscht.

1.3. Ergänzung der Ausgangslage

Funktionelle Anforderungen

- Daten Zentral verwalten und warten.
- Kundendaten (de- und Zentral) erfassen, löschen und bearbeiten
- Drucken

Pascal Kern Schule Apr 6, 7:27 pm

Ergänzend.

Die meisten punkte stimmen zwar auf unsere Software als „System“ betrachtet. Doch sind sie nicht auf unsere Software selbst und deren Umfang/Inhalt bezogen...?!

Pascal Kern Schule Apr 6, 7:33 pm

Funktionelle Anforderungen sind oft genau die Anforderungen die von Anfang an als selbstverständlich erscheinen. Bsp. Der Drucker – Druckt!

Pascal Kern Schule Apr 6, 7:38 pm

Begründungen fehlen teilweise! Ergänzt.

Belege und Kundeninformationen

Nicht-Funktionelle Anforderungen

- Weboberfläche
Die Weboberfläche muss von allen Standorten erreichbar sein und das mit einer annehmbaren Geschwindigkeit (Standleitung).
- Datenzugriff
Einfacher Zugriff auf die/alle Daten über Web-Browser.
- Generisches Interface
pro Benutzer konfigurierbare, Benutzeroberfläche.
- Systemüberwachung
Systemadministration kann Ressourcen, Leistung, usw kontrollieren.
- Um-Systeme anbinden
Zugriff auf Daten aus dem Finanz-, Benutzerverwaltung, Webshop und Auftragswesen.
- ~~Die Erfassung und Bearbeitung von Kundeninformationen.~~
- Simultaner Zugriff
Die Daten müssen von allen Standorten und verschiedenen Benutzern gleichzeitig bearbeitet werden können (Datenqualität).

1.4. Vier mögliche Risiken

- Datenverlust
durch fehlendes Backup.
- Sicherheit
Datenschutz intern wie extern (nicht geklärt/hoch) sowie keine Ausfallsicherheit (Gesamt- oder Teilsysteme).
- Einschränkungen bei Entwicklung
Die Anforderung, die alte Datenbank weiter zu betreiben, schränkt die Möglichkeiten für die Entwicklung des neuen Systems ein.
- Abhängigkeiten
Durch die Erweiterung der Kundenverwaltung und Anbindung an Benutzer-, Finanz-, Webshop und Auftragsverwaltung besteht eine Abhängigkeit (zukünftige Änderung der Fremdsysteme, Inkompatibilität, Mehraufwände durch Anpassungen, etc).
- Akzeptanz
Mögliche fehlende Akzeptanz der Mitarbeiter und internen IT Abteilung.
- Anforderungsänderungen
Mit der Einführung des Systems könnten sich sogleich die Anforderungen ändern da neue Möglichkeiten erkannt werden.

Weiter Risiken:

- Service Abhängigkeit
Die Software, da über Weboberfläche ansteuerbar, ist vollständig von der Internetkonnektivität abhängig. Fällt das Internet aus, kann die Software nicht mehr genutzt werden, dies kann einen oder alle Standorte betreffen.
- Fehlende Skalierung des Datendurchsatzes/Resourcenauslastung

Pascal Kern Schule Apr 6, 7:31 pm

Anforderungen die fälschlicherweise bei den funktionellen gelandet sind.

Nicht-Funktional
Bearbeitung der Daten =
Funktional

Pascal Kern Schule Apr 6, 7:44 pm

Begründungen fehlen Teilweise.
Ergänzt und zusätzliche Risiken ergänzt.
Merke: Die ergänzten Risiken kommen bei praktisch allen Projekten vor!

Lokales Repository löst das Problem

Bei wachsender Anzahl Zugriffe/Benutzer kann die Betriebsgeschwindigkeit der Software leiden, da diese von der Internetanbindung abhängig ist. Internetanbindung müsste angepasst werden.

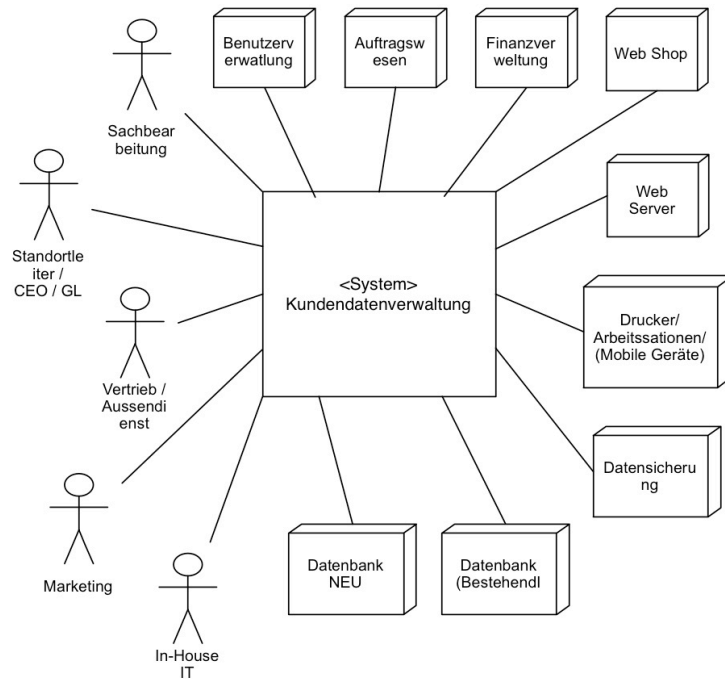
1.5. Systemkontextdiagramm

Pascal Kern Schule Apr 6, 7:51 pm

Die Firewall und Internet-Standleitung haben nichts direkt mit unserem System zu tun – sprich sie beeinflussen die Entwicklung und unser System nicht direkt. => Hardware hat meistens nicht viel im Sys-Kontextdiagramm verloren (Ausnahme von der Software angesteuerte wie Drucker, etc!)

Dafür hat der Web-Shop gefehlt!

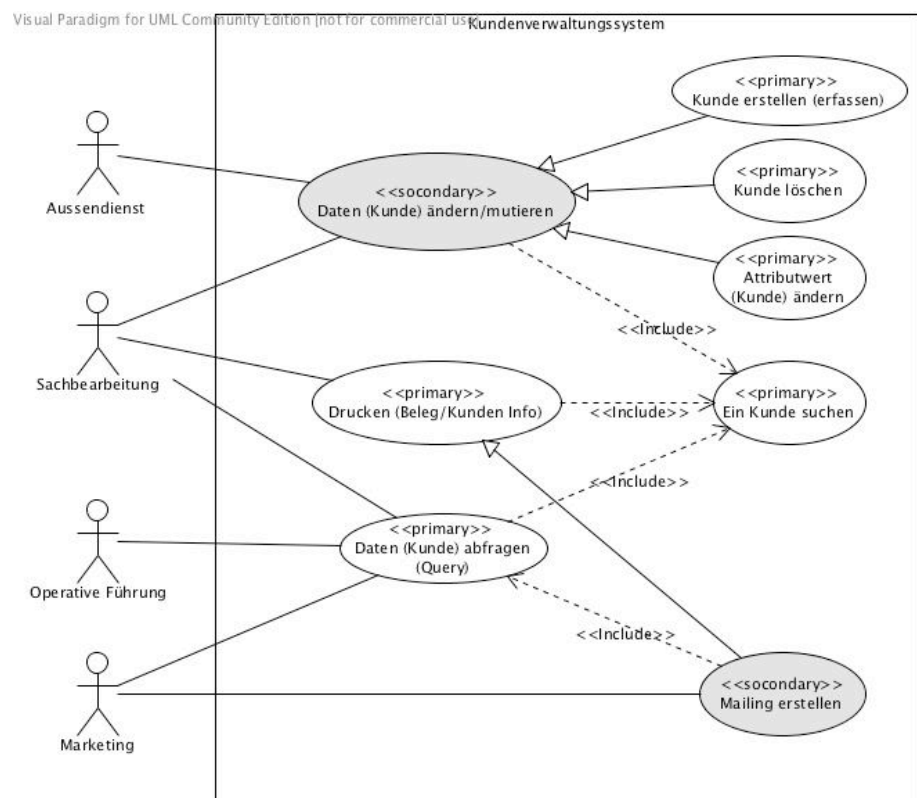
Gut ist auch die Aufteilung der Akteure in deren Rollen, ein Akteur USER würde auch gehen ist aber nicht korrekt.



2. Aufgabe: Anforderungsanalyse (Anforderungen beschreiben)

2.1. Anwendungsfalldiagramm

Anwendungsfall „Kunde Erstellen“ und fünf weitere



2.2. Priorisieren Anwendungsfälle und nicht-funktionale Anforderungen

Erläuterung Kriterien

Berechnungsformel der Prioritäten

Für jeden Anwendungsfall werden die Kriterien Systemrelevanz und Geschäftsrelevanz mit Werten zwischen 1 für kleine Priorität und 6 für die höchste Priorität bewertet.

Die gesamte Priorität wird dann wie folgt berechnet:

$$\text{Priorität} = \sqrt{A^2 + B^2}$$

Pascal Kern Schule Jun 9, 3:28 pm

Bezeichnungen geändert:
Kunde erstellen (erfassen) -> Kunde erstellen
Kunde löschen
Ein Kunde suchen -> Kunde suchen
Daten (Kunde) ändern -> Kundendaten ändern
Mailing erstellen
Daten (Kunde) abfragen (Query) -> Kundendaten abfragen
Drucken (Beleg/Kundeninfo) -> Kundendaten-Beleg drucken
~~Attributwert (Kunde) ändern~~

Pascal Kern Schule Jun 9, 3:26 pm

Kategorisierung Prim/Sec (Daten ändern und Mailing) entfernt.
Anwendungsfälle „Daten ändern“ entfernt!

Pascal Kern Schule Jun 9, 3:27 pm

Bezeichnungen geändert:
Datenzugriff (einfach) -> Einfacher Datenzugriff

Pascal Kern Schule Jun 9, 3:22 pm

Die Vorbedingung Benutzer Anmelden fehlte – war auch als Schritt nicht vorhanden!
Gilt auch für die anderen zwei UCs

Titel gemäss Änderungen in UC Liste angepasst.

Aufgabe 2: Anforderungsanalyse (Anforderungen beschreiben)

Verwendete Kriterien und Begründung

A = Systemrelevanz Für das System und dessen Funktionsweise benötigte Anwendungsfälle.
B = Geschäftsrelevanz Für den Geschäftsablauf benötigte Anwendungsfälle.

Anwendungsfälle

(Erstellt aus den Anforderungen von Auftrag 4 der Aufgabe 1)

Anwendungsfall	Prio Total	Kriterien	
		A	B
Kunde erstellen	8.49	6	6
Kunde löschen	7.81	6	5
Kunde suchen	7.21	6	4
Kundendaten ändern	6.95	3.5	6
Mailing erstellen	6.32	2	6
Kundendaten abfragen	5.39	2	5
Kundendaten Belegdrucken	4.61	3	3.5

Nicht-funktionale Anforderungen

(Übernommen aus der Aufgabe 1 vom Auftrag 4)

Anforderungen	Prio Total	Kriterien	
		A	B
Weboberfläche	8.49	6	6
Um-System Anbindungen	8.49	6	6
Simultaner Zugriff	8.49	6	6
Systemüberwachung	4.92	4.5	2
Einfacher Datenzugriff	4.74	1.5	4.5
Generisches Interface	3.61	2	3

2.3. Beschreiben Anwendungsfälle

„Kunde erstellen“

Fallname:	Kunde erstellen		Fall-ID:	1
Änderungsgshistorie:				
21.04.13		Erfassung des Falles durch Pascal Kern		
09.06.13		Vorbedingung ergänzt (fehlte als Schritt oder eben Verbindung)		
Ziel:		Ein Kunde wird im System durch Eingabe seiner Daten erfasst. Dem Kunden-Datensatz wird eine eindeutige Nummer vergeben.		
Kategorie:	Geschäfts	System	Primär	Sekundär
		X	X	
Akteur(e):	Sachbearbeitung, Aussendienst und eft. Marketing			

Aufgabe 2: Anforderungsanalyse (Anforderungen beschreiben)

Fallname:	Kunde erstellen	Fall-ID:	1
Geschäftliche Auslöser: (fachliche Gründe dafür, dass der Anwendungsfall ausgeführt wird):	Es wurde ein neuer Kunde (vom Marketing oder Aussendienst) Akquiriert.		
Vorbedingungen: (optional. Erwarteter Zustand, der gegeben sein muss, damit der Anwendungsfall ausgeführt werden kann.)	<ul style="list-style-type: none">•Kunde existiert noch nicht im System (auch nicht inaktiv)•Benutzer ist angemeldet		
Nachbedingungen: (optional. Erwarteter Zustand nach erfolgreichem Ablauf des Anwendungsfalls)	Der Kundendatensatz ist mit einer eindeutigen Kundennummer identifiziert.		
Invarianten: (optional. Bedingungen, die durch den Anwendungsfall nie verändert werden dürfen)			
Eingehende Daten (optional):	Kundendaten Adresse, Telefon (und Kundenkategorie)		
Standardablauf (Aktivitäten, Schritte):			
1. Eingabeformular Anzeigen Der Mitarbeiter öffnet auf der Weboberfläche die Eingabemaske (GUI)			
2. Verbindung zur Datenbank Das System prüft die Verfügbarkeit der Datenbank, sprich die Verbindung			
3. Daten eingeben Die Adressdaten des Kunden eingeben			
4. PLZ prüfen Zusammen mit dem eingegeben Land und der Ortschaft wird die PLZ überprüft.			
5. Telefonnummer Format prüfen Das Format der eingegebenen Telefonnummer überprüfen			
6. Eindeutige Kundennummer generieren Die nächste freie Kundennummer im System ermitteln und vergeben.			
7. Datensatz erstellen Datensatz mit der neuen Kundennummer erstellen.			
8. Kundendaten speichern Die eingegeben Daten im Datensatz mit der neuen Kundennummer speichern.			
Sonderfälle/Erweiterungen zum Standardablauf:			
2a Datenbankverbindungs Fehler			
2b Timeout abwarten Erneut versuchen zu verbinden.			
4a. PLZ ist nicht gültig			
4b Fehlermeldung Aufforderung die PLZ nochmals korrekt ein zu geben Zurück zu Eingabe			
5a. Telefonnummer hat ein ungültiges Format			
5b Fehlermeldung Aufforderung die Telefonnummer korrekt ein zu geben Zurück zu Eingabe.			
Nichtfunktionale Anforder. (optional):			
Verweise auf Ressourcen (optional):			
Offene Punkte (optional):			
Priorität:	8.49	Risiko:	tief
Version (optional):		Aufwand:	1.5 Manntage
Domäne-Experte (Autor):		Sachbearbeitung / Operative Leitung	

Pascal Kern Schule Jun 9, 3:17 pm

Falscher Anwendungsfall verwendet! Vertauscht mit Zusätzlichem UC. Dieser hier (Kunde Suchen) war nicht höchste Prio sondern dritte.

2.4. Beschreibung weiterer Anwendungsfall (Priorität hoch)

Fallname:	Kunde löschen	Fall-ID:	2	
Änderungshistorie:				
21.04.2013		Erfasst von David Marmy		
Ziel:	Jegliche Kundendaten aus der Datenbank „entfernen“ durch inaktiv setzten der Kunden-ID.			
Kategorie:	Geschäfts	System	Primär	Sekundär
	(X)	X	X	
Akteur(e):	Operative Leitung und Sachbearbeitung			
Geschäftliche Auslöser: (fachliche Gründe dafür, dass der Anwendungsfall ausgeführt wird):		Die Geschäftsbeziehungen zum Kunden werden beendet Der Kunde wurde falsch (oder doppelt) erfasst		
Vorbedingungen: (optional. Erwarteter Zustand, der gegeben sein muss, damit der Anwendungsfall ausgeführt werden kann.)		•Der zu löschende Kunde muss in der Datenbank vorhanden sein. •Benutzer ist angemeldet		
Nachbedingungen: (optional. Erwarteter Zustand nach erfolgreichem Ablauf des Anwendungsfalls)		Kundendaten aus der Datenbank entfernt		
Invarianten: (optional. Bedingungen, die durch den Anwendungsfall nie verändert werden dürfen)		Kunden ID und Name inaktiv setzten (Statistik, Umsysteme (FiBu) behalten Zugriffsmöglichkeit)		
Eingehende Daten (optional):		Kunden ID und allenfalls Name.		
Standardablauf (Aktvitäten, Schritte):				
1. Kunde suchen				
2. Kunden ID inaktiv setzen				
3. Löschung bestätigen				
4. Bestätigung				
Meldung über ausgeführte Löschung/Inaktivierung				
Sonderfälle/Erweiterungen zum Standardablauf:				
1a Kunde nicht gefunden				
1b Fehlermeldung				
1c Übersicht anzeigen				
Zurück zur Eingabeaufforderung				
Nichtfunktionale Anforder. (optional):				
Verweise auf Ressourcen (optional):				
Offene Punkte (optional):		Berechtigung(en) intern		
Priorität:	7.81	Risiko:	tief	
Version (optional):		Aufwand:	2 Manntage	
Domänenexperte (Autor):		Leiter Sachbearbeitung		
...				

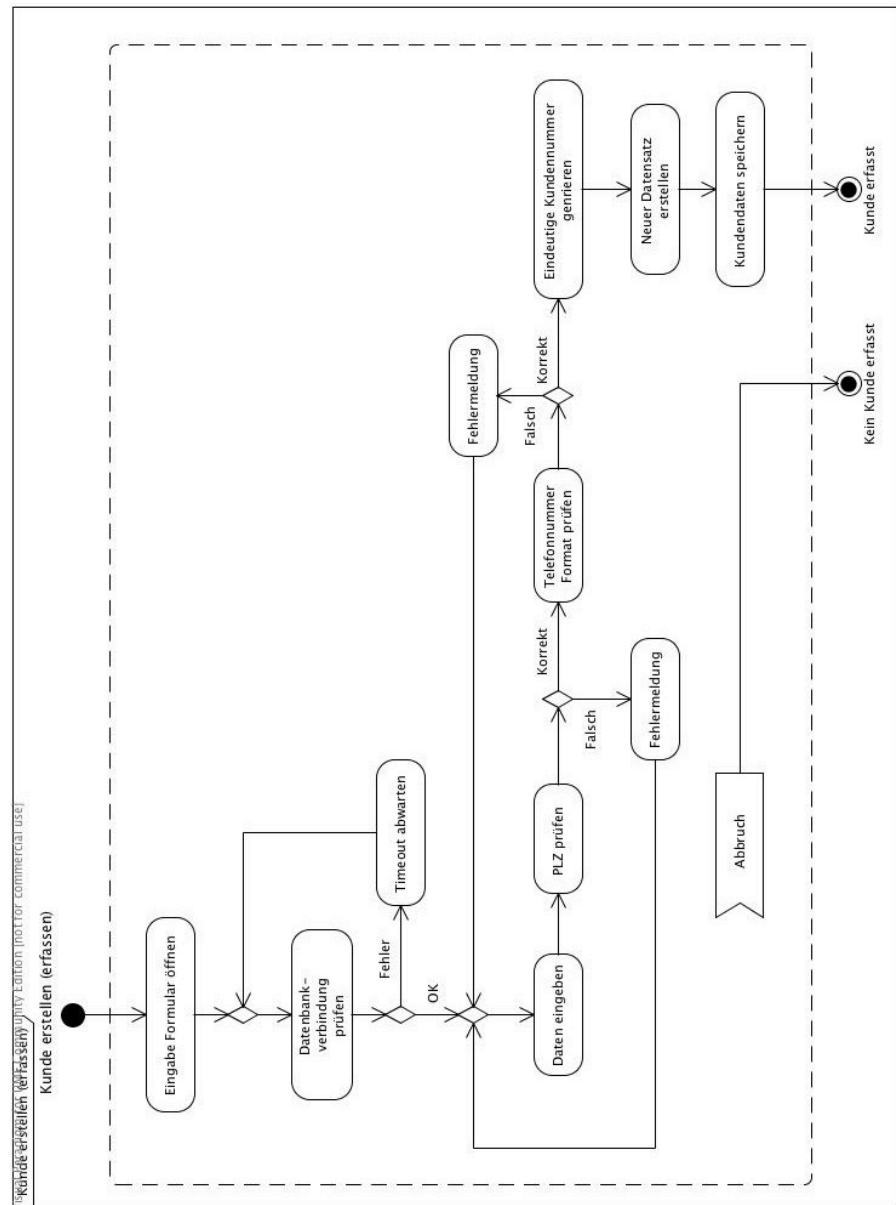
Weiterer Anwendungsfallbeschreibung (zusätzliche Übung)

Fallname:	Kunde Suchen	Fall-ID:	3
Änderungshistorie:			
21.04.2013 Erstellt durch David Marmy			

Aufgabe 2: Anforderungsanalyse (Anforderungen beschreiben)

Fallname:	Kunde Suchen	Fall-ID:	3	
Ziel:	Einen Kunden im System finden.			
Kategorie:	Geschäfts	System	Primär	Sekundär
	(X)	X	X	
Akteur(e):	Operative Führung, Sachbearbeitung und Marketing			
Geschäftliche Auslöser: (fachliche Gründe dafür, dass der Anwendungsfall ausgeführt wird):	Ein Kunde soll gelöscht oder bearbeitet werden Abruf von Kundendaten Drucken von Kundendaten Neuen Kunden erfassen			
Vorbedingungen: (optional. Erwarteter Zustand, der gegeben sein muss, damit der Anwendungsfall ausgeführt werden kann.)	<ul style="list-style-type: none"> •Der Kunde-Datensatz muss vorhanden sein. •Benutzer ist angemeldet 			
Nachbedingungen: (optional. Erwarteter Zustand nach erfolgreichem Ablauf des Anwendungsfalls)	Gesuchter Kunde gefunden / nicht gefunden			
Invarianten: (optional. Bedingungen, die durch den Anwendungsfall nie verändert werden dürfen)				
Eingehende Daten (optional):	Kunden ID und oder sonstige Attribut wie Name möglich			
Standardablauf (Aktivitäten, Schritte):				
1. Kunde suchen Möglich per Name oder ID. 2. Kunde gefunden Identifikation eindeutig über die ID und allenfalls zusätzlich Name 3. Kundendaten anzeigen Die Attributwerte zur gefunden ID anzeigen.				
Sonderfälle/Erweiterungen zum Standardablauf:				
1 Kunde suchen Möglich per Name oder ID. 2a Kunde nicht gefunden 2b Kunde muss/kann erfasst werden 3 Kudendaten anzeigen Die Attributwerte zur gefunden ID anzeigen.				
Nichtfunktionale Anforder. (optional):				
Verweise auf Ressourcen (optional):				
Offene Punkte (optional):				
Priorität:	7.21	Risiko:	tief	
Version (optional):		Aufwand:	1.5 Manntage	
Domänenexperte (Autor):	Leiter Sachbearbeitung			
...				

2.5. Aktivitätsdiagramm „Kunde erstellen“ Normal- und Sonderfall



2.6. Iterationen im Entwicklungsprozess

Anwendungsfälle

Iterationen	Anwendungsfall/-fälle	
#1 Erfassen	Kunde erstellen	Kunde suchen
#2 Bearbeiten	Kunde löschen	Kundendaten ändern
#3 Abfrage	Kundendaten abfrage	
#4 Ausgabe	Kundendaten Belegdrucken	Mailing erstellen

Ergänzend

Der „Vollständigkeitshalber“ führen wir hier noch die Nichtfunktionalen-Anforderungen (Anwendungsfälle) sowie einige zusätzliche Cases (Tasks) die unserer Meinung nach zum Entwicklungsprozess gehören.

Iterationen	Nichtfunktionale Anforderungen und weitere Cases		
#0 Schnittstellen	DB-Anbindung	Um-Systeme Anbindung	
#4.1 Erster Systemtest	<i>Gesamtes bestehendes System und dessen Funktionen</i>		
#4.2 Anbindung Umsysteme	Web-Shop	FiBu	
#5 Oberfläche (GUI)	Weboberfläche	Simultaner Zugriff	
#6 Usability	Einfacher Datenzugriff	<i>Simultaner Zugriff</i>	
#7 Sicherheit und Einstellungen	Generischer Interface	Backup	Systemüberwachung
#7.1 System Belastungstest	<i>Testen des Gesamtsystems unter Last (Simulation in Betrieb)</i>		

Erläuterung zu den Iterationen

Grundsätzliches zur Ordnung - Erläuterungen

Da nicht bei allen Anwendungsfällen der Aufwand bekannt ist, spricht die Anwendungsfälle nicht ins Detail ausgearbeitet werden, ist es schwer den Umfang der jeweiligen Iterationen vernünftig zu gestalten. Will heissen es ist „nur“ möglich die Anwendungsfälle in deren Abhängigkeit sinnvoll zu Ordnen, nicht aber den Umfang der einzelnen Iterationen in etwa gleich gross zu halten.

Da infolge der vorliegenden Aufgabenstellung nicht bei allen Anwendungsfällen der Aufwand geschätzt und das Risiko ist, könnten dies in die Ordnung der Iterationen einfließen. Es könnten also die Risikoreicheren Anwendungsfälle eher in den ersten Iterationen erledigt werden und die risikoärmeren, erst danach; wenn auch die Feedbacks der Risikoreicheren eingearbeitet werden können.

Aufgabe 2: Anforderungsanalyse (Anforderungen beschreiben)

Weiter könnten pro Iteration besser Gruppen gebildet werden also die Anzahl Anwendungsfälle mit kleinerem Aufwand pro Gruppe erhöht und dafür aufwändigere Fälle oder Risikoreichere als einzelne Iterationen geplant werden.

Zusatzinfo zum Inhalt der einzelnen Iterationen

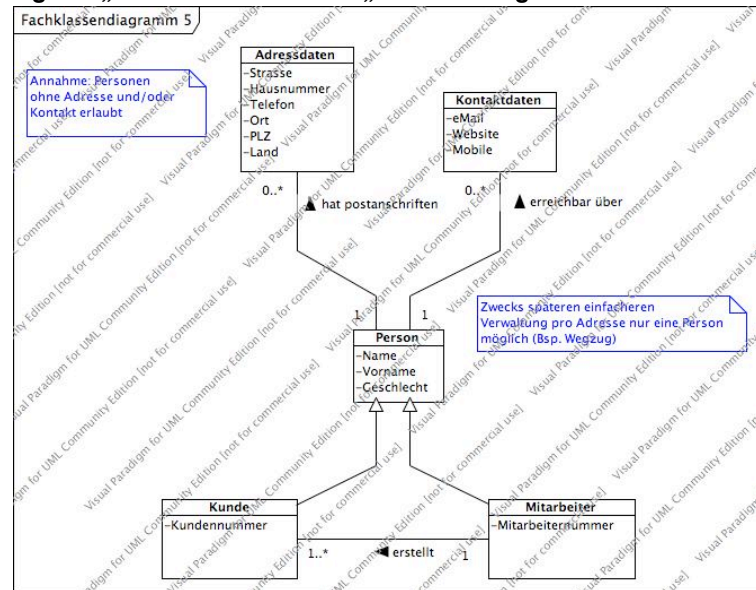
Jede Iteration setzt sich aus folgenden Schritten zusammen:

- Analyse
- Modulation
- Implementation
- Testing mit Feedback (Das Feedback der Tests haben Auswirkungen auf dieselbe und allenfalls die folgende Iteration.)

3. Aufgabe: Objektorientierte Analyse (OOA) (Analysemodell/Fachklassenmodell erstellen)

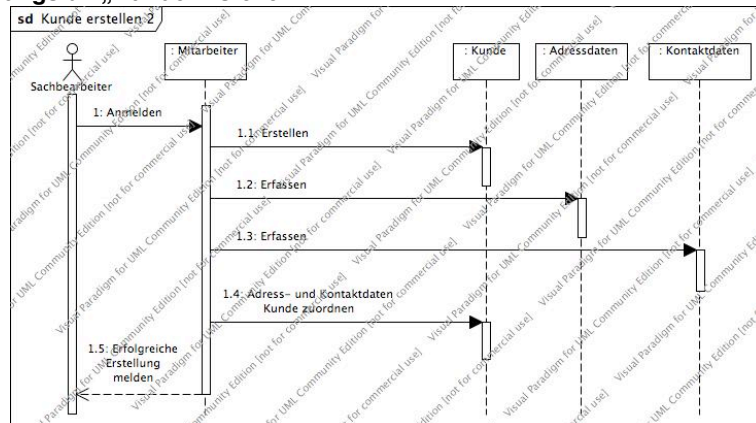
3.1. Fachklassenmodell zu Anwendungsfälle

Anwendungsfall „Kunde Erstellen“ und „Kunde anzeigen“

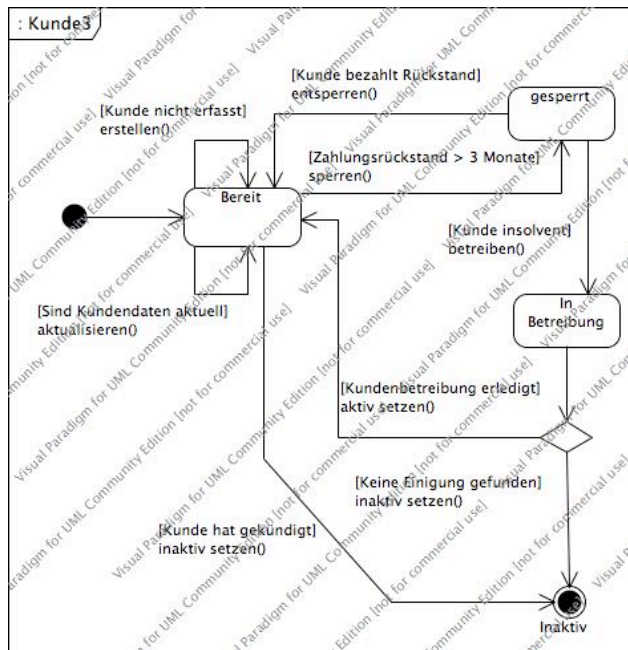


3.2. Sequenzdiagramm für Normalablauf

Anwendungsfall „Kunde Erstellen“



3.3. Zustandsdiagramm für die Abstraktion „Kunde“



3.4. Glossar zur Fachliche Verantwortlichkeit

Bezeichnung	Fachliche Verantwortung der Klasse	Synonyme (nicht bewerten)
Kunde	Identifikation eines Bedarfsträgers beinhaltet Informationen zur Geschäftsbeziehung zwischen ihm (dem Kunden) und der Unternehmung. Bezieht gegen ein entsprechendes Entgelt eine Leistung vom (unserem) Unternehmen.	Klient, Konsument, Verbraucher
Person	<u>Abstrakte Darstellung!</u> Stellt den Zugang zu zusätzlichen Daten wie Anschrift, Kontaktinformationen eines (Geschäfts-) Partner zur Verfügung. Beispiel: Gemeinsamkeiten von Kunden, Mitarbeitern aber kann auch für Lieferanten oder Geschäftspartner stehen.	Leute, Mensch
Mitarbeiter	Identifiziert jemand, der für die Unternehmung mit anderen zusammenarbeitet; Daten pflegt und Änderungen im Kundensystem vornimmt.	Personal, Belegschaft, Angestellte(r)
Adressdaten	Trägt Informationen zur Bezeichnung des Wohnorts, der Straße und der Hausnummer einer schriftlichen Kontaktmöglichkeit. Postalische Domizil-Daten einer Person oder Unternehmung.	Anschrift, Kontaktstelle
Kontakt	Daten für die direkte Kommunikation mit einer Person. Ergänzend zu den Adressdaten wie allfällige Telefon, eMail, Website, Ansprechperson usw.	Koordinaten, Kontakt-Personalien

4. Aufgabe: Objektorientiertes Design & Programmieren (OOA/D/P) (Entwurfsmodell erstellen und implementieren)

4.1. Glossar ergänzen englische Namen der Klassen

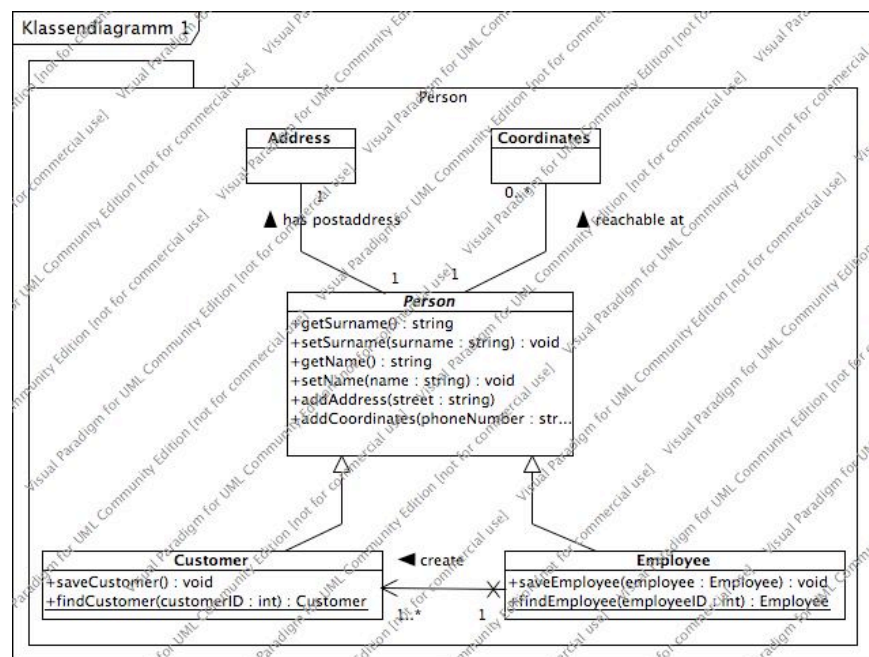
Der Vollständigkeit halber haben wir neben den Namen und Synonymen auch die Bedeutung übersetzt. Dies muss nicht unbedingt bewertet werden.

Bezeichnung	Übersetzung	Synonyme (nicht bewerten)	Übersetzung	Beschreibung Description
Kunde	Customer	Klient, Konsument, Verbraucher	Client, Consumer, --	Identifikation eines Bedarfsträgers beinhaltet Informationen zur Geschäftsbeziehung zwischen ihm (dem Kunden) und der Unternehmung. Bezieht gegen ein entsprechendes Entgelt eine Leistung vom (unserem) Unternehmen. Identification of a carrier includes information required for business relationship between him (the client) and the company. Refers to rates of pay, a performance from (our) company.
Person	Person	Leute, Mensch	People, Human	Abstrakte Darstellung! Stellt den Zugang zu zusätzlichen Daten wie Anschrift, Kontaktinformationen eines (Geschäfts-) Partner zur Verfügung. Beispiel: Gemeinsamkeiten von Kunden, Mitarbeitern aber kann auch für Lieferanten oder Geschäftspartner stehen. Abstract representation! Provides access to additional information such as address, contact information of a (business) partner. For example, similarities of customers, employees but can also stand for suppliers or business partners.
Gender	Geschlecht			Angabe zum Kunden ob er männlich oder weiblich ist. Informations about the client if it's a male or female.
Mitarbeiter	Employee	Personal, Belegschaft, Angestellte(r)	Personnel, Workforce, Employee(s)	Identifiziert jemand, der für die Unternehmung mit anderen zusammenarbeitet; Daten pflegt und Änderungen im Kundensystem vornimmt. Identifies someone who works for the company with others, maintains and modifies data in the customer system.
Adressdaten	Address	Anschrift, Kontaktstelle	--, Point of contact	Trägt Informationen zur Bezeichnung des Wohnorts, der Straße und der Hausnummer einer schriftlichen Kontaktmöglichkeit. Postalische Domizil-Daten einer Person oder Unternehmung. Carries information about the name of the place, the street and the house number written contact information. Postal domicile of a person or company data.
Kontakt	Coordinates	Koordinaten, Kontakt-Personalien	Coordinates, Contact-Personnel	Daten für die direkte Kommunikation mit einer Person. Ergänzend zu den Adressdaten wie allfällige Telefon, eMail, Website, Ansprechperson usw. Data for direct communication with a person. In addition to the address as any telephone, email, website, contact person, etc.

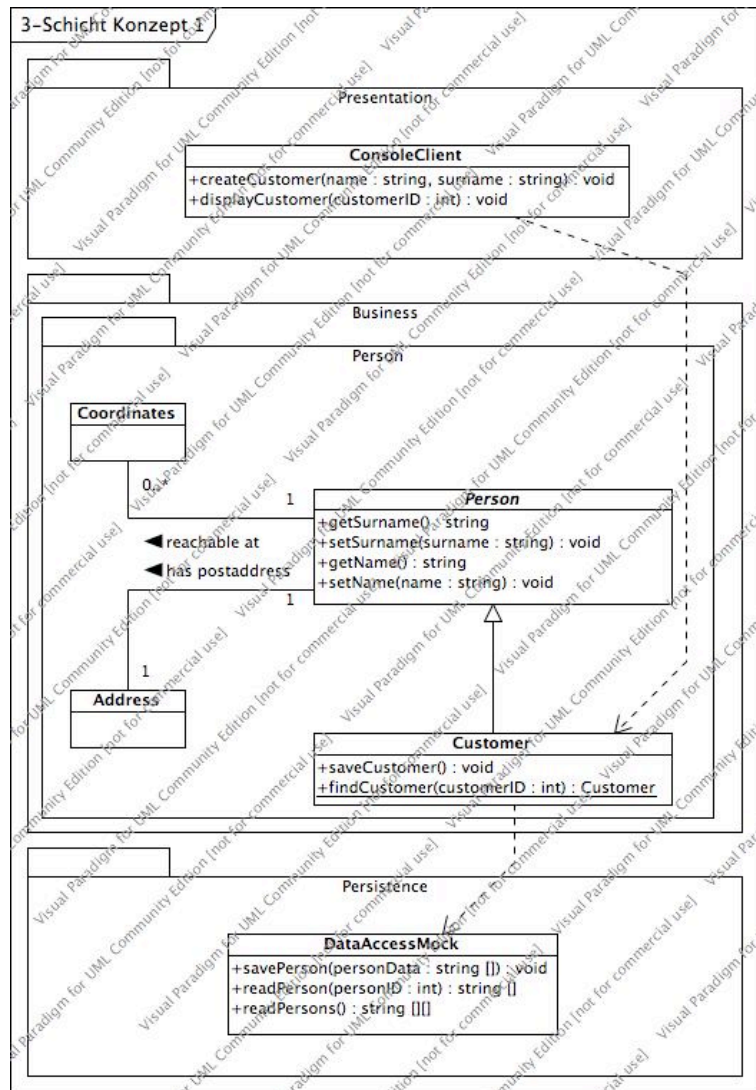
Bemerkung zu den folgenden Diagrammen

Gemäss Aufgabenstellung wurden nur die Eigenschaften Vor- und Nachname aufgeführt. Die im Analysemodell vorhandenen Eigenschaften Geschlecht (gender), Kundennummer (clientID) und Mitarbeiternummer (staffID) wurden also nicht "vergessen" sondern bewusst weg gelassen. Bei den Methoden wurden die für die Klassen Adresse (Address) und Kontakt (Coordinates) aber dennoch aufgenommen um die Logik einigermaßen komplett dar zu stellen.

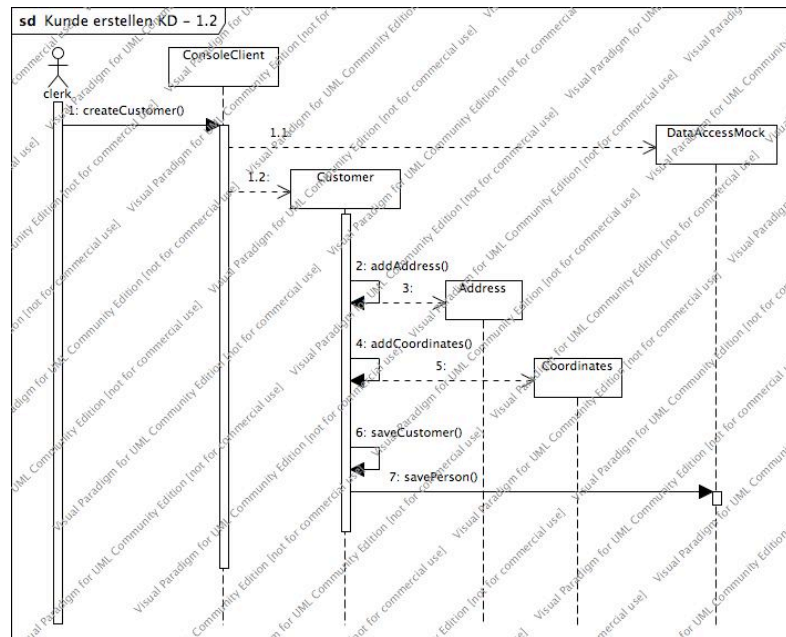
4.2. Fachklassendiagramm 1. Iteration in Entwurfsmodell Fachklassenmodell überführen



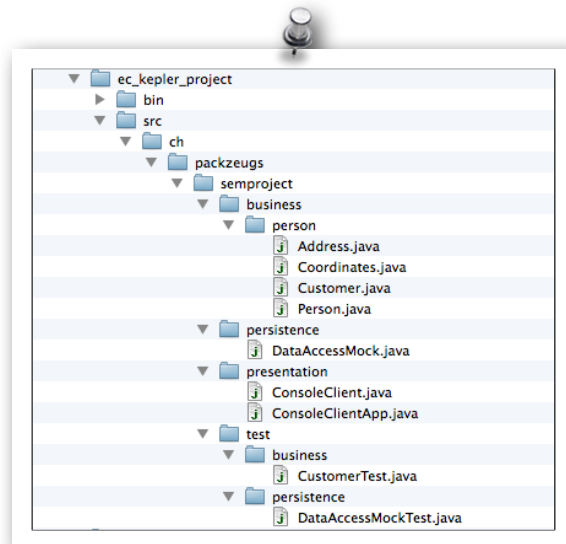
4.3. Fachklassenmodell in 3-Schichtenarchitektur integrieren



4.4. Entwurfsmodell mit Sequenzdiagramm überprüfen



4.5. Entwurfsmodell in Prototyp implementieren

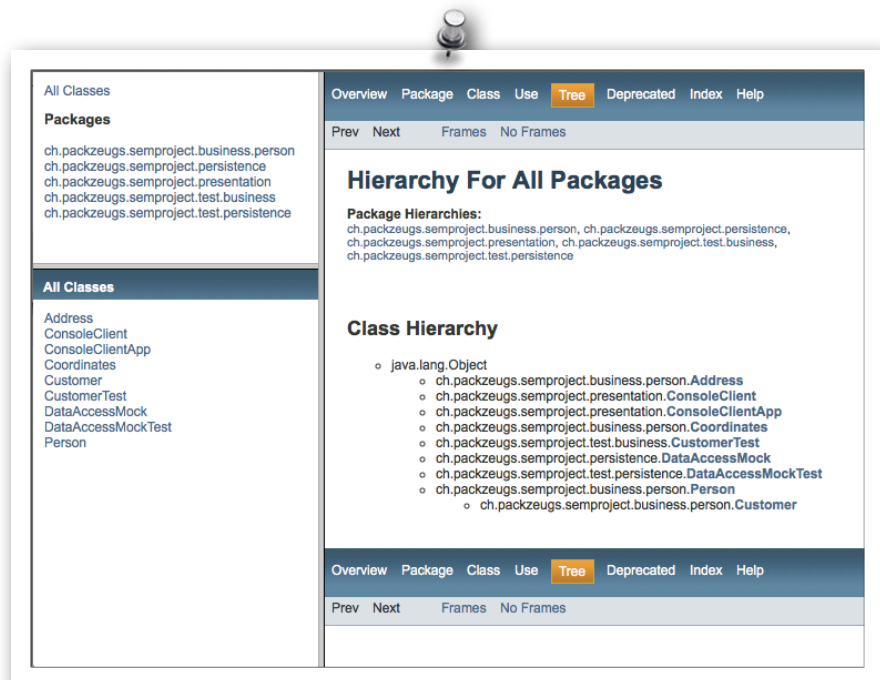


4.6. Prototyp testen

Einzelne Pakete (Schichten vom 3-Schicht Konzept) einzeln erfolgreich mit den Tests im Paket "ch.packzeugs.semproject.test" getestet.

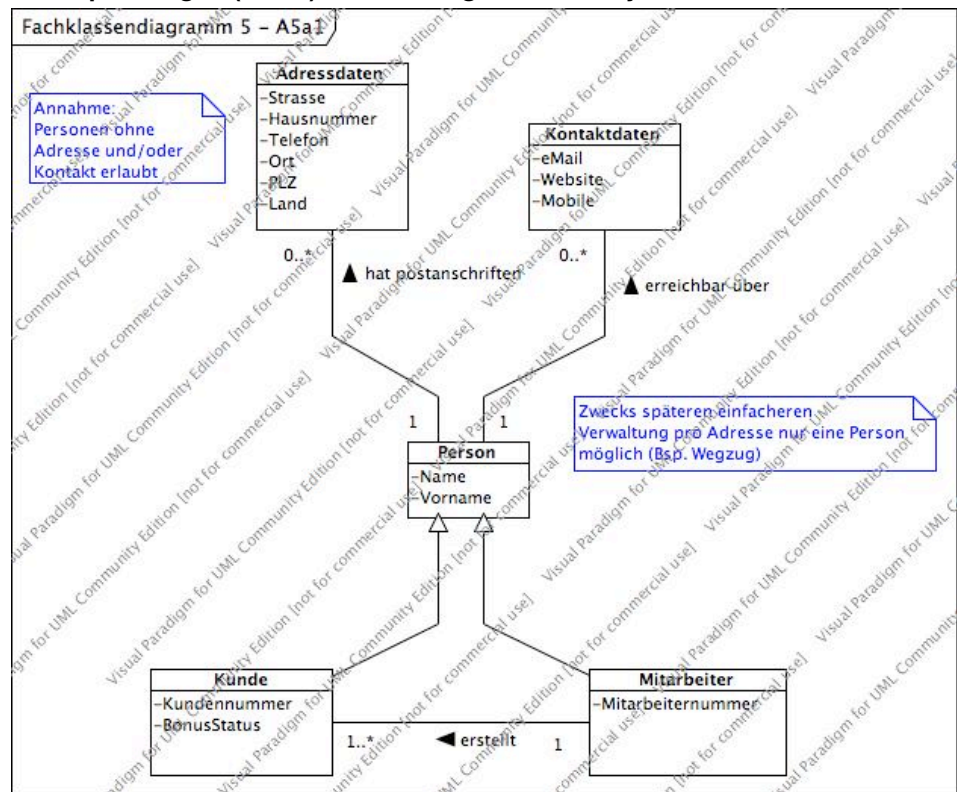
Das Zusammenspiel der drei Schichten wurde mit der Klasse "ConsoleClientApp" erfolgreich durchgeführt und kann somit auch als erfolgreich getestet betrachtet werden.

4.7. Quelltext per Javadoc als HTML-Dokument erstellen

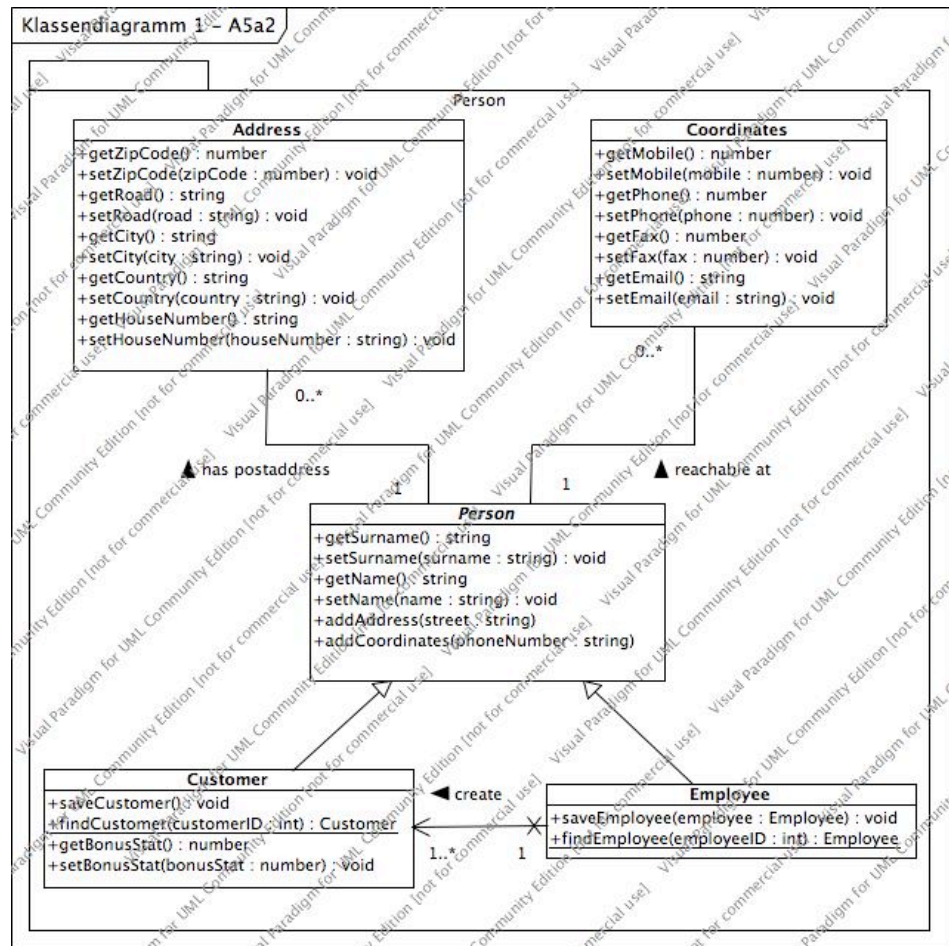


5. Aufgabe: Objektorientierte Analyse, Design & Programmieren (OOA/D/P) (Analyse- und Entwurfsmodell anpassen, Entwurfsmodell implementieren)

5.1. Anpassungen (Fach-) Klassendiagramm - Analysemodell



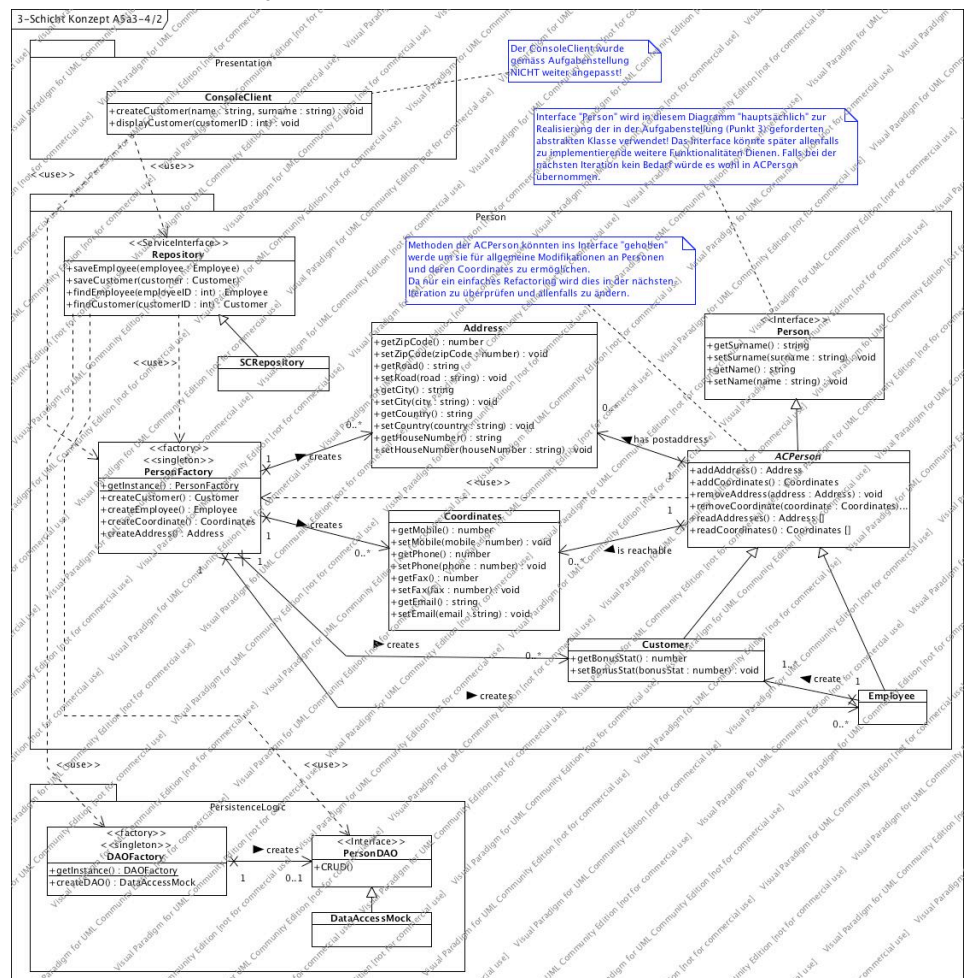
5.2. Angepasstes Klassendiagramm - (Entwurf-) Designmodell



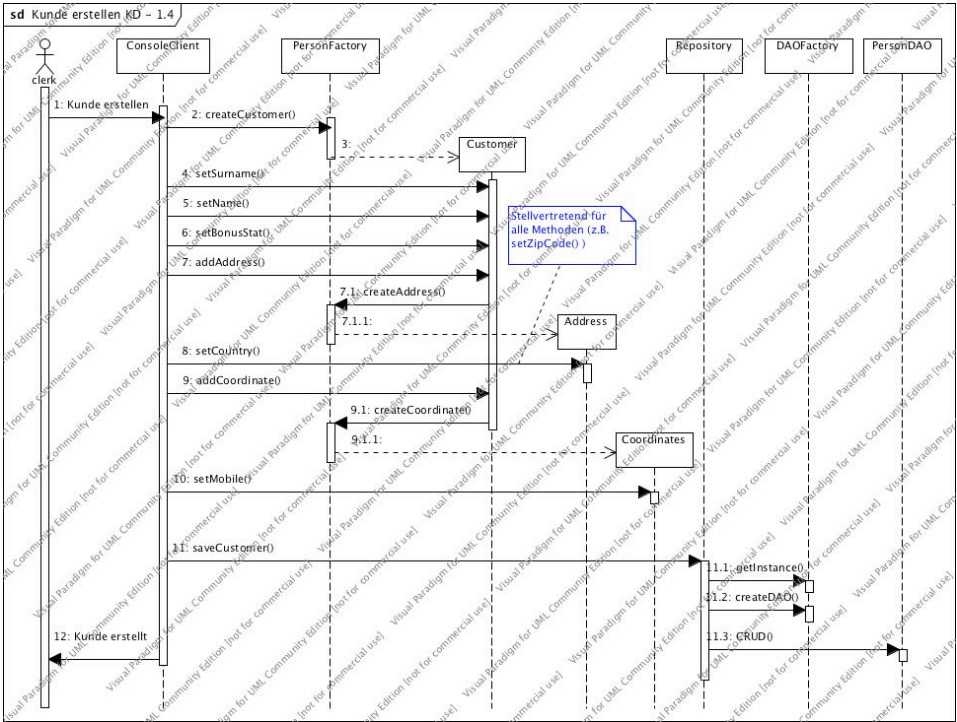
5.3. Erweiterung Design- Entwurfsmodell in “Business” und “Persistence” Schicht um abstrakten Klassen und Schnittstellen

Siehe nächste Aufgabe (5.4). Zusammengefasst zu einer Aufgabe da für uns die Unterscheidung der Aufgabeninhalte nicht nachvollziehbar ist!

5.4. Klassen- und Komponentendiagramm mit “Singleton” und “Factory” (GoF) Design Pattern auf Business und Persistence Schicht (3-Schichten Modell)



5.5. Sequenzdiagramm zur Überprüfung Normalfall “Kunde erstellen” - Korrekturen ableiten



5.6. Glossar ergänzen um Klassen und Schnittstellen aus Design-Entwurfsmodell [5.3]

Layer Presentation

Class/Interface (Technical term)	Synonyme	Beschreibung Description
ConsoleClient		Stellt einfachste Funktionalität für die Erstellung und Anzeige von Kunden zur Verfügung. Provides simple functionality for creating and displaying customer.

Layer BusinessLogic

Class/Interface (Technical term)	Synonyme	Beschreibung Description
Customer	Client, Consumer, --	Identifikation eines Bedarfsträgers beinhaltet Informationen zur Geschäftsbeziehung zwischen ihm (dem Kunden) und der Unternehmung. Bezieht gegen ein entsprechendes Entgelt eine Leistung vom (unserem) Unternehmen. Hält Informationen zu Adresse(n), Koordinate(n) und Bonusstufe. Identification of a carrier includes information required for business relationship between him (the client) and the company. Refers to rates of pay, a performance from (our) company. Holds information about address (es) coordinate (s) and bonus level.

Aufgabe 5: Objektorientierte Analyse, Design & Programmieren (OOA/D/P) (Analyse- und Entwurfsmodell anpassen, Entwurfsmodell implementieren)

Class/Interface (Technical term)	Synonyme	Beschreibung Description
Person <<interface>>	People, Human	Stellt den Zugang zu Vor- und Nachname eines (Geschäfts-) Partner zur Verfügung. Beispiel: Gemeinsamkeiten von Kunden-, Mitarbeitern aber kann auch für Lieferanten oder Geschäftspartner stehen. Provides access to first and last name of a (business partner). For example, similarities of customers, employees but can also stand for suppliers or business partners.
ACPerson		Funktionalität zum einfügen, halten und entfernen von Adressen- und Koordinatendaten einer Person. To add functionality, hold and remove addresses and coordinate data of a person.
Employee	Personnel, Workforce, Employee(s)	Identifiziert jemand, der für die Unternehmung mit anderen zusammenarbeitet; Daten pflegt und Änderungen im Kundensystem vornimmt. Identifies someone who works for the company with others, maintains and modifies data in the customer system.
Address	--, Point of contact	Trägt Informationen zur Bezeichnung des Wohnorts, der Straße und der Hausnummer einer schriftlichen Kontaktmöglichkeit. Postalische Domizil-Daten einer Person oder Unternehmung. Carries information about the name of the place, the street and the house number written contact information. Postal domicile of a person or company data.
Coordinates	Coordinates, Contact-Personnel	Daten für die direkte Kommunikation mit einer Person. Ergänzend zu den Adressdaten wie allfällige Telefon, eMail, Website, Ansprechperson usw. Data for direct communication with a person. In addition to the address as any telephone, email, website, contact person, etc.
Repository <<interface>>		Zugriff auf Funktionalität der Klasse Repository hauptsächlich für Testzwecke der Repository Funktionalität. (einfacher Einsatz eines von Test-MOKs) Access to functionality of the class repository mainly for test purposes, the repository functionality. (simple use of a test-Mok)
SCRepository	Toolbox, Toolservice	Funktionen zur Verwaltung und Benutzung von Objekten der Klassen Kunde und Angestellter. Delegiert den Zugriff auf die Datenbank an die Schnittstelle PersonDAO und bildete Objekte des Layer Business über die PersonFactory. Functions for the management and use of objects of class customer and employee. Delegates to access the database at the interface personDao and formed objects of Layer Business about the person Factory.
PersonFactory		Kontrolliert die Erstellung von Objekten im Package Person. Controls the creation of objects in the package person.

Layer Persistence

Class/Interface (Technical term)	Synonyme	Beschreibung Description
DAOFactory		Kontrolliert den Lebenszyklus von Objekten mit der PersonDAO Schnittstelle. Controls the life cycle of objects with the personDAO interface.
PersonDAO <<interface>>		Database-Access-Object (Baustein) Ermöglicht das Schreiben, Lesen, Ändern und Löschen von Angestellten- oder Kundendaten. CRUD Funktionalität (Create-Read-Update-Delete) Database Access Object (block) Allows you to write, read, modify, and delete employee or customer data. CRUD functionality (Create-Read-Update-Delete)
DataAccessMock		Verarbeitet die Angestellten- und Kundendaten In-Memory. Processes the employee and customer data in-memory.

Aufgabe 5: Objektorientierte Analyse, Design & Programmieren (OOA/D/P) (Analyse- und Entwurfsmodell anpassen, Entwurfsmodell implementieren)

5.7. ~~Angepasstes Design-Entwurfsmodell implementieren [5.3]~~

5.8. ~~JUnit Test auf "Business" und "Persistenz" Schicht für Normalfall "Kunde erstellen"~~

5.9. ~~Normalfall "Kunde erstellen" und "Kunde anzeigen" testen~~

5.10. ~~JavaDoc generieren~~

6. Anhang

6.1. Aufgabenstellungen

Aufgabenstellung 1:

Ausgabetermin: 14.03.2013

Abgabetermin: 26.03.2013

Ausgangslage:

Die Firma PackZeug AG mit Stammsitz in Basel stellt in Zusammenarbeit mit einer Auswahl an Partnern massgeschneiderte Rucksäcke verschiedener Kategorien her und vertreibt diese an Privatpersonen und ist mit ca. 70 Mitarbeitern an mehreren Standorten in der Schweiz, Österreich und Deutschland vertreten. PackZeug pflegt eine enge Kundenbindung. Deshalb sind Kundeninformationen hier von zentraler Bedeutung. Ganz unterschiedliche Mitarbeiter (vom CEO bis hin zum Marketing) benötigen den Zugriff auf sämtliche Kundeninformationen. Bisher wurden diese Informationen mittels Tabellenkalkulation an den Standorten dezentral verwaltet/gepflegt und nur unregelmässig via E-Mail an den Stammsitz übermittelt und dort in einer einfachen zentralen Datenbank eingepflegt. Die seit einigen Jahren sehr gute Geschäftsentwicklung hat dazu geführt, dass der Kundenstamm stark angewachsen ist und aktuell einige 1000 Kunden umfasst. Der bisherige Ansatz zur computergestützten Kundendatenverwaltung stösst nun zunehmend an seine Grenzen: Die Datenqualität verschlechtert sich zunehmend (veraltete, redundante und unvollständige Kundeninformationen). Abhilfe soll hier nun ein neu zu entwickelndes Kundenverwaltungssystem schaffen. Hierzu wird die externe IT-Beratungsfirma SoftThings AG in St. Gallen beauftragt, welche den Projektleiter und die benötigten IT-Fachleute bereitstellt und nach Fertigstellung auch den Betrieb organisieren sowie die Wartung und Weiterentwicklung übernehmen soll.

Die Anforderungen an das Kundenverwaltungssystem gemäss Lastenheft sind:

- Verwaltung folgender Kundeninformationen: Name, Vorname und Kontaktinformationen (Anschrift, Telefon, Natel, Fax und E-Mail)
- Sicherstellung der Datenqualität
- Kunden haben eine Anschrift, ein Telefon, eine E-Mail etc.
- Die Bedienung erfolgt über eine Web-Oberfläche
- Anbindung an zentrale Benutzerverwaltung, Finanzbuchhaltung, Auftragsverwaltung und Webshop
- Zunächst Weiterverwendung des bestehenden relativ einfachen Datenbanksystems. Für die Zukunft ist geplant auf ein leistungsfähigeres Datenbanksystem umzustellen
- Als Technologie kommt Java zum Einsatz

Auftrag:

1. Formulieren Sie eine prägnante Vision für das Kundenverwaltungssystem in maximal drei Sätzen.
2. Identifizieren Sie sechs mögliche Interessenvertreter (Stakeholder) und begründen Sie diese.
3. Identifizieren Sie jeweils ein spezifisches (eindeutiges) Ziel für jeden Interessenvertreter in Bezug auf das zu entwickelnde IT-System.
4. Ergänzen Sie die in der Ausgangslage angegebenen Anforderungen um weitere für das geplante IT-System sinnvolle zwei funktionale und zwei nicht-funktionale Anforderungen und begründen Sie diese.
5. Identifizieren Sie vier mögliche Risiken, die sich aus der Ausgangslage ergeben und begründen Sie diese.
6. Erstellen Sie für das zu entwickelnde System ein Systemkontextdiagramm, das sich aus der Ausgangslage ergibt (kein UML-Diagramm, sondern eine Box-and-Lines-Grafik).

Bewertungskriterien Aufgabe 1

Bewertung (mittels Kontrolle und/oder Gespräch, max. 18 Punkte):				
Bewertungskriterien (Punkte: 0 = nicht erfüllt; 1 = weniger als 50% erfüllt; 2 = mehr als 50% erfüllt; 3 = 100% erfüllt)	Punkte			
	0	1	2	3
Vision ist korrekt und vollständig.				
Interessenvertreter werden begründet und sind korrekt sowie vollständig.				
Ziele der Interessenvertreter (Akteure) sind spezifisch (eindeutig), nachvollziehbar und vollständig.				
Die zusätzlichen funktionalen und nicht-funktionalen Anforderungen werden begründet und sind korrekt, sinnvoll sowie vollständig.				
Risiken werden begründet und sind spezifisch, eindeutig sowie vollständig.				
Das Systemkontextdiagramm ist korrekt und vollständig.				

Bewertung Aufgabe 1 eMail vom 14.03.2013:

Hallo zusammen,

bei der Aufgabe 1 der Semesterarbeit habt Ihr 15 Punkte erreicht.

Die Punkte sind wie folgt von links nach rechts auf die angegebenen Bewertungskriterien von links nach rechts (bzw. im Aufgabenblatt von oben nach unten) verteilt:

2 3 3 2 3 2

Vision - Interessenvertreter - Interessenvertreterziele - Anforderungen Risiken - Systemkontext

Beste Grüsse
Arif

15 Punkt von max 18! = 5.18

Aufgabenstellung 2:**Ausgabetermin:** 11.04.2013**Abgabetermin:** 09.05.2013**Ausgangslage:**

Die Vision und die grundsätzlichen Rahmenbedingungen (Ziele, Anforderungen und Risiken aus Geschäftssicht etc.) des Kundenverwaltungssystems für die PackZeug AG kennen Sie nun. Zudem wurde entschieden, dass Kundenverwaltungssystem mit einem iterativen Entwicklungsprozess zu entwickeln. Bevor Sie mit der objekt-orientierten Analyse beginnen können, müssen Sie zunächst noch die Anforderungen systematisch und genau dokumentieren.

Auftrag:

1. Erstellen Sie für das zu entwickelnde Kundenverwaltungssystem ein UML Anwendungsfalldiagramm, das den Anwendungsfall "Kunde erstellen" und mindestens fünf weitere Anwendungsfälle, die sich aus den Anforderungen ergeben, sowie die entsprechenden Akteure umfasst.
2. Priorisieren Sie die Anwendungsfälle aus Auftrag 1 der Aufgabe 2 und die nicht-funktionalen Anforderungen aus Ausgangslage und Auftrag 4 von Aufgabe 1 (Vision und Rahmenbedingungen) nach aufsteigender Priorität (das Wichtigste zuerst) in zwei getrennten Tabellen, die zusätzlich noch je Anwendungsfall die Werte für Priorität und die entsprechenden Kriterien enthalten. Die Prioritäten sind als Zahlen auf Basis von zwei selbst gewählten und einheitlichen Kriterien (keine Anforderungen!) zu berechnen. Kriterien sind kurz zu erläutern und die Berechnungsformel für die Prioritäten ist aufzuführen. Gleiche Prioritäten können mehrfach vergeben werden.
3. Beschreiben Sie den Anwendungsfall "Kunde erstellen". Folgende Elemente sind dabei zu beschreiben: ID, Name, Kurzbeschreibung (Ziel), Akteure, geschäftlicher Auslöser, Vor- und Nachbedingungen, Normalablauf sowie Sonderfälle.
4. Beschreiben Sie einen weiteren Anwendungsfall mit der höchsten Priorität.
5. Erstellen Sie für den Anwendungsfall „Kunde erstellen“ ein UML Aktivitätsdiagramm, das Normalablauf und Sonderfälle umfasst.
6. Erstellen Sie als Vorbereitung für die Projektplanung im Rahmen des iterativen Entwicklungsprozesses eine Tabelle, in welcher die Anwendungsfälle aus Auftrag 1 der Aufgabe 2 mindestens vier Iterationen sinnvoll zugeordnet werden. Die einzelnen Aktivitäten des Entwicklungsprozesses in den Iterationen sind dabei nicht aufzuführen!

Bewertungskriterien Aufgabe 2:

Bewertung (mittels Kontrolle und/oder Gespräch, max. 24 Punkte):					
#	Bewertungskriterien (Punkte: 0 = nicht erfüllt; 1 = weniger als 50% erfüllt; 2 = mehr als 50% erfüllt; 3 = 100% erfüllt)	Punkte			
		0	1	2	3
1	Das UML Anwendungsfalldiagramm ist korrekt und vollständig.				
2	Die Priorisierung der Anwendungsfälle ist vollständig und erfolgt aufgrund nachvollziehbarer, einheitlicher und fachlich korrekter Kriterien.				
3	Die Priorisierung der nicht-funktionalen Anforderungen ist vollständig und erfolgt aufgrund nachvollziehbarer, einheitlicher und fachlich korrekter Kriterien.				
4	Die Beschreibung des Anwendungsfalls "Kunde erstellen" ist korrekt, vollständig und erfolgt auf Basis eines einheitlichen Vorgehens.				
5	Die Beschreibung eines weiteren Anwendungsfalls mit der höchsten Priorität, ist korrekt, vollständig und erfolgt auf Basis eines einheitlichen Vorgehens.				
6	Das UML Aktivitätsdiagramm zum Anwendungsfalls „Kunde erstellen“ ist korrekt und vollständig.				
7	Die Zuordnung der Anwendungsfälle an Iterationen ist korrekt und vollständig.				
8	Namen bzw. Bezeichnungen sind aussagekräftig und die bekannten UML Namenskonventionen werden beachtet.				

Bewertung Aufgabe 2 eMail vom 05.06.2013:

Hallo zusammen,

bei der 2. Aufgabe der Semesterarbeit habt Ihr 19 Punkte erreicht.

Die Punkte sind wie folgt von links nach rechts auf die in der Aufgabe angegebenen Bewertungskriterien von oben nach unten zu verteilen (1. von links auf 1. von oben, 2. links auf 2. von oben usw.):

2 3 2 2 2 3 3 2

Da ich Euch aus zeitlichen Gründen keine komplette bzw. keine mündliche Rückmeldung im Unterricht geben konnte, findet Ihr meine Rückmeldung zu den einzelnen Beurteilungskriterien unten.

Beste Grüsse
Arif

+++ Rückmeldung +++

1. Das UML Anwendungsfalldiagramm ist korrekt und vollständig:

aufgabe_1+2+3+4+5_team_3

24.03.2013

Seite 29 von 41

include macht keinen Sinn
Falsche Vererbungen zwischen Anwendungsfällen
Akteure können z.T. Anwendungsfälle nicht benutzen

2. Die Priorisierung der Anwendungsfälle ist vollständig und erfolgt aufgrund nachvollziehbarer, einheitlicher und fachlich korrekter Kriterien:

3. Die Priorisierung der nicht-funktionalen Anforderungen ist vollständig und erfolgt aufgrund nachvollziehbarer, einheitlicher und fachlich korrekter Kriterien:

Nicht-funktionale Anforderungen fehlen

4. Die Beschreibung des Anwendungsfalls "Kunde erstellen" ist korrekt, vollständig und erfolgt auf Basis eines einheitlichen Vorgehens:

Beschreibung enthält nicht geforderte Elemente
Ablauf: Beschreibungen sind z.T. zu detailliert
Schritt oder Vorbed. "Benutzer anmelden" fehlt
Sonderfälle fehlen
Ablauf: Beschreibungen sind z.T. redundant

5. Die Beschreibung eines weiteren Anwendungsfalls mit der höchsten Priorität, ist korrekt, vollständig und erfolgt auf Basis eines einheitlichen Vorgehens:

Falsche Sonderfälle
Beschriebener Anwendungsfall hat nicht die höchste Prio
Schritte im Normalablauf fehlen
Schritt oder Vorbed. "Benutzer anmelden" fehlt

6. Das UML Aktivitätsdiagramm zum Anwendungsfalls „Kunde erstellen“ ist korrekt und vollständig:

7. Die Zuordnung der Anwendungsfälle an Iterationen ist korrekt und vollständig:

8. Namen bzw. Bezeichnungen sind aussagekräftig und die bekannten UML Namenskonventionen werden beachtet:

Namen z.T. nicht einheitlich

19 Punkt von max 24! = 4.95

Aufgabenstellung 3:**Aufgabe 3: Objektorientierte Analyse (OOA)
(Analysemodell/Fachklassenmodell erstellen)****Ausgabetermin:** 17.09.2013**Abgabetermin:** 30.09.2013**Ausgangslage:**

Die Anforderungen an das Kundenverwaltungssystem liegen systematisch dokumentiert vor. Aus den Anforderungen sollen Sie nun ein objektorientiertes Analysemodell herleiten, dass in UML und textuell zu beschreiben ist. Dabei ist die Kundensprache Deutsch zu verwenden.

Auftrag:

1. Erstellen Sie ein Fachklassenmodell, dass die Realisierung der Anwendungsfälle "Kunde erstellen" und "Kunde anzeigen" ermöglicht. Das Fachklassenmodell muss aus mindestens zwei Klassen bestehen. Attribute sowie mögliche Assoziations- und Vererbungsbeziehungen sind anzugeben, Methoden nicht.
2. Überprüfen Sie Ihr Fachklassenmodell mittels eines UML Sequenzdiagramms für den Normalablauf des Anwendungsfalls „Kunde erstellen“ hinsichtlich Vollständigkeit, Verantwortlichkeiten der Klassen und Interaktionen der entsprechenden Objekte. Ergänzen Sie bei Bedarf das Fachklassenmodell nach dieser Überprüfung.
3. Überprüfen Sie Ihr Fachklassenmodell mittels eines UML Zustandsdiagramms für die Abstraktion "Kunde". Ergänzen Sie bei Bedarf das Fachklassenmodell nach dieser Überprüfung.
4. Beschreiben Sie die fachliche Verantwortlichkeit jeder Klasse in einem Glossar
Hinweise:
a) Objektorientierte Konzepte und/oder UML Diagramme sind nicht zu erläutern
b) Informationen aus dem UML Klassendiagramm sind nicht zu wiederholen

Bewertungskriterien Aufgabe 3**Bewertung (mittels Kontrolle und/oder Gespräch, max. 18 Punkte):**

#	Bewertungskriterien (Punkte: 0 => nicht erfüllt; 1 => <50% erfüllt; 2 => >50% erfüllt; 3 => 100% erfüllt)	Punkte			
		0	1	2	3
1	Die Klassen im UML Klassendiagramm sind fachlich korrekt und vollständig				
2	Assoziations- und/oder Vererbungsbeziehungen im UML Klassendiagramm sind fachlich korrekt und vollständig				
3	Das UML Sequenzdiagramm ist fachlich korrekt und vollständig				
4	Das UML Zustandsdiagramm ist fachlich korrekt und vollständig				
5	Die Verantwortlichkeiten der Klassen sind fachlich korrekt und vollständig im Glossar beschrieben				
6	Namen bzw. Bezeichnungen sind aussagekräftig und die bekannten UML Namenskonventionen werden beachtet				

Bewertung Aufgabe 3 (Teil 1,2 und 4) eMail vom 07.10.2013

Hallo zusammen,

bei der 3. Aufgabe (ohne Auftrag 3) des Semesterprojekts habt Ihr 12 Punkte erreicht.

aufgabe_1+2+3+4+5_team_3

24.03.2013

Seite 31 von 41

Die Punkte sind wie folgt von links nach rechts auf die in der Aufgabe angegebenen Bewertungskriterien von oben nach unten zu verteilen (1. von links auf 1. von oben, 2. links auf 2. von oben usw.):

2 3 2 0 3 2

Beste Grüsse
Arif

Zwischenergebnis:
12 Punkt von max 15! = 5.0

Bewertung Aufgabe 3 (Teil 3) eMail vom 30.10.13:

Hallo zusammen,

bei der 3. Aufgabe (jetzt mit Auftrag 3) des Semesterprojekts habt Ihr 14 Punkte erreicht.

Die Punkte sind wie folgt von links nach rechts auf die in der Aufgabe angegebenen Bewertungskriterien von oben nach unten zu verteilen (1. von links auf 1. von oben, 2. links auf 2. von oben usw.):

2 3 2 2 3 2

Beste Grüsse
Arif

Endergebnis:
14 Punkt von max 18 = 4.88

6.2.Aufgabenstellung 4:**Aufgabe 4:
Objektorientiertes Design & Programmieren (OOA/D/P)
(Entwurfsmodell erstellen und implementieren)****Ausgabetermin:** 05.10.2013**Abgabetermin:** 28.10.2013**Ausgangslage:**

Auf Basis des vorliegenden Analysemodells müssen Sie unter besonderer Beachtung der nicht-funktionalen Anforderungen für die 1. Iteration ein Entwurfsmodell erstellen und dieses als Prototyp in der Programmiersprache Java implementieren. Die 1. Iteration umfasst die Klasse (bzw. Klassen) Ihres Analysemodells, die die Abstraktion "Kunde" modelliert. Für die 1. Iteration realisieren Sie die Normalabläufe der Anwendungsfälle "Kunde erstellen" und "Kunde anzeigen". Es genügt, wenn Sie dabei die Eigenschaften Vor- und Nachname von Kunden berücksichtigen.

Auftrag:

1. Ergänzen Sie das Glossar um die ins Englische übersetzten Namen der Klassen des Fachklassenmodells.
2. Überführen Sie für die 1. Iteration das Fachklassenmodell des Analysemodells in ein Fachklassenmodell für das Entwurfsmodell. Methoden sowie mögliche Assoziations- und Vererbungsbeziehungen sind anzugeben, Attribute nicht. Stellen Sie das angepasste Entwurfsmodell mit einem UML Klassendiagramm dar.
3. Integrieren Sie das Fachklassenmodell in die gegebene logische 3-Schichtenarchitektur (siehe Seite 3).
4. Überprüfen Sie mit einem UML Sequenzdiagramm Ihr Entwurfsmodell für den Normalablauf des Anwendungsfalls „Kunde erstellen“ hinsichtlich Vollständigkeit und Verantwortlichkeiten der Klassen sowie Interaktionen der entsprechenden Objekte. Ergänzen Sie bei Bedarf das Entwurfsmodell nach dieser Überprüfung.
5. Implementieren Sie für die 1. Iteration den Teil Ihres Entwurfsmodells, der es ermöglicht einen Kunden mit seinem Vor- und Nachnamen zu speichern. Implementieren Sie dazu einen Prototypen in Java.
6. Testen Sie Ihren Prototypen für die Normalabläufe der Anwendungsfälle "Kunde erstellen" und "Kunde anzeigen", indem Sie mehrere Kunden erstellen, speichern, lesen und auf der Konsole deren Vor- und Nachnamen ausgeben.
7. Dokumentieren Sie den Java-Quelltext (nur Dateiköpfe und Methoden) mittels Javadoc und erzeugen Sie die zugehörigen HTML-Dokumentationsseiten.

Bewertungskriterien Aufgabe 4**Bewertung (mittels Kontrolle und/oder Gespräch, max. 36 Punkte):**

#	Bewertungskriterien (Punkte: 0 => nicht erfüllt; 1 => <50% erfüllt; 2 => >=50% erfüllt; 3 => 100% erfüllt)	Punkte			
		0	1	2	3
1	Das UML Klassendiagramm zum Fachklassenmodell ist fachlich korrekt und vollständig.				
2	Die Integrations des Fachklassenmodells in die gegebene logische 3-Schichtenarchitektur ist fachlich korrekt und vollständig.				
3	Das UML Sequenzdiagramm ist fachlich korrekt und vollständig.				
4	Die ins Englische übersetzten Namen der Klassen des Fachklassenmodells sind fachlich korrekt und vollständig ins Glossar eingetragen.				
5	Die Implementation der Persistence-Schicht (DataAccessMock) ist korrekt und vollständig.				
6	Die Implementation der Business-Schicht (Fachklassenmodell) ist korrekt und vollständig.				
7	Die Implementation der Presentation-Schicht (ConsoleClient) ist korrekt und vollständig.				
8	Der Prototyp läuft fehlerfrei (Namen der Kunden werden gespeichert und gelesen).				
9	Die Quellcode-Dokumentation mit Javadoc ist korrekt und vollständig.				
11	Namen bzw. Bezeichnungen in den UML Diagrammen sind aussagekräftig und die bekannten UML Namenskonventionen werden beachtet.				
12	Namen bzw. Bezeichnungen im Java-Quelltext sind aussagekräftig und die bekannten Java Namenskonventionen werden beachtet.				

Bewertung Aufgabe 4 eMail vom 27.11.13

Hallo zusammen,

Ihr habt 30 Punkte erreicht.

Die Punkte sind wie folgt von links nach rechts auf die in der Aufgabe angegebenen Bewertungskriterien von oben nach unten zu verteilen (1. von links auf 1. von oben, 2. links auf 2. von oben usw.):

2 3 2 3 3 3 3 3 2 3 3

Beste Grüsse
Arif

Ergebnis:
30 Punkt von max 33 = 5.5

Aufgabenstellung 5:

Aufgabe 5:
Objektorientierte Analyse, Design & Programmieren (OOA/D/P)
(Analyse- und Entwurfsmodell anpassen, Entwurfsmodell implementieren)

Ausgabetermin: 30.11.2013

Abgabetermin: 10.12.2013, 02.01.2014

Ausgangslage:

Folgende neuen Anforderungen der Firma PackZeug AG liegen für die 2. Iteration vor:

- Kunden können mehr als eine Anschrift, ein Telefon, eine E-Mail etc. haben
- Je nach Anzahl Bestellungen im Jahr haben Kunden einen Bronze-, Silber- oder Goldstatus. Der Status bringt den Kunden Vorteile wie z.B. Preisrabatte

Die 2. Iteration umfasst folgende Aufgaben:

- Umsetzung der neuen Anforderungen
- Vollständige Behandlung der Kundendaten (Name, Vorname und Kontaktinformationen)
- Automatisiertes Testen der Module
- Umsetzung der nicht-funktionalen Anforderungen Wartbarkeit und Wiederverwendbarkeit

Im Rahmen dieser Aufgaben müssen Sie das vorliegende Analyse- und Entwurfsmodell anpassen und das angepasste Entwurfsmodell in der Programmiersprache Java implementieren.

Aufträge:

1. Passen Sie das UML Klassendiagramm (Fachklassenmodell) des Analysemodells an für die neuen Anforderungen.
2. Passen Sie das UML Klassendiagramm des Entwurfsmodells an das angepasste UML Klassendiagramm des Analysemodells an.
3. Erweitern Sie Ihr Entwurfsmodell auf den Schichten "Business" und "Persistence" um abstrakte Klassen und Schnittstellen. Stellen Sie das angepasste Entwurfsmodell mit einem UML Klassen- und Komponentendiagramm dar.
4. Wenden Sie auf den Schichten "Business" und "Persistence" die Entwurfsmuster "Singleton" und "Factory" an. Stellen Sie das angepasste Entwurfsmodell mit einem UML Klassen- und Komponentendiagramm dar.
5. Überprüfen Sie mit einem UML Sequenzdiagramm Ihr Entwurfsmodell für den Normalablauf des Anwendungsfalls „Kunde erstellen“ hinsichtlich Vollständigkeit und Verantwortlichkeiten der Klassen und Schnittstellen sowie Interaktionen der entsprechenden Objekte. Ergänzen Sie bei Bedarf das Entwurfsmodell nach dieser Überprüfung.
6. Ergänzen Sie das Glossar um die neuen abstrakten Klassen sowie Schnittstellen aus dem Entwurfsmodell.
7. Implementieren Sie Ihr angepasstes Entwurfsmodell.
8. Implementieren Sie für die Schichten "Business" und "Persistence" Unit-Tests mit JUnit für den Anwendungsfall „Kunde erstellen“.
9. Testen Sie Ihre Implementierung für die Normalabläufe der Anwendungsfälle "Kunde erstellen" und "Kunde anzeigen", indem Sie mehrere Kunden erstellen, speichern, lesen und auf der Konsole deren Daten ausgeben.
10. Dokumentieren Sie den Java-Quelltext (nur Dateiköpfe und Methoden) mittels Javadoc und erzeugen Sie die zugehörigen HTML-Dokumentationsseiten.

Bewertungskriterien Aufgabe 5:

Bewertung (mittels Kontrolle und/oder Gespräch, max. 45 Punkte):					
#	Bewertungskriterien (Punkte: 0 => nicht erfüllt; 1 => >0% & <50% erfüllt; 2 => >=50% erfüllt; 3 => 100% erfüllt)	Punkte			
		0	1	2	3
1	Die Erweiterung des Fachklassenmodell (UML Klassendiagramm) des Analysemodells um die neuen Anforderungen ist korrekt und vollständig.				
2	Das Entwurfsmodell (UML Klassendiagramm) auf Basis des angepassten Analysemodells ist korrekt und vollständig.				
3	Die Erweiterung des Entwurfsmodells um abstrakte Klassen und Schnittstellen ist korrekt und vollständig.				
4	Die Anwendung der Entwurfsmuster "Singleton" und "Factory" auf das Entwurfsmodell ist korrekt und vollständig.				
5	Szenario (UML Sequenzdiagramm) für das Entwurfsmodell für den ausgewählten Anwendungsfall ist korrekt und vollständig.				
6	Abstrakte Klassen sowie Schnittstellen aus dem Entwurfsmodell sind korrekt und vollständig ins Glossar eingetragen.				
7	Implementation der Persistenzschicht (DataAccessMock) ist korrekt und vollständig.				
8	Implementation der Geschäftsschicht (fachliches Entwurfsmodell) ist korrekt und vollständig.				
9	Implementation der Präsentationsschicht (ConsoleClient) ist korrekt und vollständig.				
10	Die Implementation der Unit-Tests für die Persistence-Schicht ist korrekt und vollständig.				
11	Die Implementation der Unit-Tests für die Business-Schicht ist korrekt und vollständig.				
12	Die Implementation läuft fehlerfrei (Daten der Kunden werden gespeichert und gelesen).				
13	Quelltext-Dokumentation mit Javadoc ist korrekt und vollständig.				
14	Namen in den UML Diagrammen sind aussagekräftig und die bekannten UML Namenskonventionen werden beachtet.				
15	Namen im Java-Quelltext sind aussagekräftig und die bekannten Java Namenskonventionen werden beachtet.				

Bewertung Aufgabe 5 (Teil 1 und 2) eMail vom

Zwischenergebnis:

... Punkt von max ... = ...

Bewertung Aufgabe 5 eMail vom

Endergebnis:

... Punkt von max ... = ...

6.3. Abgaberichtlinien

Die Abgabe der bearbeiteten Projektteilaufgaben erfolgt per E-Mail an den Dozenten. Folgender Betreff ist für die E-Mail zu verwenden:

- SE AD: Team <#>: Aufgabe <#>
Beispiel: SE AD: Team 3: Aufgabe 3
- SE AD: <Vorname> <Nachname>: Aufgabe <#>
Beispiel: SE AD: Beat Müller: Aufgabe 2

Erlaubte Dateiformate

- PDF
- ODF-Formate (ODT, ODS)
- JPG
- ZIP
- Microsoft-Formate (DOCX, XLSX)

Regeln für Dateinamen

Dateien sind nach folgendem Schema zu benennen:

- aufgabe_<#>_team_<#>.<erweiterung>
Beispiel: aufgabe_1_team_3.odt
- aufgabe_<#>_<vorname>_<nachname>.<erweiterung>
Beispiel: aufgabe_1_beat_mueller.odt

Hinweis: Keine Leer- oder Sonderzeichen für Dateinamen verwenden

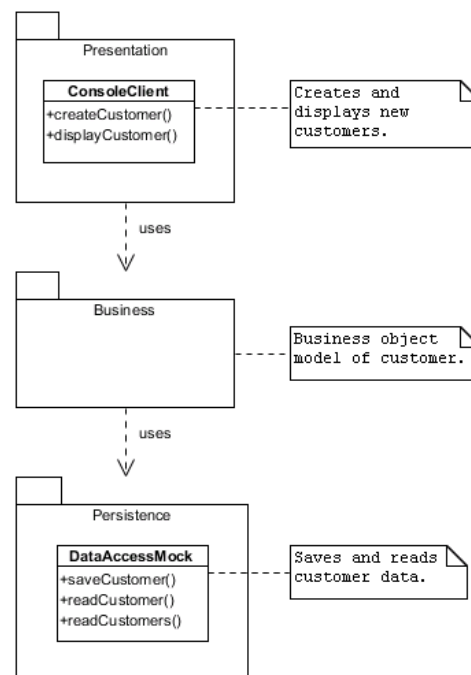
Dokumentenform

- Die Lösungsteile (Texte, Tabellen, Diagramme etc.) zu einer Aufgabe sind in einem einzelnen Dokument zusammenzufassen
- Ein Dokument für alle Aufgaben ist zu verwenden
- Das Titelblatt des Dokuments muss folgende Angaben beinhalten
 - Fach/Vorlesungstitel: Technikerschule HF Zürich – Software Engineering (A&D)
 - Haupttitel Projektarbeit, Untertitel gemäss Aufgabe
 - Datum (TT.MM.JJJJ)
 - Vorname(n) und Nachname(n) des Autors (der Autoren)
 - Klasse
- Je Aufgabe ein Hauptkapitel mit einem entsprechenden Namen
- UML-Diagramme sind als Grafiken in das Dokument zu integrieren. Falls dies nicht vernünftig darstellbar möglich ist, sind die betreffenden UML-Diagramme als JPG-Dateien beizulegen und im Dokument entsprechend zu referenzieren
- Texte, Tabellen, Diagramme etc. sind durch Überschriften bzw. Titeln den einzelnen Aufträgen einer Aufgabe zuzuordnen

6.4.Zusatzinfos zu den Aufgaben

Aufgabe 4

Gegebene logische 3-Schichtenarchitektur:



Hinweise zu den Aufträgen 5 und 6:

- Allgemein:
 - o Implementieren Sie nicht alles auf einmal, sondern gehen Sie nach dem Prinzip vor „Code a little, test a little“.
 - o Implementieren Sie nur die Anwendungsfälle „Kunde anlegen“ und „Kunde anzeigen“.
 - o Speichern und lesen Sie nur Vor- und Nachname der Kunden.
- Vorbereitung:
 - o Erstellen Sie ein neues Eclipse-Projekt.
 - o Legen Sie im neuen Eclipse-Projekt folgende Java-Packages an:
 - Presentation-Schicht
ch.<WURZEL>.persistence
 - Business-Schicht
ch.<WURZEL>.business
 - Persistence-Schicht
ch.<WURZEL>.presentation
- 1. Persistence-Schicht:
 - Implementieren Sie die Klasse DataAccessMock.
 - Speichern Sie im DataAccessMock Kundendaten (Vor- und Nachname) nicht im Dateisystem, sondern in-memory (im Hauptspeicher mittels Arrays und Strings).
- 2. Business-Schicht
 - Implementieren Sie aus Ihrem Fachklassenmodells nur die Teile, die für die Anwendungsfälle "Kunde erstellen" und "Kunde anzeigen" benötigt werden.
- 3. Presentation-Schicht
 - Implementieren Sie die Klasse ConsoleClient.
 - o Erstellen und speichern Sie mehrere Kunden
 - Verzichten Sie auf Benutzereingaben über die Tastatur und arbeiten Sie stattdessen mit "harten" Testdaten für die Daten der Kunden
 - o Lesen Sie die Kunden und geben Sie diese wieder auf der Konsole aus.

Hinweise zum Auftrag 7:

- Nur Dateiköpfe und Methoden dokumentieren.
- getter- und setter-Methoden dürfen zusammengefasst dokumentiert werden.

Aufgabe 5

Hinweise:

- Allgemein:
 - Implementieren Sie nicht alles auf einmal, sondern gehen Sie nach dem Prinzip vor „Code a little, test a little“.
 - Implementieren Sie nur die Anwendungsfälle „Kunde anlegen“ und „Kunde anzeigen“.
 - Speichern und lesen Sie nur Vor- und Nachname der Kunden.
- Auftrag 7:
 - Optional können Sie Array durch ArrayList ersetzen, um die Implementierung zu vereinfachen
- Auftrag 8:
 - Legen Sie für die Unit-Tests folgende neuen Java-Packages an:
 - Business-Schicht
ch.<WURZEL>.test.business
 - Persistence-Schicht
ch.<WURZEL>.test.presentation
- Auftrag 10:
 - Nur Dateiköpfe und Methoden dokumentieren
 - getter- und setter-Methoden dürfen zusammengefasst dokumentiert werden