

Pflichtenheft -

Tatjana Gartner, Andreas Frey

10. Januar 2017

Inhaltsverzeichnis

1	Zielbestimmungen.....	1
1.1	Kernfunktionalität	1
1.2	Wunschfunktionalität.....	2
1.3	Abgrenzungskriterien	2
2	Produkteinsatz.....	2
2.1	Anwendungsbereiche.....	2
2.2	Zielgruppen.....	2
3	Produktfunktionen.....	3
3.1	UML Diagramme.....	4
3.2	Beschreibung UML Diagramme.....	7
4	Grafische Benutzeroberfläche (GUI).....	8
4.1	Allgemeines zur Grafischen Benutzeroberfläche.....	8
4.2	Details zur Benutzeroberfläche.....	8
4.2.1	Kunden-Fenster	8
4.2.2	Mitarbeiter-Fenster	9
5	Qualitätszielbestimmungen.....	9

1 Zielbestimmungen

Die Web-Applikation, dient der Auftragsabwicklung des Pizza Lieferservices "MafiatortenExpressService". Der Pizzashop bietet verschiedene Produkte an, welche nach Kategorien eingeteilt sind. Die Web-Applikation ermöglicht es den Kunden sich online zu registrieren und Produkte zu bestellen. Kundendaten, sowie Kundenbestellungen und Produktdaten werden in einer Datenbank gespeichert. Darüber hinaus werden auch Dienste für die Mitarbeiter des Lieferdienstes bereitgestellt. Sie nutzen die Web-Applikation über einen separaten Zugang, dieser ermöglicht es einem Mitarbeiter gespeicherte Kundendaten, sowie Daten zu den Produkten, zu verwalten. Hierfür werden verschiedene Auswertungsmöglichkeiten zur Verfügung gestellt.

1.1 Kernfunktionalität

- Login über Dropdown-Liste (ohne Passwortabfrage). Ein Kunde mit Namen Emil und ein Mitarbeiter mit Namen Padrone soll standardmäßig eingerichtet sein
- Unterscheidung Mitarbeiter/Kunde bei der Nutzung des Dienstes
- Kunden können sich selbst registrieren
- Kunden können Produkte im Sortiment bestellen (inkl. Preisberechnung und Lieferzeitangabe); pro Bestellung kann nur ein Produkt bestellt werden (dieses aber in größerer Anzahl)
- Kunden können ihre eigenen Bestellungen anschauen (inkl. Gesamtsumme über alle Bestellungen)
- Mitarbeiter können Produkte anzeigen
- Mitarbeiter können User verwalten (anzeigen, hinzufügen, ändern)
- Mitarbeiter können Bestellungen ansehen nach folgenden Kriterien:
 - Alle Bestellungen anzeigen (mit Angabe von Gesamtumsatz und Durchschnittswert aller Bestellungen)
 - Alle Bestellungen pro Kunde anzeigen (mit Gesamtumsatz und Durchschnittswert für diesen Kunden)

1.2 Wunschfunktionalität

- Produkt-Verwaltung (anzeigen, hinzufügen, ändern)
- Kategorien-Verwaltung
- Löschen von Kunden und Produkten (Löschen jedoch nur, wenn es noch keine zugehörige Bestellung gibt, andernfalls nur deaktivieren)
- Login-Verfahren mit Name und Passwort
- Pro Bestellung können mehrere Produkte bestellt werden
- Status der Bestellung mitführen (z.B. bestellt, in Lieferung, ausgeliefert, storniert)
- Weitere Auswertungen über Bestellungen
- Weitere Funktionen nach eigenem Wunsch

1.3 Abgrenzungskriterien

Die aktuelle Version zum 10.1.2017 umfasst alle unter dem Punkt 1.1 genannten Funktionalitäten. Darüber hinaus wird das Login-Verfahren mittels einer Passwortabfrage durchgeführt. Produkte und Kunden können verwaltet und auch aus vollständig aus der Datenbank entfernt werden, sofern ein Produkt noch nicht bestellt wurde bzw. ein Kunde noch nie eine Bestellung getätigt hat.

2 Produkteinsatz

2.1 Anwendungsbereiche

Die Web Applikation erleichtert dem Lieferdienst die Auftragsabwicklung. Kundenbestellungen können nun online durch Mitarbeiter eingesehen werden und auch zu einem späteren Zeitpunkt erneut aufgerufen werden. Der Kunde kann bequem online Bestellen und wird automatisch durch das System über den Preis einer Kundenbestellung informiert. Der Web-Shop steigert die Kundenzufriedenheit durch eine rasche und visualisierte Auftragsabwicklung.

2.2 Zielgruppen

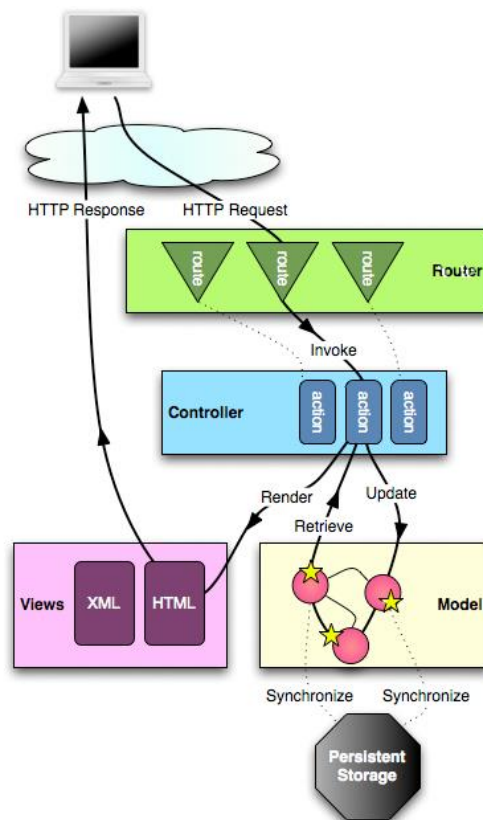
Aufgrund seiner einfachen Bedienung ist die Web-Applikation für jeden Benutzer geeignet. Zielgruppen dieses Produkts sind grundsätzlich in allen Altersgruppen zu finden.

3 Produktfunktionen

Nachstehendes Diagramm beschreibt zunächst, das Architekturmuster der Web-Applikation.

Sendet ein User eine HTTP-Anfrage ab, wandelt ein Router diese Anfrage in eine Aktion um und ruft die Klasse *Controller* auf. Ein Controller dient der Programmsteuerung, hier werden Auswertungen der Benutzereingaben durchgeführt und anschließend entsprechend an die *Model* Klasse weitergeleitet. Die Modelklasse enthält die Geschäftsobjekte, diese werden nach Erhalt der Benutzerdaten aktualisiert und zusätzlich in einer Datenbank (*persistent storage*) gesichert. Die Datensicherung (Synchronisation) wird in dieser Web-Applikation durch eine weitere Klasse *Service* realisiert. Nach diesem Vorgang werden die aktualisierten Daten zurück an den Controller gesendet.

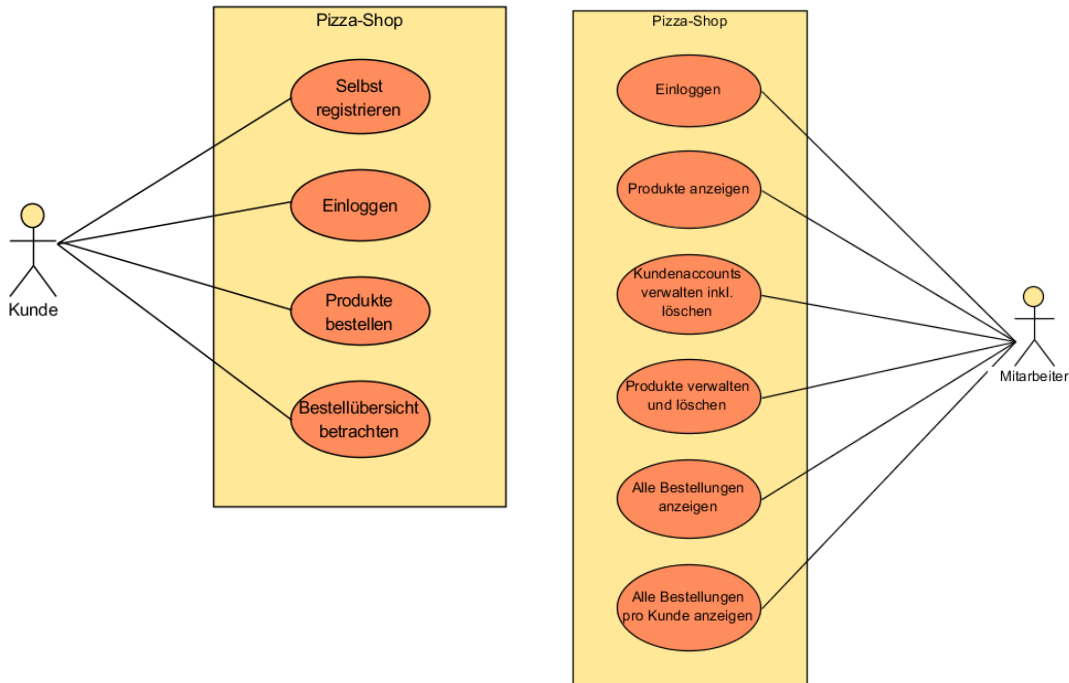
Die Darstellung der Web Applikation ist in der Klasse *View* festgelegt. Nachdem der Controller die aktualisierten Daten zurück an die View übergeben hat, wird die Benutzerseite aktualisiert.



Play Framework

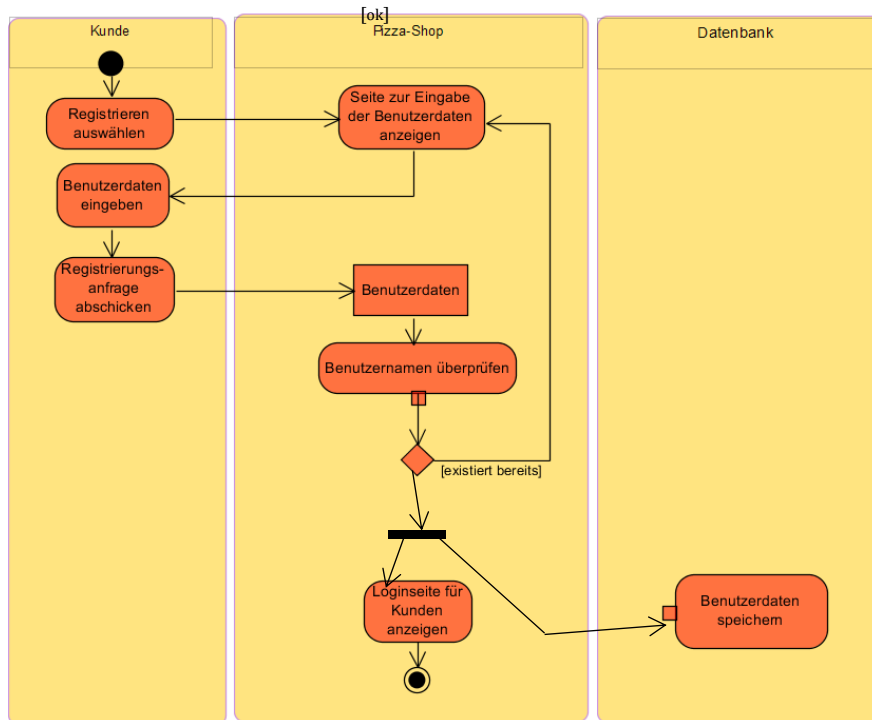
Quelle: <https://www.playframework.com/documentation/1.0/main> Abgerufen am 5.1.2017

3.1 UML Diagramme

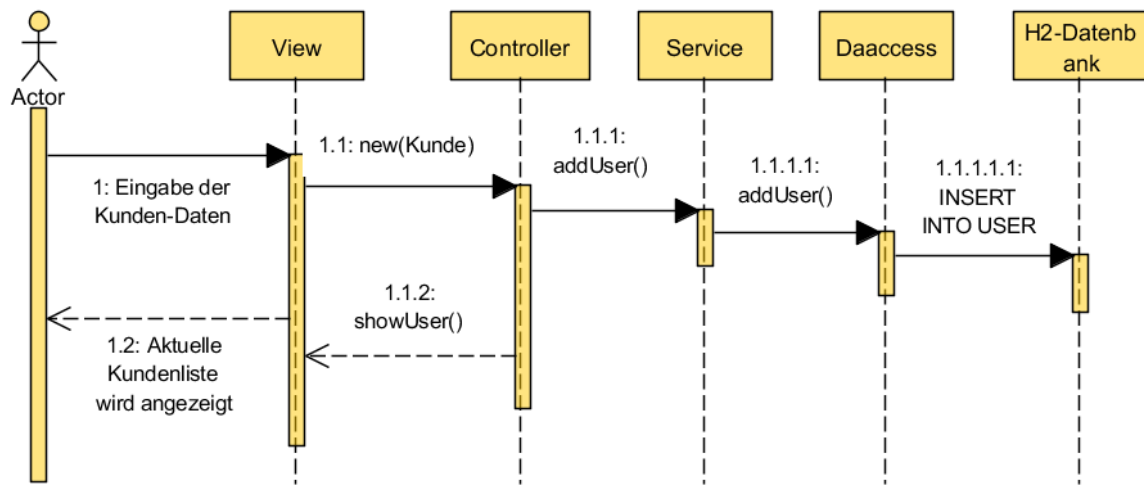


Use-Case-Diagramm 1: Kundendienste

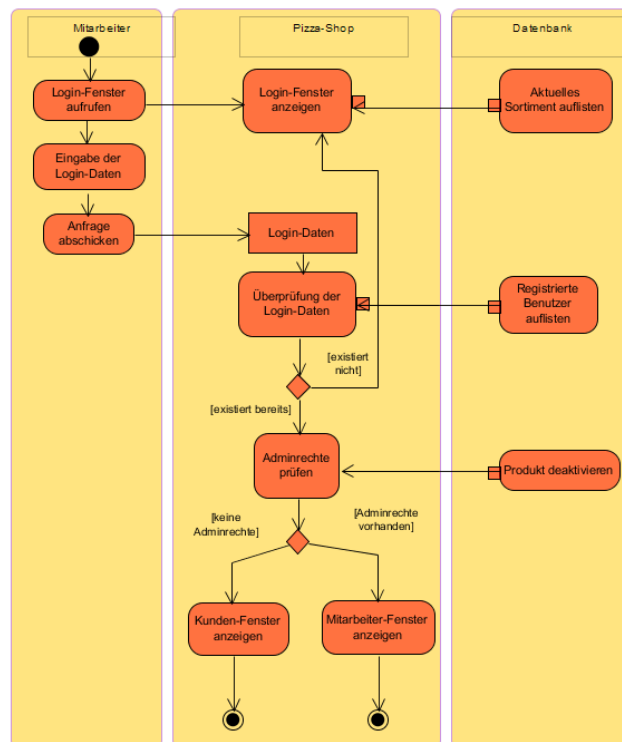
Use-Case-Diagramm 2: Mitarbeiterdienste



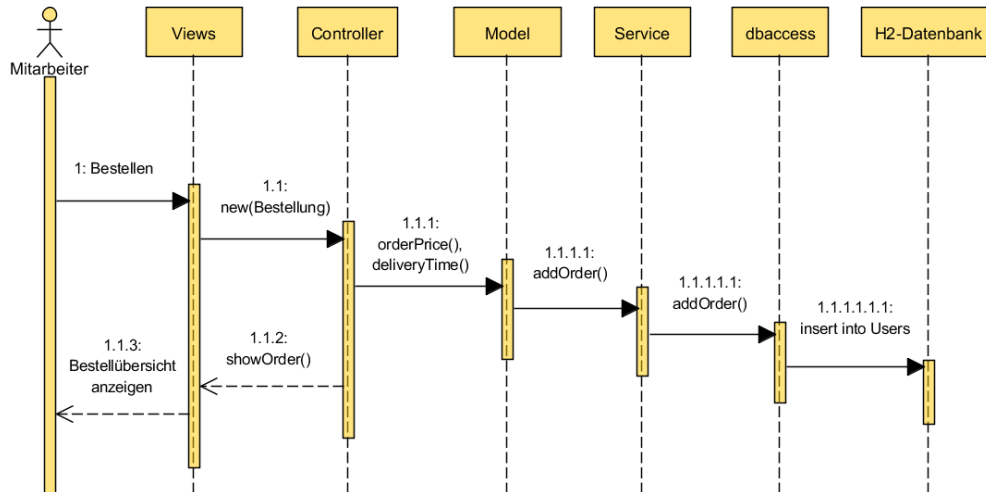
Aktivitätsdiagramm 1: Selbst-Registrierung eines Kunden



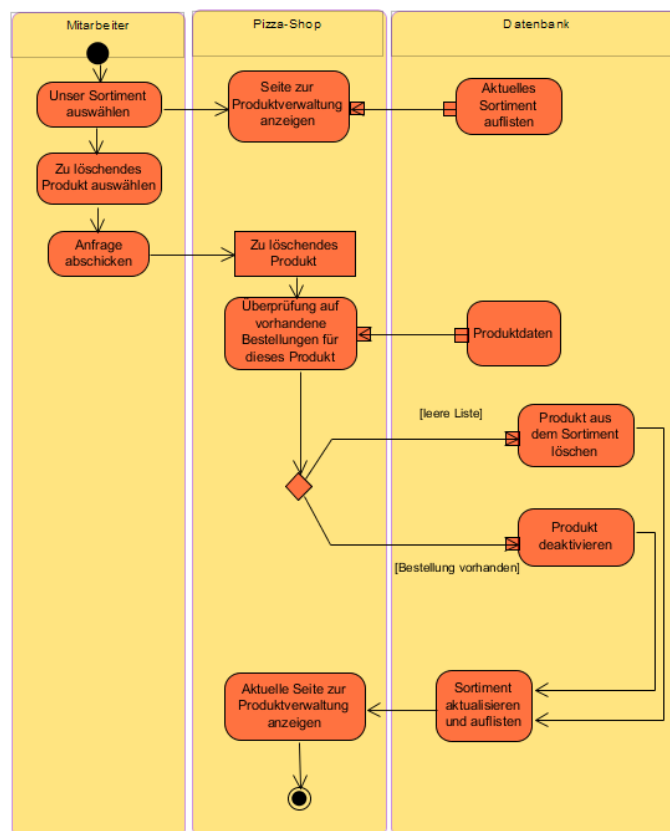
Sequenzdiagramm 1: Registrierung eines Kunden durch einen Mitarbeiter



Aktivitätsdiagramm 2: Differenzierung beim Login



Sequenzdiagramm 2: Bestellvorgang



3.2 Beschreibung UML Diagramme

Die Use-Case-Diagramme 1 und 2 zeigen, dass es vier wesentliche Use-Cases gibt, welche die Kunden betreffen, sowie sechs Use-Cases, welche die Mitarbeiter betreffen.

Der Ablauf einer Neuregistrierung eines Kunden wird im Aktivitätsdiagramm 1 gezeigt. Ein Kunde gelangt zur Registrierungsseite, indem er über einen Button „registriere dich“ auf der Startseite wählt. Nun können die notwendigen Benutzerdaten in eine Maske eingetragen und der Registrierungsantrag abgesendet werden. Datenbankgestützt überprüft das System zunächst, ob der gewählte Benutzername bereits vorhanden ist, um Duplikate zu vermeiden. Sofern der Benutzername bereits existiert, wird der Registrierungsantrag nicht erfolgreich abgeschlossen und das System leitet zurück zur Registrierungsseite. Falls der Benutzername noch nicht vergeben ist, werden die beim Antrag übergebenen Daten in der Datenbank gespeichert und der Kunde wird auf die Login-Seite weitergeleitet.

Das Sequenzdiagramm 1 beschreibt den Ablauf einer Kundenregistrierung durch einen Mitarbeiter. Nachdem ein Mitarbeiter die Kundendaten an das System gesendet hat werden diese von der Klasse *Views* an die Klasse *Controller* übergeben und dort ausgewertet. Mit der Methode `addUser()` werden die Daten an die Klasse *UserService* weitergeleitet. Die Klasse *UserDao* ist die Schnittstelle zur H2-Datenbank, hier werden die Daten gespeichert. Nach diesem Vorgang wird die aktuelle Benutzerliste abgerufen und dem Mitarbeiter angezeigt.

Das „Einloggen“ betrifft sowohl den Kunden als auch den Mitarbeiter. Daher muss das System die verschiedenen Nutzer der Web-Applikation unterscheiden können, das Aktivitätsdiagramm 3 beschreibt diesen Vorgang näher. Nachdem der Nutzer seine Login-Daten an das System übergeben hat, prüft das System, ob die Angaben mit einem Datenbankeintrag übereinstimmen. Zu jedem registrierten Nutzer ist eine Variable in der Datenbank vermerkt, welche anzeigt, ob dieser zum Zugang auf die Mitarbeiterdienste bzw. auf die Kundendienste hat. Nachdem die Variable überprüft wurde, leitet das System den Anwender dementsprechend zum Mitarbeiter bzw. Kundenfenster weiter.

Möchte ein registrierter Kunde nun eine Bestellung durchführen, wählt er im Kundenfenster über einen Button „Bestellen“. Der Kunde kann nun ein Produkt in beliebiger Menge und Größe wählen und die Bestellung kann abgesendet werden. Die Klasse *Controller* empfängt die Bestellung und wertet diese aus. Eine Bestellung mit einer Pizza bzw. ohne eine Pizza werden zu unterschiedlichen Methoden der Klasse *Model* weitergeleitet, hier werden weitere Variablen berechnet (Gesamtpreis, Lieferzeit). Die Daten werden mit Hilfe der Klasse *Service* synchronisiert und über die

Schnittstelle dbaccess zur H2-Datenbank hinzugefügt. Im Folgefenster(View) werden die aktuellen Daten der Kundenbestellungen in einer Bestellübersicht (inkl. Gesamtpreis und Durchschnittspreis) angezeigt. Die Übersicht wird über die Methode `showOrder()` aus der Datenbank abgerufen).

Mitarbeiter können Produkte verwalten (hinzufügen, ändern, hinzufügen, deaktivieren) und unter anderem auch löschen. Dieser Vorgang wird durch das Aktivitätsdiagramm 3 näher beschrieben. Wählt ein Mitarbeiter den Button „*Unser Sortiment*“, gelangt er zu einer Übersicht aller Produkte des Lieferdienstes und deren Status. Diese Informationen werden mittels Datenbankaufruf in einer Auflistung aufgelistet. Über die Maske „*Produkt löschen*“ kann der Mitarbeiter wählen und die Anfrage auf Entfernung des Produkts absenden. Das System überprüft zunächst datenbankgestützt, ob das Produkt bereits zu einem früheren Zeitpunkt bestellt wurde, in diesem Fall wird das Produkt lediglich deaktiviert, aber nicht gelöscht. Die Änderung wird in der Datenbank gespeichert. Die Produktübersicht wird aktualisiert und das Produkt wird nun entweder unter dem Status deaktiviert angezeigt, bzw. verschwindet es, falls es erfolgreich und unwiderruflich gelöscht wurde.

4 Grafische Benutzeroberfläche (GUI)

Die Grafische Benutzeroberfläche ist eine Kernanforderung an die Benutzbarkeit der Software. Sie soll das Programm möglichst einfach bedienbar machen und dennoch anschaulich den vollen Funktionsumfang der Software widerspiegeln.

4.1 Allgemeines zur Grafischen Benutzeroberfläche

Kernelement der GUI ist die Login-Seite. Um dieses Hauptfenster den Anforderungen anzupassen wird eine Schaltfläche „*Registriere dich*“ (zum Anlegen eines Benutzer-Accounts), sowie eine Login-Maske (um die Dienste des Pizza-Shops nutzen zu können) vorhanden sein.

4.2 Details zur Benutzeroberfläche

4.2.1 Kunden-Fenster

Im Kunden-Fenster, werden verschiedene Buttons zur Verfügung gestellt: „*Bestellen*“, „*Bestellungen*“, „*Logout*“. Durch Wählen gelangt der Kunde zu den jeweiligen Diensten.

Bestellen: Leitet zum Bestellformular weiter

Bestellungen: Zeigt eine Übersicht über bereits getätigte Bestellungen

Logout: beendet die Session des Kunden

4.2.2 Mitarbeiter-Fenster

Im Mitarbeiter-Fenster, werden verschiedene Buttons zur Verfügung gestellt: „*Unser Sortiment*“, „*Kundenverwaltung*“, „*Bestellungen ansehen*“, „*Logout*“. Durch Wählen gelangt der Mitarbeiter zu den jeweiligen Diensten.

Unser Sortiment: Zeigt eine Übersicht über alle Produkte und deren Status, alle Produkte können über Masken verwaltet werden

Kundenverwaltung: Zeigt eine Übersicht über alle Kunden und deren Status, alle Kunden können über Masken verwaltet werden

Bestellungen ansehen: Eine (kundenbezogene) Bestellübersicht kann aufgerufen werden

Logout: beendet die Session Mitarbeiters

5 Qualitätszielbestimmungen

	sehr wichtig	wichtig	weniger wichtig
Benutzerfreundlichkeit	x		
Effizienz	x		
Fehlertoleranz		x	
Robustheit	x		
Sicherheit			x
Zuverlässigkeit		x	