

Exercise Sheet C

Birgit Pohl 574353 (MO. 9-11)
Kevin Trogant 572451 (Mo. 15-17)
Ronja Enseleit 572404 (Mo. 15-17)
Dustin Hebecker 571271 (MO. 9-11)



Group: *Iron Man*

15. Januar 2017

Aufgabe 1 (\approx Systemtest)

Im folgenden findet sich eine Liste mit Akzeptanztests die vom Kunden durchgeführt werden sollten. Da die Natur möglicher Falscheingaben (z.B. sind Numerische Rezeptnamen erlaubt etc.) nicht bekannt ist, wird angenommen, dass solche Falscheingaben bereits im Funktionstest getestet wurden und abgesehen vom Format der Fehlerausgabe diese nicht Teil des Akzeptanztests ist. Werden alle Tests in der gegebenen Reihenfolge ausgeführt kann auf vorherigen Eingaben aufgebaut werden.

Teste als Mitarbeiter:

- Erstelle drei verschiedene Angebote mit Produkt Namen, Rezept für die drei Größen und entsprechendem Preis.
- ⇒ Die Angebote erscheinen wie eingegeben im Kunden-Menü.
- Ändere eines der zuvor erstellten Angebote bezüglich Namen, Rezept und Preis.
- ⇒ Das Angebot erscheint wie geändert im Kunden-Menü.
- Lösche eines der zuvor erstellten Angebote.
- ⇒ Das Angebot verschwindet aus dem Kunden-Menü.

Teste als Kunde:

- Wähle acht verschiedene Pizzen: Eine Pizza je Größe mit festem Rezept, für Large und X-Large bestelle je eine Pizza mit gemischter Belegung und wähle zusätzliche Toppings für je eine Pizza ohne andere Beläge, einem festen Rezept und gemischter Belegung.
- ⇒ Die Pizzen erscheinen wie ausgewählt in der Bestellübersicht.
- Gib eine Bestellung je zur Abholung und zur Lieferung auf und das je einmal als Neukunde, einmal als Bestandskunde und einmal als gesperrter Kunde.
- ⇒ Beim bestandskunden erscheinen die Pizzen vor der Haustüre und im Geschäft, nach einem positiven Bestätigungsanruf passiert das gleiche beim Neukunden. Beim gesperrten Kunden passiert nichts oder eine Sperrmeldung erscheint auf der Website.
- Ändere jede Bestellung einmal vor der Makierung "Backend" und hinzufügen weiterer Pizzen nach der Makierung "Backend".
- ⇒ Die geänderten Bestellungen erscheinen in der Bestellübersicht und später auf dem Teller.

Teste als Mitarbeiter:

- Teste das Akzeptieren/Hinzufügen eines Neukunden als auch das Ablehnen eines Neukunden.
- ⇒ Der Mitarbeiter kann die Daten korrekt im System hinterlegen.

Teste als Koch:

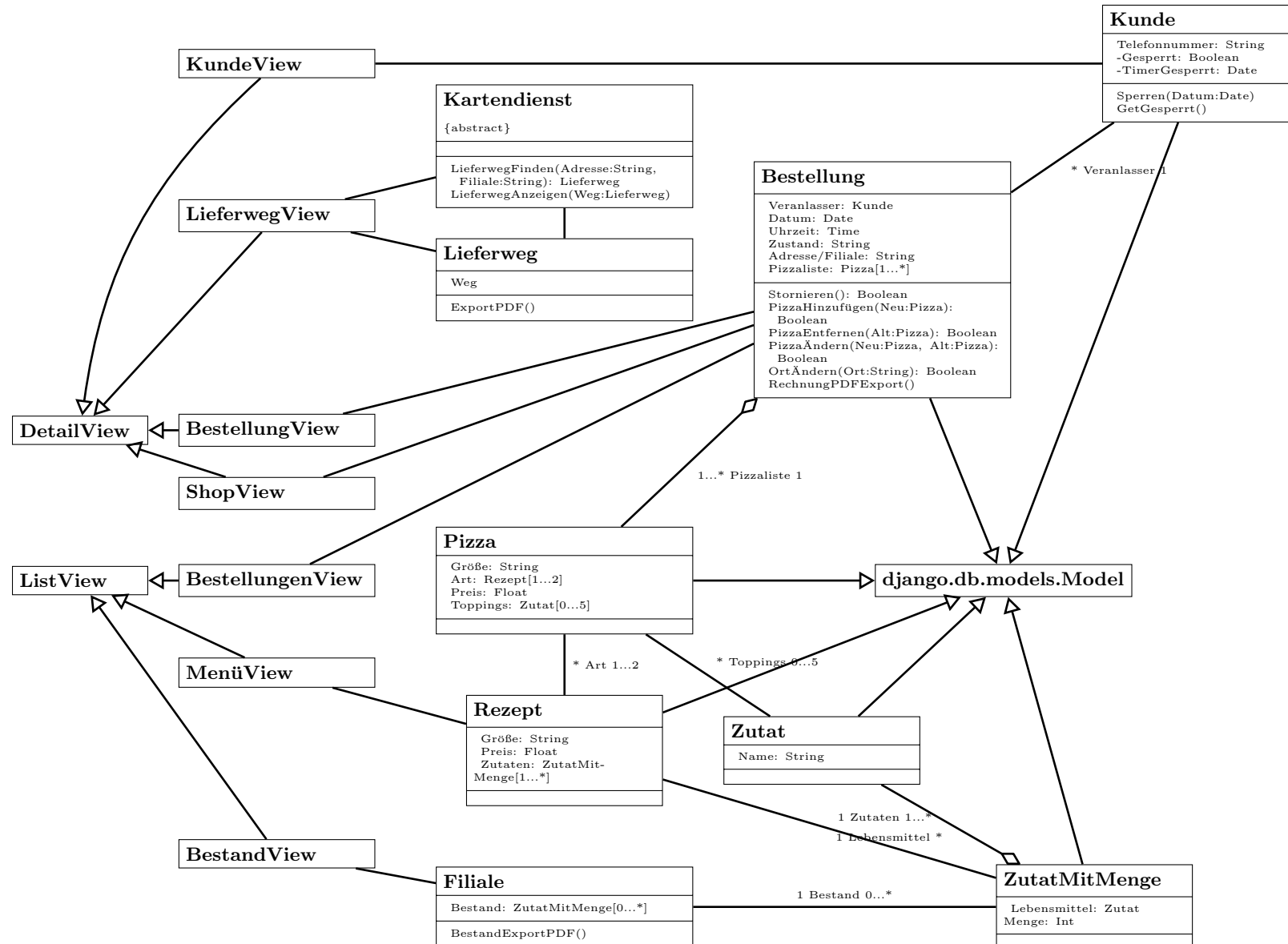
- Das Aufrufen der Bestellliste mit und ohne verfügbaren Filtern.
- ⇒ Eine komplette Bestellübersicht über alle Bestellungen erscheint und die Filter können erfolgreich angewendet werden.
- Ändere je eine Bestellung auf fertig gebacken und abgebrochen.¹
- ⇒ Die Eisntellungen werden erfolgreich ins System übernommen.

Teste als Mitarbeiter:

- Liefere eine Bestellung aus sowohl mit Ausdruck einer Karte als auch ohne.
- ⇒ Das System druckt die Rechnung einmal mit und einmal ohne aus und übernimmt den Status der Lieferung.
- Einsicht der Fillialvorräte.
- ⇒ Eine Liste aller Fillialvorräte erscheint.
- Aktualisierung der Fillialvorräte.
- ⇒ Die Änderungen werden erfolgreich ins System übernommen.
- Export der Fillialvorräte.
- ⇒ Ein PDF-Dokument mit allen Fillialvorräten wird erstellt.

¹Eine fehlende Vorratsaktualisierung bei einem Abbruch nach Backende ist hier bedenklich, die Recourcen wurden ja verbraucht.

Aufgabe 2



Team: *Iron Man*

B. Pohl, K. Trogant, R. Enseleit, D. Hebecker

15. Januar 2017

Die Klassen `DetailView`, `ListView` und `django.db.models.Model` stammen aus einem Python-Framework für die Entwicklung von Web-Anwendungen mit dem Namen Django und werden nicht von uns implementiert.