

Fiori App Entwicklung 2

Aufgabe 8 (20 Punkte):

Einführung

In dieser Aufgabe soll die in Aufgabe 7 erstellte Applikation erweitert werden. Zur beispielhaften Illustration sollen die folgenden Screenshots der in der Vorlesung gezeigten Anwendung dienen:

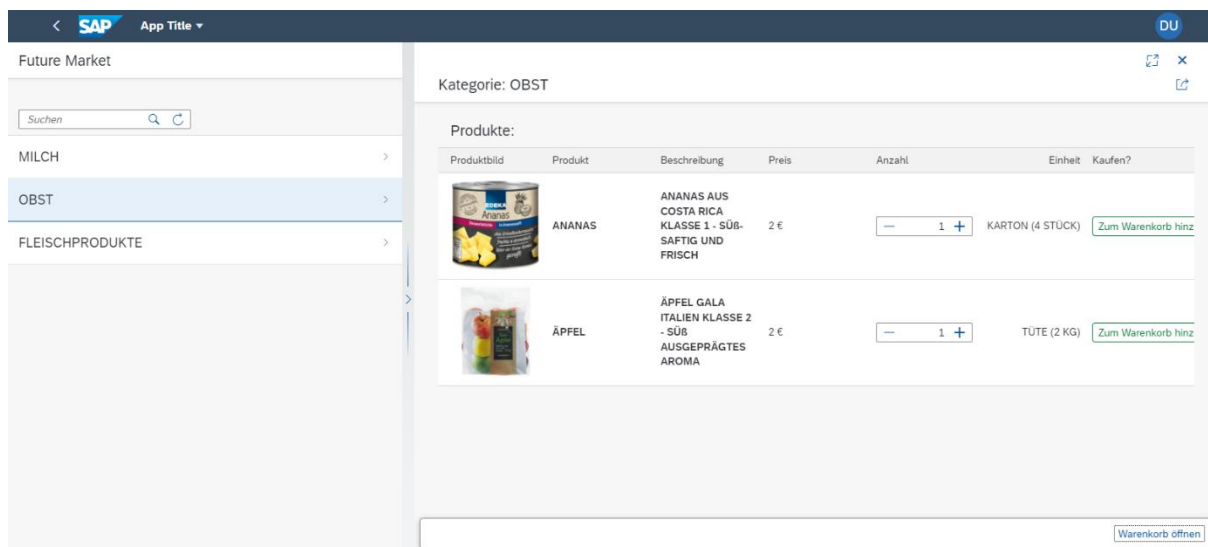


Abbildung 1: Screenshot des Hauptscreens der finalen FutureMarket Anwendung

Die Applikation unterteilt sich grundsätzlich in zwei Bereiche. Im linken Bereich (1) wird die Liste der Produktkategorien angezeigt. Klickt der Nutzer auf eine Produktkategorie, erscheinen im rechten Bereich (2) die der Kategorie zugeordneten Produkte. Für jedes Produkt werden die wichtigsten Eigenschaften direkt angezeigt.

- Ein Produkt kann über den Button „Zum Warenkorb hinzufügen“ zum Warenkorb hinzugefügt werden.
- Am unteren rechten Bildschirmrand kann über den Button „Warenkorb öffnen“ der Warenkorb geöffnet werden.

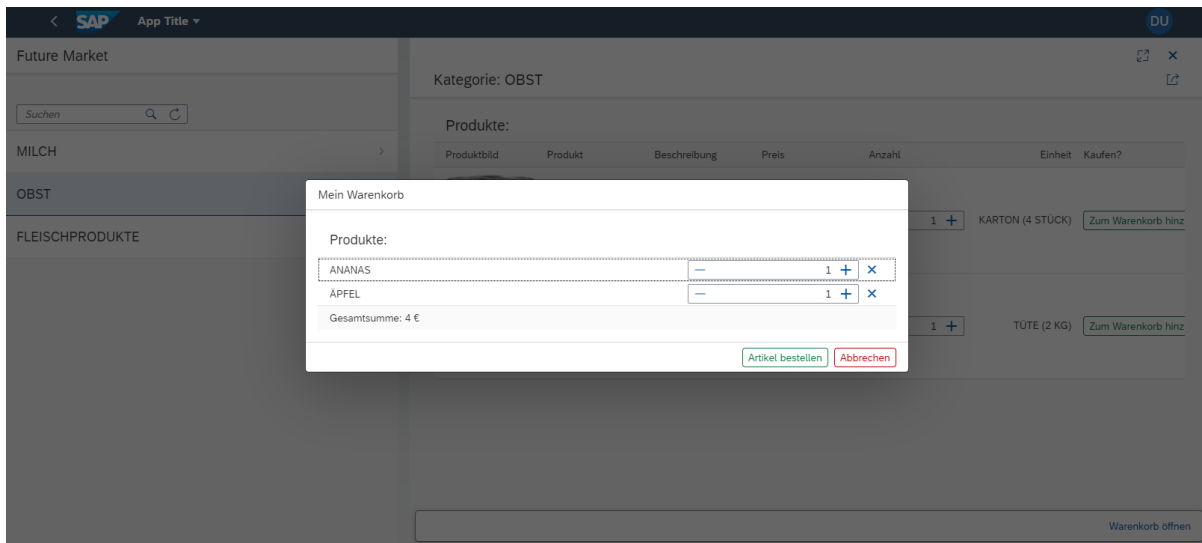


Abbildung 2: Screenshot des Warenkorbs der finalen FutureMarket Anwendung

Nach dem Klick auf „Warenkorb öffnen“ erscheint ein Popup welches den Inhalt des Warenkorbs zeigt. Für jeden Artikel wird der Artikelname sowie die aktuelle Menge angezeigt.

- Für jedes Produkt kann über einen „-“ Button die Menge reduziert, über einen „+“ Button die Menge erhöht werden.
- Des weiteren befindet sich am ende einer jeden Produktzeile ein „x“ Button, der das Produkt vollständig aus dem Warenkorb entfernt. Wurde das Produkt erfolgreich entfernt soll eine Nachricht per `MessageToast.show()` mit dem Inhalt „Löschvorgang erfolgreich“ ausgegeben werden.
- Im unteren Teil des Warenkorbs wird der aktuelle Warenkorbwert angezeigt. Dieser aktualisiert sich automatisch, sollte sich die Menge der angezeigten Produkte ändern.

Im Footer des Dialogs befinden sich die Buttons „Artikel bestellen“ und „Abbrechen“.

- Wird der Button „Artikel bestellen“ gedrückt, wird eine Bestellanforderung an den Server geschickt. Bei einer erfolgreichen Ausführung der Bestellung soll die Meldung „Bestellung erfolgreich“ ausgegeben und der Warenkorb geleert werden. Bei einem Fehler soll die Meldung „Fehler bei der Bestellung“ ausgegeben werden. Im Fehlerfall soll der Warenkorb nicht geleert werden.
- Wird der Button „Abbrechen“ gedrückt, wird der Warenkorb geschlossen.



Aufgabe

1. Im Git Repository <https://git.octavia.de/uni-kassel/bpe-ws-21-22/ha8.git> befindet sich eine Version der FutureMarket Applikation, die als Basis für die weiteren Aufgaben verwendet werden soll.

Tip: Verwendet folgende Git Befehle um den Startcode in euren Workspace zu klonen und anschließend das fertige Ergebnis in eurem eigenen Repository in einem separaten Branch zur Verfügung zu stellen:

```
// Annahme: Arbeit beginnt im Projekt aus HA7

git remote add ha8-template https://git.octavia.de/uni-kassel/bpe-ws-21-22/ha8.git
git fetch --all
git checkout -b ha8 ha8-template/master

// Änderungen machen

// Beim 1. Push
git push -u origin ha8

// Bei jedem weiteren Push reicht
git push
```

2. Fülle alle Stellen die im Code mit „TODO“ annotiert sind mit dem entsprechenden Code, um das Aussehen und Verhalten aus der im Einführungskapitel beschriebenen FutureMarket Applikation zu erreichen.

TODOs befinden sich in folgenden Dateien:

- Detail.controller.js
- Detail.view.xml
- DialogShoppingCard.fragment.xml
- List.view.xml

Stellt sicher, dass sich euer Projekt ohne Fehler starten lässt!

Abgabeformat: Checkt eure Abgaben im OctaVIA GIT ein und benutzt den Tag „Aufgabe-8“!

Deadline:

Montag, 20.12.2021, 14:00 Uhr

Donnerstag, 23.12.2021, 14:00 Uhr