

Projektarbeit Web-Programmierung

Prof. Dr. Jörg Hettel
Hochschule Kaiserslautern
Studiengang IT-Analyst
– Version: 6. August 2020 –

SPEZIFIKATION V 1.0

Inhaltsverzeichnis

| | | |
|----------|---------------------------------------|----------|
| 1 | Vorbemerkung | 1 |
| 2 | Technologien | 1 |
| 3 | Anforderungskatalog | 2 |
| 3.1 | Anforderungen an die Anwendungslogik | 2 |
| 3.2 | Anwendung: Online Pizza-Bestellung | 2 |
| 3.3 | Anforderungen an den Web-Client | 3 |
| 3.4 | Anforderungen an die Server-Anwendung | 3 |
| 3.5 | Abgrenzungen | 4 |
| 4 | Bewertungskriterien | 4 |
| 5 | Abgabe | 4 |

1 Vorbemerkung

Ziel der Projektarbeit ist die Anwendung der in dem Modul besprochenen Web- und Server-Technologien. Im Rahmen der Projektarbeit soll eine Web-Anwendung entwickelt werden, die verschiedenen Anforderungen genügt (siehe unten). Im Vordergrund stehen die Funktionalität und die Anwendungsarchitektur und nicht das Design der Oberfläche.

2 Technologien

Folgende Technologien sollen eingesetzt werden:

Client: Auf dem Client sollen HTML5 und CSS (Version 3.0) zum Einsatz kommen. Weiter soll JavaScript verwendet werden. Der Einsatz von Frameworks bzw. Bibliotheken (auch Auszüge oder Teile davon) von Drittanbietern (z.B. JQuery, Bootstrap, etc.) ist nicht erlaubt.

Server: Auf Server-Seite soll Java bzw. der Web-Container der JEE zum Einsatz kommen. Optimalerweise wird Tomcat (Version 8.x oder höher) eingesetzt. Auch hier sollen

nur die Basistechnologien Servlets und JSP (mit entsprechenden Standard-Tags und Standard-Expression-Language) eingesetzt werden. Frameworks wie Spring, Struts oder auch JavaServer Faces¹ sind nicht erlaubt.

Die Anwendung soll mit transienten Daten arbeiten, so dass keine Datenbank benutzt werden muss. Die Daten können über eine Zugriffsfassade gekapselt werden (siehe z.B. die Klasse `ModelFacade` aus den Übungen).

3 Anforderungskatalog

Im Folgenden werden die Anforderungen an die Anwendung näher spezifiziert.

3.1 Anforderungen an die Anwendungslogik

Der abzubildende Geschäftsfall ist die Online-Bestellung von Pizzen (siehe. Abschnitt 3.2)

- Die Web-Anwendung sollte mindesten 7 HTML- bzw. JSP-Seiten umfassen. Die Navigationsstruktur muss eine gewisse Komplexität besitzen. Fasst man die Web-Seite als Graph auf, so soll der Graph mindestens 11 Kanten besitzen, wobei nicht nur „Rückwärtskanten“ eingebaut werden sollen. Sie können sich diesbezüglich an der *Bookshop*-Anwendung aus den Übungen orientieren.
- Für einige Benutzereingaben (INPUT-Felder) soll eine Eingabevalidierung mit JavaScript durchgeführt werden.
- Die Anwendung soll AJAX benutzen.
- Die Anwendung muss mit benutzerspezifischen Daten, die über mehrere Requests hinweg verwaltet werden, umgehen (z.B. Warenkorb).

3.2 Anwendung: Online Pizza-Bestellung

Es soll folgender Anwendungsfall umgesetzt werden: Eine Pizzeria möchte es künftig erlauben, dass Pizzen auch online bestellt werden können. Hierzu soll eine kleine Web-Anwendung entwickelt werden.

Interessenten sollen sich individuell Pizzen konfigurieren und dann bestellen können. Die Konfigurationsparameter sind (vgl. Abb. 1)²:

- Auswahl von verschiedener Beläge aus vorgegebenen Kategorien
- Auswahl der Größe: Klein, Mittel, Groß

Der Preis der Pizza berechnet sich aus der Auswahl der Beläge (0,5 € pro Auswahl) und dem Grundpreis, der über die Größe festgelegt ist (Klein € 4,-; Mittel € 5,- groß € 6,-). Ist eine Pizza konfiguriert, kann sie in einen Warenkorb gelegt werden. Pizzen können auch wieder aus dem Warenkorb entfernt werden bzw. deren Anzahl kann auch geändert werden.

Für den Abschluss der Bestellung muss man entweder schon am System registriert sein oder man kann sich als Kunde registrieren. Von den Kunden werden der Namen und die

¹JavaServer Faces gehören zwar zum JEE-Standard, ist aber trotzdem nicht erlaubt.

²Die Abbildung zeigt lediglich einen Design-Vorschlag, der so nicht übernommen werden muss.



Abbildung 1: Der *Pizza-Konfigurator*.

Adresse registriert. Beim Anzeigen des Warenkorbs werden, je nach Inhalt des Warenkorbs auch drei weitere Bestellvorschläge gezeigt, die bei Gefallen gleich mit in den Warenkorb mit aufgenommen werden können. (Ist der Warenkorb leer, werden keine Vorschläge angezeigt.) Nach dem Einloggen bzw. der Registrierung wird der Inhalt des Warenkorbs nochmal angezeigt. Nachdem die Bestellung abgesendet wurde, kann die Bestellung (Session) gelöscht werden, d.h. Bestellungen werden nicht historisiert.

3.3 Anforderungen an den Web-Client

Der Web-Client muss folgenden Anforderungen genügen:

- Die HTML- bzw. JSP-Seiten sollten mehrere HTML-Standard-Tags enthalten, also neben `form`-Elementen auch Tabellen, `div`-Container, Überschriften, etc.
- Mit Hilfe von CSS soll zum einen das Layout der Standard-Tags verändert werden. Zum anderen sollen auch Formatklassen eingesetzt werden.
- Die Web-Seiten sollten valide sein.
- JavaScript soll „unaufdringlich“ eingesetzt werden, d.h. der eigentliche HTML-Code soll möglichst kein explizites JavaScript beinhalten.
- Weiter soll AJAX eingesetzt werden, wobei der entsprechende Service auf dem Server ebenfalls implementiert werden muss. Die personenbezogene Vorschlagsliste für weitere Pizzen auf der Warenkorbseite soll über einen AJAX-Request ermittelt werden.

3.4 Anforderungen an die Server-Anwendung

Die Server-Anwendung muss folgenden Anforderungen genügen:

- Auf dem Server soll durchgehend das MVC-Pattern eingesetzt werden, d.h. in den Servlets steht kein HTML-Ausgabe-Code und in den JSPs kein expliziter Java-Code. Die JSTL und die JSP-Expression Language darf auf der JSP eingesetzt werden.
- Die Anwendung muss eine Session verwalten (Warenkorb).

3.5 Abgrenzungen

Im Folgenden sind die Abgrenzungen aufgelistet

- Die Web-Anwendung besitzt keine Datenbankanbindung. Die Datenhaltung soll entsprechend *ge-mocked* werden.
- Das Design der Web-Seiten und die Benutzerführung sind sekundär und stehen nicht im Vordergrund. Natürlich ist es nicht verboten, dass Sie Ihre „Usability“-Kenntnisse anwenden.
- Eine Benutzerverwaltung mit verschiedenen Benutzergruppen ist nicht vorgesehen. Es gibt also nur eine Benutzergruppe (Kunden).

4 Bewertungskriterien

Folgende Kriterien werden für die Bewertung herangezogen:

- Saubere Trennung von Layout und HTML, d.h. sinnvoller Einsatz von CSS.
- Saubere Trennung von HTML und JavaScript.
- Eingabevalidierung mit JavaScript
- Einsatz von AJAX.
- Saubere Umsetzung des MVC-Pattern auf dem Server, d.h. Trennung von Java- und HTML.
- Verständlicher Code (insbesondere bei JavaScript und Java).

5 Abgabe

Abzugeben ist ein sinnvoll dokumentierter Anwendungscode. Eine explizite Projektdokumentation ist nicht erforderlich, kann aber auch mitgeliefert werden. Die Standarduser bzw. -passwörter können in einer Readme-Datei hinterlegt werden, die Sie ins WebContent-Verzeichnis legen.

Exportieren Sie die Anwendung bitte als war-Datei (Web-Archiv). Benutzen Sie hierfür bitte folgende Namenskonvention:

`WebProg_<<Nachname>>_<<Vorname>>.war`

Wichtig: Achten Sie darauf, dass die Java-Sourcen in der war-Datei enthalten sind. Die war-Datei muss lauffähig sein, d.h. sie darf keine Abhängigkeiten zu externen Bibliotheken oder ähnlichem besitzen.

Die Abgabe erfolgt über OLAT. Sie finden einen entsprechenden Abgabeordner auf der OLAT-Seite zum Modul.