

DSE BDG2, Übung 5

Wintersemester 2019/20

Deadline Abgabe: 3. November 2019, 23:59

Name: _____

Punkte: _____/24

Hinweise

- Die Abgabe der Übung erfolgt über eLearning
- Abzugeben ist ein pdf Dokument, das zu jeder Aufgabe den ausgearbeiteten Code enthält

1) namegen Service (12 Punkte)

namegen Service ist eine REST-basierte Anwendung deren Architektur wie folgt aussieht:



namegen ist eine ASP.NET Core Anwendung, deren Aufgabe es ist, zufällig generierte Namen zu liefern.

web ist ein Node.js Service, das zum Einen eine Webseite ausliefert und zum Anderen eine REST Methode zum Zurückliefern eines generierten Namens anbietet. Intern leitet web diese Anfrage an das namegen Service weiter und liefert den zurückgelieferten Wert an den Benutzer aus.

Aufgaben

a) Erstellen Sie das Dockerfile für namegen

- Verwenden Sie als Basis Image `mcr.microsoft.com/dotnet/core/sdk:3.0`
- .NET Core Anwendungen können mit dem Befehl `dotnet build` kompiliert und mit `dotnet run` ausgeführt werden
- Stellen Sie sicher, dass beim Starten des Containers `dotnet run` mit dem Parameter `--urls "http://0.0.0.0:5000"` ausgeführt wird

b) Erstellen Sie das Dockerfile für web

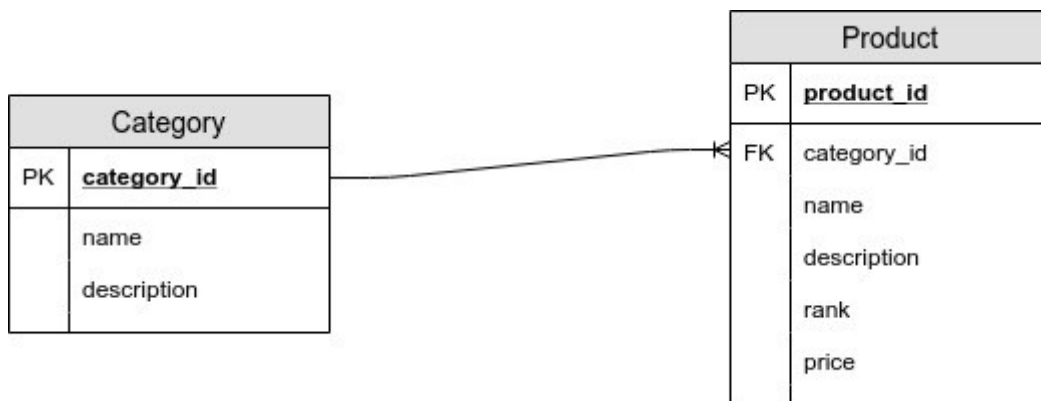
- Verwenden Sie als Basis Image `node:latest`
- Zum Herunterladen der Abhängigkeiten der Anwendungen verwenden Sie den Befehl `npm install`
- Die Node.js Anwendung kann mit `node app.js` ausgeführt werden

- web ist unter Port 3000 erreichbar

c) Erstellen Sie eine docker-compose Datei, die beide Services startet. Beachten Sie, dass web mit namgen kommunizieren muss. web nimmt an, dass namegen unter diesem DNS Namen und Port 5000 erreichbar ist (<http://namegen:5000>). Geben Sie den Port von web zum Hostbetriebssystem frei, damit Sie testen können, ob die Anwendung funktioniert. Wenn Sie die URL <http://localhost:3000> in einem Browser aufrufen, wird Ihnen ein zufällig generierter Name angezeigt.

2) Webshop (12 Punkte)

Die folgende Grafik zeigt einen Teil des ER Diagramms eines Webshops:



Ein Produkt hat einen Namen, eine Beschreibung und einen Preis. Rank gibt den Verkaufsrank des Produktes an, also wie häufig das Produkt verkauft wurde. Ein Produkt ist einer Kategorie zugeordnet. In eine Kategorie können mehrere Produkte fallen. Zum Beispiel befinden sich in der Kategorie „Elektronik“ Laptops, Fernseher oder Mobiltelefone.

Implementieren Sie REST Endpunkte in Java mit Spring Boot, um die in den nachfolgenden Aufgaben beschriebene Funktionalität umzusetzen. Es soll als Datenspeicher Redis verwendet werden.

Hinweis:

- Die im ER Diagramm dargestellten Entitäten stellen einen Teil des Datenmodells dar und dienen zum besseren Verständnis. Es müssen nur jene Entitäten und Eigenschaften in Redis abgebildet werden, die für die nachfolgenden Aufgaben benötigt werden.

Aufgaben

a) Für die Betreiber des Webshops ist eine der wichtigsten Funktionalitäten das Anlegen von neuen Produkten. Implementieren Sie daher einen Endpunkt mit dem neue Produkte inklusive Kategorien angelegt werden können.

b) Für Benutzer ist eine häufig benötigte Funktionalität das Abrufen von Informationen (Beschreibung, Preis,...) zu einem Produkt. Implementieren Sie daher einen Endpunkt, der zu einem Produktnamen die Details liefert. Sie können davon ausgehen, dass Produktnamen eindeutig sein müssen.

c) Es soll Benutzern ermöglicht werden, zu einer Kategorie alle in dieser Kategorie enthaltenen Produkte zu listen. Implementieren Sie einen Endpunkt, der zu einer Kategorie die enthaltenen Produktnamen liefert.

d) Es soll eine Liste der Top 5 billigsten Produkte ausgegeben werden. Implementieren Sie daher einen Endpunkt, der diese Top 5 Liste zurückliefert.