

Klausur Architektur von Informationssystemen

Sommersemester 2013 – 04.07.2013

Prof. Dr. Stefan Sarstedt <stefan.sarstedt@haw-hamburg.de>

Raum 1085, Tel. 040/42875-8434

Bearbeitungshinweise

- Bei Verständnisfragen heben Sie bitte den **Arm**; ich bemühe mich dann um eine Klärung.
- Es sind **keine Hilfsmittel** zugelassen.
- Schreiben Sie **leserlich**! Nicht lesbare oder unklare Teile werden mit 0 Punkten bewertet.
- Versehen Sie dieses Deckblatt mit Ihrem Namen und Ihrer Matrikelnummer.
- Die Klausur besteht aus insgesamt 3 Aufgaben auf 4 Seiten.
- Zeit zur Bearbeitung der Klausur: 120 Minuten
- Viel Erfolg!

Name:	
Matrikelnummer:	

Aufgabe 1: Allgemeine Fragen

- a) Erläutern Sie den Begriff „Software-Architektur“.
- b) Erläutern Sie folgende Begriffe aus dem Bereich der Software-Architektur und geben Sie jeweils ein Beispiel an:
- Konnektor
 - Aggregat
- c) Erläutern Sie das Konzept (Funktionsweise, sowie die Vor- und Nachteile) der Master-Slave-Replikation im Kontext von NoSQL-Datenbanken.

Aufgabe 2: Architekturentwurf & Visualisierung

Die aufstrebende Supermarktkette UNREAL beauftragt Sie mit der Konzeption eines innovativen Supermarkt-Servers zur Verwaltung Ihrer Supermärkte. Es gelten die folgenden Anforderungen und Projekt-Rahmenbedingungen:

- UNREAL-Supermärkte bestehen aus Einkaufswagen (mit eindeutiger Nummer), Gängen (mit Gang-Nr.), Regalen (Regal-Nr.) mit Produkten (EAN, Bezeichnung, Gewicht, Preis) und Kassen (Kassen-Nr.) Jedes Produkt ist genau einem Regal zugeordnet.
Hinweis: EAN = „European Article Number“ (Artikelnummer)
- In UNREAL-Supermärkten werden die in einen Einkaufswagen gelegten Produkte automatisch mit Hilfe der in den Produkten enthaltenen NFC-Chips erfasst. Diese EANs werden dann vom Einkaufswagen-Bordcomputer (der auch die Erfassung übernimmt) an den Supermarkt-Server übermittelt. Der Gesamtpreis aller Waren (berechnet auf dem Supermarkt-Server) wird stets auf dem Display des Einkaufswagens dargestellt.
- Falls ein Produkt in einem passierten Regal zu einem bereits im Einkaufswagen liegenden Produkt passt, wird durch den Bordcomputer eine Werbenachricht auf dem Display des Einkaufswagens eingeblendet. Beispiel: Falls im Einkaufswagen Grillfleisch liegt, wird beim Passieren eines Regals mit Saucen eine entsprechende Meldung gezeigt („Kunden, die das Produkt ‚Grillfleisch‘ gekauft haben, haben oftmals auch das Produkt ‚Death-by-Chili-Sauce XXL‘ gekauft. Seien Sie kein Weichei! ☺“). Informationen, welche Produkte zu welchen anderen Produkten passen, sollen auf dem Supermarkt-Server liegen.
- Die Einkaufswagen-Bordcomputer (inkl. Display, NFC-Empfänger und „Positions-Erkenner“ des Einkaufswagens) laufen unter Linux 3.1. Dieses System ist komplett durch die Fremdfirma „0815-Systems“ realisiert.
- Die Einkaufswagen-Bordcomputer sind über WLAN mit dem Supermarkt-Server verbunden. Die Kommunikation erfolgt mittels Java RMI.
- Beim Erreichen einer Kasse (erkannt durch den Bordcomputer) wird der aktuelle Einkauf durch den Supermarkt-Server an die Kasse (laufen unter Windows RT) gesendet. Die Anbindung der Kassen erfolgt über ein 100 MBit-LAN. Als Protokoll wird REST verwendet.
- Der Kunde kann an der Kasse mit Kreditkarte, Bar oder Lastschrift bezahlen (die Zahlungs-Abwicklung erfolgt nur durch die Kasse). Er erhält dann einen Rechnungsbeleg von der Kasse. Dieser Rechnungsbeleg wird auch auf dem Supermarkt-Server gespeichert und der Einkauf dort als „abgeschlossen“ markiert.
- Das Kassensystem wird ebenfalls durch die Firma „0815-Systems“ realisiert.
- Der Supermarkt-Server läuft unter Windows Server 2003 SP2. Als Persistenzmanager und Datenbank kommen Hibernate 4 und Oracle 11g zum Einsatz. Implementierungssprache ist Java 6.

Entwerfen Sie eine Architektur für den Supermarkt-Server. Erstellen Sie dazu **Sichten** und **begründen** Sie, falls angebracht, Ihre Architektur- und Entwurfsentscheidungen textuell mit **aussagekräftigen Stichworten**! Schnittstellen der Nachbarsysteme können Sie beliebig definieren.

Gefordert sind (**in UML-Notation**, außer Teilaufgabe a):

- a) Kontextsicht
- b) Fachliches Datenmodell
- c) Bausteinsicht
 - Es müssen **mehrere** fachliche Komponenten erkennbar sein.
 - Innensichten der Komponenten müssen **nicht** gezeigt werden.
- d) Laufzeitsichten für die Abläufe der in den Anforderungen der beschriebenen Szenarien (ohne Fehlerfälle)
- e) Verteilungssicht

Aufgabe 3: Architekturstile

Ergänzen Sie **alle leeren Einträge!**

Mögliche Problemstellung	Welcher Architekturstil aus der Vorlesung eignet sich am besten?
Das System soll eine Datei mit Auftragsdaten in mehreren voneinander unabhängigen Schritten performant und parallel verarbeiten.	
Ihr System soll über eine Web-Schnittstelle mit einem anderen System interagieren können. Die Web-Schnittstelle ist schlank und simpel zu halten (und darf somit bspw. nicht auf SOAP/WSDL beruhen).	
Ihr System soll umfangreiche Rechnungsdaten zuverlässig und lose gekoppelt an ein externes Abrechnungssystem schicken.	
	Blackboard
Es ist eine Klimaanlagesteuerung zu erstellen.	
	Broker