

# SE2-Uebungsaufgaben

**Bearbeiten Sie bitte die Aufgaben 1 und 2 und zwei der Aufgaben 3-5. Kennzeichnen Sie deutlich, welche Aufgabe Sie abgewählt haben!**

Als Hilfsmittel ist ein einseitig handgeschriebenes DinA4-Blatt zugelassen, das mit abzugeben ist. Die Aufgaben sind ungefähr gleich aufwändig und sind je 40 Punkte wert. Bei Unklarheiten fragen Sie den Dozenten (nicht Ihren Nachbarn!). Schreiben Sie die Lösungen auf diese Blätter, bzw. auf nummerierte leere Blätter mit Ihrem Namen; kennzeichnen Sie die Aufgabennummer eindeutig. Geben Sie alle Blätter (auch die Schmierblätter) ab: Für teilweise richtige Lösungen können Sie Teilpunkte bekommen, auch aufgrund ihrer Schmierblätter! Durch Kommunikation oder Abschreiben während der Klausur verlieren Sie die Möglichkeit der Teilpunkte.

Klausur-Punkte:

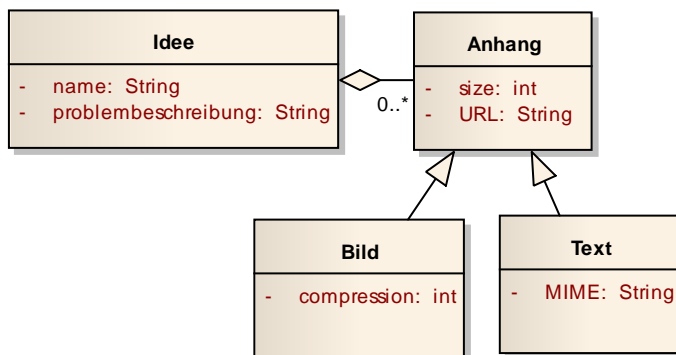
Klausur-Note:

Übungsnote:

**Gesamtnote:**

## Java-Webanwendung und JSF:

1. Erläutern sie den Unterschied zwischen client- und serverseitigem Scripting. Sollte Datenvalidierung client- oder serverseitig erfolgen? Bitte begründen Sie Ihre Auffassung.
2. Was versteht man unter AJAX? Diskutieren Sie, ob AJAX die Umsetzung eines klassischen MVC-Modells für Webanwendungen ermöglicht.
3. Zeichnen sie das MVC2-Modell für Webanwendungen! Wodurch verbessert es die Wartbarkeit und Skalierbarkeit von webanwendungen? Ordnen Sie ihrer Skizze die Begriffe JSP, Servlet und Managed Bean zu.
4. Was versteht man unter Daten-Validierung? Warum ist es grundsätzlich erforderlich, Eingabedaten zu validieren?
5. Erläutern Sie den Lebenszyklus einer JSP-Seite vom Deployment bis zum Aufruf.
6. Erläutern Sie das Prinzip der Managed Beans im JSF-Framework. Was versteht man unter einer "Managed Property"?



## ORM und Persistenz

1. Erläutern Sie die Aufgabe eines Persistenz-Frameworks (z.B. JPA). Was ist ein Persistenz-Kontext?
2. Nennen Sie 3 mögliche Transaktionszustände persistenter Objekte im Cache.
3. Was versteht man unter OR-Mapping? Kann man aufgrund der generierten Java-Klassen eines Modells automatisch die erforderlichen JPA-Annotationen generieren?
4. Nennen sie zwei Gründe, warum die Zahl der erzeugten relationalen Tabellen in einem ORM von der Zahl der Modellklassen abweichen kann.
5. Skizzieren sie ein Klassendiagramm, für das beim RM weniger relationale Tabellen erzeugt werden als es Klassen enthält, und schreiben Sie die erforderlichen Annotationen für die Klassenköpfe.
6. ... mehr ... und begründen sie, wodurch zusätzliche Tabellen entstehen.
7. JPA verfolgt die Strategie "Convention over Configuration" , d.h. es werden hinsichtlich der Gestaltung der relationalen Tabellen sinnvolle Annahmen gemacht, die aber durch Annotation oder XML-Konfiguration geändert werden können. Nennen sie 3 Beispiele.
8. Schreiben Sie für das nebenstehende Klassendiagramm JPA-annotierte Java-Klassen und skizzieren sie das dazugehörige relationale Schema mit Relationen (=Tabellen), Attributen und Fremdschlüsseln.
9. Was versteht man unter Lazy Materialization? Warum ist es bei Verwendung eines Persistenzframeworks wichtig, die Navigationsrichtungen problemgemäß zu wählen?
10. Warum müssen Collections, die als Ergebnis einer Objektsuche entstehen, in der regel nicht als Entities modelliert werden?
11. Was ist der Unterschied zwischen einer JDBC-Query und einer JPA-Query?

## Entwurfsprinzipien und Entwurfsmuster:

1. Zeichnen und erläutern Sie das Entwurfsmuster XXX am Beispiel XXX. Wozu dient es und was sind seine Vorteile?
2. Erläutern sie das Entwurfsprinzip YYYY und nennen Sie die Vorteile!

# SE2-Ubungsaufgaben

## Architektur und Web-Frameworks

1. Was versteht man unter einer Referenzarchitektur, und wozu ist sie sinnvoll? Erläutern Sie die Bedeutung von Service-Klassen in der Referenzarchitektur dieses Kurses. / Kann man die Referenzarchitektur dieses Kurses als Drei-Schichten-Modell bezeichnen? / Entspricht die Referenzarchitektur ganz oder teilweise dem MVC-Schema?
2. Erläutern Sie die Begriffe "Frontend" und "Backend" im bezug auf den von Ihnen erstellten Prototyp.
3. Erläutern Sie das Prinzip "Dependency Injection". Wozu dient es und wo kommt es in Ihrem Prototyp vor?
4. Was versteht man unter Scaffolding? Was ist Boiler-Plate-Code? Erläutern Sie, wie durch die Generierung von Code die Qualität der Software verbessert werden kann!
5. Was leisten ein "typisches" Webframework? Warum ist es sinnvoller generierten Code zu ergänzen als alles von Grund auf selbst zu programmieren? Gibt es auch Nachteile dieses Vorgehens?
6. Erläutern Sie, warum es sinnvoll sein kann, aus einem Klassenmodell zunächst ein vielfaches an Klassen zu generieren und dann den generierten Code weiter zu entwickeln.
7. Warum kann es sinnvoll sein, im Analysemodell eine Vererbungsstruktur zu modellieren und sie im Entwurfsmodell durch ein einfaches Statusattribut zu ersetzen?
8. Warum sollte man im Entwurfsmodell Navigationsrichtungen festlegen?

Viel Erfolg 😊