Dipl.-Inform. Thomas Ziemer
www.tfh-berlin.de/~ziemer
Stand: 22. Oktober 2009

0 Inhalt

OOD / OOP

OBJEKTORIENTIERTE ENTWURFSPHASE (OOD)

1 Vom Analysemodell zum Entwurf (Einführungsveranstaltung)

1.1 Vom Analysemodell zum Entwurf
1.2 Objekt und Klasse
1.3 Generische Klasse
1.4 Containerklasse
1.5 Schnittstelle
1.6 Attribut
1.7 Operation
1.8 Assoziation
1.9 Polymorphie
1.10 Mehrfachvererbung
1.11 Paket
1.12 Anforderungen an unsere Klassendiagramme (Meilenstein)

2 Vom Analysemodell zum Entwurf (Fortsetzung)

Vom Analysemodell zum Entwurf (Zweiter Teil der Einführungsveranstaltung)

1.13 Grundlegende Entwurfsentscheidungen
1.14 Architekturentwurf
1.15 Entwurf der Fachkonzeptschicht
1.16 Entwurf der Oberflächenschicht und Anbindung an die Fachkonzeptklassen

Anhang

1.17 Schritte zum OOD-Modell
1.18 Ursachen für Entwurfsrevisionen

Wir bauen uns eine Web-Applikation (Erster Teil)

2.1 Technische Voraussetzungen
2.2 Überprüfung der Installationen
2.3 Zugriff auf die Oracle-Datenbank (Erste Fingerübung)
2.4 Überprüfung der Web-Applikation

Wir bauen uns eine Web-Applikation (Zweiter Teil)

2.5 Persistente Klassen mit JPA (Zweite Fingerübung)
2.6 Die Persistenzeinheit (*persistence unit*) konfigurieren
2.7 Die Verwendung des Entity Managers
2.8 SQL-Abfragen nutzen
2.9 Überprüfung der Web-Applikation

Wir bauen uns eine Web-Applikation (Dritter Teil)

2.10 Eine Benutzungsoberfläche mit JSF (Dritte Fingerübung)
2.11 Die generelle Struktur von JSF
2.12 Die Eigenschaftsdatei (messages.properties) erstellen
2.13 Die JSP-Dateien bearbeiten
2.14 Tomcat konfigurieren
2.15 Überprüfung der Web-Applikation
2.16 Erweiterung der Funktionalität

3 Entwurfsmuster (lange Vorlesung)

Erster Teil

3.1 Entwurfsmuster
3.2 Begriffsbildung
3.3 Klassifikation
3.4 Entwurfsmuster auswählen und anwenden
3.5 Beispielhafte Entwurfsmuster (Beobachter, Kompositum, Strategie)
3.6 Weitere wichtige Entwurfsmuster im Überblick (Fassade, Dekorierer)

4 Model-View-Controller-Konzept

Entwurfsmuster (Zweiter Teil der Veranstaltung)

3.7 Weitere beispielhafte Entwurfsmuster (Fabrikmethode, Singleton, Proxy)
3.8 Weitere wichtige Entwurfsmuster im Überbl. (Schablonenmeth., Abstr. Fabrik, Interpreter)
3.9 Anforderungen an unsere Entwurfsmuster (Meilenstein)

Anhang

3.10 Entwurfsmuster für Plug-Ins
3.11 Ein Stellvertreter für Plug-Ins
3.12 Persistenz für Plug-Ins
3.13 Mehrere Implementierungen ohne Vererbung

Model-View-Controller-Konzept

4.1 Modellbildung des Model-View-Controller-Konzepts
4.2 Anforderungen an unser Model-View-Controller-Konzept (Teil des Meilensteins)

5 Datenbanken (lange Vorlesung)

Erster Teil

5.1 Datenbanksysteme
5.2 Relationale Datenbanksysteme
5.3 Abbildung des objektorientierten Modells auf Tabellen (OR-Mapping)
5.4 Objektorientierte Datenbanksysteme

Zweiter Teil

5.5 Entwurf der Datenhaltungsschicht mit relationalen Datenbanksystemen (weitere wichtige Entwurfsmuster Database-Broker, Cache, Virtual Proxy)
5.6 Entwurf der Datenhaltungsschicht mit objektorientierten Datenbanksystemen
5.7 Anforderungen an unsere Datenhaltungsschicht (Meilenstein)

Persistenz durch Entities

5.8 Persistenz durch Entities
5.9 Objektrelationale Abbildung (O/R-Mapping)

6 Wir testen eine Web-Applikation

7 Qualitätsmanagement und Qualitätssicherung in der Softwareentwicklung

7.1 Warum scheitern Softwareprojekte?
7.2 Softwarequalität
7.3 Qualitätsmanagement
7.4 Qualitätssicherung
7.5 Qualitätssicherungsmodell

Testmanagement in Softwareprojekten

7.6 Testkonzept
7.7 Testmanagement
7.8 Testplan
7.9 Testvorbereitung

Methodischer Aufbau und Durchführung von Fach- und Abnahmetest

7.10 Methodischer Aufbau und Durchführung von Fach- und Abnahmetest
7.11 Entscheidungstabellen
7.12 Testmethoden

OBJEKTORIENTIERTE IMPLEMENTIERUNG (OOP)

8 Vom Entwurfsmodell zur Implementierung

8.1 Implementierung
8.2 Strategien zur Implementierung (Top-down, Bottom-up, Hardest-first)
8.3 Prinzipien der Implementierung
8.4 Die Psychologie des Programmierens, Verhaltens- und Denkmodell
8.5 Operationalisierbare Qualitätskriterien für die Programmierung (Präsentation)
8.6 Ein weniger schönes Programmierbeispiel

Anhang

8.7 Schritte zur OOP-Phase
8.8 Ursachen typischer Programmierfehler

9 Abnahme von Softwaresystemen

9.1 Systemtest (Funktionstest, Leistungstest**s**, Benutzbarkeitstest, Sicherheitstest, Interoperabilitätstest, Installationstest, Wiederinbetriebnahmetest)
9.2 Abnahmephase
9.3 Anforderungen an unser Softwaresystem (Meilenstein)

DIE LETZTEN ETAPPEN DES SOFTWAREPROJEKTS

10 Agile Softwareentwicklung

10.1 Eine neue Sicht auf die Softwareentwicklung
10.2 Agile Softwareentwicklung
10.3 eXtreme Programming

11 Die 25 gefährlichsten Programmierfehler

11.1 Einführung
11.2 Die 25 gefährlichsten Programmierfehler

12 Abnahme unserer Softwareprojekte

13 Präsentation unserer Softwareprojekte