

Vorlesungsbeschreibung Objektorientierter Systementwurf

Lernergebnisse

Die Teilnehmer erhalten eine Einführung und Vertiefung in OO-Analyse, OO-Design, Design Muster, Design Prinzipien und UML-Modellierung. Dabei sollen die Teilnehmer nicht nur Erfahrungen mit objektorientierten SW-Plattformen sammeln, sondern auch in die Lage versetzt werden, SW-Plattformen zu analysieren und zu bewerten. Sie kennen und verstehen wesentliche SW-Architekturen und sind in der Lage, den Zusammenhang von Designentscheidungen auf den wirtschaftlichen Erfolg einer SW-Lösung zu erkennen und deren Auswirkungen zu beurteilen.

Inhalte

- Klassen (abstrakt, konkret), Attribute, Methoden, Objekte, Botschaften, Schnittstellen, Assoziationen, Kapselung u. Geheimnisprinzip
- Late Binding und Polymorphismus
- Überladen
- Überschreiben
- Einführung in UML (Klassendiagramme, Objektdiagramme, Paketdiagramme, Sequenzdiagramme)
- versch. Entwurfsmuster (Singleton, Composite, ...)
- Design Prinzipien (Single-Responsibility, Open-Close Principle, Liskovsches Substitutionsprinzip, Dependency Inversion Principle, Interface Segregation Principle, Model View Controller)
- Softwaretechnologien und Architekturprinzipien
- Middleware
- Webanwendungen
- verteilte Systeme
- nebenläufige Programmierung

Lehrmethode

Vorlesung mit gemischten Medien (Folien, Tafelarbeit, Demonstrationen, etc.), Übungen u.a. im Labor, am Computer etc.

Lehrsprache

Deutsch, Englisch

Studien-/Prüfungsleistung

Entweder eine benotete Klausur am Ende des Semesters (90 Minuten) oder eine mündliche Prüfung in Form eines Prüfungsgesprächs am Ende des Semesters. Welche der beiden Prüfungsleistungen erbracht werden muss, wird vom Dozenten am Anfang des Semesters festgelegt. Falls ein Prüfungsgespräch stattfindet, legt der Dozent dessen Dauer zu Beginn des Semesters fest. Die Dauer liegt dabei innerhalb des durch die Prüfungsordnung vorgegebenen Rahmens.

Credits

5 (150 h = 60 h Präsenz- und 90 h Eigenstudium)

Literatur

Oesterreich, B.: Objektorientierte Softwareentwicklung - Analyse und Design mit der UML 2.1, Oldenbourg Verlag, 2006.

Gamma, E.; Helm R.; Johnson, R.; Vlissedes, J.M.: Entwurfsmuster. Elemente wiederverwendbarer objektorientierter Software (Programmer's Choice), Addison-Wesley, 2004.

Coad, P.; Yourdan, E.: Objektorientierte Analyse, Englewood Cliffs, 1994.

Coad, P.; Yourdan, E.: Objektorientiertes Design, Englewood Cliffs, 1994.

Booch, G.; Rumbaugh, J.; Jacobson, I.: The Unified Modeling Language User Guide. Addison-Wesley Longman 2005.

Balzert, H.: Lehrbuch der Software-Technik (Software-Entwicklung), Spektrum Akademischer Verlag, 2000.

Balzert, H.: Lehrbuch Grundlagen der Informatik. Spektrum Akademischer Verlag, 2005.

Martin R.: Agile Software Development, Prentice Hall, 2003.

Schiedermeier, Reinhard (2010): Programmieren mit Java. 2., aktualis. Auflage. München: Pearson Studium (Pearson Studium - IT).

Ullienboom, Christian (2014): Java SE 8 Standard-Bibliothek. Das Handbuch für Java-Entwickler; 2., aktualisierte und erw. Aufl. Bonn: Galileo Press (Galileo computing).

Besonderes

Exemplarische Transferaufgaben:

Objektorientierte Modellierung fachlicher Unternehmenskonzepte als Teil einer geschäftlichen Softwareanwendung

Weiterentwicklung einer betrieblichen Software mittels eines objektorientierten Ansatzes

Alle öffnen Alle schließen