

Modulcode (1.)	Modulbezeichnung (2.)	Zuordnung (3.)
BAI4010	Programmierung Java 2 (PRGJ2)	
	Studiengang (4.)	Bachelor Angewandte Informatik/ Bachelor Angewandte Informatik DUAL
	Fakultät (5.)	Gebäudetechnik und Informatik

Modulverantwortlich (6.)	Prof. Dr.-Ing. Steffen Avemarg
Modulart (7.)	Pflicht
Angebotshäufigkeit (8.)	SS
Regelbelegung / Empf. Semester (9.)	BA4
Credits (ECTS) (10.)	5 CP
Leistungsnachweis (11.)	SL (N)
Unterrichtssprache (12.)	Deutsch
Voraussetzungen für dieses Modul (13.)	BAI3010: Programmierung Java 1
Modul ist Voraussetzung für (14.)	BAI6010: Programmierung mobiler Endgeräte
Moduldauer (15.)	1 Semester
Notwendige Anmeldung (16.)	-
Verwendbarkeit des Moduls (17.)	-

Lehrveranstaltung (18.)	Dozent/in (19.)	Art (20.)	Teilnehmer (maximal) (21.)	Anzahl Gruppen (22.)	SWS (23.)	Workload	
						Präsenz (24.)	Selbststudium (25.)
1 Programmierung Java 2	Rhöse	V	75	1	2	30	15
2 Programmierung Java 2	Rhöse	Ü	25	3	2	30	50
Summe					4	60	65
Workload für das Modul (26.)						125	

Qualifikationsziele	<p>Die Studierenden können...</p> <ul style="list-style-type: none"> • Frameworks zur Entwicklung serverseitige Java-Anwendungen benennen und einsetzen • serverseitige Anwendungen mit sinnvoller Architektur planen und implementieren • die Notwendigkeiten und Vorteile einer modularen Anwendungsstruktur benennen und praktisch umsetzen • die theoretischen Grundlagen und Herausforderungen der objekt-relationalen Abbildung verstehen sowie praktisch mit Java JDBC bzw. JPA einsetzen • die Einsatzmöglichkeiten von RESTful WebServices in einem komplexen Softwaresystems verstehen und erläutern sowie RESTful WebServices erfolgreich mit Java umsetzen • Design-Entscheidung bei der Umsetzung eines konkreten Anwendungsfalls begründen und überzeugend präsentieren • die Konzepte und Notwendigkeit von Unit-Tests erläutern sowie Unit-Test erfolgreich bei der Software-Entwicklung einsetzen • wichtige objektorientierte Entwurfsmuster benennen, erklären und im Programmcode erkennen sowie implementieren • die Notwendigkeit eines Versionskontrollsystems erläutern und ein solches praktisch einsetzen • die Vorteile des containerbasierten Deployments von serverseitigen Anwendungen benennen und haben erste Erfahrungen mit der Umsetzung gesammelt
Inhalte	<ul style="list-style-type: none"> • Java EE API (JDBC, JPA, RESTful Webservices) • Unit-Tests • Architektur von serverseitigen Anwendungen (u.a. Client/Server, Monolith, Microservices) • Containerbasiertes Deployment • Tool-Unterstützung (IDE, Versionskontrolle, Debugging) • Entwurfsmuster
Vorleistungen und Modulprüfung	<p>Die Modulprüfung besteht aus einem semesterbegleitenden Teamprojekt</p> <ul style="list-style-type: none"> • Fortsetzung des Teamprojektes aus Programmierung Java 1 • Fokus auf eine serverseitige Anwendung und deren Architektur inkl. Verwendung von Webservices, Datenbankanbindung & Containern • 3 Präsentationen im Laufe des Semesters zum aktuellen Stand des Projektes <p>Die Modulnote setzt sich wie folgt zusammen:</p> <ul style="list-style-type: none"> • 60% Sourcecode • 10% Präsentationen • 10% Unit-Tests • 10% Code-Dokumentation • 10% Allgemeine Dokumentation
Literatur	<ul style="list-style-type: none"> • Gallardo R. et.al., The Java Tutorial: A Short Course on the Basics (6th Edition), Addison-Wesley Professional 2014 / Online und als kostenloses eBook verfügbar • Evans B.J. et.al., Java in a Nutshell (6th Edition), O'Reilly 2014 • Ullenboom C., Java ist auch eine Insel: Insel 1: Das

	<p>umfassende Handbuch, Galileo Computing 2020</p> <ul style="list-style-type: none">• Ullенboom C., Java SE 9 Standard-Bibliothek, Galileo Computing 2017• Gamma, E.; Helm, R.; Johnson, R. E.: Entwurfsmuster. Elemente wiederverwendbarer objektorientierter Software, Addison Wesley, München 2004• Osherove R., The Art of Unit Testing, mitp Verlags GmbH 2015• Preißel R. & Stachmann B., Git, dpunkt.verlag GmbH 2018• Urma R.-G. et al, Modern Java in Action, Manning 2019• Bloch J., Effective Java: Best Practices für die Java-Plattform, Addison Wesley 2018• Newmann S., Vom Monolithen zu Microservices: Patterns, um bestehende Systeme Schritt für Schritt umzugestalten, O'Reilly 2020• Martin R.C., Clean Architecture, Prentice Hall 2018
--	---