

SW Eng Web: Lernziele & Leistungsnachweis

Lernziele

Nachfolgend sehen Sie die Lernziele der praktisch orientierten Vorlesung "Software-Engineering Web" (vom 16.05.2019 und dem 26.06.2019):

ID	Lernziel	Worum geht es?
QAS-01	Ich weiss, welche drei wichtigsten Arten es gibt, um meine Applikationen zu testen und kann jeweils mindestens eine Testing-Software einsetzen.	<ul style="list-style-type: none">• Für alle Tools gilt:<ul style="list-style-type: none">• Was ist die Zielgruppe?• Auf welche Art (bspw. Sprache, Grafik, etc.) definiere ich die Tests?• Wie stabil sind die Tests, die ich entwickle?• Die "Schnittstellen" von Komponenten müssen getestet werden:<ul style="list-style-type: none">• Welche Komponenten eignen sich denn dafür?• Welche Komponenten eignen sich nicht?• Ebenfalls muss das Zusammenspiel der Komponenten getestet werden:<ul style="list-style-type: none">• Gibt es dort mehrere Ebenen?• Wie stelle ich sicher, so wenig Redundanzen wie möglich zu haben?• Zudem sollte die Benutzerschnittstelle getestet werden:<ul style="list-style-type: none">• Muss ich jedes Feld testen?
ARC-01	Ich kenne den Aufbau der verwendeten Multiple-Tier-Architektur für Web-Applikationen und kann jeder Schicht eine Technologie zuordnen.	<ul style="list-style-type: none">• Wodurch kennzeichnet sich eine Schicht?• Wie viele Schichten empfehle ich für eine qualitativ gute Web-Applikationen?• Welches sind aus welchen Gründen die stabilsten Schichten, welche nicht?
ARC-02	Ich kenne die vier wichtigsten Log-Level und weiss, wann ich sie einsetze und was ich damit erreichen will.	<ul style="list-style-type: none">• Welche Levels gibt es?• Was muss ich ausgeben?• Was darf ich nicht ausgeben?• Welche Fehler bezüglich Logging gilt es zu vermeiden?

ARC-03	Ich kenne die wesentlichen Probleme von Exceptions in einem Web-Applikations-Umfeld und weiss, was ich dagegen unternehmen kann.	<ul style="list-style-type: none"> • Wie sieht es mit mehreren Usern aus? • Wie stellen sie fest, was bei kaskadierenden Aufrufen zusammengehört? • Wie sieht es an Schnittstellen zu Backends, vor allem im Hinblick auf Redundanzen aus? • Wie sieht es mit Problemen auf der Browser-Seite aus?
ARC-04	Ich kenne mindesten drei der wichtigsten Merkmale einer Web-Applikation, in denen sie sich von anderen unterscheidet, und weiss, wie sie sich konkret in der Umsetzung auswirken.	<ul style="list-style-type: none"> • Anzahl/ Art Benutzer? • Sessions? • lokal vs Netz? • volatil vs stabil?
PMG-01	Ich kann jeweils die vier wichtigsten Erfolge und Optimierungsmöglichkeiten meine Arbeiten im Projekt zeigen/ benennen.	<ul style="list-style-type: none"> • Was habe ich gemacht? • Worauf bin ich stolz? • Wo denke ich, kann ich beim nächsten mal etwas verbessern? • Wie finde ich das gesamte Produkt? • Was würde ich anders machen?
PMG-02	Ich kann Aufgaben in einem Ticket-System deren Lösungen so detailliert wie nötig und so kurz wie möglich dokumentieren.	<ul style="list-style-type: none"> • Welche drei Aspekte sollte die Beschreibung haben? • Was sollte in den Kommentaren stehen?

Bewertung

Software-Entwicklung wird immer stärker zur "Team-Sache": aus diesem Grund wird der Leistungsnachweis in dieser Vorlesung überwiegend auf dem Team-Erfolg basieren. Der Leistungsnachweis besteht aus zwei Teilen:

Praktische Arbeit

Der erste Teil ist die praktische Arbeit, die eine Note für alle Teilnehmer ergibt. Darin wird die Applikation wie folgt bewertet (EP = Extrapunkte):

Kriterium	Punkte (Gesamt)	Teil-Punkte	Beschreibung
Funktionalität	30	30 (+max 5 EP)	Abbildung der kompletten, geforderten Funktionalität gemäss Aufgabenstellung. Übertreffen der geforderten Funktionalität (Stichwort "Delivery+1").

Architektur	20	10	Die Architektur der Applikation zur Laufzeit ist ausreichend und klar beschrieben.
		10	Die verwendete Entwicklungsinfrastruktur ist ausreichend und klar beschrieben.
		(+max 5 EP)	Der Ablauf einer Suche mit dem Speichern als Favorit wird mittels geeignetem Diagramm als Laufzeitsicht dargestellt.
Code Qualität	20	5	Der Code weist ausreichend Kommentare für Klassen, Attribute und Methoden auf.
		5	Klassen, Methode, Attribute haben sinnvolle Namen.
		5	Die Quellcode-Dateien haben eine sinnvolle Grösse
		5	Die Modularisierung ist wartungsfreundlich und sinnvoll.
		(+max 5 EP)	Die Code-Qualität wird automatisiert überprüft (z. B. via Checkstyle, SonarQube).
Tests	30	10	Es existieren UI-Tests, welche die wichtigsten Funktionen an der Benutzerschnittstelle testen.
		10	Es existieren Integrationstests, welche die die Services und DB zusammenführen und testen.
		10	Es existieren Unit-Tests, welche Klassen und deren Schnittstelle ausreichend testen.
		(+max 5 EP)	Die Code Coverage wird ausgewiesen und weist mehr als 70% auf.
Dokumentation	40	10	Jedes Teil-Projekt enthält ein readme.md, die aufzeigt, was die Komponente macht, welche Besonderheiten umgesetzt werden, etc.
		30	Die Artefakte Web-Applikation, REST Service, und Datenbank (je 10) können <u>auf Basis der Dokumentation</u> ausgechecked (1), gebildet(3), gestartet(3) und getestet(3) werden.
Präsentation (ca. 20-30 min)	20	5	Jeder im Team stellt einen Teil-Aspekt der Umsetzung vor.
		5	Die Applikation funktioniert auf einem explizit gewählten Testsystem und kann gezeigt werden.
		5	Die Installation der Applikation wird vorgeführt.
		5	Fragen des Kunden werden ausreichend beantwortet (was dieser bestätigt).
Endnote	160	180	Über die Extra-Punkte können Schwächen in anderen Teilen wett gemacht werden.
		(inkl. EP)	Sollte die maximale Punktzahl übertroffen werden, bleibt es bei einer 6.0.
			(Genaue Skala wird der finalen Bewertung beigelegt)

Mündliche Prüfung

Die mündliche Prüfung orientiert sich ausschliesslich an den im Unterricht behandelten oder in der Applikation umzusetzenden Inhalten - es wird also keine Überraschungen geben.

Jeder Teilnehmer kann sich hierfür an den oben aufgeführten Lernzielen orientieren: neben den beiden Fragen zur Reflexion (PMG-01 und PMG-02) wird es je zwei Fragen zu den weiteren, o.g. Lernzielen geben.

Endnote

Die Endnote wird zu je 50% aus der gemeinsamen praktischen Arbeit und der individuellen mündlichen Prüfung gebildet.