

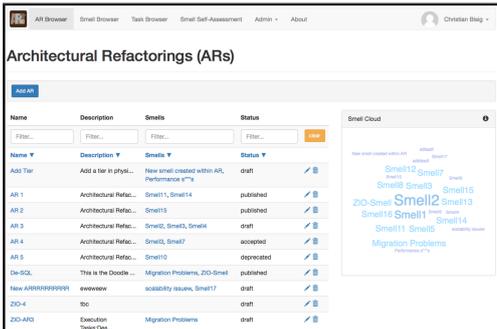


Christian Bisig

Studenten/-innen	Christian Bisig
Dozenten/-innen	Prof. Dr. Olaf Zimmermann
Co-Betreuer/-innen	Prof. Dr. Olaf Zimmermann
Themengebiet	Software and Systems

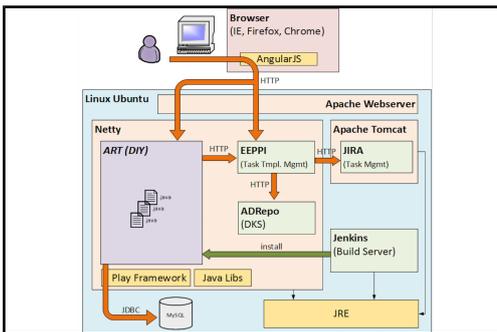
Ein Werkzeug für Architectural Refactoring

Design und Implementierung



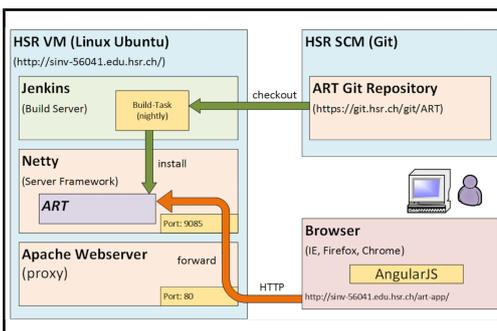
Screenshot des Architectural Refactoring Tools Version 1.0 (Architectural Refactoring Browser inkl. Smell-Tag Cloud)

Einleitung: Beim Bau neuer Software wird im Rahmen des Software Engineering oft auf bewährte Methoden wie Software Design Patterns oder Blueprints zurückgegriffen. Die Literatur bietet eine breite Unterstützung bei der Entwicklung neuer Systeme. Dieses Wissen ist weit verbreitet, und Patterns wie jene der Gang Of Four gelten als Quasi-Standards. Weniger geläufig sind dagegen Methoden zum Umbau von bestehenden Systemen. Ein Umbau kann nötig sein, wenn Leistung und Qualität eines aktuellen Systems nicht mehr den Erwartungen entsprechen. Die Behebung eines Problems soll vorzugsweise auf dem bestehenden System aufbauen und keine Neuimplementierung erfordern (aus Kosten- und Kompatibilitätsgründen). An dieser Stelle kommen Architectural Refactorings ins Spiel. Diese beschreiben Softwarearchitekturumbauten, bei denen ein Architekturpattern in ein anderes überführt wird. Für den Umbau von bestehenden Softwaresystemen mit Hilfe von Architectural Refactorings fehlt jedoch ein Werkzeug, um Wegleitungen und Hilfestellungen für ein Projekt zu finden.



Architekturübersicht

Ziel der Arbeit: Das Ziel dieser Projektarbeit ist die Implementierung des in der Projektarbeit 1 erarbeiteten Konzepts und Designs für ein Architekturumbau-Werkzeug. Das Architectural Refactoring Tool soll mit den evaluierten Techniken, Werkzeugen und Programmiersprachen implementiert werden, so dass das Werkzeug das Wissen über Architekturumbauten bestmöglich zentral festhalten und präsentieren kann. Die Benutzer sollen entsprechend Ihrer Rollen die Applikation bewirtschaften und benutzen können. Die Implementation erfolgt in mehreren Iterationen und mit fortlaufendem Benutzerfeedback durch den Advisor.



Übersicht über die automatische Softwareausbreitung

Ergebnis: Das Resultat der Projektarbeit enthält die Umsetzungsschritte Risikoanalyse, Anforderungsüberarbeitung, Designvertiefung, Implementation, Testing, Ausbreitung, Benutzerhandbuch sowie auch eine Liste von Anforderungen und Änderungen, welche in der abschliessenden Masterarbeit umzusetzen sind. Die Implementation des Architectural Refactoring Tools basiert, wie im Dokument der Projektarbeit 1 beschrieben, auf einer webbasierten Dreischichten-Architektur, welche mit dem Play Java Framework und MySQL als Backend sowie AngularJS für das Client Frontend implementiert ist. Die Applikation läuft mit dem Play-eigenen Webserver-Framework Netty.