

Automatisierte Erzeugung konkreter Testfälle für Webanwendungen aus einem textbasierten Modell

Arne-Michael Törsel

Arne-Michael.Toersel@fh-stralsund.de

Gerold Blakowski

Gerold.Blakowski@fh-stralsund.de

Der Einsatz automatisierter Testfallsuiten für den Blackboxtest von Webanwendungen ist im Unternehmensumfeld üblich. Allerdings ist der Aufwand zur Wartung solcher Testfallsuiten gewöhnlich sehr hoch. Modellbasiertes Testen verspricht Effizienzvorteile bei der Testfallerstellung und -wartung unter der Annahme, dass die Ableitung der Testfälle aus dem Modell und ihre Konkretisierung quasi „kostenlos“ möglich sind. Bisher vorgestellte Ansätze beschreiben geeignete Modelle und Verfahren zur Ableitung abstrakter Testfälle für Webanwendungen. Offen ist jedoch die automatisierte Übertragung in konkrete, mit Testwerkzeugen ausführbare Testfälle. Dies schließt aus dem Modell generierte und automatisiert konkretisierbare Testorakel ein.

Im Beitrag wird dazu ein textbasierter Modellansatz vorgeschlagen. Die Webanwendung wird durch das Modell logisch als zusammenhängende, gerichtete Graphstruktur abgebildet. Knoten repräsentieren „Sichten“ der Anwendung und Kanten Übergänge zu anderen Sichten, die durch Nutzereingaben ausgelöst werden. Variablen können zur Modellierung von globalen Zustandsinformationen, Datenflüssen und zur Formulierung von Constraints für Übergänge genutzt werden. Aus den Anwendungsmodellen leitet ein Testfallgenerator zunächst abstrakte Testfälle ab, die anschließend in ausführbare Testskripte transformiert und konkretisiert werden. Der Testtreiber ist somit prinzipiell austauschbar. Wegen der Abbildung von Daten und Datenflüssen im Modell ist es möglich, bei der Testfallableitung automatisiert Testorakel zu generieren. Bei der Konkretisierung der Testfälle werden die Testorakel in Textmatchingoperationen auf dem HTML-Quelltext der Anwendungsoberfläche umgesetzt.

Der vorgeschlagene Ansatz wurde in einem Prototyp umgesetzt. Dazu wurden die Programmiersprache Java und mehrere quelloffenen Frameworks genutzt (zum Beispiel open-ArchitectureWare Xtext für das Anwendungs-Metamodell, Canoo Webtest als Testwerkzeug). Die Minimierbarkeit des erforderlichen Interventionsaufwand seitens des Testentwicklers von der Testfallgenerierung bis hin zu konkreten, ausführbaren Testfällen konnte demonstriert werden. In Erprobungen des Prototyps zeigte sich eine Eignung des Ansatzes vor allem für Testszenarien mit einer hohen „Sichtbarkeit“ der Testauswirkung auf der Nutzeroberfläche, zum Beispiel Szenarien, die auf Datenein- und korrespondierender Datenausgabe beruhen. In der weiterführenden Entwicklung sollen das Metamodell und der Testfallgenerator erweitert werden, um zum Beispiel die automatisierte Erzeugung von Negativtestfällen sowie komplexeren Positivtestszenarien zu ermöglichen.