

Themen für 2010

Diplom-/Master-/Bachelor-Arbeiten

bei

Sogeti Deutschland GmbH

Datum des Dokuments: 10. Juli 2009
Version: 1.0
Status des Dokuments: abgestimmt
Autoren: Dr. Matthias Hamburg

1 TPI Rückwärtskompatibilität

1.1 Themen

Testmanagement, Prozessreifegradmodelle

1.2 Einstufung

Einstufung: Diplom-/ Bachelor-Arbeit

1.3 Aufgabe

TPI[®] ist ein Reifegradmodell für Software-Testprozesse. Das Akronym ist die Abkürzung von Test Process Improvement. Die erste Version wurde in 1997 publiziert (hier einfach TPI genannt), die zweite Version wird in Q4 2009 veröffentlicht und hat die Versionsbezeichnung BD TPI (Business Driven Test Process Improvement).

Aufgabe der Arbeit ist, ein Verfahren zu entwickeln, mit dem Ergebnisse der Testprozessbewertung mit der neuen Version BD TPI mit denjenigen von früheren TPI-Ergebnissen der ersten Version verglichen werden.

Die Arbeit soll klassifizieren, welche Aspekte von BD TPI formal aus einem früheren TPI Ergebnis ableitbar sind, welche nur informell, und welche wiederum nicht ableitbar sind. Dazu ist zunächst zu entscheiden, in wie weit eine Formalisierung der Modelle sinnvoll ist, und anschließend eine geeignete formale Darstellung der Modelle auszuarbeiten.

An Hand der formal definierten Elemente beider Modelle soll ein Modell (ggf. partiell) auf das andere reduziert werden. Dabei ist die Äquivalenz beider Modelle oder Teile der Modelle zu beweisen oder zu widerlegen.

Das Ergebnis der Bewertung der Testprozessreife wird in beiden Varianten formal in einer Testreifematrix dargestellt. Die Matrizen der zwei Versionen sind zwar vom Aufbau her ähnlich, aber nicht identisch. Der formale Teil der Aufgabe wird in erster Linie die Analyse der Beziehungen zwischen den zwei Reifegradmatrizen enthalten, und der Spezifikation eines formalen Übersetzungsverfahrens der TPI-Matrix in eine BD-TPI-Matrix. Die formale Transformation einer Reifegradmatrix Version 1 nach Version 2 (liegen jeweils im Excel-Format vor) soll entsprechend der Spezifikation für Matrizen im TPI Scoring Tool implementiert werden.

Der informelle Teil der Aufgabe enthält die Erarbeitung eines Verfahrens, mit dem die neuen Elemente der Modellversion zwei aus Bewertungsberichten (Assessments) der ersten Version abgeleitet werden. Neue Elemente sind zum Beispiel die Business Driver und die variable Clusterung von Kontrollpunkten.

2 TPI und ISO/IEC 15504

2.1 Themen

Testmanagement, Prozessreifegradmodelle

2.2 Einstufung

Einstufung: Diplom-/ Bachelor-Arbeit

2.3 Aufgabe

TPI[®] ist ein Reifegradmodell für Software-Testprozesse. Das Akronym ist die Abkürzung von Test Process Improvement. Die aktuelle Version wird in Q4 2009 veröffentlicht und hat die Versionsbezeichnung BD TPI (Business Driven Test Process Improvement).

Die ISO/IEC 15504 (auch SPICE genannt) ist ein Reifegradmodell für den gesamten Software-Entwicklungsprozess, das auch Prozessattribute und Aktivitäten des Testens berücksichtigt.

Aufgabe der Arbeit ist, die Beziehungen zwischen den Elementen von TPI und SPICE zu analysieren, und zu bewerten, wie das Erreichen einer Reifestufe in SPICE durch den Einsatz von TPI unterstützt werden kann.

SPICE ist ein Stufenmodell und stellt die Aktivitäten des Testens nur auf hoher Ebene dar. Dies führt dazu, dass die Kenntnis des Ziels der Optimierung (Erreichen der nächsten Stufe) den Unternehmen keine Hilfe bietet, wie sie dieses Ziel erreichen können. Hier kann TPI sinnvoll eingesetzt werden. TPI ist ein kontinuierliches Modell, hat einen wesentlich höheren Detaillierungsgrad und bietet eine Reihe von Optimierungsvorschlägen.

2.4 Betreuung durch Sogeti

Sogeti stellt dem Studenten Literatur, Beispielmaterial aus der Praxis und Beratung durch erfahrene TPI Assessoren zur Verfügung.

3 Modellbasierter funktionaler Abnahmetest mit COVER

3.1 Themen

Testentwurf, Testmethodik, modellbasiertes Testen

3.2 Einstufung

Einstufung: Diplom-/ Master-Arbeit

3.3 Aufgabe

Testbasis eines Abnahmetests sind die funktionalen Anforderungen des Auftraggebers der Softwareentwicklung. Diese Anforderungen werden nicht in Modellen sondern meist in Textform von Anwendungsfällen dokumentiert. In vielen Fällen wird der natürlichsprachliche Text strukturiert und semiformalen Regeln unterworfen, damit die wesentlichen Elemente erkennbar sind.

Der manuelle Entwurf von funktionalen Testfällen auf dieser Basis kann im Testprozess einen erheblichen Aufwand beanspruchen.

Modellbasierte Testverfahren sehen die Entwicklung eines Verhaltensmodells des Testobjekts vor. Aus diesem Modell werden anschließend durch ein MBT-Werkzeug Testfälle generiert, die eine vorgegebenes Testabdeckungsmaß des Modells erreichen.

Aufgabe der Arbeit ist, den Testentwurf durchgängig zu automatisieren. Hierzu ist ein Parser zu entwickeln, der strukturierte natürlichsprachliche Anwendungsfallbeschreibungen (die der Vorgabe einer Spezifikationsrichtlinie für Anforderungen entsprechen) in ein Verhaltensmodell (z.B. UML Aktivitätendiagramm) automatisch umsetzt. Dieses Verhaltensmodell soll alle relevanten Entscheidungspunkte, Entscheidungen, die beteiligten Attribute, und Pfade im Anwendungsfall darstellen. Die vollständige Abdeckung der Elemente der Spezifikation muss konstruktiv nachgewiesen werden. Der Parser soll gewisse noch festzulegende Anforderungen an die Benutzbarkeit und Robustheit (gegenüber Abweichungen von der Spezifikationsrichtlinie) erfüllen.

Das generierte Verhaltensmodell soll anschließend an das MBT-Werkzeug COVER übergeben werden, um nach verschiedenen Testverfahren nach TMap (z.B. Geschäftsprozess-test oder Elementarer Vergleichstest) logische Testfälle zu generieren. Dabei ist der Parser mit COVER so zu integrieren, dass zu jedem generierten Testfall die abgedeckten Elemente der ursprünglichen Anforderungsspezifikation rückverfolgbar sind.

Optional kann das Verfahren um den nächsten Schritt ergänzt werden, um die so generierten logischen Testfälle in das Testframework zur automatischen Testdurchführung zu integrieren.