

Tutorium: Mit Fehlerkennzahlen Softwareentwicklungsprojekte erfolgreich steuern

Oral Avci

SQS Software Quality Systems AG
Stollwerckstr. 11
51149 Köln
oral.avci@sqqs.de

Abstract: Fehler in der Softwareentwicklung werden nach ihrer Entdeckung meist nur korrigiert, ohne aus ihnen zu lernen. Es ist folglich nicht unwahrscheinlich, dass vergleichbare Fehler wieder auftreten, so lange die Ursachen im Prozess fortbestehen. Wie aber kann man in Softwareentwicklungsprojekten aus Fehlern lernen und Verbesserungspotenziale im Prozess aufdecken? Das gleichnamige Tutorium auf der Fachtagung Software-Management 2008 stellt ein Verfahren vor, um aus Fehlern Prozessverbesserungen ableiten zu können. Dieser Beitrag gibt einen Überblick über die Inhalte dieses Tutoriums.

1 Einführung

Fehler sind in Softwareentwicklungsprojekten allgegenwärtig. Unter einem Fehler wird ein unzureichendes Merkmal oder ein erwartetes, jedoch fehlendes Merkmal eines Arbeitsergebnisses der Softwareentwicklung verstanden, sofern es eine Änderung in diesem Ergebnis notwendig macht. Mängel im Prozess der Softwareentwicklung verursachen häufig viele Fehler in Arbeitsergebnissen (Fachkonzept, Softwarecode etc.). Je später ein Fehler entdeckt wird, desto mehr Folgefehler entstehen und folglich steigt der Aufwand für die Fehlerkorrektur.

Wie aber kann man in Softwareentwicklungsprojekten aus Fehlern lernen und Verbesserungspotenziale im Prozess aufdecken? Im Tutorium „Mit Fehlerkennzahlen Softwareentwicklungsprojekte erfolgreich steuern“ auf der Fachtagung Software-Management 2008 wird ein Verfahren vorgestellt, um aus Fehlern Prozessverbesserungen ableiten zu können. Es wird beschrieben, wie Fehler sinnvoll klassifiziert und ausgewertet werden können, um Prozessmängel als Ursachen der Fehler aufdecken zu können. Neben dem Verfahren werden praktischen Erfahrungen aus einer Fallstudie aufgezeigt.

Der vorliegende Beitrag gibt einen Überblick über die Inhalte dieses Tutoriums. Er ist wie folgt aufgebaut: In Kapitel 2 wird die Motivation für das Fehleranalyseverfahren beschrieben. Kapitel 3 beschreibt die wesentlichen Merkmale des Fehleranalyseverfahrens. Der Nutzen des Verfahrens wird in Kapitel 4 dargestellt. Abschließend wird in Kapitel 5 eine Fallstudie vorgestellt, bei der praktische Erfahrungen bei der Anwendung des Verfahrens gewonnen werden konnte. Eine detaillierte Beschreibung des Verfahrens und der Fallstudie findet sich in [Av08].

2 Das Problem: Fehler in Softwareentwicklungsprojekten

In Softwareentwicklungsprojekten entstehen viele Fehler in Arbeitsergebnissen wie beispielsweise dem Fachkonzept oder dem Softwarecode. Diese Erkenntnis ist zunächst nicht überraschend. Interessant ist jedoch, dass verschiedene empirische Untersuchungen seit Mitte der siebziger Jahre bis heute feststellen, dass ein Großteil der Fehler, nämlich 15% bis 50%, durch die Anforderungsanalyse verursacht werden [AW05]. Hierbei handelt es sich zum einen um Anforderungsfehler und zum anderen um Folgefehler im Entwurf und der Implementierung. Die Untersuchungen zeigen zudem auf, dass es sich bei den Anforderungsfehlern häufig um übersehene oder falsch verstandene Anforderungen handelt, also Fehler, die meist erst sehr spät entdeckt werden. Je später aber ein Fehler entdeckt wird, desto mehr Folgefehler können entstehen und desto aufwändiger wird die Korrektur. Die Kosten für Überarbeitungen können bis zu 41% der gesamten Projektkosten ausmachen [Di93]. Die Auswirkungen dieser erforderlichen Überarbeitungen sind vertraut: Software wird zu spät ausgeliefert und Budgets werden überschritten. Werden hingegen die Überarbeitungen nicht durchgeführt, erhalten die Kunden eine Software minderer Qualität.

Fehler finden Eingang in Arbeitsergebnisse, weil sie durch den Prozess, also der Art und Weise wie eine Aufgabe durchgeführt wird, begünstigt werden. Obwohl Fehler relevante Informationen über den Prozess geben können, werden diese nicht genutzt. Selbst dann nicht, wenn eine Fehlerverfolgung gestützt durch ein Softwarewerkzeug praktiziert wird (meist bezeichnet als „bug tracking tool“ oder „defect tracking tool“). Entsprechende Werkzeuge unterscheiden in der Regel nicht zwischen Fehlern und ihren Symptomen, z. B. dem Fehlverhalten der Software. Außerdem legen sie nahe, ausschließlich rein administrative Informationen zu erfassen: Wann ist das Fehlverhalten aufgetreten? Wurde der Fehler bereits behoben? Wer ist für die Behebung verantwortlich? etc. Beschränkt man sich jedoch auf diese administrativen Informationen, versäumt man es, aus Fehlern zu lernen. Fehler werden dann nur behoben und geraten anschließend schnell wieder in Vergessenheit. Eine nachhaltige Prozessverbesserung bleibt aus.

3 Die Lösung: Prozessrelevante Informationen zu Fehlern erfassen und auswerten

Aus Fehlern lernen heißt, die Ursachen dieser Fehler im Prozess zu identifizieren. Ist ein Mangel im Prozess bekannt und verstanden, kann er ggf. behoben werden. In diesem Fall werden zukünftig vergleichbare Fehler vermieden. Sofern ein Mangel im Prozess nicht behoben werden kann, ist die Kenntnis des Prozessmangels trotzdem wertvoll, um den Prozess der Qualitätssicherung anzupassen. So kann nach Fehlern gezielt gesucht werden, die durch den Prozessmangel verursacht worden sind. Entsprechende Fehler werden folglich früher gefunden und weitere Folgefehler vermieden.

Wie kann man nun aus Fehlern lernen? Hierzu wurde durch den Autor ursprünglich am Seminar für Wirtschaftsinformatik und Systementwicklung an der Universität zu Köln ausgehend von der Orthogonal Defect Classification [Ch96] ein Verfahren entwickelt, das auf Fehlerkennzahlen aufbaut und im Wesentlichen aus drei Schritten besteht.

Schritt 1: Fehlererfassung

Im Kern des Verfahrens steht ein etabliertes Klassifikationsschema für Fehler im Entwurf und der Implementierung, das Anfang der neunziger Jahre durch Ram Chillarege entwickelt und seither verfeinert worden ist. Anhand dieses Klassifikationsschemas werden die Fehler erfasst. Im Unterschied zu anderen Klassifikationsschemata, die in der Literatur oder in Fehlerverfolgungstools vorgeschlagen werden, extrahiert es prozess- und produktrelevante Informationen aus einem Fehler. So werden beispielsweise für jeden Fehler folgende Fragen beantwortet:

- Welches strukturelle Merkmal wurde geändert, um den Fehler zu beheben?
- Wie wurde der Fehler behoben? Indem etwas eingefügt, etwas gelöscht oder etwas geändert worden ist?
- Welches Prüf- oder Testkriterium hat zur Entdeckung des Fehlers geführt?

Schritt 2: Identifikation von Fehlermustern mit Handlungsbedarf

Unter wirtschaftlichen Gesichtspunkten ist es nicht sinnvoll, eine Ursachenanalyse für jeden Fehler einzeln vorzunehmen. Die semantischen Informationen zu den erfassten Fehlern aus dem ersten Schritt erlauben durch einfache ein- und mehrdimensionale Auswertungen Fehlermuster zu identifizieren. Ein Fehlermuster ist eine Menge von Fehlern mit gemeinsamen Merkmalen. Den Fehlern eines Fehlermusters liegen mit großer Wahrscheinlichkeit die gleichen Ursachen zugrunde. Im Rahmen einer Arbeitssitzung werden besonders auffällige Fehlermuster mit ausgewählten Projektbeteiligten analysiert. Diese Arbeitssitzung muss durch eine Person, die das Verfahren beherrscht, moderiert werden. Ziel dieser Arbeitssitzung ist es, dass die Projektbeteiligten festlegen, für welche Fehlermuster Maßnahmen eingeleitet werden sollen.

Schritt 3: Aufdecken der Ursachen der identifizierten Fehlermuster

In einer weiteren moderierten Arbeitssitzung bestimmen Projektmitarbeiter die Ursachen von ein bis zu maximal drei ausgewählten Fehlermustern. Der Moderator konkretisiert hierzu zunächst jeweils die Fehlermuster und stellt exemplarisch einige Fehlerberichte zur Diskussion.

Genauso wie ein Arzt auf der Grundlage festgestellter Krankheitssymptome, die Ursachen für die Krankheit eines Patienten erkennen und schließlich behandeln will, können die Fehlermuster als Symptome aufgefasst werden, die Rückschlüsse über Mängel im Prozess erlauben [Ca98]. Daran schließen Überlegungen an, wie ein Fehlermuster zukünftig vermieden oder entsprechende Fehler früher gefunden werden können. Da die Prozessmängel mit dem Verfahren präzise konkretisiert werden, erleichtert dies eine effektive und effiziente Entwicklung von wirksamen Verbesserungsmaßnahmen mit Projektbeteiligten. Das Ergebnis sind Verbesserungsmaßnahmen, die unternehmensspezifische und projektspezifische Besonderheiten berücksichtigen und in der Summe den Prozess nachhaltig verbessern.

4 Nutzen für Praktiker

Eine Prozessanalyse auf der Grundlage von Fehlerkennzahlen ermöglicht eine leichte, systematische, kontinuierliche Verbesserung der Prozesse. Das Verfahren fördert auf der Grundlage quantitativer Daten eine sachliche Diskussion mit Beteiligten über die Prozesse. Da zudem Fehler bereits während einer Softwareentwicklung erfasst und ausgewertet werden können, eröffnet sich die Möglichkeit, das beschriebene Verfahren begleitend zur Entwicklung einzusetzen. So können bereits während der Laufzeit eines Projekts Mängel im Prozess aufgedeckt werden. Einem Entwicklungs- bzw. Projektleiter liefert das Verfahren Informationen über den Stand des Entwicklungsprozesses und des Softwareproduktes. Einem Testmanager zeigt es auf, wie erfolgreich die Prüf- und Testaktivitäten sind. Ein Prozessverantwortlicher kann durch den Vergleich von historischen mit aktuellen Fehlerdaten untersuchen, ob Änderungen im Prozess zu den gewünschten Verbesserungen geführt haben.

5 Die Fallstudie: Anwendung des Verfahrens in einem großen Softwareentwicklungsprojekt

Das Verfahren wurde im Rahmen einer Fallstudie in einem realen Softwareentwicklungsvorhaben bei dem IT-Dienstleister einer großen deutschen Versicherung angewandt. Es wurden nachträglich reale Fehler von zwei Releaseentwicklungen zu einer geschäftskritischen Anwendungssoftware für Lebensversicherungen erfasst und analysiert, um Mängel im Prozess der Anforderungsanalyse dieses Entwicklungsvorhabens aufzudecken. Im Tutorium werden praktische Erfahrungen und Beispiele aus dieser erfolgreichen Anwendung des Verfahrens vorgestellt.

6 Kurzüberblick über die Inhalte des Tutoriums

Im Tutorial wird ein Kennzahlensystem zur Steuerung laufender Softwareentwicklungsprojekte vorgestellt. Auf der Grundlage von Fallbeispielen wird den Teilnehmern des Workshops aufgezeigt, wie Fehlerkennzahlen die Beantwortung folgender Fragen bei laufenden Projekten unterstützen:

- Ist die Software bzw. ein Zwischenstand der Software reif für die nächste Teststufe?
- Ist die Software reif für die Auslieferung?
- Sind zusätzliche Maßnahmen der Qualitätssicherung (Prüfen und Testen) erforderlich? Wenn ja, welche?
- Was kann im Entwicklungsprozess geändert werden, damit Fehler mit bestimmten Merkmalen vermieden werden?
- Was kann im Testprozess geändert werden, damit entstandene Fehler mit bestimmten Merkmalen früher entdeckt werden?
- Wie weit fortgeschritten sind die Tests?

Ziel des Tutorials ist, nach einer kurzen Einleitung, vor allem der „angeleitete“ Erfahrungsaustausch der Teilnehmer, um neue – praxiserprobte - Aspekte für die eigene Arbeit zu erhalten. Zielgruppen sind Testmanager, QS-Beauftragte, Qualitätsmanager und Personen, die Prozesse durch Kennzahlen überwachen und steuern.

Literaturverzeichnis

- [Av08] Avci, O.: Aus Fehlern in der Softwareentwicklung lernen. Anforderungsanalyse und Qualitätssicherung mit Fehleranalysen verbessern. Saarbrücken 2008. VDM Verlag Dr. Müller, ISBN-10: 3836476878, ISBN-13: 9783836476874.
- [AW05] Avci, O; Wagner, H.: Das chronische Problem der Anforderungsanalyse und die Frage: Fehler vermeiden oder früh entdecken? In: Cremers, A.B.; Manthey, R.; Martini, P.; Steinhage, V. (Hrsg.): Informatik 2005 - Informatik LIVE, Band 2, Lecture Notes in Informatics (LNI), Vol. P-68. Bonn 2005, S. 279-283.
- [Ca98] Card, D.N.: Learning from Our Mistakes with Defect Causal Analysis. In: IEEE Software. Nr. 1, 1998, S. 56-63.
- [Ch96] Chillarege, R.: Orthogonal Defect Classification. In: Lyu, M. R.: Handbook of Software Reliability Engineering. Los Alamitos (California) u. a. 1996, S. 359-400.
- [Di93] Dion, R.: Process Improvement and the corporate balance sheet. In: IEEE Software. Nr. 4, 1993, S. 28-35.