Vorlesung „Softwaretechnik“

Manuelle Prüfmethoden

Dr. Lutz Prechelt

Manuelle Prüfmethoden

Verfahren der analytischen Qualitätssicherung:

1. Arbeitsergebnisse werden gründlich auf Mängel hin durchgesehen
   - Mängel können sein:
     - Defekte
     - Verletzungen von Standards
     - Verbesserungswürdige Lösungen
2. Mängel werden zusammengetragen
3. Mängel werden beseitigt
   - evtl. nicht alle
4. Nach Korrektur evtl. erneute Prüfung

Beispiel: Kleine Codedurchsicht

- Entwickler A hat 4 zusammengehörige Klassen fertig gestellt
  - Und erfolgreich übersetzt
- Er sendet Entwickler B eine Email, mit der Bitte, die 4 Klassen (insgesamt 600 Zeilen Code) zu begutachten
  - B kennt die Anforderungs- und Entwurfsdokumente, aus denen sich ergibt, was die Klassen leisten sollten
- B nimmt sich dafür 3 Stunden Zeit
- B meldet entdeckte Mängel per Email an A zurück:
  - 2 vergessene Funktionen
  - 2 Zweifel an Bedeutung von Anforderungen
  - 4 Fehler in Steuerlogik
  - 5 übersehene Fehlerfälle
  - 4 Vorschläge zur Verbesserung der Robustheit
  - 3 Verstöße gegen Entwurfs-/Kodier-/Kommentierrichtlinien

Begriffe

- Manuelle Prüfmethoden laufen unter vielen verschiedenen Bezeichnungen:
  - Durchsicht (review), Inspektion, Durchgang (walkthrough), Formelle Technische Durchsicht (FTR) und anderen
- Bei manchen Leuten haben manche oder alle dieser Begriffe eine recht genau festgelegte Bedeutung
- Leider nicht überall die gleiche

- Vorschlag: Meistens von „Durchsicht“ sprechen
  - Und bei Bedarf präzisieren, was man damit meint

Arten und Freiheitsgrade

- Art und Umfang des Dokuments
  - Für alle Arten von Zwischen- und Endprodukten geeignet:
    - Anforderungsbeschreibungen (von vage bis mathematisch)
    - Entwurfsdokumente (von grob bis detailliert)
    - Sonstige Dokumentation
    - Code (von einzelnen Methoden bis zu großen Systemen)
    - Daten (Konfigurationsdaten, Steuerungstabellen, ...)
    - Testpläne, Testdatensätze
    - Etc.
- Anzahl Beteiligter
    - Nur Dokumentautor
    - 1 Fremder
    - mehrere Fremde
    - Jeder mit gleicher Rolle oder mit verschiedenen Rollen

Arten und Freiheitsgrade (2)

- Formalisierungsgrad
  - Von:
    - Völlig informell, nirgends geplant oder protokolliert, kein festgelegtes Verfahren oder Ergebnis
  - Bis:
    - Hochstrukturiert in festgelegten Teilschritten, mit mehreren Treffen, Protokollen, mit Planung und Endbericht
- Form des Ergebnisses
  - Dokumentationsgrad:
    - Mündliche Weitergabe, Notizzettel, Computerprotokoll
  - Strukturgrad:
    - Freitextkommentare oder Einordnung in festes Problemartenschema
  - Bewertungsgrad:
    - Alle Probleme gleich gewichtet oder klare Prioritäten

Fagan-Inspektionen

- Bekanntestes Verfahren
- Vorgeschlagen von Michael Fagan (IBM Ingenieur, ehemals im HW-Bereich) im Jahr 1976
- Ziel: Möglichst viele Schwächen im Dokument finden
- Sehr aufwändiges Vorgehen:
  1. Planung: Team zusammenstellen (Autor, Moderator, Schriftführer, mehrere Gutachter)
  2. Einführungstreffen: Autor und Moderator erklären Produkt und Inspektionsziele
  3. Lesephase: Gutachter sehen Produkt durch und machen sich damit vertraut
  4. Inspektionstreffen: Team sucht unter der Leitung des Moderators nach Mängeln, Schriftführer protokolliert Ergebnisse, Autor fragt ggf. nach Klarstellung
  5. Autor korrigiert Produkt, Moderator kontrolliert

Fagan-Inspektionen: Kritik

- Verfahren ist übertragen von Durchsicht von elektronischen Schaltungen
  - Schaltungen weniger klar beurteilbar als Software
  - Für Software weniger angemessen
- Fagan-Inspektionen sind recht effektiv, aber wenig effizient
  - Lesephase sollte der Mängelsuche dienen
  - Inspektionstreffen „unterdrückt" manchmal gefundene Defekte in der Diskussion
  - Hoher Verwaltungsaufwand
  - Sinnvollste Zahl von Gutachtern unklar
    - sinkender Grenznutzen durch Überlappung

Selbst-Begutachtung

- Anderes Extrem: Der Autor sieht sein Produkt allein durch
- Kann sehr effektiv sein
  - Reales Beispiel:
    - 4 Monate Arbeit, 2 Tage Durchsicht,
    - 45% Defekt-Entdeckungsquote,
    - Rest durch 2 Monate Test
  - Effektiv vor allem, wenn typische persönliche Fehlerarten bekannt sind
  - Und wenn die Begutachtungsgeschwindigkeit empirisch optimiert wird
- Nutzen hängt stark von Einstellung ab
  - „Ich will viele Defekte aufdecken"
- Ein paar Tage Abstand sind empfehlenswert

Perspektiven-basiertes Lesen

- Besonders für frühe Phasen geeignet
  - Anforderungsdokumente
  - Architektur, Grobentwurf, GUI-Entwurf
- Es gibt mehrere Gutachter
- Jeder analysiert das Dokument aus einer anderen Perspektive
  - Endbenutzer (evtl. verschiedene Benutzergruppen)
  - Entwerfer, Implementierer
  - Tester
  - Etc.
- Zu jeder Perspektive gehören andere Fragen und Schwerpunkte und andere Arten von möglichen Mängeln
  - Jeder Gutachter bekommt eine Beschreibung, die seine Perspektive erklärt
- Erhöht Effektivität durch Senken der Überlappung

Empirische Resultate

- Codedurchsichten finden mehr Defekte pro Stunde als Test
- Codedurchsichten finden andere Defekte als Test
- Wichtigste Steuergröße: Begutachtungsgeschwindigkeit
- Fagan-Codeinspektionen sind ineffizient
  - große Überschneidung der Resultate verschiedener Gutachter
  - Kein Mehrwert durch Treffen: Separate Durchsicht reicht
  - Lange Laufzeit (Kalendertage): Prozessverzögerung
- Auch Selbstbegutachtung kann sehr effektiv sein
  - Verlangt jedoch den Willen und die Geduld
- Durchsichten in frühen Phasen sparen besonders viel Kosten
- Perspektiven-basiertes Lesen:
  - Effektiver als wenn jeder alles sucht
  - Effizient durch hohe Gesamtausbeute

Danke!