Aspekte der Projektdurchführung

Dozent: Prof. Dr. Michael Eichberg
Kontakt: michael.eichberg@dhbw.de
Version: 1.0.1

Folien: [HTML] https://delors.github.io/lab-aspekte-der-projektdurchfuehrung/folien.de.rst.html
[PDF] https://delors.github.io/lab-aspekte-der-projektdurchfuehrung/folien.de.rst.html.pdf
Fehler melden: https://github.com/Delors/delors.github.io/issues

Übersicht

Aspekte, die im Rahmen der Durchführung des Lehrprojektes von besonderer Bedeutung sind:

- Risikomanagement
- Architektur
- Qualitätssicherung
- Build-Prozess

1. Risikomanagement

Kategorien von Risiken (hier)

Projektrisiken: Risiken, die den Projektfortschritt/-plan betreffen.
Beispiele:

- Ausfall von Personal
- Ausfall der Server(-infrastruktur)/des GIT Server
- ...

Produktrisiken: Risiken, die die Qualität/Performance des Produktes betreffen.
Beispiele:

- Anforderungen sind unvollständig, nicht-richtig erfasst oder ändern sich
- Spezifikation verzögert
- Eingesetzte Bibliotheken, Frameworks, etc. entsprechen nicht den Erwartungen
- ...

Risikomanagementplan

Erstellen Sie für Ihr Projekt ein Risikomanagementplan. d. h. betrachten Sie nur solche, die im Rahmen Ihres Projekts tatsächlich auftreten können.

- Identifizieren und bewerten Sie mögliche Risiken
- Beschreiben Sie für Risiken, die eine hohe Eintrittswahrscheinlichkeit haben, wie sie diesen Risiken begegnen; d. h. wie Ihre Strategie aussieht.

2. Architektur

Architektur und nicht-funktionale Anforderungen

- Dokumentieren Sie die grundlegende Architektur Ihrer Anwendung.
- Bedenken Sie auch die nicht-funktionalen Eigenschaften, die von der Architektur mit- bzw. maßgeblich beeinflusst werden:
  - Reaktionsgeschwindigkeit (*responsiveness*): Gibt es die Resultate in der erwarteten Zeit?
  - Zuverlässigkeit (*reliability*): Verhält es sich wie erwartet?
  - Verfügbarkeit (*availability*): Ist das System verfügbar, wenn es gebraucht wird?
  - Sicherheit (*security*): Wie werden Nutzer identifiziert? Wo werden welche Daten wie gespeichert?
  - Benutzerfreundlichkeit (*usability*)
  - Wartbarkeit (*maintainability*): Beheben von Fehlern/welche (Arten von) Erweiterungen sind vorgesehen?
  - Resilienz (*resilience*): Verhalten bei Teilausfall?

Beispielhafte Fragestellungen, die eine Anwendung als solches Betreffen

Nicht-funktionale Anforderungen

- Sicherheitskonzept (Welche Angriffe sind möglich? Welche Sicherheit (vor wem) wollen wir garantieren? ...)
- Datenkonzept (Länge von Nachrichten? Bilder? Videos? ... Eventual Consistency?)
- Datenschutzkonzept (Wo werden die Nachrichten gespeichert? Wer darf die Nachrichten einsehen? Umgang mit illegalen Inhalten?)
- ...

Bemerkung

Einige Fragen können rein rechtlicher Natur sein. Andere haben jedoch konkrete Auswirkungen auf die Entwicklung oder den Betrieb.

3. Qualitätssicherung

Effekte Mangelnder Qualitätssicherung[1]

> Auf Tausenden von Windows-Rechnern weltweit ist der Bluescreen of Death zu sehen. Neben Privatpersonen und Firmen sind auch Banken, Krankenhäuser und Flughäfen betroffen. Es entsteht ein Milliardenschaden. [...]
>
> —Juli 2024 - Das Crowdstrike Desaster

Wo fehlten die Qualitätsmaßnahmen (mind./vermutlich)?

- passende QS Maßnahmen fehlten (vermutlich) bei der Entwicklung
- passende QS Maßnahmen fehlten vor dem Rollout

[1] Es handelt sich in diesem Falle eindeutig nicht um einen Cybersecurity Vorfall. Es ist ein Beispiel für mangelnde Qualitätssicherung an *mehreren* Stellen.

Durchzuführende Qualitätssicherung

- Auswahl von Qualitätszielen

> **Achtung!**
>
> Es ist Ihre Aufgabe die/das wirklich wichtigste QM Ziel zu identifizieren, und für dieses eine entsprechende Planung durchzuführen, die es Ihnen ermöglicht das Ziel im Rahmen des Projektes aus zu erreichen. Bedenken Sie die Projektdauer und Ihre Möglichkeiten.

- Qualitätsmaßnahmen leben (und dokumentieren)
- Beleg und Präsentation der durchgeführten Qualitätsmaßnahmen

Beschreibung eines Qualitätsziels

Qualitätsziel: Sichere Webanwendung

Projektspezifische Motivation	Im Rahmen des Projektes … wird eine Webanwendung entwickelt, auf die über das Internet zugegriffen wird. Da diese Anwendung … personenbezogene Daten verarbeitet und potentiellen Angriffen ausgesetzt ist, ist ein wesentliches Qualitätsziel, dass die Anwendung keine Sicherheitslücken aufweist über die Angreifer Daten anderer Benutzer abgreifen können.
Umfang	Im Rahmen dieses Projektes können wir jedoch nur gewährleisten, dass die Webanwendung keine „Standardlücken“ wie zum Beispiel SQL Injections aufweist. Um dieses Ziel zu erreichen, setzen wir die folgenden Tools/Prozesse: … ein.
Durch wen/wann?	Darüber hinaus wurde ein Entwickler benannt, der sich maßgeblich um das Thema „Sicherheit in Webanwendungen“ kümmert und …
Wie wird reagiert?	Die automatisierte Analyse des Codes der Webanwendung erfolgt im Rahmen des regelmäßigen „Nightly Builds“. Sollte ein Problem gefunden werden, so geht eine Mail an alle Entwickler und im Rahmen des nächsten (gruppeninternen) Meetings wird dann ein Entwickler bestimmt, der den Fehler beseitigt.

Qualitätssicherungsdokumentation am Projektende

- Die Abgabe muss belegen, dass die beschriebenen Qualitätsmaßnahmen und Prozesse auch durchgeführt wurden.
- Es ist darauf zu achten, dass …
  1. erkenntlich ist, dass der Prozess eingehalten wurde (d. h. wann und wie häufig etwas getan wurde) und auch, dass
  2. die Maßnahmen im beschriebenen Umfang durchgeführt wurden.

(exemplarisch) Qualitätssicherungsdokumentation - Automatisierte Tests

Wurden als QS Maßnahme automatisierte Tests geplant, so ist die vollständige Liste der Tests abzugeben und es ist zu belegen welche Teile des Codes getestet wurden. Weiterhin ist die Relation der Tests zu den User Stories zu zeigen.

Dies kann insbesondere dadurch geschehen, dass ein Auszug eines Codeabdeckungstools gezeigt wird; z. B. aggregiert auf Klassen-/Dateiebene.

Bitte halten Sie die Möglichkeit vor die Testsuite im Rahmen der Abschlusspräsentation zu zeigen.

(exemplarisch) Qualitätssicherungsdokumentation - Benutzerstudie

Die Abgabe soll zeigen wann diese Studie(n) von wem und mit welchen Probanden durchgeführt wurde und wie der genaue Ablauf war.

Wurden den Probanden Aufgaben geben und diese danach gebeten einen Fragebogen auszufüllen? Fand ein (geschlossenes/offenes) Interview statt? Wurden die Probanden nur beobachtet?

Insbesondere ist kurz zu präsentieren, welche Ergebnisse aus der Benutzerstudie abgeleitet wurden und welche Konsequenzen gezogen wurden.

(exemplarisch) Qualitätssicherungsdokumentation - Dokumentation des Quellcodes

Ist eine Maßnahme, die versprochen wurde, dass der Code dokumentiert wurde, so ist hier ein Auszug des Codes zu zeigen.

Die gezeigten Dateien sollten repräsentativ für das Projekt sein. Die gewählten Dateien müssen weiterhin von herausgehobener Bedeutung für das Projekt sein.

Der restliche Code sollte vorgehalten werden, falls im Rahmen der Präsentation Rückfragen kommen.

(exemplarisch) Qualitätssicherungsdokumentation - Code Reviews

Falls die geplante Maßnahme systematische Code Reviews waren, dann ist diesbezüglich die Checkliste zu zeigen, auf die die Reviewer zu achten hatten.

Weiterhin ist exemplarisch ein Stück Code zu zeigen, der den Prozess durchlaufen hat.

Der weitere Code ist vorzuhalten, um ggf. im Rahmen der Präsentation die Effektivität der Maßnahme zu belegen. Sollten nicht alle Teile einem Review unterzogen worden sein, so ist dies im Vorfeld - ohne Aufforderung - im Rahmen der Präsentation zu erklären.

4. Build-Prozess

Automation des Build-Prozess

Ziel
Stabile Builds

Um stabile Builds zu erhalten ist es notwendig, dass ...

- die Laufzeitumgebung(en) fest definiert ist
- alle Einstellungen festgelegt sind (insbesondere die Compiler-Einstellungen)
- alle Abhängigkeiten wohl definiert (inkl. Versionsnummer) sind:
  - Abhängigkeiten zum Build-System
  - Abhängigkeiten zu den verwendeten Bibliotheken
  - Abhängigkeiten zu den verwendeten Tools

Grundlegend zu automatisierende Tätigkeiten

- Codeabdeckung
- Quellcode Formatierung
- Überprüfung des Stils
- (Lightweight) Bug Detection
- Dokumentationsgenerierung
- Packaging

Automatisierbare Tätigkeiten

- Überprüfung auf veraltete Bibliotheken und Werkzeuge
- Generierung der Webseite
- Veröffentlichung (zum Beispiel in einem Repository, auf einem Webserver, …) und/oder Deployment