

1 Skizze / Phasen des derzeitigen Entwicklungs- und Auslieferungsprozesses

Bitte skizzieren Sie den derzeitigen und Prozess der Softwareerstellung von der Erstellung bis zur Auslieferung.

Antwort: *Verweis auf das Unternehmens-Wiki von adesso. Hierunter fallen demnach die Phasen Build, Smoke-Test, Integrationstest, Regressionstest und Web-Test¹*

Antwort:

- Setup Entwicklungsumgebung
- Entwicklung – > Subphasen Checkin – > Build & Unittests – > Automatisierte Deployments – > Automatisierte (Integrations-)Tests
- Integrationstests
- Delivery

Antwort:

1. Entwicklung, meist TDD
2. Build
3. Unit-Tests
4. Integrations Tests
5. ggf. Deploy auf Testsystem / Staging System
6. ggf manuelles Testen oder exploratives Testen
7. Release bzw. Auslieferung manuell, wenn 1-7 ok

Welche Phasen, von der Erstellung bis zur Auslieferung von Software, werden unterschieden?

Zustimmung:

¹Vgl. https://www.adesso.de/wiki/index.php/Continuous_Integration#Build-Pipeline

Kompilierung des Quellcode	2	67%
Unit- und Komponententests / Analyse	2	67%
automatisierte Akzeptanztests	1	33%
automatisierte Kapazitätstests	2	67%
manuelle Testsstage	1	33%
Konfigurieren von Test- und Produktivumgebung	1	33%
Installation und der Release der Software	2	67%

Antwort: Integrations-Tests

Antwort: diese sind in jedem Projekt ganz individuell!

2 Commit-Stage

Aussagen zur Commit-Stage

Zustimmung:

Gewöhnlich wird die Software auf einem Entwickler-System gebaut und paketierr.	0	0%
Es gibt einen standardisierten Prozess für das Bauen / Paketieren einer Anwendung.	2	67%
Wenn Quellcode kompiliert wurde, wird dieser in allen nachfolgenden Phasen verwendet und nicht noch einmal kompiliert.	3	100%
Erstellte Softwarepakete werden mit einer fortlaufenden Versionsnummer versehen.	2	67%
Alle im Prozess erstellten Softwarepakete werden mit einer fortlaufenden Versionsnummer versehen.	1	33%
Alle Artefakte, die beim Kompilieren und Testen entstehen, werden in einem zentralen Repository aufbewahrt.	3	100%
Dateien und Information zur Konfiguration der erstellten Software werden in einem geschützten Speicher zentral verwaltet.	2	67%
Es werden übersichtliche Berichte erstellt, die eine qualitative Aussage einer bestimmten Version ermöglichen.	1	33%

Anmerkung: auch hier: ist in jedem Projekt ganz individuell!

Wie werden Softwareentwicklungsprojekte beim Konfigurationsmanagement unterstützt?

Antwort: Es gibt Vorlagen für Ant und Maven

Antwort: Das Startup (vormals CI) team unterstützt durch Beratung und Doing in der Startup-Phase eines Projektes. Weiterhin werden Templates für Buildsysteme z.V. gestellt damit Projekte einheitlich gebaut werden (ANT bereits vorhanden, Maven und Gradle-Templates in Vorbereitung).

Antwort:

- Beratung
- über Ticketsystem, um Einrichten von CI Jobs oder bsp. Sonar zu triggern

Wie wird mit Abhängigkeiten zu externen Bibliotheken bzw. Komponenten von Drittanbietern umgegangen?

Antwort: Über einen eigenen Nexus-Server

Antwort: 3rd-Party Bibliotheken werden auf einem zentralen Nexus in einem dafür vorgesehenen Repository abgelegt. Die Projekte sind verantwortlich für die Einhaltung der Compliance, z.B. dürfen Libs, deren Lizenz eine Auslieferung verhindern (z.B. SUN JDK Bibliotheken, JDBC-Treiber usw.) nicht mit ausgeliefert werden.

Antwort: Nexus als Repository Manager und Proxy zu anderen Repositories

Wie wird der Build-/ Test-Prozess ausgelöst?

Zustimmung:

Kompilierung, Paketierung und Test werden jeweils durch einzelne Kommandos auf der Konsole durchgeführt.	0	0%
Für den Prozess existieren Skripten die das Bauen, Paketieren und Testen der Anwendung durchführen.	0	0%
Wir haben automatisierte Prozesse die durch Ereignisse, z.B. Entwickler-Commit in der Versionsverwaltung, ausgelöst werden.	2	67%

Anmerkung: individuell

Anmerkung zur Commit-Stage:

Anmerkung: Jedes Projekt handhabt dies ganz individuell!

3 Acceptance-Stage

Aussagen zum Testbetrieb:

Zustimmung:

	Zustimmung	Anteil
Es gibt vereinzelt ein paar automatisierte Unit- und Komponententests.	1	33%
Bei einer Änderung des Softwarestandes werden der größte Teil der Tests wiederholt (Regressionstest).	3	100%
Es wird eine statische Code-Analyse durchgeführt.	2	67%
Funktionalen Anforderungen werden automatisierte getestet.	2	67%
Unit- und Komponententest werden mit einer hohen Abdeckung durchgeführt.	3	100%
Wenn manuelle Tests durchgeführt werden, basieren diese auf speziellen Risikoszenarien.	1	33%
Die Anwendung wird nach bestimmten Sicherheitsaspekten analysiert.	1	33%
Es wird auf eine Testabdeckung von 100% wert gelegt.	0	0%
Test- und Produktivumgebung sind zu großen Teilen identisch.	1	33%
Testberichte können von allen Projektbeteiligten eingesehen werden.	1	33%

Welche Werkzeuge und Methoden werden zur statischen Analyse des Quellcodes angeboten?

Antwort: Sonar und FindBugs

Antwort: Sonar (kapselt Findbugs, Jococo, Checkstyle, PMD, Cobertura)

Antwort: Sonar, Findbugs, PMD

Welche Werkzeuge werden zur Aufdeckung von möglichen Sicherheitslöchern verwendet bzw. angeboten?

Antwort: s. oben (Findbugs hat einige Security-Regeln)

Welche Werkzeuge nutzen Sie für automatisierte Akzeptanz- und Kapazitätstests?

Antwort: JMeter, Selenium

Antwort: JUnit, JBehave

Welche Informationen können aus den Testberichten entnommen werden?

Antwort: Alle wesentlichen ;-) Abdeckung Erfolg, Nichterfolg

Antwort: Fehlerhafte Tests, Testabdeckung

Nach welchen architektonischen Vorgaben / Brüchen wird die erstellte Software hin untersucht?

Antwort: Modularität, Komplexität, Kohäsion

Gibt es bestimmte Strategien für den Umgang mit Testdaten?

Antwort: teilweise

Anmerkungen zur Acceptance-Stage

Anmerkung: Auch hier: in jedem Projekt ganz individuell

4 Deployment

Welche Verfahren werden genutzt, um eine releasefähige Software in die Produktivumgebung auszuliefern?

Antwort: Releases werden über Jenkins ausgeliefert

Antwort: manuelles Delivery -> es ist mir nicht bekannt dass wir in eine Prod-Umgebung eines Kunden automatisiert zugreifen und deployen könnten. Zudem steht einem automatisierten Deployment in vielen Unternehmen die SOX-Compliance entgegen die besagt dass Entwicklung und Produktion strikt zu trennen ist.

Antwort: Skripte (Shell, Ant), Manuell

Wie wird das Software-Deployment durchgeführt?

Zustimmung:

	Zustimmung	Anteil
Es gibt derzeit keine Deployment-Skripte.	0	0%
Deployment-Skripte werden vorwiegend über die Konsole angestoßen.	1	33%
Deployment-Skripte werden automatisch angestoßen, sofern ein bestimmtes Quality-Gate durchschritten wurde.	0	0%
Es gibt eine Oberfläche von der aus das Deployment einer bestimmten Version angestoßen werden kann.	1	33%

Anmerkung: individuell, meist über Ant oder Maven

Werden Projektteams bei der Erstellung eines Deployment-Skriptes unterstützt?**Zustimmung:**

	Zustimmung	Anteil
Es gibt keine Unterstützung, da die Projekte dies Selbst regeln.	0	0%
Es gibt ein paar Skripte aus anderen Projekten, die als Vorlage genutzt werden können.	1	33%
Es gibt Templates für unsere Standardwerkzeuge.	1	33%
Es gibt Tools mit denen sich das Deployment konfigurieren lässt.	1	33%

Welche Werkzeuge werden für das Deployment genutzt?

Antwort: ant und Shell-Skripte

Antwort: Ant mit Shellsripten für App-Server, die keine dedizierte Schnittstelle oder Plugin für Ant/Maven bieten. Falls diese Ant-Tasks oder Maven-Plugins anbieten (z.B. Apache Tomcat) werden diese genutzt

Wie wird mit Konfiguration von Test- / Produktivumgebungen umgegangen?**Zustimmung:**

	Zustimmung	Anteil
Abhängige Komponenten werden vor dem Betrieb bzw. Test manuell installiert.	1	33%
Projekte erstellen eigene Skripte, um die benötigten Komponenten zu installieren.	2	67%
Es gibt Template-Skripte um Standardumgebungen zu installieren und konfigurieren.	0	0%

Existieren derzeit Bemühungen eine Delivery-Pipeline in einzelnen Softwareentwicklungsprojekten umzusetzen?

Zustimmung:

	Zustimmung	Anteil
Es gibt noch kein Einsatzszenario bzw. Interesse.	0	0%
Es sind Projekte in Zukunft geplant.	0	0%
Es gab / gibt schon Pilotprojekte.	1	33%
Eine Delivery-Pipeline haben wir schon öfter umgesetzt.	1	33%

Anmerkungen zum Deployment

Anmerkung: Auch hier: alles individuell, insbesondere was die Produktivsetzung angeht. Zur Frage *Wie wird mit Konfiguration von Test- / Produktivumgebungen umgegangen?* auch hier ganz unterschiedlich, aber meist manuell. Bei der AMS haben wir nun in einer Cloud-Umgebung Chef im Einsatz um Server automatisiert zu provisionieren.