Agile Testing

Grundlagen Schulung

Autor(en): Anis Ben Hamidene

Novatec Consulting GmbH

Inhalt

1 Beschreibung

Agile Methoden haben sich in den letzten Jahren zum Mainstream entwickelt. Dennoch beschreiben diese Ansätze lediglich Grundgerüste der agilen Softwareentwicklung und lassen Fragen über Umsetzungsdetails und das Thema Testing unbeantwortet. Dabei ist gerade das Agile Testing ein unverzichtbarer Aspekt, um die Qualität und den Erfolg einer Softwareentwicklung sicher zu stellen. In diesem interaktiven Workshop lernen die Teilnehmer die wichtigen Bausteine des agilen Testings und wie diese effizient in den Entwicklungsprozess integriert werden können, um die erwünschte Qualität zu erreichen.

2 Zielgruppen

Die Agile Testing Schulung wird allen Personen empfohlen, die mit den Themen Testen und Software-Qualität im agilen Kontext zu tun haben. Hierzu zählen vor allem: Tester und Qualitätsmanager, aber auch Entwickler. Zudem können nicht technische Module auch für Architekten, Projektleiter und Product Owner von Interesse sein.

3 Module

Die Schulung ist modular aufgebaut. Sie kann deswegen auf die individuellen Bedürfnisse der Teilnehmer zugeschnitten werden. Je nach Bedarf kann die Schulung entweder an einem oder zwei Tage gehalten werden.

Zusätzlich bieten wir mehrere Vertiefungsschulungen an, die auf die einzelnen Themen eingehen.

Modulübersicht

1. Agilität und deren Auswirkung aufs Testen
2. Der neue Blickwinkel
3. Bausteine von nachhaltigen Testing Strategien
4. Agile Prinzipien
5. Agile Requirements Engineering mit Specification By Example/ Behaviour Driven Development (wichtig)
6. Testarten
7. Testautomatisierung
8. Unit Tests
9. Automatisierung von GUI-Tests
10. Exploratives Testing
11. Software Craftsmanship
12. Product Quality Testing
13. Integration Patterns
14. Visualisierung und Reporting

3.1 Agilität und deren Auswirkung aufs Testen

Damit die Herausforderungen für die Qualitätssicherung im agilen Kontext gemeistert werden können, ist eine tiefgreifende Umgestaltung der Testaktivitäten und Verantwortlichkeiten notwendig.

Deswegen ist ein gemeinsames Verständnis für die Natur von agilen Projekten wichtig.

Dazu gehen wir in diesem ersten Modul auf die Bedeutung von Komplexität z. B. mit Hilfe des Stacey Modells oder des Cynefin Frameworks ein und schaffen dadurch die Basis für das neue notwendige Mindset.

Im Anschluss besprechen wir die wichtigen Herausforderungen, die das Testen im agilen Kontext bremsen oder verhindern.

3.2 Der neue Blickwinkel

Anhand von mehreren inspirierenden Zitaten und Fallbeispielen werden Impulse für einen Mindsetwechsel und für einen neuen Blickwinkel auf die Testaktivitäten und –verantwortlichkeiten vermittelt.

Insbesondere gehen wir auf eine differenzierte Definition/Betrachtung von Qualität als Konzept und Ziel ein.

3.3 Bausteine von nachhaltigen Testing Strategien

In diesem Modul werden die Dimensionen und Bereiche aufgezeigt, die in jeder nachhaltigen Agilen Testing Strategie betrachtet und angegangen werden sollen: von der Entstehung der Produktideen, über die Spezifikation, Planung, Priorisierung, Umsetzung bis hin zur Produktivsetzung der Liefergegenstände.

3.4 Agile Prinzipien

Während diesem Modul werden die wichtigsten agilen Prinzipien besprochen, die für eine erfolgreiche und nachhaltige Implementierung notwendig sind.

3.5 Agile Requirements Engineering mit Specification By Example/Behaviour Driven Development (wichtig)

Behaviour Driven Development (BDD) bzw. Specification By Example verstärkt die Zusammenarbeit aller Beteiligten an der Softwareentwicklung und definiert die zu erstellende Software vom Ergebnis her. Wie soll sich die Software verhalten? Was soll sich ändern? Die systematische Anwendung von BDD macht die Abnahme von User Stories durch automatisierte Akzeptanztests zum Kinderspiel. Ganz nebenbei entsteht eine „lebende Dokumentation“, die immer aktuell ist.

In diesem Modul lernen die Teilnehmer eine praxisorientierte Einführung in BDD. Gängige Methoden werden in einen Geschäftszusammenhang gebracht, so dass eine Integration in den Arbeitsalltag eines Product Owners problemlos gelingen kann.

Noch greifbarer wird die Thematik in einer abschließenden kurzen Demo, in der wir die gängigen BDD Werkzeuge vorstellen (z. B. Cucumber, JBehave, Serenity BDD, Concordion, Fitnesse...).

3.6 Testarten

Um die Testaktivitäten effizient und effektiv zu gestalten und in den Entwicklungsprozess zu integrieren ist ein gemeinsames Bild der unterschiedlichen Testarten essenziell.

Im Modul „Testarten" setzen sich die Teilnehmer mit den unterschiedlichen Begrifflichkeiten und Testarten auseinander und lernen die Unterschiede anhand der Agile Testing Pyramide sowie der Agile Testing Quadranten kennen.

3.7 Testautomatisierung

Testautomatisierung ist in der heutigen schnelllebigen Zeit sehr wichtig. Viele Testautomatisierungsvorhaben sind jedoch leider trotz hoher Investitionssummen gescheitert, weil diese auf die falschen Beine gestellt wurden.

In diesem Modul lernen die Teilnehmer die wichtigsten Prinzipien und Regeln, um nachhaltige und wertstiftende Testautomatisierung aufzubauen.

3.8 Unit Tests

Unit Tests sind ein wichtiger Baustein jeder nachhaltigen und wertstiftenden Testautomatisierungsstrategie. Falsch umgesetzt entwickeln sie sich jedoch schnell zu einem Wartungsalbtraum und sie werden deswegen schnell aufgegeben.

In diesem Modul lernen die Teilnehmer die wichtigsten Merkmale von Unit Tests sowie die wichtigsten Prinzipien und Regeln, um das Beste aus ihnen zu holen. Auch häufige Antipatterns werden in praktischen Übungen diskutiert.

Darüber hinaus besteht die Möglichkeit in das Thema Mocking einzusteigen, da Mocking einerseits die Erstellung von Unit Tests unterstützt, andererseits kann der ungünstige Einsatz der Mocking Tools zu unerwünschten Effekten führen und den Wartungs- und Fehlerfindungsaufwand in die Höhe treiben

3.9 Automatisierung von GUI-Tests

Aufgrund ihrer Natur sind automatisierte GUI Tests abhängig von ihrer Ausführungsumgebung (Betriebssystem, Browser, Client-Technologie, Netzwerklatenz, Client Performance, …). Deswegen sind sie anfällig für Probleme und meistens sehr instabil.

Zudem sind GUI-Tests insgesamt aufwändig und teuer. Umso wichtiger ist es deshalb solche Tests auf einer soliden Basis aufzubauen.

Bei dem Modul „Automatisierung von GUI-Tests" lernen die Teilnehmer die wichtigsten Prinzipien, Patterns und Werkzeuge, um stabile und nachhaltige automatisierte GUI Tests zu erstellen (Bsp. Page Object Pattern, Workflow Pattern, Screenplay Pattern, …). Auch setzen sich die Teilnehmer mit den üblichen Antipatterns auseinander (Record and Replay, …).

3.10 Exploratives Testing

Testautomatisierung ist zwar sehr wichtig, sie kann jedoch nicht alle Risiken mit einem vertretbaren Aufwand abdecken. Aus diesem Grund ist es weiterhin wichtig auf die Kreativität und Erfahrungswerte der Testexperten zu setzen. Es geht dabei jedoch nicht um das „Abarbeiten" von manuellen Testschritten, die in Testfallbeschreibungen festgehalten werden, sondern um das sog. explorative Testing.

In diesem interaktiven Modul lernen die Teilnehmer die Grundlagen vom explorativen Testing und wie sie diese wichtige Art des Testens in ihre täglichen Arbeiten integrieren können.

3.11 Software Craftsmanship

Qualität ist nicht nur Aufgabe der Tester, sondern eher des gesamten Delivery Teams. Dazu gehören natürlich die Entwickler, die Verantwortung für Qualität genauso übernehmen müssen.

Das bedeutet für sie, dass sie Qualität in ihre tägliche Arbeit integrieren müssen.

In diesem Modul besprechen wir Methoden und Techniken, die den Entwicklern helfen die Qualität ihres Codes zu erhöhen. Dazu gehören Clean Code Prinzipien, Test Driven Development, Coding Dojos, Pair Programming usw.

3.12 Product Quality Testing

Tests, die die Grenzen der Software untersuchen (a.k.a. nicht funktionale Tests) sind mindestens genauso wichtig wie funktionale Tests.

In der agilen schnelllebigen Welt fehlt jedoch die Zeit solche Tests wie bisher am Ende des Releases durchzuführen. Das Releasekonzept verschwindet sogar immer mehr.

Deshalb lernen die Teilnehmer In diesem Modul, wie sie solche Testarten in ihre Entwicklunsgzyklen integrieren und wie sie mit den zugehörigen Risiken umgehen können.

3.13 Integration Patterns

In diesem Modul diskutieren die Teilnehmer wie sie die bisher gewonnen Erkenntnisse in einen agilen Entwicklungslebenszyklus integrieren können. Dazu wird ein von der Novatec entwickeltes Modell zu Visualisierung als Basis verwendet.

Im zweiten Teil werden verschiedene Modelle zur Skalierung in großen Projekten vorgestellt und diskutiert.

3.14 Visualisierung und Reporting

Transparenz und schnelles Feedback sind zentrale Prinzipien in der agilen Welt. Dabei sind Reports und Visualisierungen wichtige und entscheidende Werkzeuge.

Hierbei ist es auch notwendig sich von den „alten" Gewohnheiten zu lösen und Visualisierung mit einem anderen Mindset anzugehen. Impulse dazu bekommen die Teilnehmer in diesem Modul.

Sie haben Interesse an einer Individualschulung?

Sprechen Sie uns an:

Regina Schlindwein
regina.schlindwein@novatec-gmbh.de
+49 (170) 9050-583

Anis Ben Hamidene
anis.benhamidene@novatec-gmbh.de
+49 (711) 22040-746