



# Inspektions- und Testkonzept

Team MYO

Version 1.0  
23.03.2015

# 1 WIEDERKEHRENDE TESTS:

---

## 1.1 DEFINITION OF DONE (STORY):

*„Eine User Story ist ‘done’, wenn die Ergebnisse dem PO präsentiert wurden und dieser damit einverstanden ist“.*

Dies bedeutet, dass jede User Story vor Abschluss vom Product Owner betrachtet wird und dieser überprüft ob das Ergebnis die Anforderungen wie gewünscht erfüllt. Hierdurch wird sichergestellt, dass die erforderlichen Funktionalitäten umgesetzt und (zumindest nach ihrer Umsetzung) Funktionstüchtig sind.

Diese Reviews sind Teil des Arbeitsprozesses und müssen nicht dokumentiert werden.

## 1.2 DEFINITION OF DONE (TASK):

*„Ein Task ist “Done”, wenn das Ergebnis von einem anderen Team-Mitglied oder dem kompletten Team überprüft wurde. Handelt es sich um einen Implementationstask so muss das überprüfende Team-Mitglied sicherstellen, dass der Code durch geeignete Unit Tests abgedeckt ist und den Code-Richtlinien sowie dem GUI-Konzept entspricht.“*

In dieser Definition of Done wird unterschieden, ob es sich um einen Implementationstask oder einen anderen Task, wie Dokumentation, Research oder vergleichbares handelt. Hierbei ist ein Implementationstask jeder Task, der das Schreiben oder Ändern von Code im Produkt beinhaltet. Das Erstellen eines Prototyps beispielsweise ist kein Implementationstask.

Bei einem Implementationstask werden Kriterien überprüft, die dem PO im Review der Userstory (siehe Definition of Done (Story)) nicht auffallen würde, da dieser nur auf Funktionalität achtet. Es gibt zur Erhaltung der Wart- und Erweiterbarkeit des Produktes Code-Richtlinien und die Vorgabe Unit Tests zu schreiben, um nach jeder Erweiterung zu zeigen, dass keine ältere Funktionalität beeinflusst wurde. Hinzu kommt ein GUI-Konzept welches den grundlegenden Aufbau (nicht jedoch grafisches Erscheinungsbild) der Benutzeroberfläche definiert. Dieses muss erfüllt werden, um dem Anwender eine einheitlich bedienbare Anwendung zur Verfügung zu stellen und die Umsetzung grundlegender Usability-Konzepte zu gewährleisten.

Handelt es sich nicht um einen Implementationstask, so beschränkt sich die Überprüfung darauf das Ergebnis zu betrachten und grob zu beurteilen ob das bei der Erstellung des Tasks verfolgte Ziel erreicht wurde. Dies ist besonders relevant bei Tasks aus denen nicht direkt Funktionalitäten folgen und der Product Owner im Review der User Story folglich nicht überprüfen kann, beispielsweise ein Research-Task.

Diese Reviews sind Teil des Arbeitsprozesses und müssen nicht dokumentiert werden.

## 2 EINMALIGE RELEASETTESTS:

---

### 2.1 INTEGRATIONSTEST

Vor der Auslieferung des Produkts an den Kunden findet ein abschließender Integrationstest statt, in dem die Anwendung gegen alle Anforderungen im Pflichtenheft ausführlich getestet wird. In diesen Tests werden nichtfunktionale Kriterien wie Codequalität und Abdeckung durch Unit Tests nicht berücksichtigt, sofern diese nicht im Pflichtenheft explizit gefordert wurden.

Dieser Integrationstest soll Fehler die durch Erweiterung bestehender Funktionalitäten oder das Zusammenspiel verschiedener Module entstehen aufdecken, sodass diese vor Auslieferung des Produktes behoben werden können.

Diese manuellen Integrationstests werden ausführlich dokumentiert. Hierzu wird zu jeder Anforderung des Pflichtenheftes ein Testplan erstellt in dem jeder Testfall als Punkt in einer Checkliste aufgeführt wird. Die ausgefüllte Checkliste mit zugehörigen Kommentaren dient als Testprotokoll. Schlussendlich wird das Ergebnis in einem Test-Result zusammengefasst.