

UASJ - Maps

Version 1.2

1 Historie der Dokumentversionen

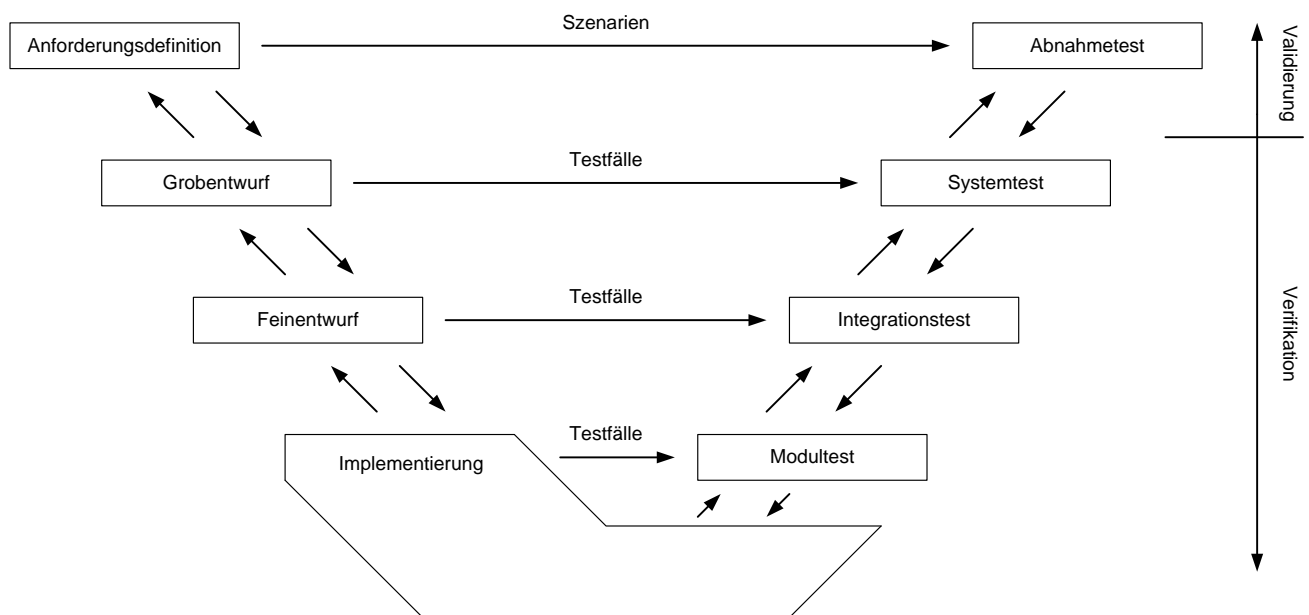
[illegible]

2	<u>Inhaltsverzeichnis</u>	
1	Historie der Dokumentversionen	2
2	Inhaltsverzeichnis.....	3
3	Vorbetrachtung.....	4
3.1	Testmodell	4
3.2	Teststrategie	4
4	Modultest.....	5
4.1	Allgemeine Beschreibung	5
4.2	Was wird getestet?.....	5
4.3	Wie wird getestet?	5
5	Integrationstest.....	6
5.1	Allgemeine Beschreibung	6
5.2	Was wird getestet?.....	6
5.3	Wie wird getestet?	6
6	Systemtest.....	7
6.1	Allgemeine Beschreibung	7
6.2	Was wird getestet?.....	7
6.3	Wie wird getestet?	7
7	Abnahmetest.....	8
7.1	Allgemeine Beschreibung	8
7.2	Was wird getestet?.....	8
7.3	Wie wird getestet?	8

3 Vorbetrachtung

3.1 Testmodell

Für die Planung der einzelnen Softwaretests wird als Grundlage das V-Modell verwendet. Diese Vorgehensweise gliedert den Softwareentwurf in einzelne Phasen. Die folgende Darstellung zeigt den Prinzipiellen Aufbau von Software Qualitätssicherung nach dem V-Modell.



Im Folgenden werden zu allen vier Teststufen, zur Sicherung der Qualität kurz verbal Beschrieben und die Grundlagen für die Durchführung der Tests mit den Entsprechenden Kriterien erläutert.

3.2 Teststrategie

Es wurde von dem Bearbeitungsteam jeden Donnerstag ein lauffähiger Build erstellt. An diesem Builds wurden Schnittstellen und Funktionstest durchgeführt, um sicher zu stellen, dass Funktionierende Funktionen in ihrer Ausführung nicht durch andere und neu hinzugekommene beeinträchtigt werden.

Die dem Handbuch beiliegenden Testlisten beziehen sich nur auf die endgültige Version der einzelnen Komponenten, um die Lauffähigkeit dieser nachzuweisen.

4 Modultest

4.1 Allgemeine Beschreibung

Der Modultest ist die erste Phase des strukturierten Softwaretests, der auf Basis des V-Modells durchgeführt wird. Er befindet sich auf der untersten Ebene des V-Modells. Er besitzt somit ein geringes Abstraktionsniveau und ist somit stark an den Quelltext angelehnt. Der Modultest dient der Softwareverifikation.

4.2 Was wird getestet?

Im Modultest werden die erstellten Java Klassen des zu Grunde liegenden UML Klassendiagramms für die gewünschten Applikationen getestet. Hierzu werden die Klassen einzeln und voneinander Unabhängig getestet.

4.3 Wie wird getestet?

Für den Test der erstellten Java Klassen werden neben den Schnittstellen auch fehlerbehandelnde Pfade getestet, die eine Falscheingabe von Werten verhindern und interne Fehlerbehandlungen durchführen, Eine eventuelle Weiterleitung solcher Ausnahmesituationen wird, soweit vorhanden auch überprüft. Für die Schnittstellentests der Klassen soll überprüft werden, ob Eingabewerte richtig geschrieben wurden und, ob die Eingabewerte die erwarteten Ausgangswerte generieren. Neben diesen so genannten Black Box Test wird ein Funktionstest der einzelnen Klasse durchgeführt. Da die, in der Applikation verwendeten Klassen einen geringen Umfang haben, wird der Funktionstest auf dem Android Smartphone oder im Debug-Modus der verwendeten Entwicklungsumgebung durchgeführt. Es werden somit keine komplexen Testprogramme eingesetzt, welche eine Vielzahl an Testszenarien mit einer genügend großen Wiederholung durchführen, da der Testaufwand den Entwicklungsaufwand um ein Vielfaches übertreffen würden. Für den Modultest wird aufgrund schweren Nachweisbarkeit der Korrektheit von verwendeten Algorithmen kein White Box Test durchgeführt.

5 Integrationstest

5.1 Allgemeine Beschreibung

Auf der Ebene des zu Grunde liegenden V-Modells. Ist die zweite Teststufe der Integrationstest. Der Integrationstest kann durchgeführt werden, sobald alle zu der Applikation gehörigen Klassen einzeln getestet und danach in einem Programm vereint wurden. Er ist somit auf einem höheren Abstraktionsgrad wie der Modultest.

5.2 Was wird getestet?

Im Integrationstest wird lediglich die Kommunikation von zwei oder mehreren Klassen untereinander getestet. Bei einer bereits vorhandenen, grafischen Benutzeroberfläche können hierbei ebenfalls die Menüführung und Klickstreams zur Auswahl verschiedener Menüpunkte getestet werden.

5.3 Wie wird getestet?

Für den Integrationstest wird getestet, ob die Schnittstellen der Klassen untereinander fehlerfrei funktionieren. Daher wird überprüft, ob Daten korrekt von einer Klasse zu anderen übertragen werden und auch eventuelle Fehlerbehandlungen korrekt an folgende Klassen weitergegeben werden. Weiterhin wird auch hier ein Funktionstest des Klassenkonstruktes auf dem entsprechenden Android Smartphone durchgeführt. Da der Abstraktionsgrad des Integrationstestes höher ist, wird ein weitaus umfangreicherer Funktionstest als bei dem Modultest durchgeführt. Im Funktionstest wird die Menüführung zur Erreichung aller Funktionen über entsprechende Klickstreams getestet.

6 Systemtest

6.1 Allgemeine Beschreibung

In den Ebenen des V-Modells stellt der Systemtest die Vorletzte Testeinheit. Als Grundlage der Tests dienen die Produktspezifikationen, welche im Pflichtenheft vereinbart sind. Im Systemtest wird somit nachgewiesen, dass alle Funktionen, welche im Pflichtenheft spezifiziert worden sind im vollen Umfang erfüllt werden. Der Systemtest ist somit der letzte Schritt der erforderlichen Software Verifikation.

6.2 Was wird getestet?

Im Systemtest werden in einem fertigen vorübergehenden oder auch endgültigen Build der Applikation alle im Pflichtenheft vereinbarten Funktionen getestet. Der höhere Abstraktionsgrad als im Integrationstest erlaubt einen Umfassenderen Überblick über das ganze System. So müssen hier nicht mehr Tests an einzelnen Klassen oder Klassenkonstrukten vorgenommen werden.

6.3 Wie wird getestet?

Im Systemtest werden an den vorübergehenden oder auch endgültigen Build auf dem Smartphone oder dem Simulator getestet. Es werden dort die Entsprechenden Klickstreams durchgeführt. Für die Testung des Multitouch auf dem Display ist Lauffähige Programm auf dem Smartphone zu testen, da der Simulator diese Möglichkeit nicht bietet.

7 Abnahmetest

7.1 Allgemeine Beschreibung

Der letzte der Tests, welcher auf Basis des V-Modells durchgeführt wird ist der Abnahmetest. Er dient zur Softwarevalidierung gegenüber dem Kunden. Eine enge Zusammenarbeit mit dem Kunden ist beim Abnahmetest deshalb von großer Bedeutung, weil der Kunde mit der Grundlage des Lastenheftes Prüfen muss, ob die entwickelte Applikation seinen Anforderungen entspricht. Der Abnahmetest besitzt im Vergleich zu den anderen Tests den höchsten Abstraktionsgrad, da der Endanwender für eine Nutzung der Applikation keine Kenntnisse weder über Klassenstrukturen, Algorithmen noch über interne Mechanismen besitzen muss.

7.2 Was wird getestet?

An der fertigen Applikation werden alle im Lastenheft spezifizierten Leistungen mit den entsprechenden zeitlichen oder sonstigen Anforderungen getestet.

7.3 Wie wird getestet?

Für den Abnahmetest wird die Applikation auf mehreren Modellen von Smartphones zum Laufen gebracht. Hierdurch soll nachgewiesen werden, dass eine Wiederverwendbarkeit der Software auf allen Geräten mit der Android Softwareplattform gegeben ist. Der hohe Abstraktionsgrad ermöglicht es, dass in dem Abnahmetest nur die, vom Kunden Spezifizierten Leistungen sich in den Testprotokollen wiederfinden und somit getestet werden müssen. Eine Qualitative Abweichung des Endproduktes von den spezifizierten Anforderungen ist somit nur nach oben hin Zulässig.