



Mobile App Testing

Der Kunde ist unterwegs
und hat nur 2 Sekunden

José Díaz & Björn Lemke Seite 4

Kontextabhängiges Testen
von Apps

Matthias Schulte & Tim A. Majchrzak Seite 8

Testen mobiler
Applikationen

Markus Gärtner Seite 18



Díaz Hilterscheid

Mobile Update Testing

Der Autor

Daniel Knott



Daniel Knott hat einen technischen Hintergrund in verschiedenen Programmiersprachen und Software Quality Assurance Tools. Nach seiner Ausbildung bei der

IBM Deutschland GmbH studierte er an der Fachhochschule Wiesbaden Informatik mit dem Schwerpunkt Softwarequalitätssicherung. Anschließend arbeitete er bei der XING AG in Hamburg. In verschiedenen agilen Projekten (XING Suche, XING Empfehlungen) war er verantwortlich für das Testmanagement, für die Ausführung der Testfälle sowie für die Testautomatisierung. Als Team Lead Quality Assurance im XING Mobile- und API-Team entwickelte er voll automatisierte Testumgebungen für die Android- und iPhone-App. Aktuell arbeitet Daniel Knott als Software Test Manager bei der AOE GmbH in Wiesbaden, wo er für verschiedene Kunden das Software-Testmanagement der Web- und Mobileanwendungen übernimmt.

Zusätzlich zu seiner Tätigkeit bei der AOE GmbH ist Daniel ein Sprecher auf verschiedenen agilen Konferenzen sowie Organisator der Software Test User Group Rhein Main (www.stugrm.de).

Sein XING-Profil:

www.xing.com/profile/Daniel_Knott

Sein Blog: www.adventuresinqa.com

Twitter: [@dnlkntt](https://twitter.com/dnlkntt)

Manuelles sowie automatisiertes Testen von mobilen Anwendungen ist während der Entwicklungsphase ein sehr wichtiger grundlegender Bestandteil. Je nach Anwendung sollten die richtigen Tools miteinander kombiniert werden, um eine stabile und qualitativ hochwertige App zur Verfügung zu stellen. Ebenfalls sollten die Usability, die Performance und die Sicherheit der App während der Entwicklung nie vernachlässigt werden.

Neben den genannten Bereichen gibt es einen anderen wichtigen Punkt, der oft bei der Entwicklung sowie vor einem Release von mobilen Anwendungen nicht bedacht wird: das Testen des Updateprozesses.

Dieser Punkt ist sehr wichtig, da dabei das Verhalten der App getestet wird, wenn sie von der aktuellen Live-Version auf die neue Release-Version aktualisiert wird. Mit diesem Test wird sichergestellt, dass die neue Version die installierte Version nicht negativ beeinflusst. Nachfolgende Situationen werden unter anderem geprüft, die bei Usern zu einer schlechten User Experience oder zu einem Fehlverhalten der App führen können.

Testen, dass das Update ...

- ... den zuvor eingeloggten User nicht mit dem Update ausloggt.
- ... die lokale Datenbank nicht löscht oder so modifiziert, dass die Daten eventuell defekt sind.
- ... korrekt auf dem Gerät installiert wird.
- ... von einer noch älteren Version auf die aktuelle Version funktioniert.

Für die beiden größten Plattformen Android und iOS wird nachfolgend beschrieben, wie der Updateprozess einer App getestet werden kann.

iOS

Für iOS-Apps gibt es derzeit zwei Wege, wie der Updateprozess getestet werden kann. Der erste Weg führt über iTunes (dazu die Technical Note TN2285 von Apple beachten [APLo1]).

1. Im ersten Schritt muss die aktuelle Live-Version gebaut werden. Hinweis: Die Version muss den gleichen Bundle Identifier haben wie die neue App.
2. Es muss sichergestellt werden, dass keine ältere Version im iTunes oder auf dem Testgerät verfügbar ist.
3. Die gebaute App aus Schritt 1 per Drag-and-drop in iTunes einfügen und mit dem Testgerät synchronisieren.
4. Die App starten, einloggen und sicher sein, dass dies die aktuelle Live-Version ist.
5. Jetzt wird der Release-Kandidat gebaut und ebenfalls per Drag-and-drop in iTunes eingefügt und mit dem Testgerät synchronisiert. iTunes sollte einen Dialog zeigen und sagen, dass eine ältere Version gelöscht wird. Hinweis: Die alte Version nicht löschen, da ansonsten der Updateprozess nicht durchgeführt wird. Anschließend installiert iTunes die neue Version über die alte Version und simuliert somit das Update.
6. Die neue App starten und testen, dass das Update den vorherigen Stand der App nicht verändert hat. Die Daten sollten die gleichen sein.

Der zweite Weg, um den Updateprozess zu testen, führt über das Tool iPhone Configuration Utility [APLo2]. Dieser Weg ist deutlich komfortabler, gerade wenn der Updatetest auf mehreren iOS-Geräten durchgeführt werden soll.

1. Im ersten Schritt muss die aktuelle Live-Version gebaut werden. Hinweis: Die Version muss den gleichen Bundle Identifier haben wie die neue App.
2. Es muss sichergestellt werden, dass keine ältere Version auf dem Testgerät verfügbar ist.
3. Die gebaute App aus Schritt 1 über das Tool auf dem Testgerät installieren.
4. Die App starten, einloggen und sicher sein, dass dies die aktuelle Live-Version ist.
5. Jetzt wird der Release-Kandidat gebaut und über das Tool installiert, dabei wird direkt der Updateprozess getestet.

Schreiben Sie für uns!

Testing Experience DE Nr. 4:
„Testautomatisierung“ (Januar 2014)

Für unsere nächste Ausgabe zum Thema „Testautomatisierung“ können auch Sie gern Ihr Wissen und Ihre Erfahrungen in Form eines Artikels einreichen.
Der Einsendeschluss für Artikel ist der 15. November 2013.



www.testingexperience.de

6. Die neue App starten und testen, dass das Update den vorherigen Stand der App nicht verändert hat. Die Daten sollten die gleichen sein.

Android

Den Updateprozess auf Android-Geräten zu testen ist deutlich einfacher und schneller als bei iOS. Mit Hilfe von adb [ADBo1] kann dieser Test auf der Kommandozeile durchgeführt werden.

1. Im ersten Schritt muss die aktuelle Live-Version aus dem Google Play Store auf dem Testgerät installiert werden.
2. Die App starten, einloggen und sicher sein, dass dies die aktuelle Live-Version ist.
3. Die neue Release-Version der App bauen. Hinweis: Es muss sichergestellt werden,

dass die neue Release-Version mit dem gleichen Schlüssel signiert wird wie die aktuelle Live-Version.

4. Mit dem Befehl `./adb install -r APPNAME.apk` die App auf dem Testgerät installieren. Wichtig ist der Parameter `-r`: Dieser gibt an, dass die bestehenden Daten erhalten bleiben und nur die App aktualisiert wird. Es wird somit ein Update simuliert.
5. Die neue App starten und testen, dass das Update den vorherigen Stand der App nicht verändert hat. Die Daten sollten die gleichen sein.

Mit dieser einfachen und schnellen Methode wird sichergestellt, dass der Updateprozess für den User zu keinen Problemen führt. Probleme, die beim Update auftreten, können zum Vertrauensverlust an die App oder die Firma, die diese App entwickelt, führen.

Referenzen

- [APLo1] *Apple TN2285*, http://developer.apple.com/library/ios/#technotes/tn2285/_index.html#//apple_ref/doc/uid/DTS40011323
- [APLo2] *iPhone Configuration Utility*, <http://support.apple.com/kb/DL1465>
- [ADBo1] *Android Debug Bridge*, <http://developer.android.com/tools/help/adb.html> ■