



Hochschule Osnabrück
University of Applied Sciences

Fakultät
Ingenieurwissenschaften und Informatik

Projektdokumentation

für das
Masterprojekt
mit dem Thema

Entwicklung eines Android Usability Testing Frameworks

Autor: Albert Hoffmann
albert.hoffmann@hs-osnabrueck.de

Prüfer: Prof. Michaela Ramm, M.A.

Abgabedatum: 06.10.2014

Kurzfassung

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Abstract

Das ganze auf Englisch.

I Inhaltsverzeichnis

II	Abbildungsverzeichnis	III
III	Tabellenverzeichnis	IV
IV	Listing-Verzeichnis	IV
V	Abkürzungsverzeichnis	V
1	Einleitung	1
1.1	Motivation	1
1.2	Problemstellung	1
1.3	Stand bei Projektbeginn	1
2	Stand der Technik	2
2.1	Mobile Usability Testing Methoden	2
2.1.1	Thinking Aloud	2
2.2	Mobile Usability Testing Anbieter	2
2.2.1	UserTesting	2
2.2.2	UserZoom	3
2.3	Mobile Usability Testing Werkzeuge	4
2.4	Relevante Technologien	4
2.4.1	Chromecast Screen Mirroring	4
2.4.2	Android Bildschirmaufzeichnung	4
3	Konzept	4
3.1	Anforderungen	4
4	Realisierung des Servers	4
4.1	RESTFul Webservice	4
5	Realisierung der Android Bibliothek	4
6	Realisierung des Testleiter-Clients	4
7	Ergebnisse	4
8	Fazit	4
8.1	Weiterentwicklung	4
8.1.1	Integrierte Blickverfolgung	4
9	Quellenverzeichnis	6
	Anhang	I
A	GUI	I

II Abbildungsverzeichnis

III Tabellenverzeichnis

IV Listing-Verzeichnis

V Abkürzungsverzeichnis

SDK	Software Development Kit
REST	Representational State Transfer
URL	Uniform Resource Locator

1 Einleitung

1.1 Motivation

1.2 Problemstellung

1.3 Stand bei Projektbeginn

Dieses Projekt basiert auf einer Arbeit, welche zusammen mit Oliver Erxleben (oliver.erxleben@hs-osnabrueck.de) im Modul *Mensch-Maschine-Kommunikation* (Wintersemester 2013/2014) des Informatik–Masterstudiengangs *Verteilte und mobile Anwendungen* erstellt wurde. Die Ergebnisse zum Stand der Abgabe können aus dem Git-Repository von Oliver Erxleben unter der folgenden URL¹ abgerufen werden:

https://github.com/olivererxleben/hipsterbility/releases/tag/v1.0_semester_final

Der Vollständigkeit halber wird der Stand der vorausgehenden Arbeit an dieser Stelle kurz wiedergegeben. Es wurde ein Prototyp erstellt, welcher aus einer Android Bibliothek und einem kleinen REST²-Server besteht. Zum Testen der Bibliothek wurde weiterhin eine kleine Android Applikation erstellt. Insgesamt wurde folgender Funktionsumfang skizziert:

- Android 4.x Bibliothek
 - Abrufen von Testsitzungen und Testaufgaben vom Server.
 - Benutzeroberfläche zum Auswählen und Starten von Testsitzungen.
 - Aufzeichnen des Kamerabildes der Frontkamera mit Ton, nach dem Starten einer Testsitzung.
 - Erstellen von Screenshots der Applikation bei Benutzereingaben auf dem Touchscreen und Visualisierung der Eingaben auf dem Screenshot.
 - Hochladen der gesammelten Daten zum Server.
- Node.js basierter Server mit REST-Schnittstelle und Webinterface
 - Bereitstellen von Testsitzungen und -aufgaben.
 - Rudimentäres Benutzermanagement.
 - Weboberfläche zum Erstellen von Testsitzungen und -aufgaben
 - Speichern der hochgeladenen Daten in einer Datenbank und im Dateisystem.
 - Aufbereiten der Screenshots und des Videos von der Frontkamera in ein Ergebnisvideo für die Auswertung.

¹Uniform Resource Locator

²Representational State Transfer

Die Idee und die ursprüngliche Planung stammen von Oliver Erxleben, welcher auch für die Serverkomponente zuständig war.

Da viele Funktionen in der Implementierung nur skizziert wurden eignet sich der Prototyp nicht für die Verwendung in einer produktiven Umgebung.

Folgende Komponenten wurden aus der vorhergehenden Arbeit übernommen und weiterentwickelt:

- das grundlegende Konzept,
- Teile des Datenmodells,
- und Quellen der Android Bibliothek und Testapplikation.

Der Server wird auf der Basis von *Java EE 7* neu aufgebaut.

2 Stand der Technik

2.1 Mobile Usability Testing Methoden

Usability-Tests können grob in zwei räumliche Kategorien eingeteilt werden, Labortests und Feldtests, welche jeweils moderiert oder unmoderiert stattfinden können.

Ein einfacher Testaufbau für Labortests lässt sich z.B. mit Hilfe eines Computers mit angeschlossener, externer Kamera realisiert werden. Vorausgesetzt werden auch entsprechende Räumlichkeiten und Benutzer, welche die Tests durchführen. Die, an den Computer angeschlossene Kamera, dient zum Aufzeichnen und Dokumentieren des Bildschirms eines mobilen Gerätes auf dem getestet wird. Über die Kamera werden auch die Benutzereingaben dokumentiert. [Vgl. Bud14]

2.1.1 Thinking Aloud

2.2 Mobile Usability Testing Anbieter

Neben Methoden und Werkzeugen zum Messen und Testen von Software-Usability gibt es auch Dienstleister, deren Geschäftsmodell auf Usability-Tests und Untersuchungen basiert. Hier werden einige Anbieter solcher Dienstleister kurz vorgestellt und das Angebot kurz beschrieben. Die Auswahl und Beschreibung beschränkt sich auf Anbieter, die das Testen von mobile Applikationen anbieten.

2.2.1 UserTesting

Der Dienstleister *UserTesting* (<http://www.usertesting.com/>) bietet, unter anderem, Usability-Studien an, welche von Testteilnehmern durchlaufen werden. Das Angebot beinhaltet auch das Testen von mobilen Webseiten und Applikationen. [Vgl. Use14c]

Der Kunden/Auftraggeber stellt Testaufgaben zusammen, welche die Testpersonen erfüllen sollen. Es gibt die Möglichkeit Geräteklasse, Anzahl der Testteilnehmer und eine Zielgruppe festzulegen. Das Ergebnis der in Auftrag gegebenen Studie kann, laut Anbieter, schon nach ca. einer Stunde vorliegen und umfasst Videos von Testteilnehmern während der Tests, schriftliche Antworten und die Möglichkeit der anschließenden Befragung der Teilnehmer. [Vgl. Use14a]

Das Angebot zum Testen von mobilen Applikationen umfasst die Plattformen Android und iOS. Pro Plattform werden als Testgeräte Tablets und Smartphones angeboten, welche bei der Spezifikation einer Teststudie angegeben werden. Auch hier werden Zielgruppen ausgewählt und Testaufgaben spezifiziert. Das Testergebnis umfasst auch eine Videoaufzeichnung vom Test, auf dem der Gerätebildschirm sichtbar ist und der Benutzer Aussagen nach der *Thinking Aloud* Methode (siehe Abschnitt 2.1.1) tätigt. Die pauschalen Kosten für kleine Studien belaufen sich auf \$49 pro Testbenutzer. [Vgl. Use14b]

2.2.2 UserZoom

Das *UserZoom* (<http://www.userzoom.de>) Angebot vereint Werkzeuge und Dienstleistungen. Letztere sind vorwiegend unterstützender Natur und umfassen z.B. Beratung, Rekrutierung von Teilnehmern und den Kunden-Support [vgl. Use13a].

Das Angebot bzgl. des Mobile Testing umfasst eine mobile Applikation für die iOS und Android Plattformen, welche sich zum Testen von Webseiten und webbasierten App-Prototypen eignet. Das Verfahren wird als „[...] Remote Unmoderated Mobile Usability Testing [...]“ [Use13b] bezeichnet und erlaubt ein standortunabhängiges Testen. Angemeldete Testpersonen werden in Zielgruppen eingeteilt und per E-Mail zu Studien und Befragungen eingeladen. Das eigentliche Testen wird über eine native mobile Applikation durchgeführt, in welcher Fragen beantwortet und Aufgaben erfüllt werden sollen. Mit der Applikation lassen sich keine nativen Applikationen testen. [Vgl. Use13b]

Im Gegensatz zu *UserTesting* gibt *UserZoom* keine Pauschalpreise an. Ein Rechenbeispiel auf der Webseite des Unternehmens vergleicht Remote Usability Testing mit Labortests. Dabei werden Kosten von bis zu 120 € pro Testbenutzer in Labortests und ca. 10 € für Remote-Testbenutzer berechnet. Die Kostengegenüberstellung am Ende des Artikels, welche eine hypothetische Ersparnis von 40 % berechnet lässt mit 12440 € zu 7600 € darauf schließen, dass sich das Angebot eher an Unternehmen richtet, als an einzelne Entwickler oder kleine Teams. [Vgl. Dar13]

2.3 Mobile Usability Testing Werkzeuge

2.4 Relevante Technologien

2.4.1 Chromecast Screen Mirroring

2.4.2 Android Bildschirmaufzeichnung

3 Konzept

3.1 Anforderungen

An das Projekt ergeben sich die folgenden Anforderungen:

-

4 Realisierung des Servers

4.1 RESTFul Webservice

5 Realisierung der Android Bibliothek

6 Realisierung des Testleiter-Clients

7 Ergebnisse

8 Fazit

8.1 Weiterentwicklung

8.1.1 Integrierte Blickverfolgung

Projekte wie *Opengazer* [Zie09] zeigen eindrucksvoll, dass Eye-Tracking bzw. Gaze-Tracking auch ohne spezielles Equipment möglich ist. Mit Hilfe einer einfachen Webcam gelang dem Entwicklerteam das Verfolgen von Augenbewegungen und die Visualisierung des Fokuspunktes auf dem Computerdisplay. Die quelloffene Software benötigt für die korrekte Funktion das Markieren von Referenzpunkten im Gesicht auf dem Kamerabild, sowie eine Kalibrierung des Blickes mit Referenzpunkten auf dem Display.

Ein schwerwiegender Nachteil der aktuellen Version ist allerdings, dass der Kopf möglichst still gehalten werden und die Webcam fixiert werden muss. Selbst wenn die Software auf mobile Betriebssysteme portiert und ausgeführt werden könnte, wäre es nur schwer möglich die zuvor genannten Bedingungen zu erfüllen. Das Testgerät müsste fixiert werden und auch die Testperson dürfte sich beim Test möglichst wenig bewegen. Die Qualität der Messungen wäre auch bei perfekten Bedingungen fraglich, da schon kleine Messungenauigkeiten den Wert der Messpunkte stark reduzieren würden, da die Displays von mobilen Geräten sehr viel kleiner sind als bei Desktop-PCs oder Notebooks.

Eine alternative zur Blickverfolgung mit der integrierten Kamera von mobilen Geräten könnte *The Eye Tribe* [The14] bieten. Diese bieten einen kompakten Eye-Tracker für \$99 an. Die Hardware ist mit 1,9 x 20 x 1,9 cm Ausmaßen sehr kompakt und mit 70 g auch leicht genug für den mobilen Einsatz, zumindest an Tablets. Aktuell ist die Software nur unter Microsoft Windows lauffähig, eine Android Version befindet sich, laut Herstellerwebsite, in der Entwicklung.

Blickverfolgung auf einem Smartphone ohne externe Hardware könnte schon in naher Zukunft möglich sein. Durch die Verwendung von mehreren Kameras in der Gerätefront kann das *Amazon Fire Phone* [ama14] schon jetzt die Perspektive des Bildschirminhalts dynamisch an den Betrachtungswinkel des Benutzers anpassen. Zwar wird das Eye-Tracking vom aktuellen *Fire Phone SDK*³ (noch) nicht unterstützt, lediglich das Head-Tracking, jedoch ist das Smartphone erst seit kurzem in den U.S.A. verfügbar und der Verkaufstart in Deutschland steht noch aus (stand September 2014). Welche Möglichkeiten die zusätzlichen Kameras bieten und ob sich mit ihnen eine präzise Blickverfolgung realisieren lässt wird die Zukunft zeigen.

³Software Development Kit

9 Quellenverzeichnis

- [ama14] amazon.de, Hrsg. *Amazon Fire Phone (Telekom)*. 2014. URL: <http://www.amazon.de/gp/feature.html?ie=UTF8%5C&docId=1000819063> (besucht am 18.09.2014).
- [Bud14] Raluca Budiu. *Usability Testing for Mobile Is Easy*. Hrsg. von Nielsen Norman Group. 2014. URL: <http://www.nngroup.com/articles/mobile-usability-testing/> (besucht am 18.09.2014).
- [Dar13] Javier Darriba. *5 Gründe, warum Remote Usability Testing günstiger ist als traditionelle Labor-Tests*. Hrsg. von UserZoom GmbH. 2013. URL: <http://www.userzoom.de/tag/kostengunstig/> (besucht am 22.09.2014).
- [The14] The EyeTribe, Hrsg. *The EyeTribe*. 2014. URL: <https://theeyetribe.com/> (besucht am 18.09.2014).
- [Use14a] Inc. UserTesting, Hrsg. *How It Works: UserTesting provides on-demand usability testing*. 2014. URL: <http://www.usertesting.com/how-it-works> (besucht am 22.09.2014).
- [Use14b] Inc. UserTesting, Hrsg. *UserTesting: Mobile Testing*. 2014. URL: <http://www.usertesting.com/mobile> (besucht am 22.09.2014).
- [Use14c] Inc. UserTesting, Hrsg. *UserTesting: The fastest way to get feedback*. 2014. URL: <http://www.usertesting.com/> (besucht am 22.09.2014).
- [Use13a] UserZoom GmbH, Hrsg. *Dienstleistungen: Mehr als eine Software: Wir bieten alles, was Ihnen zum Erfolg verhilft*. 2013. URL: <http://www.userzoom.de/dienstleistungen/> (besucht am 22.09.2014).
- [Use13b] UserZoom GmbH, Hrsg. *Software / Research-Tools: Mobile Usability Testing*. 2013. URL: <http://www.userzoom.de/software/research-capabilities/mobile-usability-testing/> (besucht am 22.09.2014).
- [Zie09] Piotr Zielinski. *Opengazer: open-source gaze tracker for ordinary webcams*. Hrsg. von The Inference Group. 2009. URL: <http://www.inference.phy.cam.ac.uk/opengazer/> (besucht am 18.09.2014).

Anhang

A GUI

Ein toller Anhang.

Screenshot

Unterkategorie, die nicht im Inhaltsverzeichnis auftaucht.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit (entsprechend der genannten Verantwortlichkeit) selbstständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form weder veröffentlicht noch einer anderen Prüfungsbehörde vorgelegt.

Datum:

.....

(Unterschrift)