Usability
im IoT
Requirements
und Test
Was sind die wichtigsten Q-Merkmale
When asked to rank the types of issues most encountered by mobile app users, the following issues were at the top of the list:

- **User Interface Issues**: 58%
- **Performance**: 52%
- **Functionality**: 50%
- **Device Compatibility**: 45%
- **Missing Functionality**: 29%
- **OS Compatibility**: 23%
- **Security**: 14%
- **Backend Services Available**: 10%
- **Backend Services Functionality**: 10%
- **Other**: 4%

*Respondents could choose more than one answer, results may add to more than 100%*
IoT vs Mobile Device

Kriterien an einen Automaten nach DIN EN ISO 9241-110 für gute Usability:
• Aufgabenangemessenheit
• Selbstbeschreibungsfähigkeit
• Erwartungskonformität
• Lernförderlichkeit
• Steuerbarkeit
• Individualisierbarkeit
• Fehlertoleranz

Kriterien gute Usability einer App nach dem Buch Mobile App Testing von Daniel Knott:
• Weniger ist mehr
• Selbsterklärend
• Fehler erlauben
• Wortlaut
• Konsistenz
• Muster
Steuerung Internet of Things
Usability

Usability im IoT

Usability Mobile Devices
Hinweise

Checkliste
Tipps
Fragt, wenn etwas unklar ist, aber ... 
Folien werden anschließend bereit gestellt
Usability im IoT
Requirements und Test
Was ist gute Usability

Gute Usability fällt nicht auf...

Schlechte schon.
Unbelievable #usability problems while changing password of the @Lufthansa_DE account. 5 tries and at the end even a red success note #noUX
British Bad Design. Is it turned on or off?
#usability #ergonomics #design #ux #redundancy

Los geht's #muc17 #usability #ucd

A user interface is like a joke. If you have to explain it, it's not that good.
(Martin LeBlanc)

2 Retweets 6 Gefällt mir^-Angaben
Usability im IoT
Requirements und Test
Zielgruppe / Personas
Fragen zur Zielgruppe

Wer ist meine Nutzerbasis?
Wie alt ist der durchschnittliche Nutzer?
Wieviele Frauen oder Männer sind in meiner Zielgruppe?
Welche Geräte werden von meiner Nutzerbasis am häufigsten verwendet?
Personas?

<table>
<thead>
<tr>
<th>Information</th>
<th>Profil</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Martin</td>
</tr>
<tr>
<td>Geschlecht</td>
<td>männlich</td>
</tr>
<tr>
<td>Alter</td>
<td>28</td>
</tr>
<tr>
<td>Monatliches Einkommen</td>
<td>3.000 €</td>
</tr>
<tr>
<td>Bildungshintergrund</td>
<td>Masterabschluss Informatik</td>
</tr>
<tr>
<td>Wohnort</td>
<td>Hamburg</td>
</tr>
<tr>
<td>Benutzt die folgenden Apps</td>
<td>Twitter, Facebook, Feedly, Spotify</td>
</tr>
<tr>
<td>Überprüft sein Smartphone ... Mal am Tag</td>
<td>alle 15 Minuten</td>
</tr>
<tr>
<td>Besitzt folgende Geräte</td>
<td>LG Google Nexus, iPad Mini</td>
</tr>
<tr>
<td>Persönliche Merkmale</td>
<td>Freundlich, intelligent, politisch, Fußballfan</td>
</tr>
</tbody>
</table>
Szenarien benutzen

Inhalt:
1) Beschreibung des Szenarios - was erwarten die Nutzer, was wird benötigt
2) Beschreibung des Ablaufs
3) Beschreibungen möglicher Fehler und der Umgang damit
4) Informationen über zeitgleich ablaufende Aufgaben
5) Beschreibung des Endes des Szenarios - Zustand der App
Szenario U-Bahn-Überweisung
Er erinnert sich, dass seine neue Bank eine App zur Verfügung stellt.
Er lädt sie mit dem zur Verfügung gestellten W-Lan in der U-Bahn herunter und startet sie.
Martin meldet sich mit den Daten vom Online-Banking an.
Er sucht die Rechnung in seinem Emailprogramm und führt die Überweisung durch.
Storyboard

Benutzerinteraktionen werden bildlich dargestellt.

Martin ist auf dem Weg von der Arbeit nach Hause...

...mit dem W-Lan...und führt die Überweisung durch...

Gut geeignet für besonderen Kontext wie Umgebungen und komplexe Konzepte!
Anwendungsfälle

UML Diagramm - weniger ist mehr!

<table>
<thead>
<tr>
<th>Name des Anwendungsfalls</th>
<th>Überweisung durchführen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ziel des Anwendungsfalls</td>
<td>Überweisung ist auf dem zweiten Konto eingegangen</td>
</tr>
<tr>
<td>Quelle der Anforderung</td>
<td>Max Müller</td>
</tr>
<tr>
<td>Akteure</td>
<td>Benutzer, Kunde</td>
</tr>
<tr>
<td>Auslösendes Ereignis</td>
<td>Benutzer will Überweisung durchführen</td>
</tr>
<tr>
<td>Voraussetzungen</td>
<td>Kunde meldet sich im Laufe des Anwendungsfalls in der App an, gültige PIN und TAN vorhanden</td>
</tr>
<tr>
<td>Interaktionsschritte des Standardfalls</td>
<td>1. Kunde loggt sich in der App ein</td>
</tr>
<tr>
<td></td>
<td>2. Kunde entscheidet sich, was er tun möchte</td>
</tr>
<tr>
<td></td>
<td>3. Kunde füllt benötigte Angaben für die Überweisung in der App aus</td>
</tr>
<tr>
<td></td>
<td>3.a. Empfänger eingeben: Luise Mustermann</td>
</tr>
<tr>
<td></td>
<td>3.b. Empfänger Kontonummer eingeben: 123456</td>
</tr>
<tr>
<td></td>
<td>3.c. Betrag eingeben: 100 €</td>
</tr>
<tr>
<td></td>
<td>3.d. Verwendungszweck Rechnung Nummer: 42</td>
</tr>
<tr>
<td></td>
<td>3.e. PIN und TAN eingeben</td>
</tr>
<tr>
<td></td>
<td>3.f. Absenden Button antippen</td>
</tr>
<tr>
<td>Alternativen</td>
<td>3.c. falscher Betrag: 100 €</td>
</tr>
<tr>
<td>Nachbedingungen</td>
<td>Überweisungsbetrag ist auf dem Konto von Frau Luise Mustermann eingegangen</td>
</tr>
<tr>
<td>Kommentare</td>
<td>Ob die Überweisung auf dem Konto eingegangen ist, kann auch im Backend validiert werden</td>
</tr>
</tbody>
</table>

Quellen: nach Guy Vollmer - Mobile App Engineering (S. 60f) - dpunkt verlag
User Stories
Prototyping
Low Fidelity prototyping
High Fidelity prototyping
Tipps zu den Anforderungen

Wichtig: Formulieren Sie die Anforderungen testbar! Verwenden Sie kurze Sätze und kurze Absätze. Nie mehrere in einer Anforderung!

Nach ISO/IEC/IEEE 29148:2011:
- Eindeutig
- Notwendig
- Konsistent
- Prüfbar
- Realisierbar
- Verfolgbar
- Vollständig

Außerdem:
- Abgestimmt
- Unabhängig
- Verhandelbar
- Nützlich
- Schätzbare
- Verständlich

Quelle: Pohl, Rupp - Basiswissen Requirementsengineering (S47ff) - dpunkt.verlag
Usability
im IoT
Requirements und Test
Usability Test im Labor
Usability Test mit einfachen Mitteln

Szenarien
Heuristiken
Error guessing
Exploratives Testen - Testtouren
Testing “In the Wild“
Tools

Benutzen Sie **alles** für den Test, was Sie bekommen können!!!

Und **alles** aus RE bzw. währendessen:

- Personas, Szenarien,
- Anwendungsfälle, Prototyping...

**Tester sind beim RE dabei!**
**Bsp.: Testen Low Fidelity Prototyp mit entworfenen Szenarien**
Erstellen Sie eigene Checklisten und lassen Sie auch echte Nutzer testen!

Quelle: Guy Vollmer - Mobile App Engineering (S. 208ff)
Heuristiken

1) Systemstatus muss sichtbar sein -> Ladebalken

2) Die Anwendung muss mit der Realität verknüpft sein -> natürliche Sprache, besser Phrasen, noch besser Bilder verwenden

3) Benutzerkontrolle und Freiheit -> Undo/Redo Funktionalität, Falscheingaben müssen schnell korrigierbar sein

4) Konsistenz und Standards -> Nicht „abbrechen“ und später „verwerfen“, Standards der jeweiligen Plattform einhalten

5) Fehler vermeiden -> bei Nutzereingabe bereits auf Fehleingabe prüfen und im Fehlerfall darstellen

6) Erkennen statt Abrufen -> Nutzer soll Funktionen erkennen anstatt sie auswendig zu lernen, intuitive Benutzung!

Quelle: https://www.nngroup.com/articles/ten-usability-heuristics/
7) Für Neulinge und erfahrene Benutzer --> versteckte Expertenfunktionen, erweiterte Einstellungen anbieten

8) Minimalistisches und ästhetisches Design --> nur anzeigen, was wirklich benötigt wird

9) Dem Nutzer helfen eine Fehler zu erkennen, zu analysieren und zu lösen --> keine Fehler-Codes, Klartext verwenden und Lösung vorschlagen

10) Hilfe und Dokumentation anbieten

Quelle: https://www.mngroup.com/articles/ten-usability-heuristics/
Error guessing

Beim “Error Guessing” oder dem “intuitiven Vorgehen” wird versucht, mittels Erfahrungswissen durch reflektives “Nachdenken” weitere Testfälle aufzustellen. Hierbei stützt sich der Tester auf gesammelte Erfahrungen im Bereich der Programmiersprachen, Compiler, Hardware-Abhängigkeiten und dergleichen.

Quelle: http://www.software-kompetenz.de/servlet/is/10751?print=true
Testtouren

Supermodel Tour
Wie bei Supermodels üblich, interessieren Sie sich in diesem Test für die Fassade, bzw. das User Interface. Die Inhalte interessieren Sie weniger. Konzentrieren Sie sich in diesem Test auf die Benutzeroberfläche, und stellen Sie fest, ob alle Bedienelemente richtig angeordnet sind, etc.

Lonely Businessman Tour

Guidebook Tour
In der „Handbuch“ Tour verwenden Sie die Dokumentation, um sich durch die Software zu hangeln (neue Dokumentationen, aber auch Kollwitz-Garders, oder Trainingsmaterialien). Mit diesem Test sollten Sie sicherstellen, dass die Software so arbeitet wie dokumentiert, und dass die Dokumentation selbst keine Fehler enthält. Da es in vielen Ländern gesetzliche Anforderungen an die Produktdokumentation gibt, ist dieser Test in seiner Wichtigkeit nicht zu unterschätzen.

Landmark Tour
Mit der Landmark Tour („Sehenswürdigkeiten“) ist nicht darauf ausgerichtet Ihre Hauptanwendung stoppte durchzutesten (das kann ein automatischer Test viel besser). Vielmehr prüfen Sie hiermit die einzelnen Features in allen möglichen Kombinationen durch, und stellen sicher, dass die Hauptfeatures Ihrer Anwendung in allen möglichen Kombinationen zusammenarbeiten. Für die Testdurchführung wählen Sie die wichtigsten Features Ihrer Anwendung aus, und führen diese jeweils in unterschiedlichen Reihenfolgen aus.

Coach Potato Tour
Coach Potatoes sitzen im echten Leben vor dem Fernseher, essen Chips und schauen einen Film. Die Testmethode zielt darauf ab, ein Feature mit möglichst geringem Testaufwand zu testen. Zum Beispiel geben Sie nur genau soviel Daten ein, wie Sie benötigen, um von einem Screen zum nächsten zu navigieren. Mit diesem Sparsansatz stellen Sie insbesondere sicher, dass die Standardfälle überprüft werden.

Quellen:
https://product-management.net/2014/exploratory-testing-example-testing-nips/
https://product-management.net/2014/exploratory-testing-example-testing-nips/

Personas, Szenarien, Anwendungsfälle aus RE benutzen!
Erstellen Sie eigene Chartas und Session Sheets!
Testing "In the Wild" 👍 🙄
Tools

reflector
ux recorder
(magitest)

oder auch einfach eine Webcam

Exploratives Testen --> Test Chartas, Session Sheets
Typische Fehler

Länderspezifische Datumsangaben

Zu viele und verwirrende Steuerungsmöglichkeiten
Zu viel Inhalt

langer Text

Versteckte Gesten
Tastatur über dem Eingabefeld
Usability im IoT
Requirement und Test
Again what learned!

über Usability re

zum Usability test

- IoT Usability == Mobile Usability (natürlich nicht immer)
- Früh in den Test einsteigen
- Alles nutzen, was vorhanden
- Personas, Szenarien, Storyboards, Anwendungsfälle, User Stories, Prototyping, Stores untersuchen, Heuristiken, Error Guessing, Expl. Testen und Testtouren, Testing "In the Wild", Tools
CPIoT

- Gebrauchstauglichkeit
- Spannungsfeld
Gebrauchstauglichkeit, Wartbarkeit und IT-Sicherheit

https://www.asqf.de/asqf/produkte/asqf-certified-professional-for-iot-cpiot/
Podcast Quality Heroes

bei iTunes, podcast.de und player.fm
Links / Quellen

- https://www.usability.de/
- https://www.usabilityblog.de/usability-tests-selbst-gemacht-die-besten-tipps-und-die-wichtigsten-vornachteile/
- https://www.nngroup.com/articles/ten-usability-heuristics/
- https://www.nngroup.com/articles/mobile-usability-update/
- https://www.interaction-design.org/literature/article/don-t-build-it-fake-it-first-prototyping-for-mobile-apps
- https://uxplanet.org/designing-perfect-text-field-clarity-accessibility-and-user-effort-d03c1e26904b
- https://blog.prototyp.io/10-usability-heuristics-with-examples-4a81ad920c
- http://www.w3.org/WAI/mobile
- https://developer.android.com/training/accessibility/testing.html
- https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/iPhoneAccessibility/Accessibility_on_iPhone/Accessibility_on_iPhone.html
Kontakt

nils.roettger@imbus.de  imbus.de  fb.com/imbus  #netweek@nilsroettger@yourasqf  xing.com/profile/Nils_Roettger

Used template by Prezibase.com