Testing of mobile solutions: „Old wine in new skins?“

Dr. Frank Simon  
Dr. Marcus Iwanowski  

2013-09-25

BLUECARAT.

Background

BLUECARAT®

• Who?
  ◦ SME, 4 Offices
  ◦ 250 Employees, 25 Mio. Euro revenue

• What?
  ◦ IT-Services+Solutions for the complete IT-System lifecycle
  ◦ 6 centers of operation:
Testing

Hetzel 88: “Any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results.” 
(http://www.ece.cmu.edu/~koopman/des_s99/sw_testing/)

Myers (1989) “Software Testing is the process of executing a program or system with the intent of finding errors.”
(http://www.ece.cmu.edu/~koopman/des_s99/sw_testing/)

To test a program is to try to make it fail.

Internet: “The process of exercising software to verify that it satisfies specified requirements and to detect errors.”
(http://www.software-testing-outsourcing.com/glossary.html)
Testing....modern definition

The process consisting of all life cycle activities, both static and dynamic, concerned with planning, preparation and evaluation of software products and related work products to determine that they satisfy specified requirements, to demonstrate that they are fit for purpose and to detect defects. “

Testing: Three characteristics

1/3: Testing is more than testing executables

• “[...] software products and related work products [...]”
• Software: “Computer programs, procedures, rules, and possibly associated documentation and data pertaining to the operation of a computer system” (IEEE Standard Glossary of Software Engineering Terminology /ANSI 83)
• Examples:
  • Libraries
  • Functions
  • Nets
  • Market Places
  • Cloud services
  • Documentation
  • Architectures…

Executable program
Testing: Three characteristics
2/3: Testing is more than testing functionality

• “[…] satisfy specified requirements […]”

It works!

Safe?
Compliant?
Maintainable?
Extendable?
Etc.

Testing: Three characteristics
3/3: Testing has to validate requirements

• “[…] they are fit for purpose […]”

Requirements are verified!

Is this fit for purpose?
Mobile devices enable localizability, location-independence and availability

- Mobile Devices: Definition

It knows where you are!

It doesn't have a home!
Apps on mobile devices don’t differ much from “traditional” software

Mobile Devices: Example

From an old-fashioned testing perspective:
Testing of mobile devices is nothing new (old wine)
The modern testing definition asks for some more adjustments (new skins)

- New Testing requirement
- New testing processes
- New testing environments
- New architecture
- Mobile ecosystem
- New testing objects

Everything exist in variations

Mobile Testing: New skins
Vision
- Mobile Devices werden effizient und effektiv in neuen Geschäftsprozessen eingesetzt und integriert.
- Spezifika mobiler Lösungen werden bewusst und gewinnbringend eingesetzt.
- Mobile Lösungen sind unabhängig von der Vielzahl von Zielplattformen und -technologien und werden in Deutschland effizient und effektiv entwickelt.

Ziel
- Erarbeitung und fortwährende Pflege von Best Practices für die Entwicklung erfolgreicher mobiler Lösungen.
- Bereitstellung einer Plattform zum Wissensaustausch zur Stärkung des IT-Standorts Deutschland.

Aktueller Stand
- Bisher über 50 Best Practices strukturiert gesammelt
- Offenes Netzwerk des Austausches initiiert

In Zusammenarbeit mit

Die Task Force freut sich auf Ihre Unterstützung!
E-Mail: m.fischer@bitkom.org
BLUECARAT.

Pitfall 1.x: Mobile Requirements ignorance
Challenge 1.x: Requirements of mobile solutions change dramatically!

Pitfall 1.x: Mobile-Requirements-Ignorance:
"Mobile solutions are just copies of desktop-apps, so the requirements are pretty much the same as well"

- 1.1: Reuse analysis and design documents and re-implement them on mobile devices
  - Mobile devices enable NEW processes
  - Mobile devices NEED modified processes
- 1.2: Weights for quality attributes can be migrated too
  - USABILITY becomes much more important today
  - SECURITY generates new hidden champions (e.g. Apple's Touch ID sensor)
- 1.3: There exist established quality attribute catalogues
  - Bad ENERGY EFFICIENCY will kill your app rating
  - Google just bought Bump...how about BUMPABILITY?

Example for changed requirements

**Pitfall 2.x: Mobile multiple assets ignorance**

- **Pitfall 2.x: Mobile Multi Assets Ignorance:**
  
  "Quality of mobile solutions depend only on delivered app for one target device in one target context"

- **2.1: There is only one device with one operating system**
  
  - You need 30 different phones to cover 50% of Android versions
  - 30 test branches?
  - Today Windows has 7% coverage; forecast 2015: 30%

- **2.2: The network is working equally world-wide**
  
  - GSM is different from EDGE, is different from UMTS, LTE…
  - Roaming demonstrates new behavior (e.g. strong limits)

- **2.3: Devices have a standardised set of available sensors**
  
  - BlackBerry doesn’t have a 3G-sensor (at least Frank’s doesn’t have one); Apple now has a Touch-ID sensor
  - Samsung Galaxy S IV has built-in temperature and humidity sensors

---

**Challenge 2.x: Each mobile asset that is relevant for quality exist in different versions**

- **Pitfall 2.x: Mobile Multi Assets Ignorance:**
  
  "Quality of mobile solutions depend only on delivered app for one target device in one target context"
Example

Android brand fragmentation

Android screen sizes
(the darker the more frequent)

Apple: OS version by devcie

From:
http://opensignal.com/reports/fragmentation.php
http://blog.mixpanel.com/page/2/

Pitfall 3.x: Mobile architecture ignorance
Challenge 3.x: Interoperability is core of mobile solutions and need architectural investment

- Pitfall 3.x: Mobile Architecture Ignorance: „Interoperability of mobile apps doesn’t need specific investments and comes for free for the whole application lifecycle“

- 3.1: Mobile architecture is built-in
  - Mobile interface landscape changes on a daily basis and have to be monitored (incl. techniques)
  - Abstraction layers only way to handle complexity

- 3.2: External interfaces can be used for testing as well
  - Interfaces are not available for free all the time, you have to care for them
  - Generation of extreme test data (e.g. negative data) coming from interfaces have to be enabled

- 3.3: Mocking of interfaces can be done ad-hoc
  - Shift-Left is not possible without service virtualization
  - Intelligent mocking is a value for its own and should be shared project wide

Example

**Pitfall 4.x: Mobile ecosystem ignorance**

1. **Central market places and MDM-tools override own rules**
   (e.g. ethic and technical constraints)

2. **Monetary "Ligging" is mandatory for all big market places**

3. **Customer's ratings are fast and brute and have to be considered**

4. **Hidden/direct modifications intensifies power of ratings**

5. **Unpredictable changes for all connected (sub-)systems**

6. **Anticipation of requirements no longer than 6 months**

---

**Challenge 4.x: Demand for proactive management of mobile solutions and its development**

- **Pitfall 4.x: Mobile ecosystem ignorance:**
  
  "Having good requirements and a strong straight forward process guarantees great business for my mobile solution"

- **4.1: We define the rules for our apps**

- **4.2: We can manage customer's communication**

- **4.3: We define our development process**

Example

For the benefit of passengers using Apple iOS 6, local area maps are available from the booking office.

Conclusion

values at work.
Conclusion

- Old-fashioned testers won’t recognise any difference between testing desktop applications and mobile apps
  - But they don’t bring much value for a new business
- Modern testers know the specifics of testing mobile apps
  - They know that mobile devices generate new/changed requirements
  - They know that multiple variants of all assets ask for new testing processes
  - They know that testing mobile devices need a centralised testing environment hosting all (virtualised) services
  - They know that new test objects occur on a daily basis and each can generate strong risks
- But modern testers know as well that testing mobile devices can be much fun as well!