

# UI-TESTING FÜR MOBILE PLATTFORMEN

HERAUSFORDERUNG UND CHANCE ZUGLEICH!

von **Markus Tiede**

# ABOUT ME

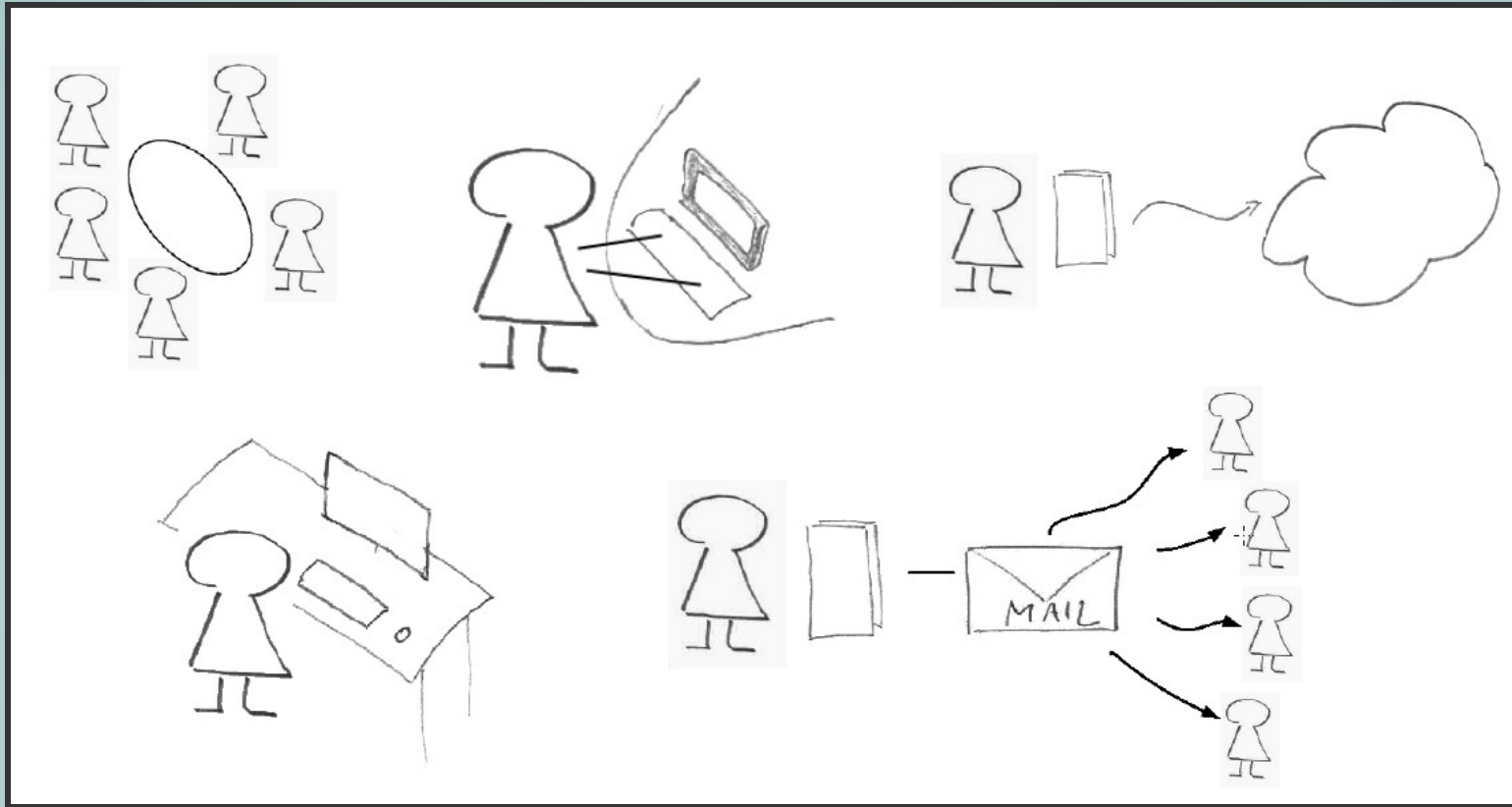
# AGENDA

1. **Das Projekt**  
der Kunde, die App und die Technik
2. **Das Tool**  
die Ziele, die Konzepte und die Technik
3. **Die Erfahrung**  
das Feedback und der Ausblick

# 1. DAS PROJEKT - BREDEX GMBH

- **Entwicklung von Enterprise Anwendungen**  
Plattformunabhängig  
25+ Jahre
- **Mobile Entwicklung:** Geschäftsanwendungen  
Plattformunabhängig  
individuelle Technologieauswahl
  - nativ
  - crossplattform
  - webbasiert
- **Qualitätssicherung & Testautomatisierung**

# 1. DAS PROJEKT - DIE USER STORY



# DEMO!

DIE APP



# 1. DAS PROJEKT - DIE APP

- Erstellung und Verteilung von Meeting-Protokollen
  - Protokoll mobil erfassen
    - Tablet (iOS Android, Windows 8), Laptop
      - Crossplattform IDE: Xamarin Mono
    - Offline Support bei Erstellung von Protokollen
  - Überarbeitung am Arbeitsplatz(optional)
  - Verteilung als PDF per Mail
  - Funktionen zur Protokollverwaltung
  - Automatisierte funktionale Regressionstests
- GUIDancer / Jubula*

## 2. DAS TOOL - DIE ZIELE

mobile UI Toolkits für **GUI**dancer / Jubula  
frei verfügbar und open-source: [testing.bredex.de](https://testing.bredex.de)

**"Code-free keyword-based black-box testing" - mobile!**  
high-level Testschritte + robuste UI Widgeterkennung

**Analog zur Desktop Welt: cross-...**

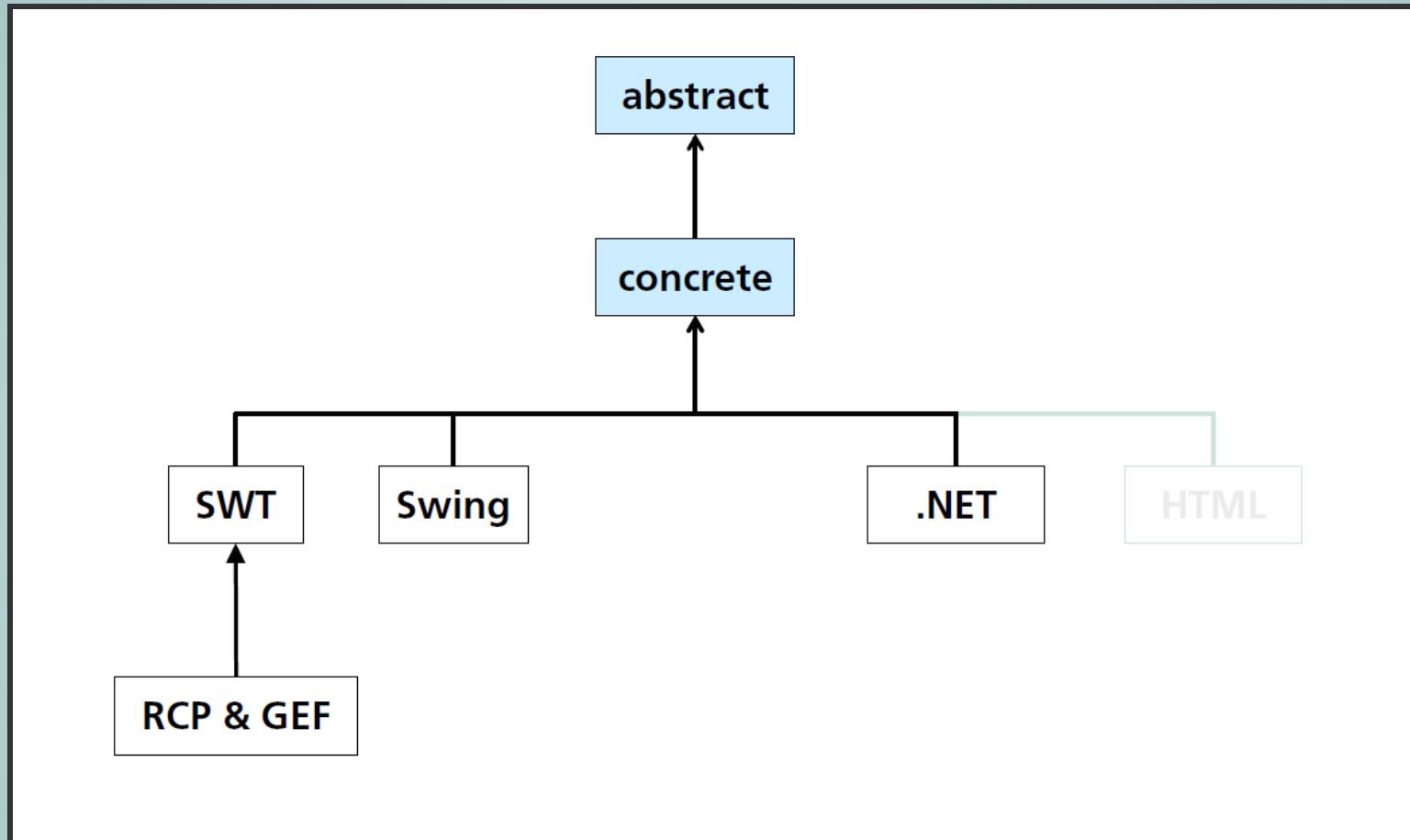
	iOS	Windows	Android
Version	5+	8+	2.3+
Klasse	Tablet / Phone / Hybrid		
Umgebung	Simulator / Hardware		
	Rotation, Netzwerk ...		



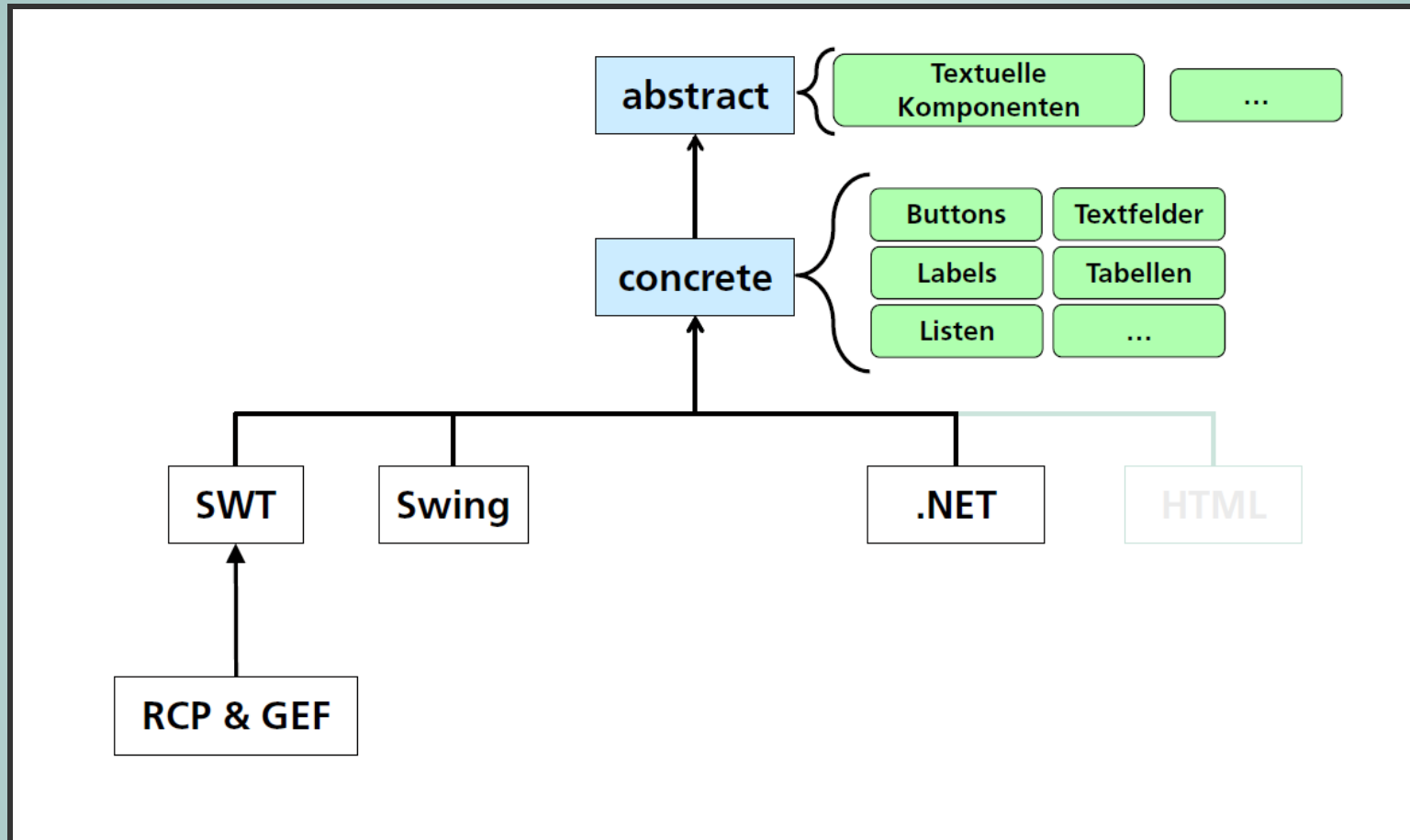
## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



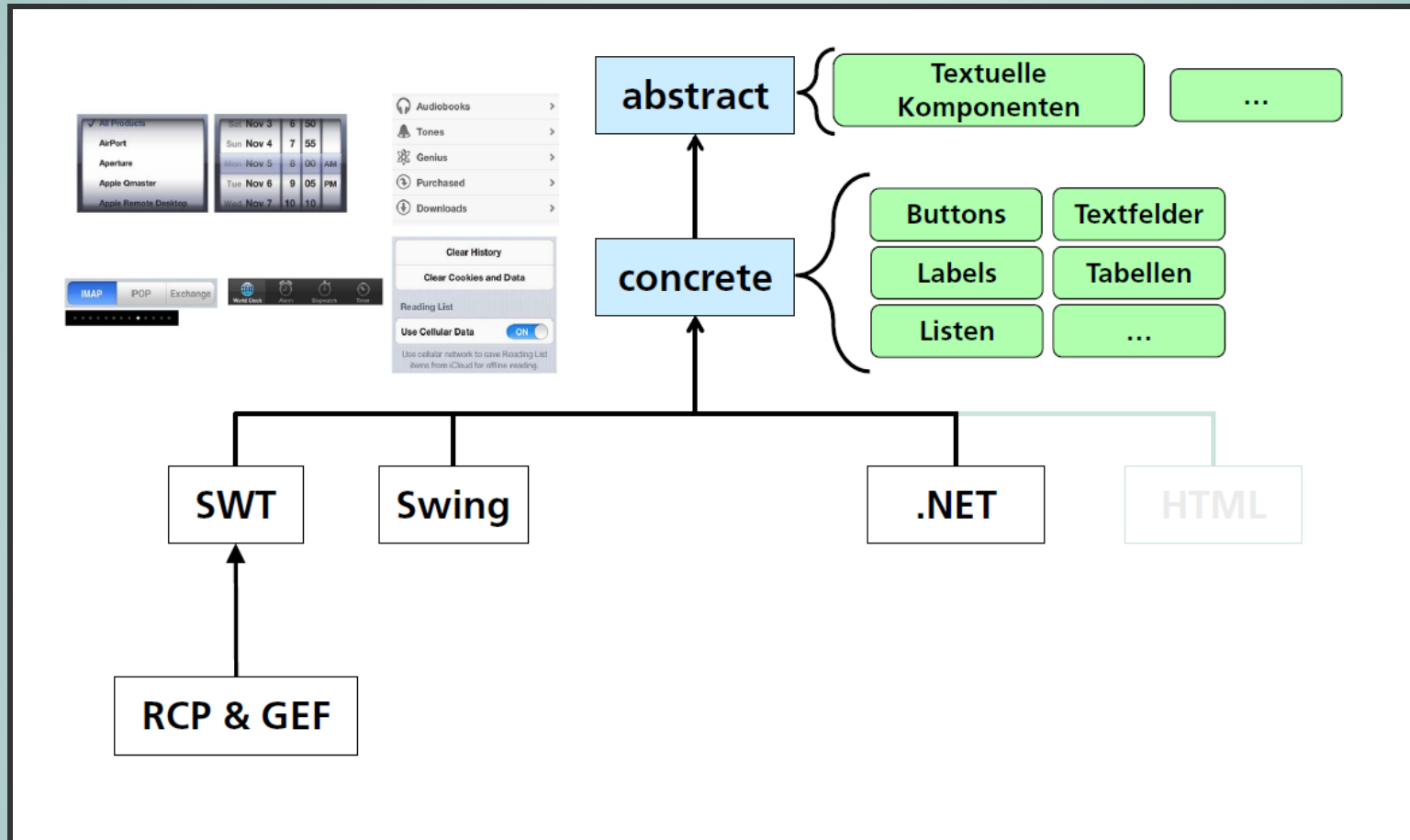
## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



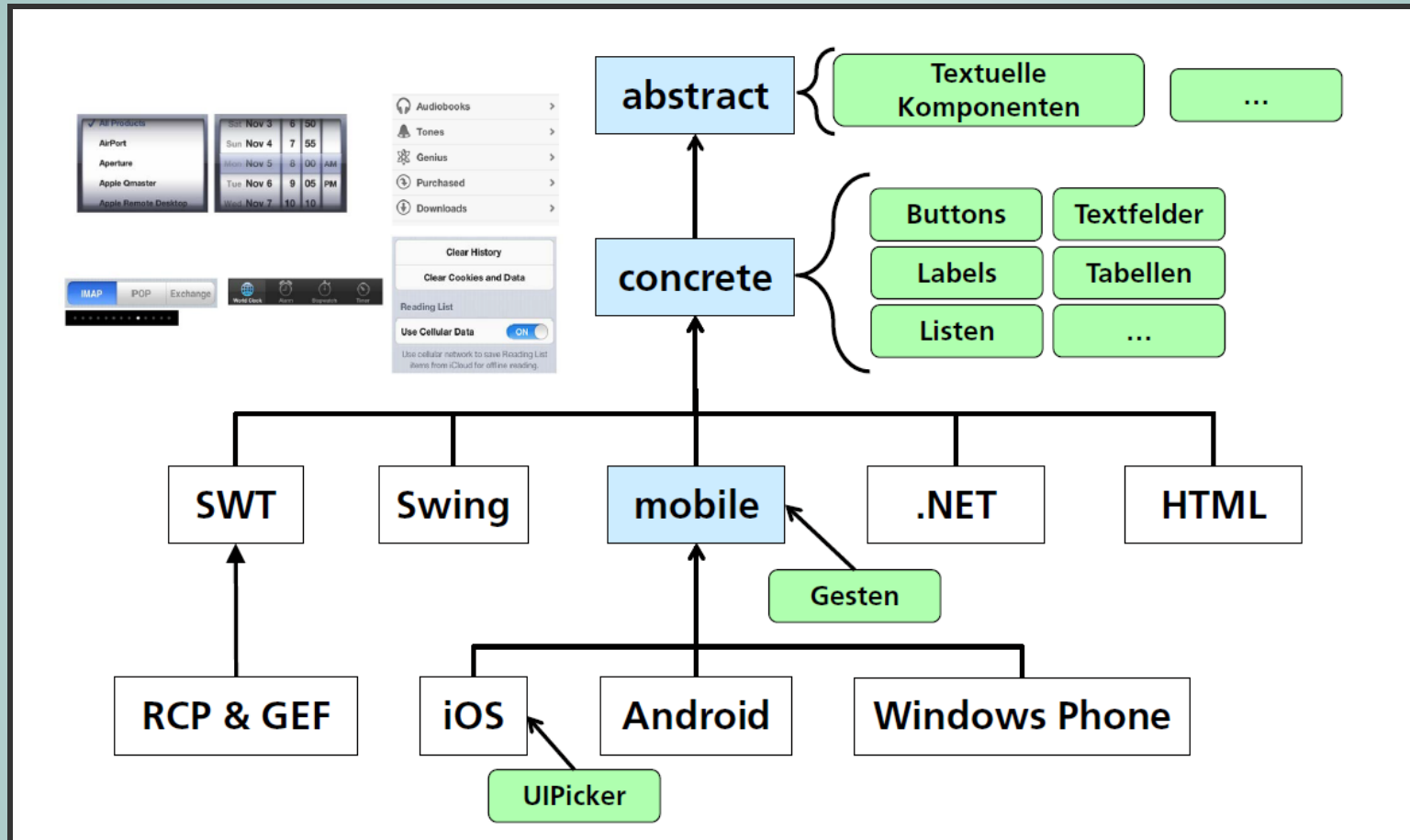
## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



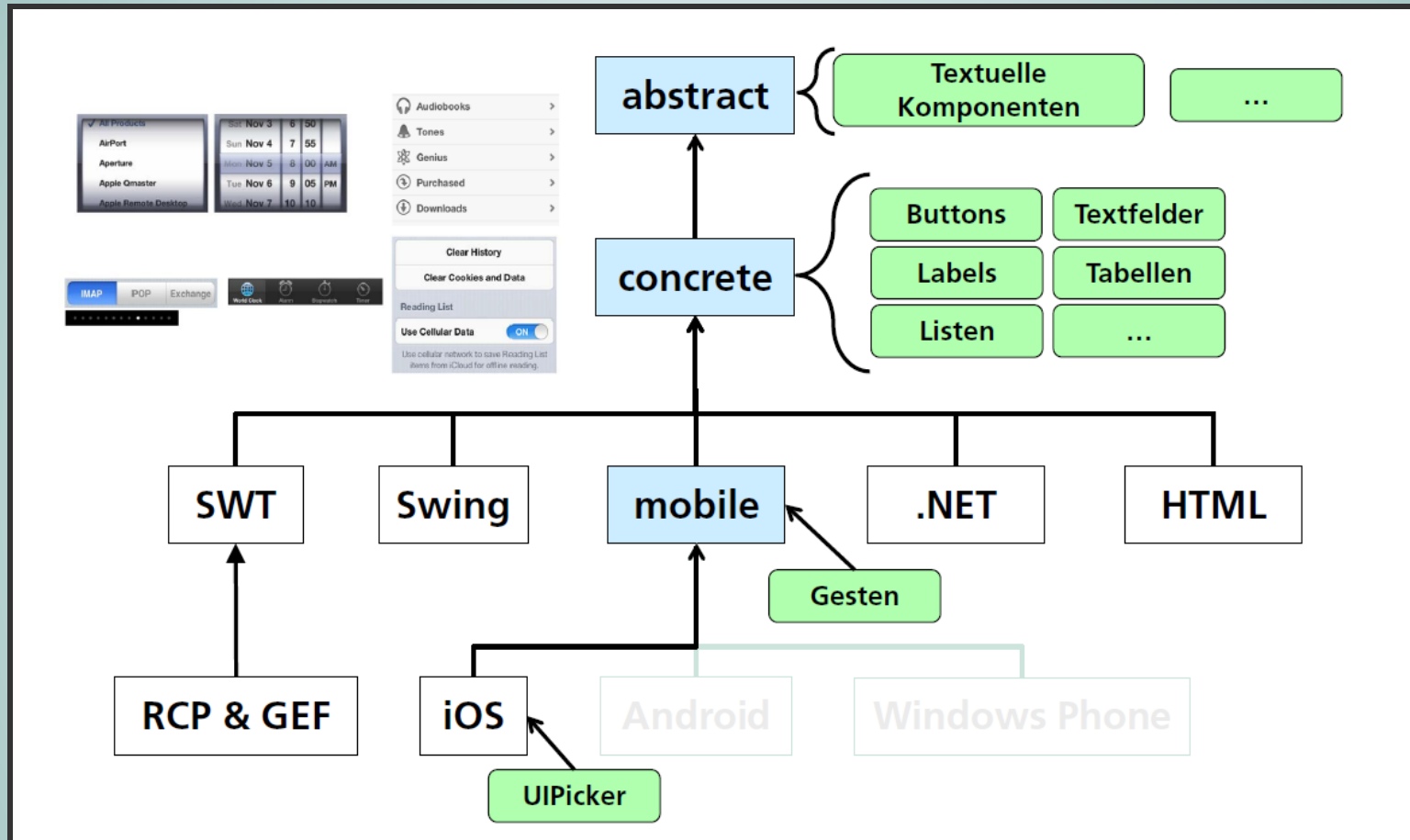
## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION

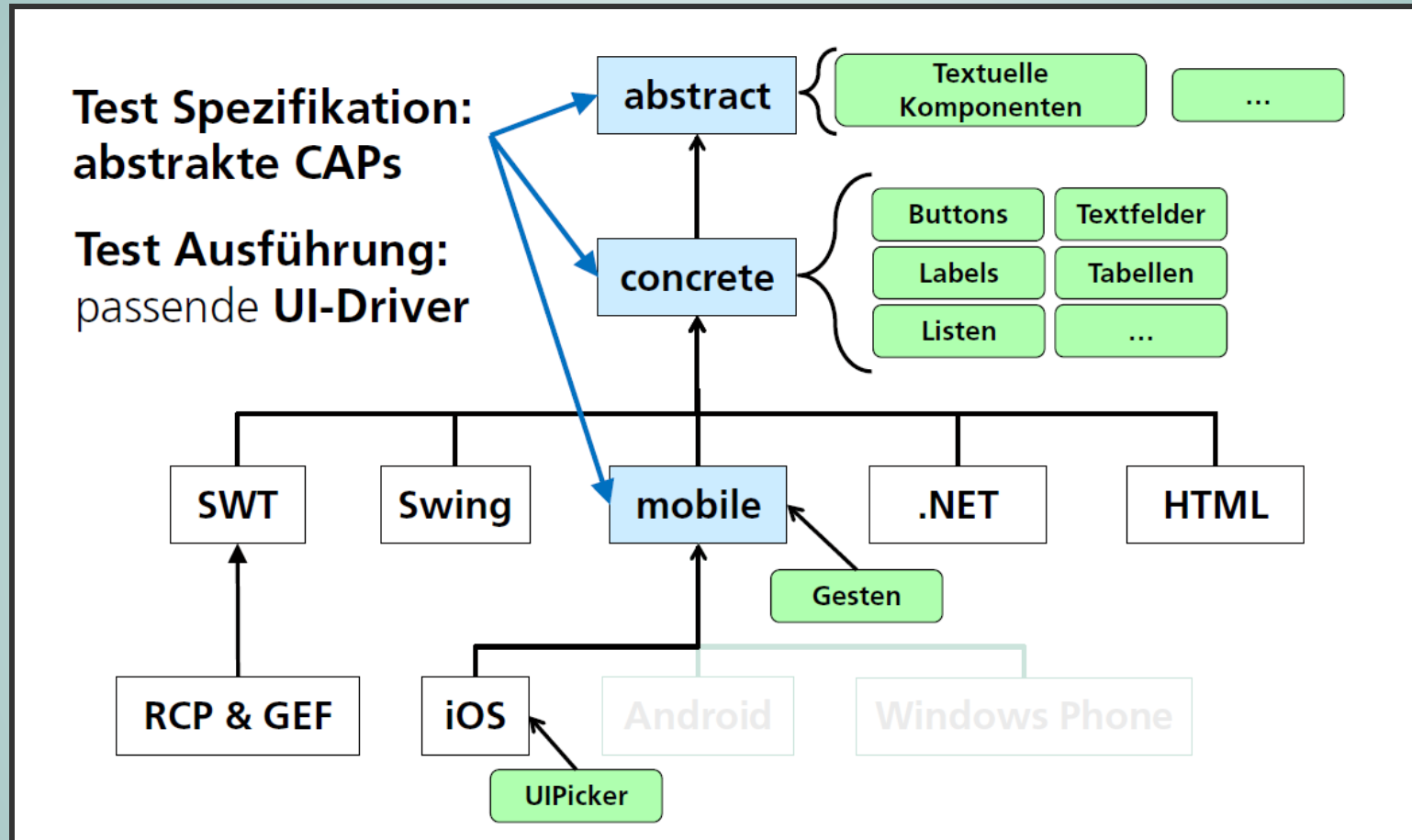


## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION

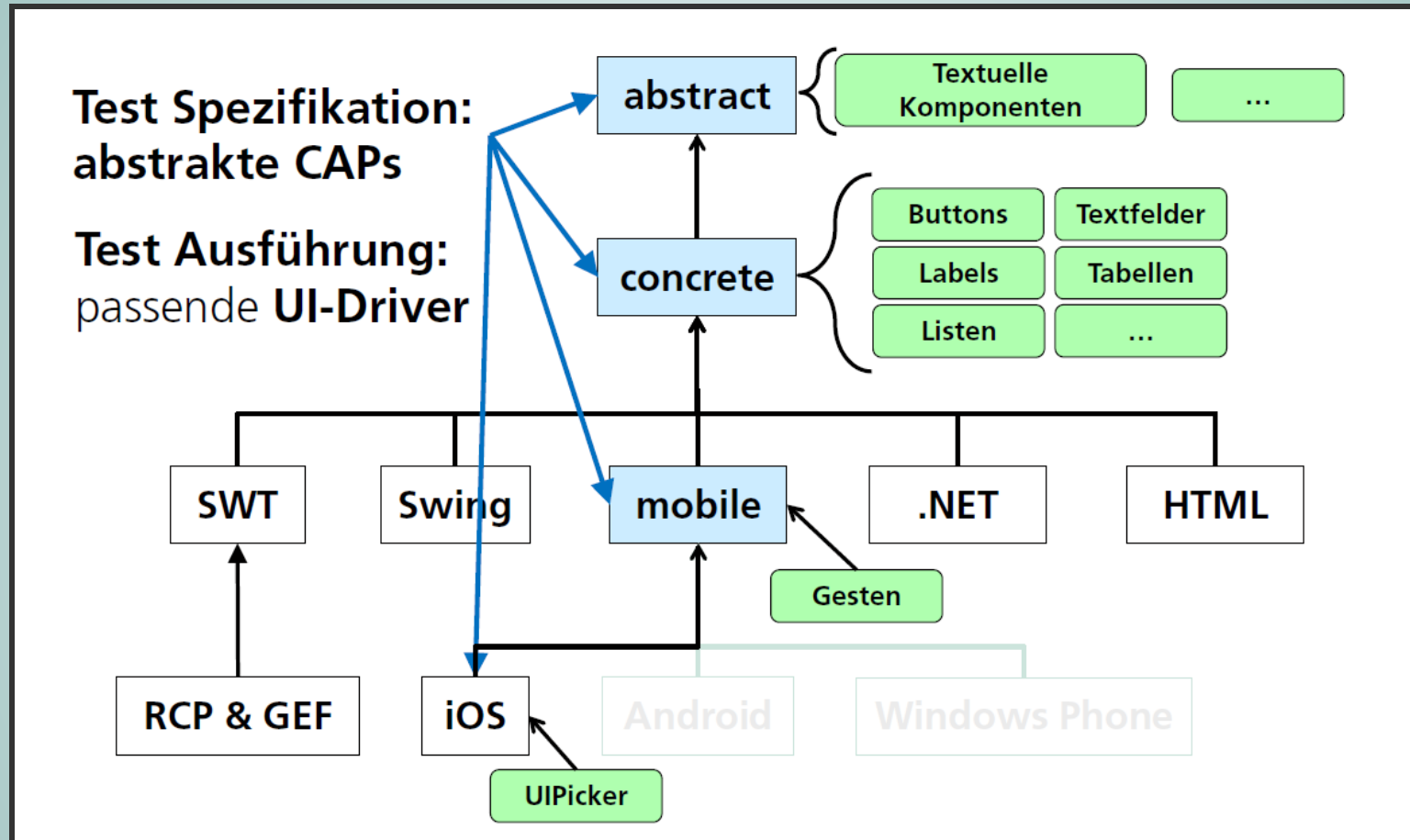




## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



## 2. DAS TOOL - DAS KONZEPT: ABSTRAKTION



## 2. DAS TOOL - DAS KONZEPT: ENTKOPPLUNG

The image shows a screenshot of a software interface divided into two main sections. The left section is titled "Test Suite Browser" and contains a hierarchical tree view of test suites. The right section shows a graphical user interface for a simple adder.

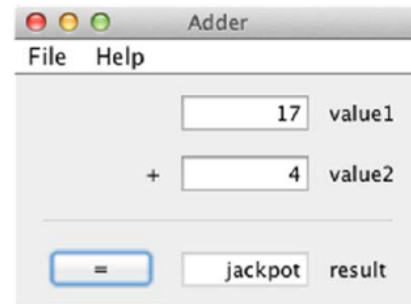
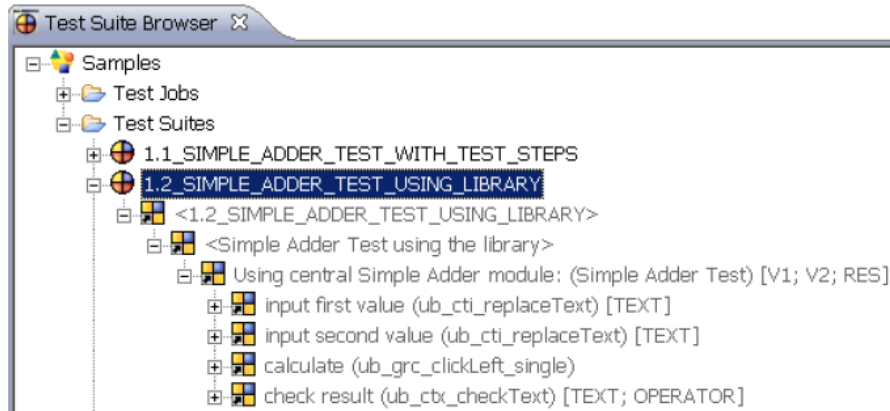
**Test Suite Browser:**

- Samples
  - Test Jobs
  - Test Suites
    - 1.1\_SIMPLE\_ADDER\_TEST\_WITH\_TEST\_STEPS
    - 1.2\_SIMPLE\_ADDER\_TEST\_USING\_LIBRARY (highlighted)
      - <1.2\_SIMPLE\_ADDER\_TEST\_USING\_LIBRARY>
        - <Simple Adder Test using the library>
          - Using central Simple Adder module: (Simple Adder Test) [V1; V2; RES]
            - input first value (ub\_cti\_replaceText) [TEXT]
            - input second value (ub\_cti\_replaceText) [TEXT]
            - calculate (ub\_grc\_clickLeft\_single)
            - check result (ub\_ctx\_checkText) [TEXT; OPERATOR]

**Simple Adder UI:**

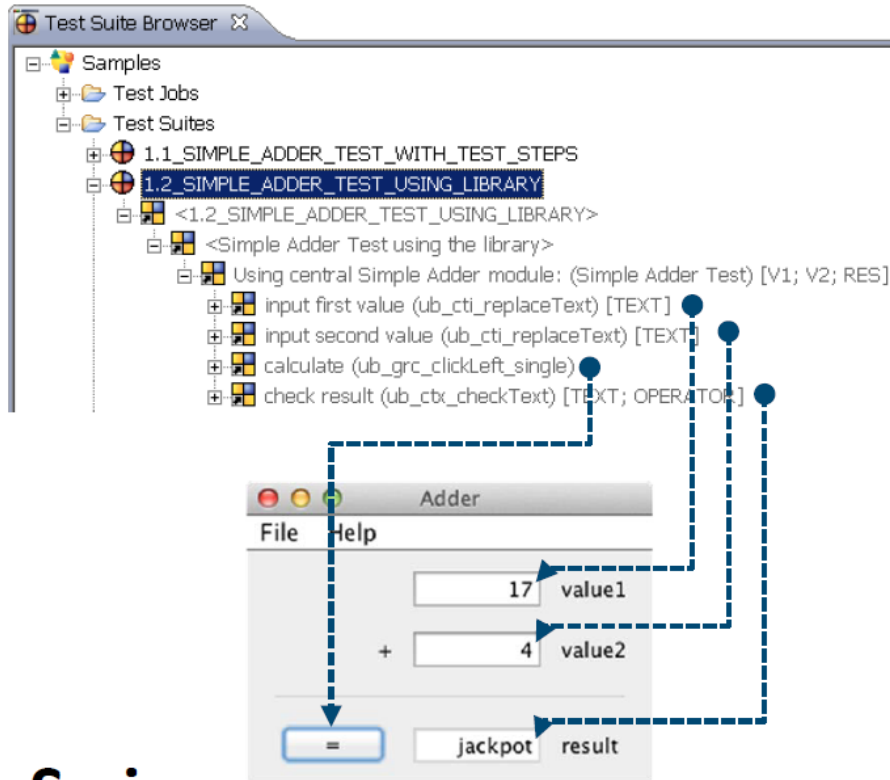
The UI consists of two input fields at the top. The first field contains the number "17" and the second field contains the number "4". A plus sign "+" is positioned between the two fields. Below the input fields is a button labeled "=" and a button labeled "jackpot".

# 2. DAS TOOL - DAS KONZEPT: ENTKOPPLUNG



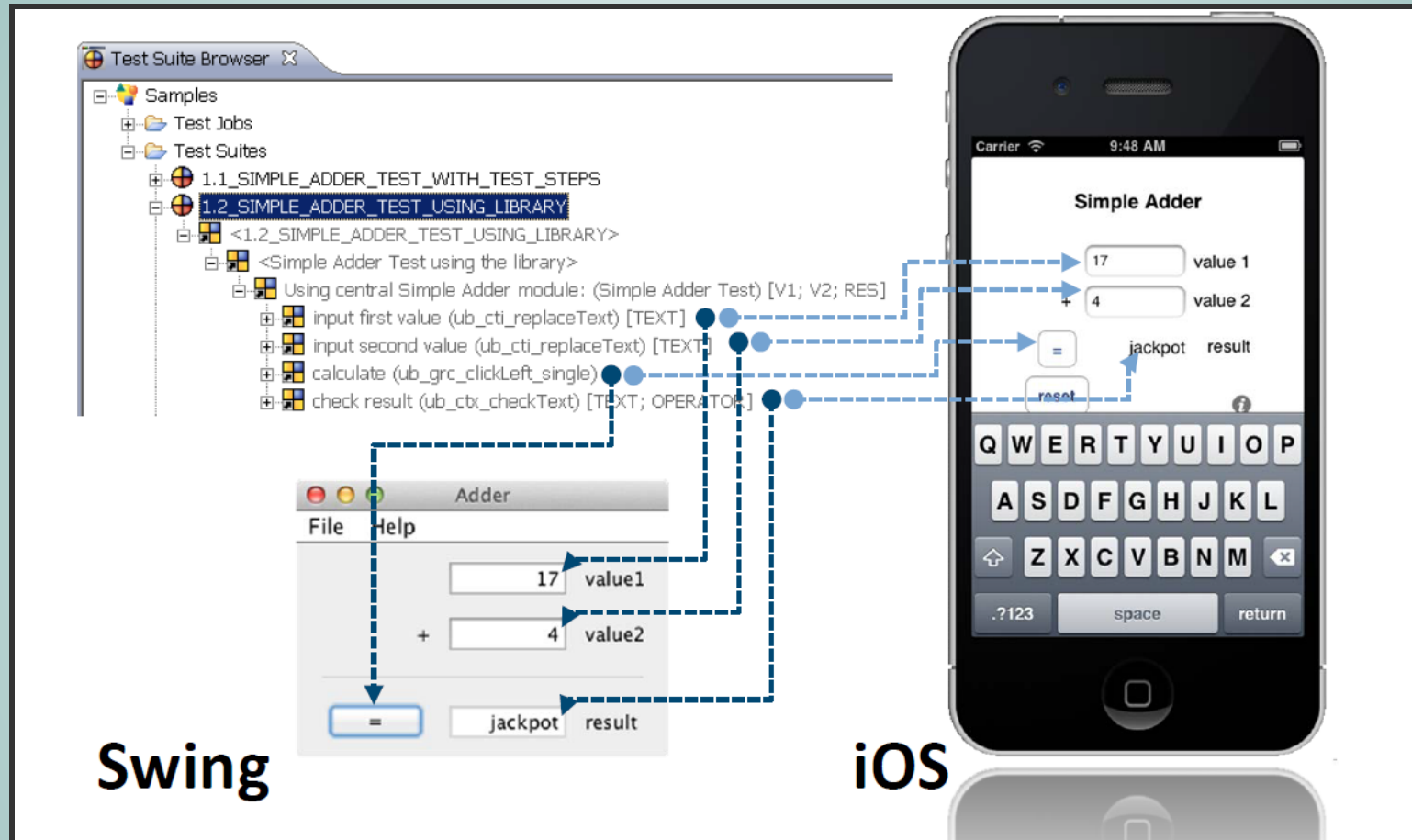
Swing

# 2. DAS TOOL - DAS KONZEPT: ENTKOPPLUNG



Swing

## 2. DAS TOOL - DAS KONZEPT: ENTKOPPLUNG





# DEMO!

## TESTSPEZIFIKATION UND MAPPING

## 2. DAS TOOL - DIE TECHNIK: IOS

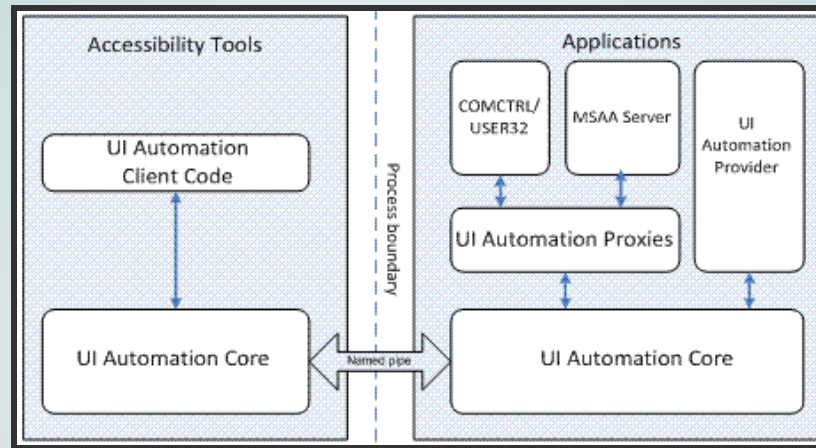
- **Driver Robot API: KIF**  
Keep it functional
- **iOS Sandbox Prinzip**  
AUT Quellcode-Modifikation notwendig  
Einschränkungen bzgl. AUT start / re-start
- **Supported**  
iOS 5+  
Simulator + Devices: iPad (mini), iPhone, iPod, ...  
non-continuous gestures  
Cross-Plattform mobile Frameworks / IDEs: Xamarin Mono

# **DEMO!**

## **TESTLAUF DER PROTOKOLL-APP**

## 2. DAS TOOL - DIE TECHNIK: WINDOWS 8

Driver Robot API: Microsoft UI Automation Framework  
keine cross-UI Toolkit Abstraktion von Microsoft



**Supported:** .NET ab 3.5+

UI Toolkit Winforms; WPF & Modern UI in Planung

OS: alle außer Windows RT

# 3. DIE ERFAHRUNGEN

Erfahrungen aus Protokoll-App und weiterem iOS  
Kundenprojekt

Plattformsupport

Windows, iOS



Android, Windows RT: offen



Aufwand für Integration: gering



**GUI**dancer: aber AUT-Quellcode muss vorliegen

Integration in CI: erfolgreich



Einstieg für Tester: schnell





# 3. DIE ERFAHRUNGEN - DIE STOLPERSTEINE

**Hürden: Keyboardlayouts**

Kontextabhängig, Plattformabhängig

Device-abhängig, Sprachabhängig

Offen: Zeichen erreichbar via "Long Tap"

**Herausforderungen: Navigationskonzepte**

Android / iOS: tabbed user interface

Windows 8 Modern UI: hub design

**Grenzen: Umgebungsparameter nachbilden**

Orientierungswechsel

Connectivity



# **3. DIE ERFAHRUNGEN - DER AUSBLICK**

**Unterstützung für weitere Toolkits / Plattformen**

**Android und Windows Modern UI**

**Kombination von plattformübergreifenden  
und -spezifischen Testschritten**

**if-then-else**

**Abstraktion für unterschiedliche Navigationskonzepte**

# Q&A!