

Betreuer: Prof. Dr. Steffen Avemarg und Dipl.-Inf. Mario Stiffel

Studiengang Angewandte Informatik, Altonaer Str. 25, 99085 Erfurt, Tel. 0361 6700 642, e-mail: informatik@fh-erfurt.de



Livia Schumm

1994 Geboren in Stuttgart
2004-2012 Philipp-Matthäus-Hahn-Gymnasium Leinfelden-Echterdingen
2012-2015 Ausbildung zur Physiotherapeutin
2016-2017 Vollzeitstellung in der Praxis physio4me
Seit 2017 Studium FH-Erfurt Bachelor Angewandte Informatik

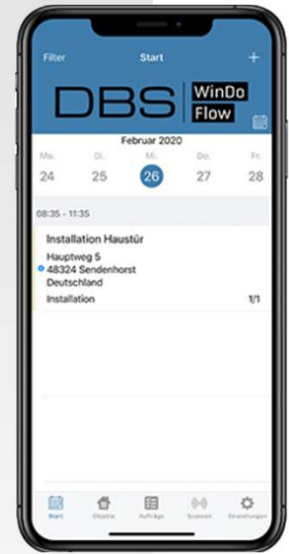
Motivation

Mobile Apps immer und überall nutzen können = ein (fast) selbstverständlicher Anspruch
Allzeit und überall bestehende Netzverbindung ≠ selbstverständlich gewährleistet

Bedeutet? Mobile Applikationen sind bestenfalls offline nutzbar!

Anwendungsfall DBS WinDo Flow – MontageProfi (Abb. rechts)

Eine digital begleitende App für Montagearbeiten, die häufig an Orten ohne zuverlässige Netzverbindung (z.B. auf Baustellen, in Kellern) genutzt wird. Als Vorarbeit für die geplante Integration eines Offlinemodus gilt das Interesse dieser Arbeit dem Informationsgewinn von Umsetzungsmöglichkeiten anhand der Entwicklung und prototypischen Umsetzung eines anwendungsspezifischen Lösungsansatzes.



Offlinefähigkeit

Art und Möglichkeiten der Umsetzung unterscheiden sich anwendungs- und anforderungsspezifisch enorm
→ einen allgemeingültigen Lösungsansatz für Offlinefähigkeit in mobilen Anwendungen gibt es nicht

Es lassen sich jedoch Grundvoraussetzungen feststellen:

1. Zugriff auf Geräteinformationen (Zustand Netzwerkverbindung)
2. Offlineverfügbarkeit von Daten (lokaler Speicher)
3. Synchronisationslogik und Konfliktresolution bei verschiedenen Datenständen eines Datenverwaltungssystems

Prototypische Umsetzung

- Cross-Plattform-App mit Xamarin.Forms
- Einbindung OData-Schnittstelle für Online-Datenzugriff
- Eigenerkennung der Gerätekonnektivität
- Verbindungsunabhängiger Login
- Lokales Speichern und Löschen von Auftragsdaten
- OfflineService-Klasse für Handling von OData-Serviceanfragen je nach Gerätekonnektivität
 - Generalisierung von Online- und Offline-Datenantworten
 - Offlinedaten als Schlüssel – Wert – Paare

OData-Query OData-Response

Evaluation

- Skalierte Bewertung qualitativer Kriterien des Endprodukts
- Gewichtung mittels paarweisem Vergleich

Kriterium	Erfüllungsgrad	Gewichtung	Maximal	Ergebnis
Ausführbarkeit	10	19	190	190
Datenpersistenz	9	19	190	171
Datenkonsistenz	8	17	170	136
Entitätsbeziehungsprinzip	5	16	160	80
Nutzung von Offlinedaten	6	13	130	78
Integrierbarkeit	8	10	100	80
Gerätekonnektivität	9	10	100	90
Datensynchronisierung	1	8	80	8
Manuelle Offlinebereitstellung	8	7	70	56
Nutzungsschnittstelle	9	6	60	54
Datensicherheit	5	6	60	30
Bilder und Dateien	0	1	10	0
Summe			1.320	973

Insgesamt **73,7% Erwartungskonformität** erfüllt

Ableitung der Verwertbarkeit für MontageProfi

- Erkennung **Gerätekonnektivität** *uneingeschränkt verwertbar*
- Integriertes Properties-Wörterbuch des Frameworks zuverlässig als persistenter Offlinespeicher aber ohne Verschlüsselungsmöglichkeit für sensible Daten
→ für **Login** in der Praxis *ohne eigene Verschlüsselung nicht geeignet*
- **Offline-Strategie mit lokalem Schlüsselwertspeicher funktioniert**
→ für exemplarische Darstellung relationaler Offlinedaten *geeignet*
→ für komplexere Datenmodelle / viele Entitätsbeziehungen *zu umständlich*
- **OfflineService-Klasse als Offline-logik integrier- und nutzbar**
→ Persistenz des Offlinespeichers sehr gut und verwertbar
- Hohe Konsistenz der Prototypdaten aber nicht gut verwertbar für Anforderungen des Originals
→ Prototyp ohne Bearbeitung von Daten
→ MontageProfi mit notwendiger Offline-Bearbeitung von Daten:
Geringes Potenzial zur **Erhaltung der Datenkonsistenz** ohne geeignete Synchronisierung und Konfliktbewältigung verschiedener Datenstände

Schlussfolgerung

Als elementare Erkenntnis, lässt sich der entwickelte Ansatz mittels Offlinedaten als Schlüssel-Wert-Paaren für den gewünschten MontageProfi-Offlinemodus als theoretisch umsetzbar, aber nicht umfassend geeignet oder effizient bewerten.

Empfehlung

Anhand der gewonnenen Informationen sollte der Offlinemodus stattdessen mit einer tatsächlich relationalen SQLite Datenbank umgesetzt werden. Die Komponenten des Prototyps können als mögliche Ergänzung dienen.

Ausblick Prototyp

- Automatisierte dauerhafte Synchronisierung
- Bearbeitungsfunktionen
- Logindaten Verschlüsselung / lokale Speicheralternativen
- Vergleich zu Umsetzung mittels lokaler Datenbank