

Systemtechnik

5HIT 2017/18

Synchronisation bei mobilen Diensten

Laborprotokoll

Marvin Ertl

18. April 2018

Note:

Betreuer: Michael Borko

Version: 1.0

Begonnen: 14.04.2018

Beendet: 17.04.2018

Inhaltsverzeichnis

1	Einführung	3
1.1	Ziele	3
1.2	Voraussetzungen	3
1.3	Aufgabenstellung	3
1.4	Bewertung	3
2	Plattform	5
2.1	Couchbase	5
3	Umsetzung	6
3.1	Couchbase Server	6
3.2	Sync Gateway	6
3.3	Couchbase Lite	7
	Literaturverzeichnis	8

1 Einführung

Diese Übung soll die möglichen Synchronisationsmechanismen bei mobilen Applikationen aufzeigen.

1.1 Ziele

Das Ziel dieser Übung ist eine Anbindung einer mobilen Applikation an ein Webservices zur gleichzeitigen Bearbeitung von bereitgestellten Informationen.

1.2 Voraussetzungen

- Grundlagen einer höheren Programmiersprache
- Grundlagen über Synchronisation und Replikation
- Grundlegendes Verständnis über Entwicklungs- und Simulationsumgebungen
- Verständnis von Webservices

1.3 Aufgabenstellung

Es ist eine mobile Anwendung zu implementieren, die einen Informationsabgleich von verschiedenen Clients ermöglicht. Dabei ist ein synchronisierter Zugriff zu realisieren. Als Beispielimplementierung soll eine Einkaufsliste gewählt werden. Dabei soll sichergestellt werden, dass die Information auch im Offline-Modus abgerufen werden kann, zum Beispiel durch eine lokale Client-Datenbank.

Es ist freigestellt, welche mobile Implementierungsumgebung dafür gewählt wird. Wichtig ist dabei die Dokumentation der Vorgehensweise und des Designs. Es empfiehlt sich, die im Unterricht vorgestellten Methoden sowie Argumente (pros/cons) für das Design zu dokumentieren.

1.4 Bewertung

- Gruppengröße: 1 Person
- Anforderungen "**Grundkompetenz überwiegend erfüllt**"
 - Beschreibung des Synchronisationsansatzes und Design der gewählten Architektur (Interaktion, Datenhaltung)
 - Recherche möglicher Systeme bzw. Frameworks zur Synchronisation und Replikation der Daten
 - Dokumentation der gewählten Schnittstellen
- Anforderungen "**Grundkompetenz zur Gänze erfüllt**"
 - Implementierung der gewählten Umgebung auf lokalem System
 - Überprüfung der funktionalen Anforderungen zur Erstellung und Synchronisation der Datensätze
- Anforderungen "**Erweiterte-Kompetenz überwiegend erfüllt**"
 - CRUD Implementierung

- Implementierung eines Replikationsansatzes zur Konsistenzwahrung
- Anforderungen **Erweiterte-Kompetenz zur Gänze erfüllt"**
 - Offline-Verfügbarkeit
 - System global erreichbar

2 Plattform

Um eine Anwendung zu entwickeln, welche es erlaubt, Daten zu synchronisieren und eine Offline-Verfügbarkeit mittels einer lokalen Client-Datenbank zu Verfügung zu stellen, gibt es einige unterschiedliche Ansätze.

Wie bereits im Unterricht besprochen bietet diff-sync eine einfache Möglichkeit mittels NodeJS Daten zu synchronisieren. Weiters gibt es unzählige Plattformen die solche Features anbieten, unter anderem Googles Firebase. Diese erlaubt es einfach, eine App zu entwickeln, welche die Daten mit einer Firebase Datenbank synchronisieren. [2]

Schlussendlich wurde Couchbase als Plattform zur Umsetzung dieser Übung gewählt, da es mittels Sync Gateway und Couchbase Lite die umfangreichste und die meisten Funktion bietet. Ebenso unterstützt Couchbase sehr viele unterschiedliche Systeme und Programmiersprachen. Unter anderem auch C# mit UWP, wobei die mit der UWP und .Net Plattform entwickelten Anwendungen mittels Xamarin auch auf Android und IOS laufen können. Da bereits einiges an Erfahrung mittels der Laborübung mit der UWP Plattform gesammelt worden ist, wurde für diese Aufgabe ebenfalls auf diese Plattform gesetzt.

2.1 Couchbase

Zur Umsetzung solch einer Anwendung mittels Couchbase werden 3 Komponenten benötigt.

- **Couchbase Server:** Ein NoSQL Server, welcher zentral alle Daten verwaltet.
- **Sync Gateway:** Dies stellt die Schnittstelle zwischen den Server und den Clients dar. Sync Gateway bietet eine REST-Schnittstelle auf den Server zur Synchronisierung der Datensätze.
- **Couchbase Lite:** Stellt die Client Datenbank zu Verfügung. Kann offline verwendet werden, sobald eine Verbindung hergestellt werden kann, werden die Datensätze mittels Sync Gateway synchronisiert.

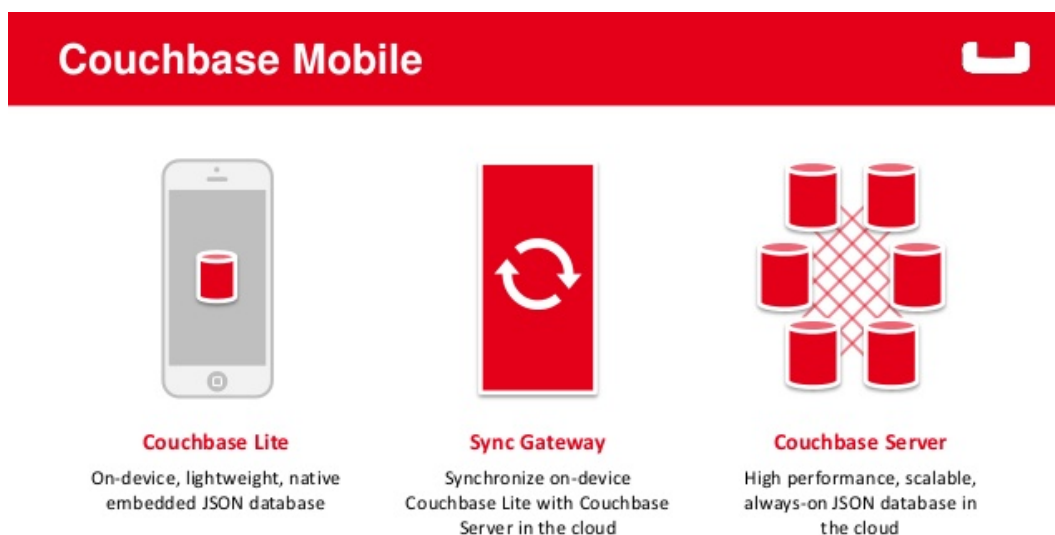


Abbildung 1: Umsetzung mittels Couchbase

Einer der großen Vorteile dieser Plattform ist, dass der Client selber nie mit dem Server im Hintergrund kommuniziert bzw. umgesetzt werden muss. Die App selber greift nur auf die lokale Datenbank zu und nur diese Zugriffe müssen umgesetzt werden, die Synchronisierung übernimmt Sync Gateway und das Abspeichern der Couchbase Server. [3]

3 Umsetzung

3.1 Couchbase Server

Zunächst wurde ein Couchbase Enterprise Server auf einen Server aufgesetzt, verwendet wurde die Version 5.1. Die Enterprise Software darf kostenlos zur Entwicklung und Testen verwendet werden, erst beim normalen Betrieb muss eine Lizenz erworben werden.

Der Server mit der Datenbank ist unter der IP-Adresse 37.252.185.24 und dem Port 8091 erreichbar. Beim Aufsetzen wurde ein neuer Cluster erstellt und die RAM Zuweisung wurde verringert, ansonsten wurden alle Einstellung bei den default Werten belassen.

Über diese Adresse kann nun auf das Admin Dashboard des Couchbase Servers zugegriffen werden. Über dieses wurde ein Benutzer *UserEin* erstellt, welche volle Zugrissrechte auf den Bucket *einkaufsliste* hat. [4]

3.2 Sync Gateway

Wie auch der Couchbase wurde nun Sync Gateway auf dem Server installiert. Auch hier muss kaum etwas konfiguriert werden, nur beim erstmaligem Starten des Services muss eine Konfigurationsdatei angegeben werden.

```
1  {
2      "log": ["*"],
3      "databases": {
4          "db": {
5              "server": "http://localhost:8091",
6              "bucket": "einkaufsliste",
7              "username": "UserEin",
8              "password": "Einkaufsliste",
9              "users": { "UserEin": { "disabled": false, "admin_channels": ["*"], "password":
10                  ↪ "Einkaufsliste" } },
11              "sync": function(doc) {
12                  channel("liste");
13              }
14          }
15      }
16  }
```

Diese JSON gibt den Namen der Datenbank an *db*, weiters muss der Couchbase Server angegeben werden, dieser ist lokal aufrufbar. Nun wurde am Server ein Bucket mit dem Namen *einkaufsliste* erstellt, dieser Name muss auch in der Konfiguration angegeben werden. Weiters muss der Benutzer sowie sein Passwort angegeben werden sein, sodass sich der Service mit den Server kommunizieren kann. Abschließend gibt eine *sync* Funktion an, welche Dokumente synchronisiert werden sollen. Hierbei kann mittels

Tags gearbeitet werden, sodass ein Client nur bestimmte Dokumente synchronisiert, in diesem Fall wurden alle Dokumente dem Channel *liste* hinzugefügt.

Ob alles richtig funktioniert hat, kann einerseits anhand der Meldung in der Konsole überprüft werden, andererseits muss die Adresse mit dem Port 4984, bei einer erfolgreichem Konfiguration folgendes zurückliefern.

```
1  {"couchdb": "Welcome", "vendor": {"name": "Couchbase Sync  
    ↳ Gateway", "version": "2.0"}, "version": "Couchbase Sync Gateway/2.0.0(832;2d8a6c0)"}
```

Wichtig zu beachten ist, sobald Sync Gateway aufgesetzt worden ist, sollte das Admin Dashboard des Couchbase Servers nicht mehr zum Verwalten der Dokumente verwendet werden, dies könnte nämlich zu Problemen bei der Synchronisierung zu den Clients führen. Sync Gateway bietet eine REST-Schnittstelle, über welche alle Dokumente im Bucket verwaltet werden können. [1]

3.3 Couchbase Lite

Wie bereits oben erwähnt wurde bei dieser Übung auf die UWP Plattform zurückgegriffen, da bei der letzten Übung bereits einiges an Erfahrung gesammelt worden ist und die Entwicklung sowie das Testen einfacher und schneller ist, da die Anwendung auch lokal getestet werden kann und nicht in einem Emulator oder auf einem Smartphone getestet werden muss.

Um C

Literaturverzeichnis

- [1] *Couchbase*. [Online; accessed 17. Apr. 2018]. 2018. URL: <https://developer.couchbase.com/documentation/mobile/2.0/installation/sync-gateway/index.html>.
- [2] *Firebase*. [Online; accessed 17. Apr. 2018]. 2018. URL: <https://firebase.google.com>.
- [3] *Offline-first apps with Couchbase Sync Gateway | The Couchbase Blog*. [Online; accessed 17. Apr. 2018]. 2018. URL: <https://blog.couchbase.com/offline-first-apps-couchbase-sync-gateway>.
- [4] *Why Couchbase?* [Online; accessed 17. Apr. 2018]. 2018. URL: <https://developer.couchbase.com/documentation/server/5.1/introduction/intro.html>.

Abbildungsverzeichnis

1	Umsetzung mittels Couchbase	5
---	---------------------------------------	---