

[Modul]

Verfasser: **Leonhard Stransky, 4AHIT**

Datum: **DD.MM.YYYY**

Aufgabenstellung CouchDB – PouchDB

Aufgabe GK/EK:

In dieser Übung wollen wir genau so ein System simulieren.

Ihr findet im Anhang ein Beispiel für ein System aus einer CouchDB (serverseitig) als auch eine PouchDB (clientseitig).

Ihr sollt dieses System mit Hilfe eines JS Frameworks (Beispiele für Frameworks findet Ihr unterhalb) jetzt dahingehend anpassen, dass es möglich ist Artikel per QR Code einzuscannen. Natürlich dürft ihr auch andere Frameworks als den im Kurs angegebenen nutzen, die beiden vorgestellten sollen euch lediglich einen Anhaltspunkt bieten.

Eure Applikation (HTML5, CSS, JS) am Mobilgerät liest entsprechend die Informationen aus dem QR Code aus und ordnet sie bestehenden Einträgen zu bzw. schlägt, wenn diese nicht vorhanden sind neue Einträge vor.

Erstellt hierfür eine minimale CouchDB in einem Docker Container, aus der ihr einfache Daten auslesen, bzw. in die ihr schreiben könnt. Sorgt dafür, dass die Daten dann synchronisiert werden wenn eine Verbindung besteht, aber über die PouchDB eine offline Verfügbarkeit auch gegeben ist.

Die Beurteilung findet wie immer in einem persönlichen Abnahmegespräch statt. Bei dieser Übung gilt "je mehr und besser ihr euer Lösung umsetzt desto - EK"

Eure Ergebnisse gebt hier als ZIP Datei zusammengepackt ab.

Umsetzung

1. Launching a CouchDB Docker Container

```
docker run --name dev-couchdb -p 5984:5984 -e COUCHDB_USER=admin -e COUCHDB_PASSWORD=root -d couchdb:latest
```

2. Check if the CouchDB is running

```
curl localhost:5984
```

3. Access the CouchDB Web Interface under:

```
http://localhost:5984/_utils/
```

4. Create a new Database

```
curl -X PUT http://admin:password@127.0.0.1:5984/storage
```

```
# Output  
# {"ok":true}
```

5. Checking List of Databases

```
curl -X GET http://admin:password@127.0.0.1:5984/_all_dbs
```

```
# Output  
# ["storage"]
```

6. Setting up CORS

```
http://localhost:5984/_utils/#_config/nonode@nohost/cors
```

7. Create Docker-Compose File

```
version: '3.8'  
  
services:  
  couchdb:  
    image: couchdb:latest  
    container_name: dev-couchdb  
    ports:  
      - "6984:6984"  
    volumes:  
      - ./ssl_keys:/opt/couchdb/etc/local.d  
    environment:  
      COUCHDB_USER: admin  
      COUCHDB_PASSWORD: root  
      COUCHDB_HTTPS_KEY_FILE: /opt/couchdb/etc/local.d/server.key  
      COUCHDB_HTTPS_CERT_FILE: /opt/couchdb/etc/local.d/server.crt  
      COUCHDB_HTTPD_BIND_ADDRESS: 0.0.0.0
```

8. Create local.ini File

[daemons]

```
httpd = {chttpd, start_link, [https]}
```

[ssl]

```
cert_file = /opt/couchdb/etc/local.d/server.crt
```

```
key_file = /opt/couchdb/etc/local.d/server.key
```

```
port = 6984
```

[cors]

```
origins = https://yourdomain.com
```

```
credentials = true
```

```
methods = GET, POST, PUT, DELETE, OPTIONS
```

```
headers = accept, authorization, content-type, origin, referer, x-csrf-token
```

[httpd]

```
enable_cors = true
```

9. Fixing Error (database_does_not_exist)

```
dev-couchdb | {database_does_not_exist,[{mem3_shards,load_shards_from_db,
[<<"_users">>],[{file,"src/mem3_shards.erl"},{line,430}]}],
{mem3_shards,load_shards_from_disk,1,[{file,"src/mem3_shards.erl"},{line,405}]}],
{mem3_shards,load_shards_from_disk,2,[{file,"src/mem3_shards.erl"},{line,434}]}],
{mem3_shards,for_docid,3,[{file,"src/mem3_shards.erl"},{line,100}]}],
{fabric_doc_open,go,3,[{file,"src/fabric_doc_open.erl"},{line,39}]}],
{chttpd_auth_cache,ensure_auth_ddoc_exists,2,[{file,"src/chttpd_auth_cache.erl"},
{line,214}]}],{chttpd_auth_cache,listen_for_changes,1,
[{file,"src/chttpd_auth_cache.erl"},{line,160}]}}}]}
```

Manually create the databases

```
curl -X PUT http://admin:password@127.0.0.1:5984/_users
curl -X PUT http://admin:password@127.0.0.1:5984/_replicator
```

1. Download OpenSSL

```
winget install --id=ShiningLight.OpenSSL -e
```

11. Generate SSL Certificates

```
# Generate a Private Key
openssl genrsa -out server.key 2048
```

```
# Generate a Self-Signed Certificate  
openssl req -new -x509 -days 365 -key server.key -out server.crt -sha256
```

12. Test HTTPS Connection

```
curl -k https://admin:password@localhost:6984/
```

Quellen:

[1] <https://www.geeksforgeeks.org/create-a-qr-code-scanner-or-reader-in-html-css-javascript/>

[2] <https://blog.minhazav.dev/research/html5-qr-code>

[3] <https://pouchdb.com/guides/setup-couchdb.html>

[4] <https://docs.couchdb.org/en/stable/>