# Anforderungsdefinition

Unser Datenbanksystem sollte eine kleine und private Datenbank für eine Jugendherberge in der Cloud zur Verfügung stellen. Hier liste ich auf, was wir dafür brauchen/machen werden.

**1. Lokale DBMS (z.B. MariaDB auf XAMPP)**

* **ERD 2.NF:**
  + Backpacker-Schema
* **Zugriffsberechtigungen:**
  + Zugriffsmatrix
  + Gemäß Zugriffsmatrix (mindestens ein Benutzer pro Gruppenrolle)
  + Testprotokolle für Rollen und Benutzer
  + SQL-Skripte (DCL)
* **DB-Daten:**
  + Import der Backpacker CSV-Dateien
  + SQL-Skripte (DML)
  + Bereinigung der Datenbank (FK, Index, Constraints)
* **Testen:**
  + Testprotokolle für Datenkonsistenz
  + SQL-Skripte (Testdaten für Migration)

**2. Remote Cloud-DBMS (z.B. MariaDB auf AWS)**

* **Setup Cloud DBMS**
* **Betrieb:**
  + Sicherstellung des Cloud DBMS für den produktiven Betrieb
  + Konfigurationen für den produktiven Betrieb -> .ini

**3. Automatisierte Migration**

* **Lokale DBMS auf Cloud-DBMS migrieren**
  + **Berechtigungen:**
    - Automatisierte Übertragung der Zugriffsberechtigungen
    - Testprotokolle für Rollen und Benutzer
    - SQL-Skripte (DCL)
  + **Daten:**
    - Automatisierte Übertragung der Struktur und Daten
    - SQL-Skripte (DDL & DML)
    - Datenbankbereinigung
  + **Testen:**
    - Testprotokolle zur Überprüfung der Datenkonsistenz anhand der Testdaten der lokalen DB
    - SQL-Skripte

**4. Protokollierung**

* Vollständige und nachvollziehbare Dokumentation

# Vergleich

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Größe** | **vCPUs** | **Arbeitsspeicher (GiB)** | **Speicher (GB)** | **Netzwerkleistung** | **Zweck** |
| db.t3.micro | 2 | 1 | bis zu 16 GiB | Bis zu 5 Gigabit | Einstiegsinstanz für Test- und Entwicklungszwecke |
| db.m5.large | 2 | 8 | bis zu 6 TB | Bis zu 10 Gigabit | Allzweck-Instanz für kleine bis mittlere Datenbanken |
| db.r5.2xlarge | 8 | 64 | bis zu 6 TB | Bis zu 10 Gigabit | Instanz mit hohem Speicher für große Datenbanken und hohe Arbeitslasten |

Wir entscheiden uns für AWS, da wir dies schon kennen. In AWS entschieden wir uns für die db.m5.large, da unsere Datenbank kein Corporate Level erreichen wird.