- Musikagentur -

SQL Datenbank

Von

Fabrice Bosshard  
 &  
 Fabian Bächli

AP15a

Matthias Vonoreilli

Inhalt

[1. Projektbeschreibung 3](#_Toc473528380)

[1.1. Aufgabe 3](#_Toc473528381)

[1.2. Beschreibung 3](#_Toc473528382)

[1.3. Projektumfang 3](#_Toc473528383)

[2. Analyse 4](#_Toc473528384)

[2.1. Anforderungen 4](#_Toc473528385)

[3. Planung 4](#_Toc473528386)

[3.1. Zeitplan 4](#_Toc473528387)

[3.2. Design 5](#_Toc473528388)

[3.3. Beziehungsbeschreibung 5](#_Toc473528389)

[4. Vorgehensbeschreibung 6](#_Toc473528390)

[5. Testung 6](#_Toc473528391)

[5.1. Tests (Views) 6](#_Toc473528392)

## Projektbeschreibung

### Aufgabe

Anhand einer selbstdefinierten Problemstellung wird eine Datenbanklösung realisiert. Dabei werden zugehörige Dokumente wie z.B. Zeitplan, Datenmodell etc. erstellt.

### Beschreibung

Wir haben uns hierbei für eine Musikagentur entschieden. Wir können in unseren Datenbank Veranstaltungen erfassen, an denen mehrere Bands spielen können und auch mehrere DJs pro Act auftreten kann. In den Bands spielen jeweils mehrere Musiker und eine Band hat einen Promoter. Ein Musiker ist eine Person. Eine Person kann ein Musiker, Promoter oder DJ sein und kann mehrere Kompetenzen haben.

Optional: Es wird ein Ticketverkauf eingebaut. Veranstaltungen haben ein

Zuschauerlimit. Und Zuschauer die an mehreren Veranstaltungen teilnehmen können. (Wurde nicht umgesetzt)

### Projektumfang

**Zeitbudget:** Über das ganze Modul ca. 20 Lektionen.  
 Zusätzlich ca. 5h Hausaufgaben.

**Hilfsmittel:** Keine Beschränkung.

#### Aufgabenaufteilung

**Fabian:**

* Arbeit an Dokumentation
* Design erstellen (Datenmodell)
* DDL implementieren
* Testbericht erstellen
* Index setzen

**Fabrice:**

* Arbeit an Dokumentation
* Zeitplanung erstellen
* Testfälle erstellen
* DML implementieren
* Views erstellen

## Analyse

### Anforderungen

Der Umfang, bezüglich der Zeitangabe für das Projekt, ist sehr eng. Deshalb ist es sehr wichtig, dass wir eine gute Planung erstellen. Wir müssen als erstes eine Zeitplanung erstellen. Um dieses Projekt effizient erfüllen zu können, müssen wir stets im Zeitplan arbeiten.

#### Arbeitsumgebung / Arbeitsmittel

Wir haben beschlossen, dass wir das Projekt mit der SQL Workbench abhandeln möchten. Dazu benützen wir einen ganz normalen MySQL-Server den wir mit dem XAMPP Control Panel hosten. Um die Datenmodelle erstellen zu können, benützen wir die Funktion «Reverse Engineering», die die Workbench anbietet.

## Planung

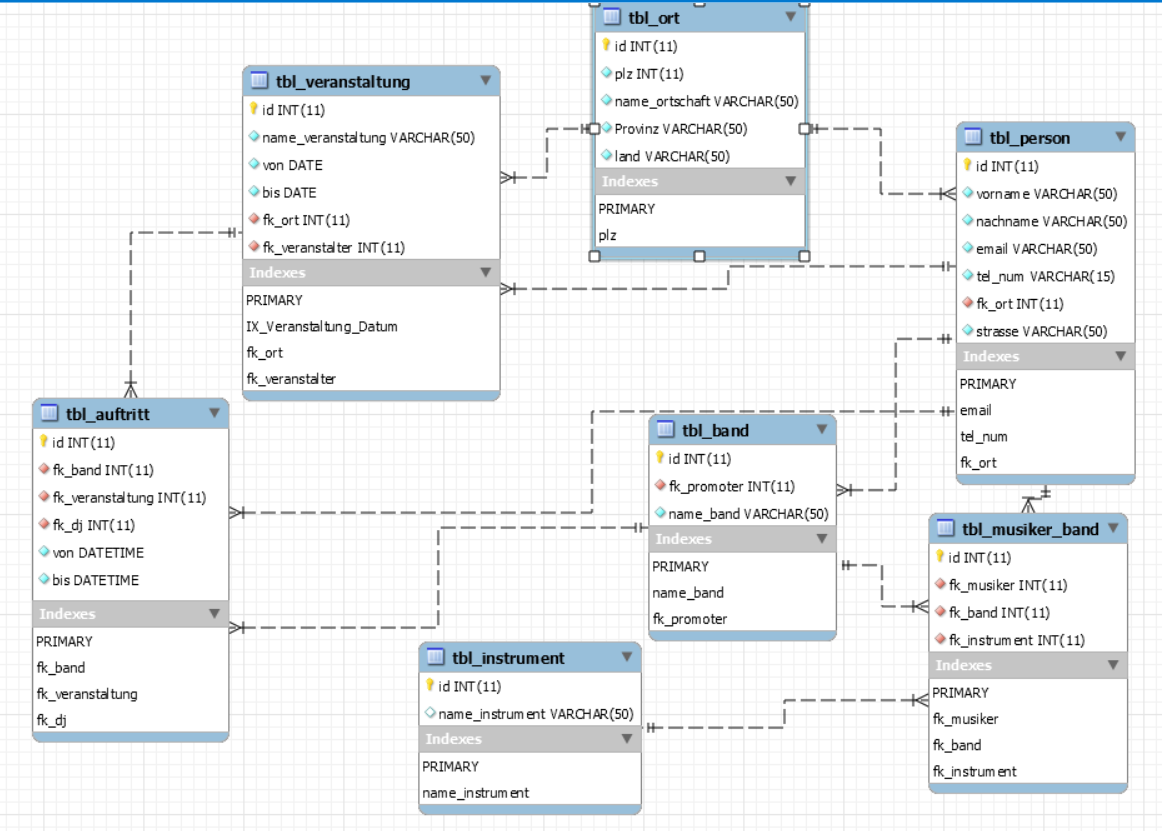
### Zeitplan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Zeitplan** | **Fabrice** | | **Fabian** | **Gemeinsam** |
| **19.12.2016** | - Dokumentation beginnen | | - ERM erstellen (Vorlage auf Papier) | - Projektbeschreibung deklarieren |
| - Bereits definierte Anforderungen dokumentieren | - DDL Implementierung beginnen | - DDL Implementierung beginnen | - Projektumfang analysieren |
| **09.01.17** | - Datenbank allenfalls ausbessern und säubern | - DDL Implementierung beenden | - DML Implementierung beginnen | - Testfälle planen |
| **16.01.17** | - Views planen | - Views implementieren | - DML Implementierung beenden | - Dokumentation überarbeiten und erweitern |
| **23.01.17** | Views Implementierung beenden | | Datenbank testen und ausbessern | - Dokumentation fertigstellen |
| **30.01.17** |  | |  |  |

1 >  
 1 <  
 2 <  
 3 <

### Design

#### ERM



#### Entscheidungen

Veranstaltung: Einen Ort, Datum von – bis, einen Veranstalter, einen Namen.

Ort: Eine PLZ, einen Namen, ein Kanton, ein Land.

Person: Namen, Vornamen, email, Telefonnummer, Strasse, Ort.

Band: Einen Promoter, einen Namen.

Musiker in Band: Eine Person, eine Band, ein Instrument (Position in Band)

Auftritt: Eine Band, eine Veranstaltung, ein DJ, Datum von- bis,

Instrument: Einen Namen.

### Beziehungsbeschreibung

|  |  |
| --- | --- |
| **Fall** | **Bezeichnung** |
| 1 | Eine Veranstaltung hat nur einen Veranstalter |
| 2 | Ein Veranstalter ist eine Person |
| 3 | Ein DJ ist eine Person |
| 4 | Ein Musiker ist eine Person |
| 5 | Eine Band hat mehrere Personen |
| 6 | Eine Person kann in mehreren Bands spielen |
| 7 | Eine Band nimmt an mehreren Veranstaltungen teil |
| 9 | Eine Person hat einen Ort und dieser ist nur einmal erfasst |
| 10 | Auftritt hat eine Band, eine Veranstaltung und einen DJ |
| 11 | Eine Person hat ein Instrument, kann aber auch mehrere haben |
| 12 | An einer Veranstaltung hat es mehrere Veranstaltungen |

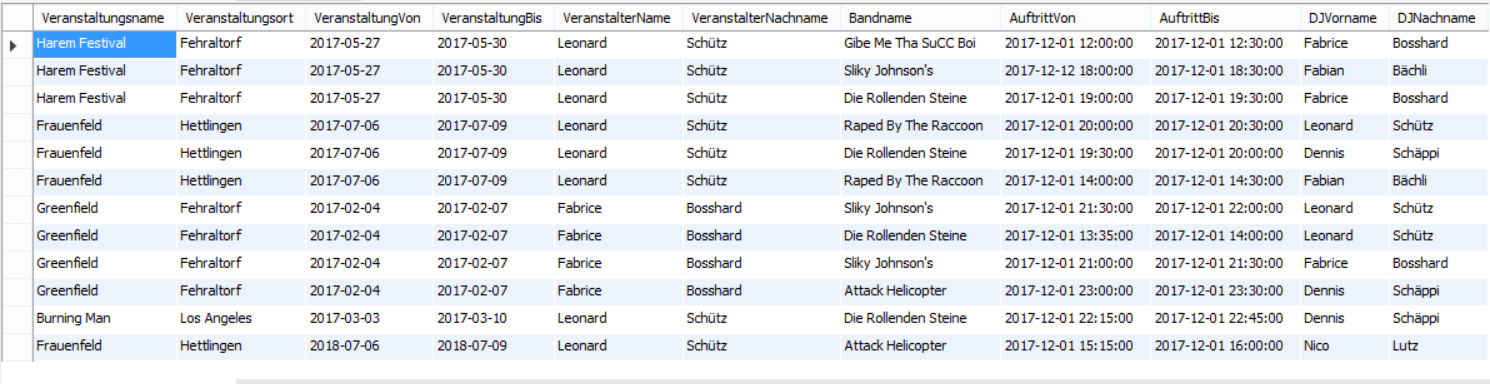
## Vorgehensbeschreibung

Wir haben als erstes versucht eine möglichst korrekte Architektur zu machen. Diese würde uns dabei helfen im Verlauf der Arbeit effizienter vorzugehen. Als wir unsere Implementation danach begannen, merkten wir, dass wir die Architektur trotzdem umstellen mussten, da wir einen Logik-Fehler gemacht hatten. Der Rest des Projekts verlief ziemlich reibungslos. Wir hatten keine weiteren Probleme im Verlauf und konnten so in ziemlich kurzer Zeit mit der Arbeit abschliessen.

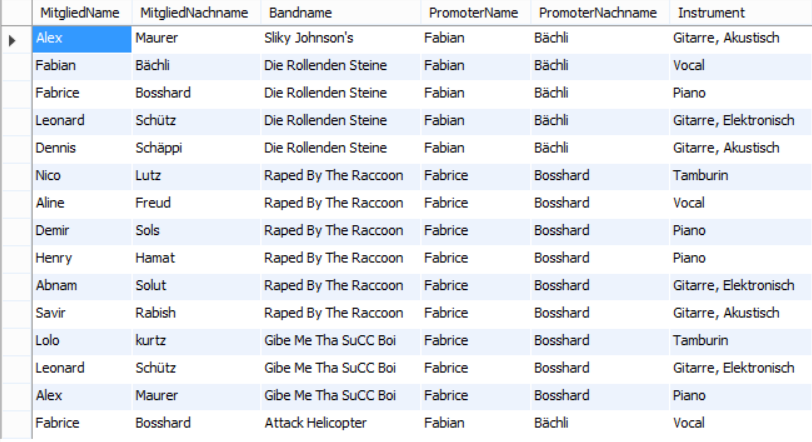
## Testung

### Tests (Views)

1. **Veranstaltung & Auftritt & Band & DJ** («VeranstaltungAuftrittBandDJ»)

* Zeigt alle Auftritte von allen Veranstaltungen.

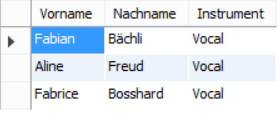
1. **Band Mitglieder** («BandMitglieder»)

* Zeigt alle Mitglieder einer Band. (Person kann in 2 oder mehreren Bands spielen)

1. **Veranstaltung zwischen Zeitrahmen** («VeranstaltungZwischenZeitrahme»)

* Zeigt alle Veranstaltungen die nach folgendem Datum stattfinden. (2017-05-01)

1. **Personen mit Gleichem Instrument** («PersonenMitGleichemInstrument»)

* Zeigt alle Personen die das gleiche Instrument spielen. (Vocal)