

SYSTEMBESCHREIBUNG

Wir haben den Auftrag bekommen, ein Informationssystem für die kommende Fußball-Weltmeisterschaft zu gestalten. Das Datenmodell wird als Backend für eine Web-Anwendung verwendet werden, in der Nutzer wichtige Informationen über die WM 2018 abrufen können. Organisiert ist das Datenmodell durch sieben Tabellen, die größtenteils miteinander verknüpft sind.

Die Aufteilung der einzelnen Tabellen ist in unserem ER-Diagramm logisch dargestellt und entspricht im Fall, dass man die N:M-Verbindungen außer Acht lässt, gleichzeitig dem physischen Datenmodell. Dieses besteht aus sieben nach Themenbereichen getrennten Tabellen.

Die Erste davon hat den Bezeichnung ‚Spieler‘ und besitzt vier Spalten, von denen jede einen eindeutigen Erkennungsnamen trägt: Spieler-ID-, Name, Mannschaft und Status. Hier werden Informationen über die einzelnen Spieler, deren Namen, zugehöriger Mannschaft und Spielerstatus bezüglich Verstößen (eventuelle Sperrung aufgrund roter Karte) gespeichert. Die eindeutige Identifikation eines jeden Spielers erfolgt anhand der Spieler_ID in Form eines Primary Keys.

Die Tabelle ‚Mannschaft‘ ist gegliedert in folgende Spalte: Name, Gruppe und WM-Titel. Damit erhält der Nutzer die Möglichkeit Datensätze über die an der diesjährigen WM teilnehmenden Mannschaften, die Gruppenzugehörigkeit und die Anzahl der WM-Siege desjenigen Teams. Hier dient der offizielle Name des Teams als Primary Key und identifiziert somit die Mannschaft.

Im Weiteren folgt unsere umfassendste Tabelle in Bezug auf die Spaltenzahl, welche alle Informationen zu den einzelnen WM-Spielen repräsentiert und den Namen Spielplan trägt. Die Gliederung erfolgt anhand der Felder Spielpaarung_ID, Datum, Veranstaltungsort, Heimteam, Gastteam, Tore_Heimteam, Tore_Gastteam, die Titel sind selbsterklärend gewählt. In diesem Fall entschieden wir uns den Parameter Spielpaarung_ID zum Primary Key zu ernennen und damit den Spielen eine ID zur eindeutigen Erkennung mitzugeben.

Die Veranstaltungsort-Tabelle enthält Ort, Zuschauerzahl, Eintrittspreise der genutzten Stadien der WM und den Namen, wobei Letzterer als Primary Key die ID zur Erkennung trägt.

Nächste Tabelle ist die Verstoß-Tabelle. Die Spalten heißen: Spieler_ID, Gelbe_Karte und Rote_Karte. Diese Tabelle regelt den Verstoß von jedem einzelnen Spieler durch Eintrag der Anzahl gelber und roter Karten. Die Spieler_ID bildet den Primary Key.

Die Beteiligten der WM werden in der Beteiligten-Tabelle eingetragen, geordnet in Beteiligten_ID, Name, Rolle und Mannschaft. Primary Key bildet die Beteiligten_ID. In die Spalte Rolle wird eingetragen, welche Rolle jede/r Beteiligte bei der WM spielt, also zum Beispiel Schiedsrichter, Trainer, Physio, Krankenhelfer etc. und ob die Person zu einer Mannschaft gehört oder nicht.

Die letzte Liste enthält Daten der Nutzer unseres Systems. Durch den Nutzernamen ist die eindeutige Zuordnung gegeben, denn dieser bildet den Primary Key, zugehörige Spalten sind Passwort und Rolle.

Auf den genannten Aufbau konnten wir uns als Team einigen, um die wichtigsten Informationen über die WM 2018 wiederzugeben.

Auf die Verknüpfungslogik innerhalb der Datenbank wird im Folgenden eingegangen:

Während die Nutzer-Tabelle eigenständig und nicht mit anderen Tabellen verknüpft ist, verbindet sich die Spieler-Tabelle sowohl mit der Verstoß- als auch der Mannschaft-Tabelle. Jedem Spieler wird logischerweise ein Status zugeordnet, welcher den aktuellen Zustand bezüglich der Spielfähigkeit oder einer Sperrung angibt. Letzteres passiert innerhalb der Tabelle Verstoß, welche mit einem Trigger arbeitet. Die Spieler-ID wird hier als Foreign Key von der Spieler-Tabelle übernommen. Jeder Spieler ist Mitglied einer Mannschaft und somit ebenfalls in der Mannschaft-Tabelle registriert. Dabei dient die Spalte Mannschaft als Foreign Key zum Primary Key der Spalte Name in Mannschaft, während die Mannschaften-Tabelle selbst Spieler und Beteiligte, also etwaige Mitarbeiter wie Trainer, Ärzte etc., hält. So wie die Spieler einer Mannschaft zugeordnet werden können, ist dies je nach Aufgabengebiet ebenfalls mit den Beteiligten möglich. Wie vorher gilt hier selbiges mit dem Foreign Key, wobei die zusätzliche Typisierung durch die Option NOT NULL gewählt wurde, um die Möglichkeit einer Unzugehörigkeit zu einer Mannschaft bei bspw. den Unparteiischen zu gewährleisten.

Die Spielpaarungen der Mannschaften finden in der Tabelle Spielplan ihren Eintrag, was eine Verknüpfung zwischen diesen Beiden bedeutet. Einer Spielpaarung sind immer genau zwei Teams zugeordnet. Da zur Identifikation der Teams deren Primary Key (Name) als Foreign Key verwendet wird, ist es ausgeschlossen, dass ein Team gegen sich selbst spielt.

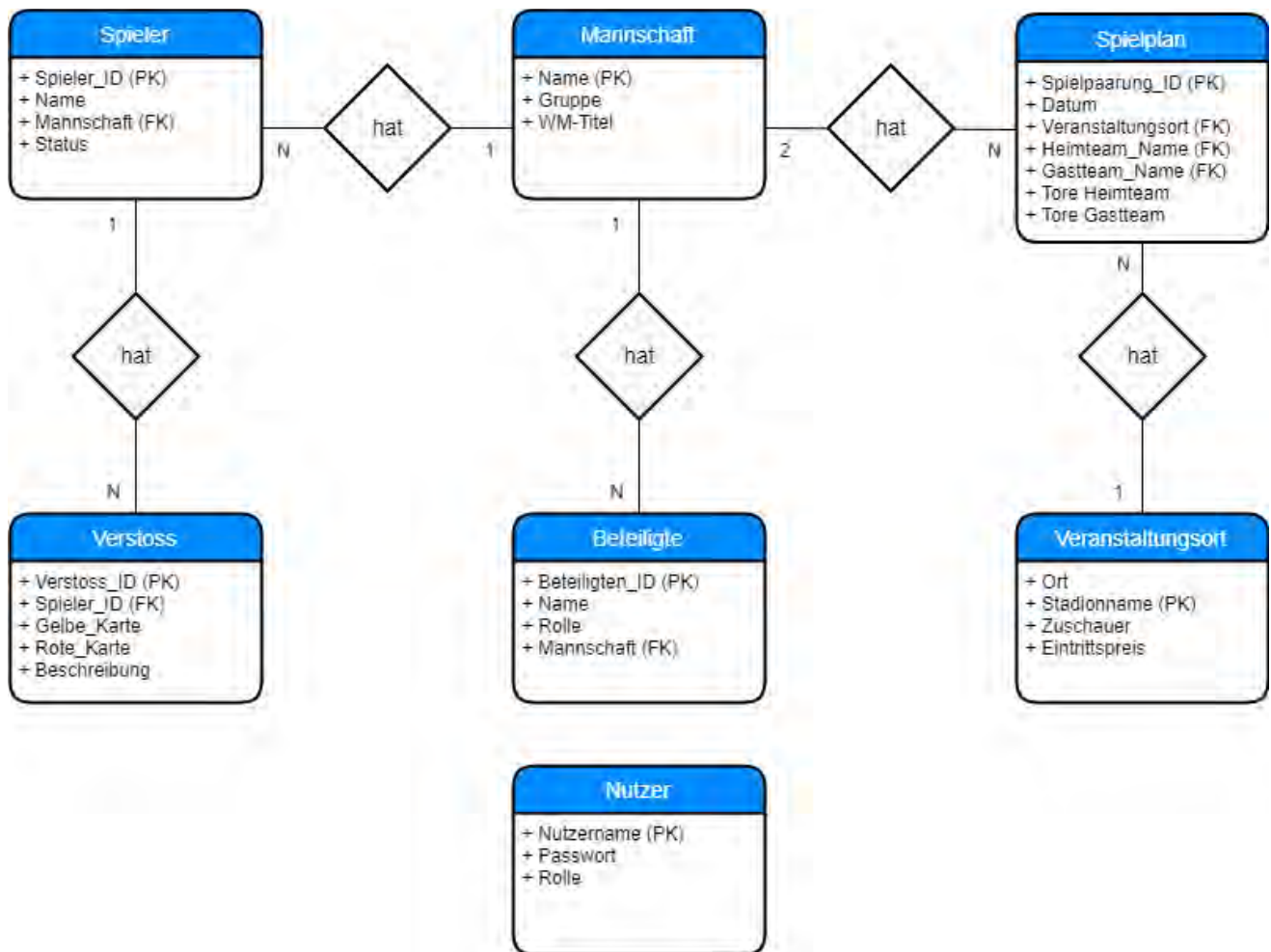
Außerdem liegt eine weitere Bindung durch die Spalte Veranstaltungsort der Spielplan-Tabelle mit Veranstaltung selbst als Foreign Key vor.

Die Verwendung von Indizes ist in den meisten Fällen sinnvoll, um mögliche Mehrdeutigkeiten durch bspw. Namen auszuschließen, größere Datenmengen zu ermöglichen und die Wartbarkeit und eine mögliche weiterführende modulare Entwicklung am Projekt zu gewährleisten.

Jeder genannte Primary Key stellt in unserem Fall eine explizite Definition einer ID dar.

Somit ist die Verwendung von Indizes als Datenbankfeature (CREATE INDEX xyz ...) auf bestimmte Spalten überflüssig.

ENTITY-RELATION-DIAGRAMM



DATENBANK-SKRIPTE

CREATE

```

DELETE FROM spielplan;
DELETE FROM beteiligte;
DELETE FROM veranstaltungsort;
DELETE FROM nutzer;
DELETE FROM verstoss;
DELETE FROM spieler;
DELETE FROM mannschaft;
    
```

```

DROP TABLE spielplan;
DROP TABLE beteiligte;
DROP TABLE veranstaltungsort;
DROP TABLE nutzer;
DROP TABLE verstoss;
DROP TABLE spieler;
DROP TABLE mannschaft;
    
```

```

CREATE TABLE mannschaft (
    
```

```

        name VARCHAR2(32) PRIMARY KEY,
        gruppe CHAR(1) NOT NULL,
        wm_titel INTEGER DEFAULT 0);

CREATE TABLE veranstaltungsort(
    stadionname VARCHAR2(32) PRIMARY KEY,
    ort VARCHAR2(32) NOT NULL,
    zuschauerzahl INTEGER NOT NULL,
    eintrittspreis FLOAT);

CREATE TABLE nutzer(
    nutzername VARCHAR2(32) PRIMARY KEY,
    passwort VARCHAR2(32) NOT NULL,
    rolle VARCHAR2(32));

CREATE TABLE spieler(
    spieler_id INTEGER PRIMARY KEY,
    name VARCHAR2(32) NOT NULL,
    mannschaft VARCHAR2(32) NOT NULL,
    status VARCHAR2(32),
    CONSTRAINT mannschaftsname_fk FOREIGN KEY (mannschaft) REFERENCES mann-
schaft(name));

CREATE TABLE beteiligte(
    beteiligten_id INTEGER PRIMARY KEY,
    name VARCHAR2(32) NOT NULL,
    rolle VARCHAR2(32) NOT NULL,
    mannschaft VARCHAR2(32),
    CONSTRAINT mannschafts_zugehoerigkeit_fk FOREIGN KEY (mannschaft) REFERENCES
mannschaft(name));

CREATE TABLE verstoss(
    verstoss_id INTEGER PRIMARY KEY,
    spieler_id INTEGER NOT NULL,
    gelbe_karte INTEGER DEFAULT 0,
    rote_karte INTEGER DEFAULT 0,
    beschreibung VARCHAR(64),
    CONSTRAINT spieler_id_fk FOREIGN KEY (spieler_id) REFERENCES spieler(spie-
ler_id));

CREATE TABLE spielplan(
    spielpaarung_id INTEGER PRIMARY KEY,
    datum DATE NOT NULL,
    veranstaltungsort VARCHAR2(32) NOT NULL,
    heimteam_name VARCHAR2(32) NOT NULL,
    gastteam_name VARCHAR2(32) NOT NULL,
    tore_heimteam INTEGER DEFAULT 0,
    tore_gastteam INTEGER DEFAULT 0,
    CONSTRAINT heimteam_name_fk FOREIGN KEY (heimteam_name) REFERENCES mann-
schaft(name),
    CONSTRAINT gastteam_name_fk FOREIGN KEY (gastteam_name) REFERENCES mann-
schaft(name),
    CONSTRAINT veranstaltungsort_fk FOREIGN KEY (veranstaltungsort) REFERENCES
veranstaltungsort(stadionname));

create or replace TRIGGER check_verstoss_rote_karte_aiu
AFTER INSERT OR UPDATE OF
gelbe_karte, rote_karte
ON verstoss
REFERENCING NEW AS NEW OLD AS OLD
FOR EACH ROW
WHEN(new.rote_karte >= 0)
BEGIN
    IF :NEW.rote_karte > 0 THEN

```

```

UPDATE spieler SET status = :NEW.beschreibung
WHERE spieler_id = :new.spieler_id;

DBMS_OUTPUT.PUT_LINE('Spieler Nr. '||:new.spieler_id||' , neuer Status:
'||:new.beschreibung||'!');

END IF;
END;

```

INSERT

```

/* Mannschaft */
INSERT INTO mannschaft VALUES('Deutschland','A', 4);
INSERT INTO mannschaft VALUES('England','A', 1);
INSERT INTO mannschaft VALUES('Schweiz','A', 0);
INSERT INTO mannschaft VALUES('Botswana','B', 0);
INSERT INTO mannschaft VALUES('Island','B', 0);
INSERT INTO mannschaft VALUES('Iran','B', 0);

/* Veranstaltungsort */
INSERT INTO veranstaltungsort VALUES('Luzhniki-Stadion', 'Moskau', 81000,
12.99);
INSERT INTO veranstaltungsort VALUES('Sankt-Petersburg-Stadion', 'St. Peters-
burg', 68000, 23.00);
INSERT INTO veranstaltungsort VALUES('Fisht-Stadion', 'Sotschi', 47659, 9.95);
INSERT INTO veranstaltungsort VALUES('Spartak-Stadion', 'Moskau', 45360, 19.50);

/* Spieler */
INSERT INTO spieler VALUES(0,'Volker Vollpfosten','Botswana', 'aktiv');
INSERT INTO spieler VALUES(1,'Torben Torgefahr','Deutschland', 'aktiv');
INSERT INTO spieler VALUES(2,'Mitchel Matchmaker','England', 'aktiv');
INSERT INTO spieler VALUES(3,'Björn Ballson','Island', 'aktiv');
INSERT INTO spieler VALUES(4,'Georg Goaly','Iran', 'aktiv');
INSERT INTO spieler VALUES(5,'Torben Tribbler','Schweiz', 'aktiv');

/* Spielplan */
INSERT INTO spielplan VALUES(0, TO_DATE('14.06.2018','DD-MM-YYYY'), 'Luzhniki-
Stadion', 'Deutschland', 'England', 0, 0);
INSERT INTO spielplan VALUES(1, TO_DATE('15.06.2018','DD-MM-YYYY'), 'Sankt-Pe-
tersburg-Stadion', 'England', 'Schweiz', 0, 0);
INSERT INTO spielplan VALUES(2, TO_DATE('16.06.2018','DD-MM-YYYY'), 'Fisht-Sta-
dion', 'Schweiz', 'Deutschland', 0, 0);
INSERT INTO spielplan VALUES(3, TO_DATE('17.06.2018','DD-MM-YYYY'), 'Spartak-
Stadion', 'Iran', 'Botswana', 0, 0);
INSERT INTO spielplan VALUES(4, TO_DATE('18.06.2018','DD-MM-YYYY'), 'Luzhniki-
Stadion', 'Island', 'Iran', 0, 0);
INSERT INTO spielplan VALUES(5, TO_DATE('19.06.2018','DD-MM-YYYY'), 'Sankt-Pe-
tersburg-Stadion', 'Botswana', 'Island', 0, 0);

/* Beteiligte */
INSERT INTO beteiligte VALUES(0, 'Peter Zwegat', 'Schiedsrichter', NULL);
INSERT INTO beteiligte VALUES(1, 'Barbara Salesch', 'Schiedsrichter', NULL);
INSERT INTO beteiligte VALUES(2, 'Jogi Löw', 'Trainer', 'Deutschland');
INSERT INTO beteiligte VALUES(3, 'Ragnar Lodbrok', 'Trainer', 'Island');
INSERT INTO beteiligte VALUES(4, 'Dicker Junge', 'Waterboy', NULL);
INSERT INTO beteiligte VALUES(5, 'Willfred Wiener', 'Würstchenverkäufer', NULL);

/* Nutzer */
INSERT INTO nutzer VALUES('admin', 'admin', 'admin');
INSERT INTO nutzer VALUES('schwaebische_zeitung', 'abcde', 'journalisten');

```

```
INSERT INTO nutzer VALUES('team_deutschland', '12345', 'teilnehmer');
INSERT INTO nutzer VALUES('nsa_germany', '00000', 'gast');

/* verstoss */
INSERT INTO verstoss VALUES(0, 2, 1, 0, NULL);
INSERT INTO verstoss VALUES(1, 3, 1, 0, NULL);
INSERT INTO verstoss VALUES(2, 1, 0, 1, '2 Spiele gesperrt');
INSERT INTO verstoss VALUES(3, 4, 1, 0, 'Beim Foul selbst verletzt');

COMMIT;
```

Kriterium		max.	ist	Kommentar
Relative Qualität		15		
Spezifikation / Systembeschreibung		15		
ER-Modell inkl. formaler Korrektheit im Sinn der Normalisierung		10		
Sinnvolle Nutzung von Normierungsdaten		5		
Lauffähigkeit		5		
Restartfähigkeit		5		
Vollständigkeit der Testdaten		5		
Umfang der Umsetzung des Datenmodells		-	-	
	Anzahl der Tabellen	5		
	Vollständige Nutzung der Datentypen	5		
	Sinnvolle Nutzung von Randbedingungen	5		
	Sinnvolle Nutzung von Indexierung	5		
Summe				