



**Hochschule für Technik
und Wirtschaft Berlin**

University of Applied Sciences

**Mai, Truc Linh 551017
Tran, Manh Cuong 552625
Nguyen, Hai Tuan Son 557376**

**Herausgabetermin: 25.04.2018
Abgabetermin: 11.07.2018**

Restaurant Management System (RMS)

**Studiengang: Internationaler Studiengang Medieninformatik
Fachbereich 4 Informatik, Kommunikation und Wirtschaft
Modul: Datenbanken (Gruppe 2)
Leitung: Prof. Dr. Rüger Oßwald
Sommersemester 2018
Fachsemester: 3**

1. Kurzbeschreibung

Das Restaurant-Management-System soll für einen reibungslosen Arbeitsprozess sorgen, vor allem aber die Übersicht und die Interaktion von allen Akteuren klar und deutlich darstellen. Zwischen den verschiedenen Akteuren wird eine klare Beziehung festgelegt und vor allem wie miteinander interagiert wird. Das Konzept ist auf kleinere bis mittlere Restaurants ausgerichtet, auch wird hiermit ein Entwurf garantiert der einfach und verständlich bleibt. Es soll den Prozess von Bestellaufnahme bis zur Servierung so einfach wie möglich gestalten und unnötig komplizierte Wege vermeiden. Somit wird die Wartezeit für die Kunden verkürzt und bleibt für die Mitarbeiter übersichtlich und dennoch effizient. Letztlich soll die Implementation unseres Systems schnell und problemlos erfolgen.

2. Voraussetzungen bezüglich Software & Technik

Für die Modellierung und Implementierung der Datenbank haben wir lediglich zwei verschiedene Softwareprodukte der Firma Sybase benutzt. Wir empfehlen bei zukünftige Erweiterungen der Datenbank weiterhin diese Programme zu benutzen.

2.1 Powerdesigner (Version 15.2)

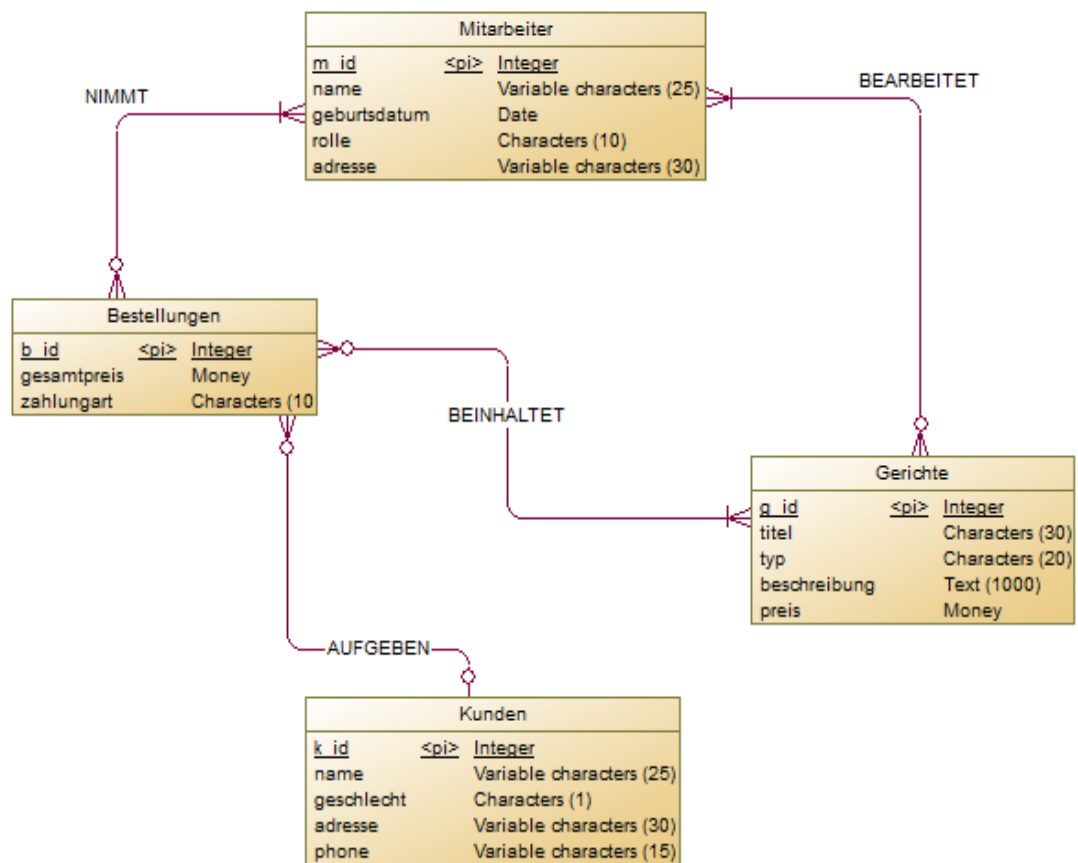
Die Modellierung erfolgte ausschließlich in dem Programm *Powerdesigner*.

2.2 SQL Anywhere (Version 17)

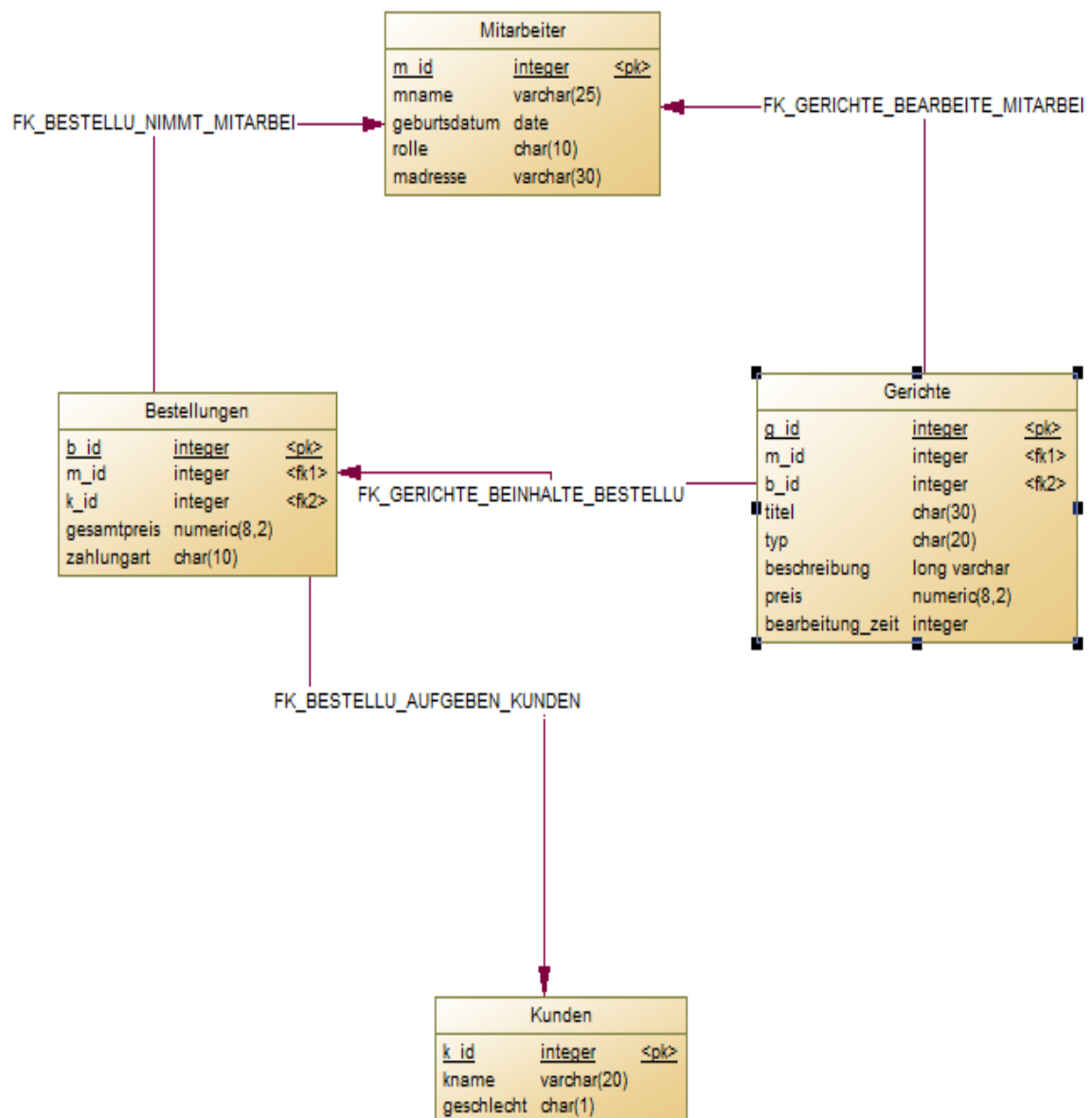
Die Implementierung hingegen wurde in *SQL Anywhere* programmiert.

3. Datenbankentwurf

3.1 Konzeptuelle Ebene



3.2 Physische Ebene



4.Funktionsumfang

Das System erfasst jeden Mitarbeiter mit einer individuellen Nummer. Es ist möglich jeden aktiven Mitarbeiter und seine persönlichen Daten zu erfassen. Im Mittelpunkt stehen jedoch die Gerichte und Bestellungen die anhand verschiedenen Attributen versehen werden können um Fehler in Bestellungen zu vermeiden. Die Zeit der Bestellung wird erfasst um die Aufbereitungszeit gering zu halten und eventuell in der Zukunft Verbesserungen im Prozess vornehmen zu können.

Die Tabellen können mit folgenden Anweisungen aufgerufen werden.

| <u>NAME</u> | <u>BESCHREIBUNG</u> |
|-------------------------------------|---|
| Top3Gerichte | Ordnet Gerichte nach den Preisen, von teuersten Gerichten bis niedrigsten Preisen und zeigt die 3 teuersten an. |
| NieBestellt | Zeigt alle Gerichte an die kein einziges Mal bestellt wurden. |
| GerichteUnter10 | Zeigt alle Gerichte an die nicht mehr als 10min dauern. |
| GerichteMitarbeiterMeistensServiert | Zeigt die Gerichte an die am meisten von Mitarbeitern serviert wurden |
| PIEP | PIEP |

5. Funktionen

| <u>NAME</u> | <u>PARAMETER</u> (mit Typ) | <u>RÜCKGABE</u> | <u>WIRKUNG</u> |
|--------------------------------------|-------------------------------|-------------------------|---|
| DurchschnittUmsatz ProMitarbeiter | MitarbeiterID: integer | Durchschnitt: double | Berechnet den durchschnittlichen Umsatz des Mitarbeiters |
| AnzahlBestellungenProKunde | KundenID: integer | Anzahl: int | Zeigt die Bestellungsanzahl jedes Kunden |

6. Prozeduren

| <u>NAME</u> | <u>PARAMETER</u> (mit Typ) | <u>WIRKUNG</u> |
|---------------------|--|--|
| GerichteUpdate | <u>gerichtelD integer</u> , <u>titel char(50)</u> , <u>typ char(50)</u> , <u>beschr varchar(250)</u> , <u>preis double</u> , <u>be_zeit integer</u> | Eintragungen der einzelnen Gerichte können geändert und angepasst werden |
| FilterBestellung | gtyp char(50), kundenID integer | Bestellungen können hiermit nach Typ und KundenID gefiltert werden |
| AnzahlServierung | name char(10) | Zählt mit wie viele Gerichte die Mitarbeiter serviert haben |
| BeschreibungAendern | <u>beschr varchar(250)</u> , <u>gerichtelD integer</u> | Beschreibungen werden hiermit geändert |

7. Verbesserungs- & Erweiterungsansätze

Wichtige Verbesserungsvorschläge die Zukunft berücksichtigt werden sollten.

| <u>NAME</u> | <u>ANSÄTZE</u> | <u>VORTEILE</u> |
|------------------|---|--|
| MostBoughtDish | PIEP | Kunden können sofort informiert werden was der Topseller unter den Gerichten ist |
| TotalSum | Select sum gerichte in bestellung <u>PIEP</u> | Gesamtsumme für alle bestellten Gerichte werden zusammengerechnet und angezeigt |
| GerichteUnter10€ | create view "DBA"."GerichteUnter10€" as select titel, typ, beschreibung from gerichte where gerichte_preis < 10 | Zeigt alle Gerichte an die weniger als 10€ kosten |
| Zutaten | Create table Zutaten(name char(32), menge double, ...) | Tabelle: zeigt an welche Zutaten vorhanden sind und wie viel noch davon übrig sind |
| Kontakte | Create table Kontakte(name char(32), telephone(16), ...) | Tabelle: alle Kontaktdaten von Mitarbeitern |

Abschließende Worte

Für den langanhaltenden Bestand des Restaurants empfehlen wir ein Bewertungssystem für Gerichte einzuführen in der Mitarbeiter und Kunden in Gerichte einsehen können welche begehrt sind und welche eher nicht. Somit können Mitarbeiter eventuell Gerichte gegen neue austauschen und Topseller zu jeder Zeit anbieten. In der Zeit der Social Media ist es wichtig auf die Meinung von Kunden zu hören und auf Trends zu achten wie der glutenfreien oder der veganen Ernährung.

Falls Sie sich entscheiden unser Bewertungssystem implementieren stehen wir Ihnen jeder Zeit zur Verfügung. Wir danken Ihnen uns die Chance zu geben mit Ihnen Geschäfte machen zu dürfen.

Quellen

- Skript der Datenbanken Vorlesung & Übung von Prof Dr. Rüger Oßwald
- Interactive SQL Help (<http://dcx.sap.com/index.html>)