

Informationssysteme - Fortgeschrittene Konzepte

Skriptum zum Vortrag - 01.09.2022

Dipl.-Ing. Msc. Paul Panhofer BSc.^{1*}

1 ZID, TU Wien, Taubstummengasse 11, 1040, Wien, Austria

Abstract:

MSC: paul.panhofer@gmail.com

Keywords:

*E-mail: paul.panhofer@tuwien.ac.at

1.) Aufgabenblatt



Kompetenz

- **Relationale Modellierung:** Datentabellen, Schlüsselstabellen, Basistabellen, identifizierende und nichtidentifizierende Beziehungen.

1.Beispiel) Relationale Modellierung

Eine Eventmanagement Firma möchte expandieren. Im Zuge dieser Expansion sollen auch die verschiedenen Arbeitsabläufe der Firma soweit wie möglich automatisiert werden. Sie werden dafür mit der Erstellung der Datenbank beauftragt.

- **Veranstaltung:** Das Unternehmen hat sich auf das Planen von Veranstaltungen spezialisiert.

Eine Veranstaltung wird über eine ID identifiziert. Sie besitzt eine Bezeichnung, sowie Beschreibung und Dauer. Weiters sollen auch der Anfangszeitpunkt und Endzeitpunkt gespeichert werden. Im System sollen auch spezielle Veranstaltungen erfasst werden können, zu welchen zusätzliche Informationen benötigt werden. Ein Film ist eine Veranstaltung und ist einem Genre zugeordnet, außerdem wird noch die Altersgrenze vermerkt. Zu einem Konzert wird die Musikrichtung und Band gespeichert und zu einem Theaterstück der Name der Theatergruppe.

- **Saal:** Eine Veranstaltung kann in genau einem Saal stattfinden, wobei in einem Saal mehrere Veranstaltungen stattfinden können. Ein Saal wird über eine ID identifiziert. Zusätzlich wird auch noch eine Bezeichnung zu dem Saal vermerkt. Ein Saal besteht aus beliebig vielen Plätzen. Ein Platz kann innerhalb eines Saals über eine Platznummer eindeutig identifiziert werden. Außerdem wird der Platzpreis gespeichert. Vom Eventmanagement System sollen verschiedene Veranstaltungsorte erfasst werden können. Ein Veranstaltungsort wird über seinen Namen eindeutig identifiziert und umfasst einen oder mehrere Säle.

- **Ticket:** Ein Ticket wird im System über eine fortlaufende Ticketnummer identifiziert. Ein Ticket umfasst immer genau einen Platz einer bestimmten Veranstaltung. Tickets können von einem Kunden

reserviert werden. Außerdem soll auch der Zeitpunkt der Reservierung erfasst werden.

- **Kunde:** Ein Kunde wird über eine fortlaufende Kundennummer eindeutig im System identifiziert. Zusätzlich wird auch noch der Nachname des Kunden gespeichert. Das System erfasst außerdem noch Premiumkunden. Ein Premiumkunde besitzt zusätzlich noch einen Vornamen, ein Geburtsdatum, eine Telefonnummer sowie ein Geschlecht. Ein Premiumkunde besitzt genau einen Account über welchen er Bestellungen tätigen kann.

Ein Account wird eindeutig über eine EMail Adresse identifiziert. Außerdem wird in der Datenbank auch noch das Passwort für den Account gespeichert. Premiumkunden sollen sich über diesen Account später mittels eines Webinterfaces am System anmelden und dort Bestellungen tätigen können.

- **Bestellung:** Über einen Account können beliebig viele Bestellungen getätigt werden, wobei jede Bestellung genau einem Account zugeordnet ist.

Eine Bestellung wird über eine fortlaufende Bestellungsnummer identifiziert. Zusätzlich werden zu einer Bestellung noch der Zeitpunkt sowie eine Anmerkung gespeichert. Eine Bestellung kann beliebig viele Tickets umfassen. Ein Ticket ist immer einer bestimmten Bestellung zugeordnet.

- **Mitarbeiter:** Die Datenbank soll außerdem noch die Mitarbeiter der Eventmanagement Firma erfassen. Ein Mitarbeiter wird über seine Sozialversicherungsnummer eindeutig identifiziert. Außerdem werden noch Vorname und Nachname vermerkt. Mitarbeiter können einem anderen Mitarbeiter untergeordnet sein, wobei ein Mitarbeiter aber beliebig viele untergeordnete Mitarbeiter besitzen kann.

Ein Veranstaltungsort wird immer von genau einem Mitarbeiter geleitet. Es soll auch die Arbeitsfunktion des Mitarbeiters vermerkt werden.

□

1.) Aufgabenblatt - Gruppe B (5.Punkte)



Kompetenz ▼

- **Relationale Modellierung:** Datentabellen, Schlüsseltabellen, Basistabellen, identifizierende und nichtidentifizierende Beziehungen.

1.Beispiel) Bugtracking (5.Punkte)

Die Softwarefirma Beta-Release möchte eine Datenbank anlegen um nicht den Überblick über ihre Produkte und deren Komponenten zu verlieren.

- **Person:** Jede Person hat eine Sozialversicherungsnummer, einen Nachnamen, einen Vornamen und eine Personalnummer. Personen arbeiten dabei in folgenden Berufen: Software Entwickler, Qualitätsmanager.
- **Projekt:** Jedes Projekt wird durch eine ID identifiziert. Für Projekte wird eine Bezeichnung gemeinsam mit einer Projektnummer gespeichert. Weiters besitzt jedes Projekt noch ein Budget. Jedes Projekt wird von genau einem Software Entwickler geleitet, wobei ein Entwickler auch mehrere Projekte leiten kann.
- **Software:** Jede Software wird eindeutig durch eine ID identifiziert. Weiters wird ein Titel und eine Beschreibung gespeichert. Darüber hinaus soll gespeichert werden welche Personen im Rahmen welcher Projekte an welcher Software gearbeitet haben. Mit Hilfe der Datenbank sollen außerdem die verschiedenen Versionen der erstellten Software verwaltet werden. Eine Version ist eindeutig identifiziert durch die entsprechende Software sowie einer Versionsnummer. Es muss zu jeder Software jeweils mindestens eine Version existieren. Jede Softwareversion beinhaltet ein oder mehrere Pakete. Zu jedem Paket wird sein Name und eine Paketversion gespeichert, wobei der Name und die Version gemeinsam ein Paket eindeutig identifizieren. Darüber hinaus werden die Abhängigkeiten zwischen Paketen vermerkt, also welches Paket welche anderen Pakete als Voraussetzungen benötigt. Zur Vermeidung von zirkulären Abhängigkeiten sind die Abhängigkeiten in Paketen als Baumstruktur aufgebaut.

- **File:** Ein Paket enthält ein oder mehrere Files, wobei ein File eindeutig identifiziert ist durch das Paket in dem es sich befindet, und seinen Namen. Zusätzlich sollen sein Inhalt und der Dateityp gespeichert werden. Es werden 2 Arten von Files unterschieden: Source-Files – für diese soll ihre Länge gespeichert werden und Grafik-Files. Für Source-Files soll vermerkt werden, in welcher Programmiersprache sie verfasst sind, wobei in jedem Source-File nur eine Programmiersprache verwendet werden darf. Eine Programmiersprache wird eindeutig identifiziert durch ihren Namen. Es soll außerdem für jeden Software Entwickler gespeichert werden, welche Programmiersprachen diese Person beherrscht, und wie gut (LEVEL - BASIC, INTERMEDIATE, MASTER).

- **Bugreport:** Zu einem Bugreport soll seine Beschreibung gespeichert werden sowie die Information, wann der Report gemeldet worden ist. Jeder Bugreport besitzt außerdem eine eindeutige Trackingnummer, und ist genau einer Softwareversion zugeordnet. Bevor ein Bug bearbeitet wird, muss er von einem Qualitätsmanager bestätigt werden. Es soll nicht nur vermerkt werden, wer einen Bug bestätigt hat, sondern auch wann das passiert ist. Ein Bugreport beschreibt dabei immer einen einzelnen Bug. Für behobene Bug wird zusätzlich ein Kommentar vermerkt, sowie von welchem Softwareentwickler der Fehler behoben wurde, und wann. Ein Bug wird dabei von genau einem Entwickler behoben. Zusätzlich kann ein behobener Bug durch maximal einen Qualitätsmanager geprüft werden. Zu jeder Prüfung soll eine Bewertung und das Datum der Prüfung gespeichert werden.

