

# Logbuch Modul 151

Sandro Ropelato

3. Oktober 2006

Version: 1.2.1

## Tag 1: 28. September 2006

Zeit:	Tätigkeit
08:30 – 09:00	Ankommen, Laptop aufstellen, Arbeitsplatz vorbereiten
09:00 – 09:30	Einführung des Lehrers
09:30 – 09:45	Ämterverteilung
10:15 – 12:00	Entwicklungsumgebung schützen
13:00 – 13:30	Statische Seite testen
13:30 – 14:10	Datenbank designen und SQL-Code erstellen
14:10 – 15:30	Datenbank + Zugangsberechtigung für jeden Schüler auf dem Webserver erstellen
15:30 – 16:00	FTP-Upload der statischen Seite
16:00 – 16:25	Fortfahren mit Aufgabe2
16:25 – 16:30	Abräumen des Arbeitsplatzes

## Tag 2: 2. Oktober 2006

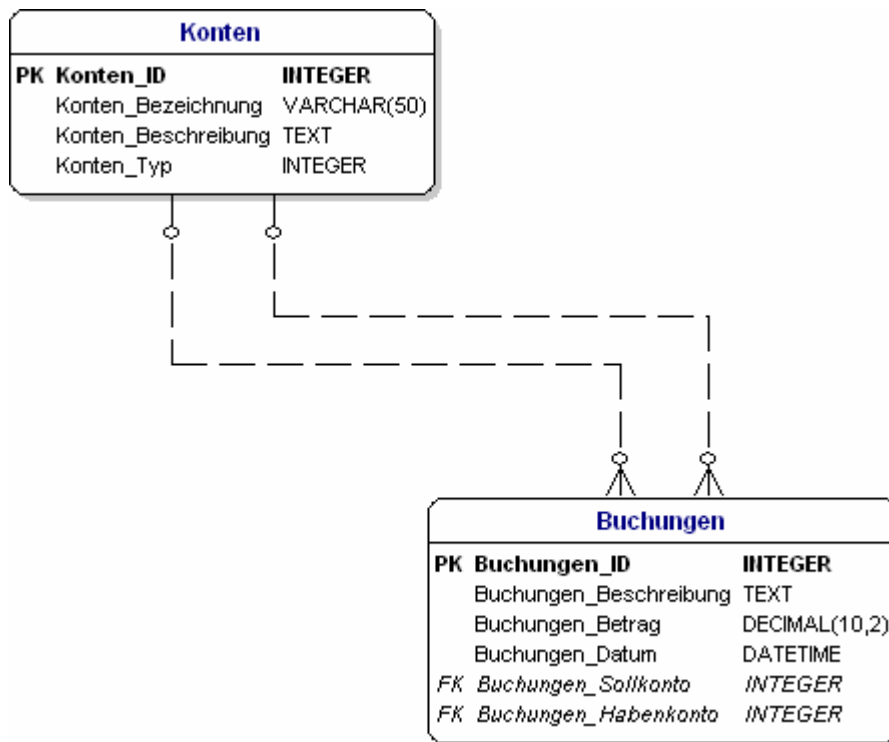
<b>Zeit:</b>	<b>Tätigkeit:</b>
08:00 – 08:30	Einrichten des Arbeitsplatzes
08:30 – 09:00	Weiterarbeiten an Aufgabe 2
09:00 – 09:20	Backup der Datenbank auf den Webserver überspielen
09:20 – 10:00	Netzwerk kontrollieren
10:30 – 12:00	Weiterarbeiten an Aufgabe 2
13:00 – 13:45	Weiterarbeiten an Aufgabe 2
13:45 – 14:30	Versuch, InnoDB zu aktivieren – leider erfolglos
14:30 – 15:00	Aufgabe 3: ERD erstellt.
15:00 – 15:30	Aufgabe 3: SQL-Dump erstellt, Datenbank eingerichtet
15:00 – 16:30	Aufgabe 3: Datenbankabfrage anpassen, Applikation erweitern
16:30 – 17:00	Backup der DB erstellen, Produktivserver abrechnen

## Tag3: 3. Oktober 2006

<b>Zeit:</b>	<b>Tätigkeit:</b>
08:00 – 08:30	Einrichten des Arbeitsplatzes
08:30 – 09:00	Aufgabe 3: Datenbankabfragen entwickeln
09:00 – 09:30	Aufgabe 3: Datenbankabfragen testen
09:50 – 10:30	Dokumente (ERD, SQL-Dump) in Logbuch übernehmen, Seite uploaden
11:00 – 12:00	Aufgabe 4: Verschlüsselungsalgorithmus studieren
13:00 – 13:30	Aufgabe 4: Übungsbeispiel zu ENCODE und DECODE
13:30 – 14:00	Aufgabe 4: Bugs beheben
14:00 – 14:30	Aufgabe 4: ENCODE und DECODE auf Airline-Buchhaltung anwenden
14:30 – 15:00	Dokumente und SQL-Statements in Logbuch nachtragen, Neue Seite uploaden

## Aufgabe 3:

ERD:



## SQL-Dump:

→ siehe: *db\_dump.sql*

Enthaltene SQL-Befehle:

- Tabellenstruktur erstellen
- Testdaten einfügen
- Benutzer erstellen, Berechtigungen setzen (siehe unten)
- Aufgabe3-spezifische Abfragen (siehe unten)

## Auszug aus db\_dump.sql:

(...)

# Benutzer mit Schreibberechtigung:

#####

```
CREATE USER 'benutzer1'@ '%' IDENTIFIED BY 'helloWorld';
```

```
GRANT ALL PRIVILEGES ON * . *
TO 'benutzer1'@ '%'
IDENTIFIED BY 'helloWorld'
WITH GRANT OPTION
MAX_QUERIES_PER_HOUR 0
MAX_CONNECTIONS_PER_HOUR 0
MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0 ;
```

# Benutzer nur mit Leseberechtigung:

#####

```
CREATE USER 'benutzer2'@ '%' IDENTIFIED BY 'phpBuch';
```

```
GRANT SELECT ON * . *
TO 'benutzer2'@ '%'
IDENTIFIED BY 'phpBuch'
WITH MAX_QUERIES_PER_HOUR 0
MAX_CONNECTIONS_PER_HOUR 0
MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0 ;
```

# Summe der Beträge auf der Soll-Seite eines Kontos:

#####

```
# SELECT SUM(Buchungen_Betrag) FROM Buchungen WHERE Buchungen_Sollkonto =
{ bitte Kontennummer einfügen};
```

# Summe der Beträge auf der Haben-Seite eines Kontos:

#####

```
# SELECT SUM(Buchungen_Betrag) FROM Buchungen WHERE Buchungen_Habenkonto =
{ bitte Kontennummer einfügen};
```

# Falsche Buchungen (Sollkonto == Habenkonto):

#####

```
# SELECT * FROM Buchungen WHERE Buchungen_Sollkonto = Buchungen_Habenkonto;
```

(...)

## Aufgabe 4:

### DB Verschlüsselung

Die Kontennamen (Konten.Konten\_Bezeichnung) besitzen den Datentyp VARCHAR, der mit den SQL-Befehlen ENCODE und DECODE verschlüsselt werden kann.

### Auszug aus db\_dump2.sql:

(...)

```
# Testeinträge für Tabelle Konten:
```

```
#####
```

```
INSERT INTO konten VALUES (1000, ENCODE('Kasse', 'lol'), NULL, 0);  
INSERT INTO konten VALUES (1500, ENCODE('Debitoren', 'lol'), NULL, 0);  
INSERT INTO konten VALUES (2000, ENCODE('Kreditoren', 'lol'), NULL, 2);  
INSERT INTO konten VALUES (2500, ENCODE('Eigenkapital', 'lol'), NULL, 2);  
INSERT INTO konten VALUES (3000, ENCODE('Warenaufwand', 'lol'), NULL, 4);  
INSERT INTO konten VALUES (3000, ENCODE('Warenertrag', 'lol'), NULL, 5);
```

(...)

### Anpassungen der Datenbank:

Da nur das Feld Konten\_Bezeichnung verschlüsselt wird und es bereits den Datentyp VARCHAR besitzt, müssen keine Änderungen an der Datenbank vorgenommen werden.

## Änderungen der Applikation:

(Auszug aus kontenplan.php):

(...)

```
include("encode.inc");
```

(...)

```
$sql = 'select Konten_ID, decode(Konten_Bezeichnung, "'. $encode_passwort. "') from
      Konten where Konten_Typ like 0 order by Konten_ID';
$result = mysql_query($sql);
while($row = mysql_fetch_array($result))
{
    $aktiv_array[$aktiv_id][0] =
    $row['decode(Konten_Bezeichnung, "'. $encode_passwort. "')'];
```

(...)

Um die Daten wieder zu decodieren wird das SQL-Statement  
DECODE({Attribut}, „{Passwort}“) verwendet.

## Testprotokoll:

Nr	Beschreibung	Soll-Wert	IST-Wert	Ok?
1	Neues Konto hinzufügen	Konten_Bezeichnung verschlüsselt in DB gespeichert	Konten_Bezeichnung verschlüsselt in DB gespeichert	ja
2	Kontenplan anzeigen	Konten_ID und Konten_Bezeichnung korrekt geschrieben	Konten_ID und Konten_Bezeichnung korrekt geschrieben	ja
3	Buchung ausführen	Inhalt der Dropdown-Menüs korrekt geschrieben.	Inhalt der Dropdown-Menüs korrekt geschrieben	ja
4	Hauptbuch anzeigen	Konten_ID, Konten_Bezeichnung und Konten_Betrag korrekt	Konten_ID, Konten_Bezeichnung und Konten_Betrag korrekt	ja
5	Kontenplan anzeigen	Testfall 3 hat Kontenstand der betreffenden Konten verändert.	Testfall 3 hat Kontenstand der betreffenden Konten verändert.	ja