

Modern Information Systems WS 11/12

Nachname	Vorname	Matrikelnummer
Hasenauer	Hannes	0930213
Fritzscht	Daniel	0930754
Temmel	Hans Christian	0930308

Inhaltsverzeichnis

1	Database Schema	2
1.1	Entity-Relationship Diagram	2
1.2	Domains	3
1.3	Relations	3
1.3.1	Additional Description	3
1.4	Functional Dependencies	3
1.4.1	Additional Description	4
2	Practical Implementation	5
2.1	Waiter	5
2.1.1	Relation	5
2.1.2	Data	5
2.1.3	Screenshot	5
2.2	Table	5
2.2.1	Relation	5
2.2.2	Data	6
2.2.3	Screenshot	6
2.3	Burger	6
2.3.1	Relation	6
2.3.2	Data	6
2.3.3	Screenshot	7
2.4	SideDish	7
2.4.1	Relation	7
2.4.2	Data	7
2.4.3	Screenshot	7
2.5	SalesSlip	8
2.5.1	Relation	8
2.5.2	Data	8
2.5.3	Screenshot	9
3	Relevant Code Snippets	10
3.1	Adding a new Burger	10
3.2	Delete a Burger	10
3.3	Get burgers that are served by a given waiter to a specific table	11
3.4	Get all burgers that have been served by a a specific waiter	12

1 Database Schema

1.1 Entity-Relationship Diagram

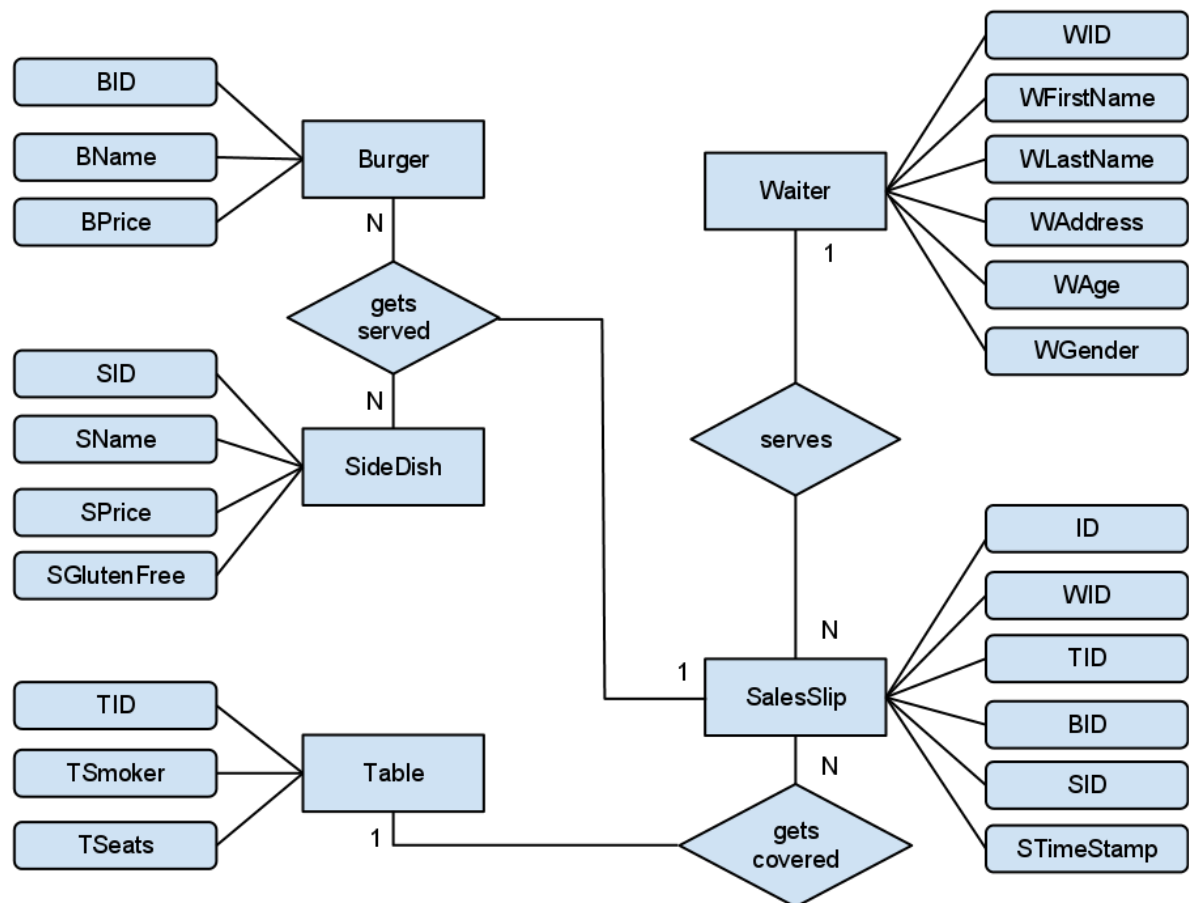


Abbildung 1: Entity-Relationship Diagram

1.2 Domains

Name	DataType	Description
WID WFirstName WLastName WAddress WAge WGender	Integer String String String Integer Integer	Waiter ID Firstname of the waiter Lastname of the waiter Address of the waiter Age of the waiter Gender of the waiter
TID TSmoker TSeats	Integer Boolean Integer	Table ID Smokertable? Amount of seats
BID BName BPrice	Integer String Integer	Burger ID Name of the burger Price of the burger
SID SName SPrice SGlutenfree	Integer String Integer Boolean	Sidedish ID Name of the sidedish Price of the sidedish Glutenfree?
ID WID TID BID SID STimeStamp	Integer Integer Integer Integer Integer TimeStamp	Sales Slip ID Waiter ID Table ID Burger ID Sidedish ID The actual timestamp

1.3 Relations

Waiter	Table	Burger	SideDish	SalesSlip
<u>WID</u> WFirstName WLastName WAddress WAge WGender	<u>TID</u> TSmoker TSeats	<u>BID</u> BName BPrice	<u>SID</u> SName SPrice SGlutenFree	<u>ID</u> WID TID BID SID STimeStamp

1.3.1 Additional Description

The domains WID, TID, BID and SID are the foreign keys in the relation SalesSlip.

1.4 Functional Dependencies

- Waiter(WID, WFirstName, WLastName, WAddress, WAge, WGender)
 - WID \rightarrow WFirstName
 - WID \rightarrow WLastName
 - WID \rightarrow WAddress
 - WID \rightarrow WAge
 - WID \rightarrow WGender
- Table(TID, TSmoker, TSeats)
 - TID \rightarrow TSmoker
 - TID \rightarrow TSeats

- Burger(BID, BName, BPrice)
 - $BID \rightarrow BName$
 - $BID \rightarrow BPrice$
- SideDish(SID, SName, SPrice, SGlutenFree)
 - $SID \rightarrow SName$
 - $SID \rightarrow SPrice$
 - $SID \rightarrow SGlutenFree$
- SalesSlip(ID, WID, TID, BID, SID, STimeStamp)
 - $ID \rightarrow WID$
 - $ID \rightarrow TID$
 - $ID \rightarrow BID$
 - $ID \rightarrow SID$
 - $ID \rightarrow STimeStamp$

1.4.1 Additional Description

Every domain in the relation is only identifiable by a unique primary key. Therefore there are no indirect key dependencies and the database is in the 3rd normal form.

2 Practical Implementation

For all primary keys auto increment values are used. This is done to assign a unique identifier to every relation who contains a primary key automatically.

2.1 Waiter

2.1.1 Relation

```
CREATE TABLE IF NOT EXISTS 'Waiter' (  
  'WID' int(5) unsigned NOT NULL AUTO_INCREMENT,  
  'WFirstName' varchar(30) NOT NULL,  
  'WLastName' varchar(30) NOT NULL,  
  'WAddress' varchar(100) NOT NULL,  
  'WAge' int(3) NOT NULL,  
  'WGender' varchar(10) NOT NULL,  
  PRIMARY KEY ('WID')  
);
```

2.1.2 Data

```
INSERT INTO 'Waiter' ('WFirstName', 'WLastName', 'WAddress', 'WAge', 'WGender') VALUES  
  ('Daniel', 'Fritzs', 'Daniels Address', 20, 'male');  
INSERT INTO 'Waiter' ('WFirstName', 'WLastName', 'WAddress', 'WAge', 'WGender') VALUES  
  ('Hannes', 'Hasenauer', 'Hannes Address', 23, 'male');  
INSERT INTO 'Waiter' ('WFirstName', 'WLastName', 'WAddress', 'WAge', 'WGender') VALUES  
  ('Hans Christian', 'Temmel', 'Hans Christians Address', 25, 'male');  
INSERT INTO 'Waiter' ('WFirstName', 'WLastName', 'WAddress', 'WAge', 'WGender') VALUES  
  ('Sandra', 'Steal', 'Sandra's Address', 15, 'female');
```

2.1.3 Screenshot

WID	WFirstName	WLastName	WAddress	WAge	WGender
1	Daniel	Fritzs	Daniels Address	20	male
2	Hannes	Hasenauer	Hannes Address	23	male
3	Hans Christian	Temmel	Hans Christians Address	25	male
4	Sandra	Steal	Sandra's Address	15	female

Abbildung 2: Waiter

2.2 Table

2.2.1 Relation

```
CREATE TABLE IF NOT EXISTS 'Table' (  
  'TID' int(5) unsigned NOT NULL AUTO_INCREMENT,  
  'TSmoker' tinyint(1) NOT NULL DEFAULT FALSE,  
  'TSeats' int(2) unsigned NOT NULL,  
  PRIMARY KEY ('TID')  
);
```

2.2.2 Data

-- NonSmoking Tables

```
INSERT INTO 'Table' ('TSeats') VALUES (4);
INSERT INTO 'Table' ('TSeats') VALUES (4);
INSERT INTO 'Table' ('TSeats') VALUES (2);
INSERT INTO 'Table' ('TSeats') VALUES (5);
```

-- Smoking Tables

```
INSERT INTO 'Table' ('TSmoker', 'TSeats') VALUES (true, 4);
INSERT INTO 'Table' ('TSmoker', 'TSeats') VALUES (true, 2);
INSERT INTO 'Table' ('TSmoker', 'TSeats') VALUES (true, 5);
```

2.2.3 Screenshot

TID	TSmoker	TSeats
1	0	4
2	0	4
3	0	2
4	0	5
5	1	4
6	1	2
7	1	5

Abbildung 3: Table

2.3 Burger

2.3.1 Relation

```
CREATE TABLE IF NOT EXISTS 'Burger' (  
  'BID'      int(5) unsigned NOT NULL AUTO_INCREMENT,  
  'BName'    varchar(25) NOT NULL,  
  'BPrice'   float(4,2) NOT NULL,  
  PRIMARY KEY ('BID')  
);
```

2.3.2 Data

```
INSERT INTO 'Burger' ('BName', 'BPrice') VALUES  
  ('Manhattan Burger', 6);  
INSERT INTO 'Burger' ('BName', 'BPrice') VALUES  
  ('NewYork Classic', 6.5);  
INSERT INTO 'Burger' ('BName', 'BPrice') VALUES  
  ('Extra Meat Burger', 8);  
INSERT INTO 'Burger' ('BName', 'BPrice') VALUES  
  ('Hot Chicken Burger', 7.25);  
INSERT INTO 'Burger' ('BName', 'BPrice') VALUES  
  ('Austrian Burger', 7.5);
```

2.3.3 Screenshot

BID	BName	BPrice
1	Manhattan Burger	6.00
2	NewYork Classic	6.50
3	Extra Meat Burger	8.00
4	Hot Chicken Burger	7.25
5	Austrian Burger	7.50

Abbildung 4: Burger

2.4 SideDish

2.4.1 Relation

```
CREATE TABLE IF NOT EXISTS 'SideDish' (  
  'SID'          int(5) unsigned NOT NULL AUTO_INCREMENT,  
  'SName'        varchar(15) NOT NULL,  
  'SPrice'        float(4,2) NOT NULL,  
  'SGlutenFree'  tinyint(1) NOT NULL DEFAULT FALSE,  
  PRIMARY KEY ('SID')  
);
```

2.4.2 Data

```
INSERT INTO 'SideDish' ('SName', 'SPrice', 'SGlutenFree') VALUES  
  ('Pommes', 2.5, true);  
INSERT INTO 'SideDish' ('SName', 'SPrice') VALUES  
  ('Potatoes', 2);  
INSERT INTO 'SideDish' ('SName', 'SPrice') VALUES  
  ('Potato Stripes', 2.7);  
INSERT INTO 'SideDish' ('SName', 'SPrice', 'SGlutenFree') VALUES  
  ('Rice', 2, true);
```

2.4.3 Screenshot

SName	SPrice	SGlutenFree
Pommes	2.50	1
Potatoes	2.00	0
Potato Stripes	2.70	0
Rice	2.00	1

Abbildung 5: SideDish

2.5 SalesSlip

2.5.1 Relation

```
CREATE TABLE IF NOT EXISTS 'SalesSlip' (  
  'ID'          int(100) unsigned NOT NULL AUTO_INCREMENT,  
  'WID'         int(5) unsigned NOT NULL,  
  'TID'         int(5) unsigned NOT NULL,  
  'BID'         int(5) unsigned,  
  'SID'         int(5) unsigned,  
  'STimeStamp' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  PRIMARY KEY ('ID')  
);
```

2.5.2 Data

```
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (1, 5, 1, 1);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (1, 6, 2, 2);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (1, 7, 3, 3);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID', 'STimeStamp') VALUES  
  (1, 5, 0, 2, '2010-11-18 12:00:30');  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (1, 6, 1, 3);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (1, 7, 3, 1);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID', 'STimeStamp') VALUES  
  (2, 3, 5, 2, '2009-08-10 10:04:30');  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (2, 4, 3, 1);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID', 'STimeStamp') VALUES  
  (2, 4, 1, 3, '2011-04-03 16:00:42');  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (2, 6, 2, 2);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID', 'STimeStamp') VALUES  
  (3, 2, 2, 2, '2008-06-05 14:05:42');  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (3, 2, 1, 4);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (4, 1, 4, 3);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID') VALUES  
  (4, 1, 2, 4);  
INSERT INTO 'SalesSlip' ('WID', 'TID', 'BID', 'SID', 'STimeStamp') VALUES  
  (4, 1, 5, 2, '2010-09-14 17:00:51');
```


2.5.3 Screenshot

ID	WID	TID	BID	SID	STimeStamp
1	1	5	1	1	2011-12-23 16:55:51
2	1	6	2	2	2011-12-23 16:55:51
3	1	7	3	3	2011-12-23 16:55:51
4	1	5	0	2	2010-11-18 12:00:30
5	1	6	1	3	2011-12-23 16:55:51
6	1	7	3	1	2011-12-23 16:55:51
7	2	3	5	2	2009-08-10 10:04:30
8	2	4	3	1	2011-12-23 16:55:52
9	2	4	1	3	2011-04-03 16:00:42
10	2	6	2	2	2011-12-23 16:55:52
11	3	2	2	2	2008-06-05 14:05:42
12	3	2	1	4	2011-12-23 16:55:52
13	4	1	4	3	2011-12-23 16:55:52
14	4	1	2	4	2011-12-23 16:55:52
15	4	1	5	2	2010-09-14 17:00:51

Abbildung 6: SalesSlip

3 Relevant Code Snippets

3.1 Adding a new Burger

```
1 <form action="addBurger.php" method="POST">
  <table width="600" border="0" cellpadding="0" cellspacing="0">
3     <tr>
        <td width="128">Name:</td>
5         <td width="472"><input type="text" name="bName"></td>
    </tr>
7     <tr>
        <td>Price</td>
9         <td><input type="text" name="bPrice"></td>
    </tr>
11  </table>

13  <input type="submit" value="Submit"></input>
</form>

15 <?php
17     if (isset ($_POST["bName"]) && $_POST["bPrice"]) {
19         $bName = $_POST["bName"];
20         $bPrice = $_POST["bPrice"];
21         $query = "INSERT INTO 'Burger' ('BID', 'BName', 'BPrice')
22                 VALUES (NULL, '$bName', '$bPrice');";
23
24         mysql_query($query) or die (mysql_error());
25         echo ("Burger $bName has been added to database");
26     }
27 ?>
```

Listing 1: addBurger.php

This file is used to add a new burger to the database.

3.2 Delete a Burger

```
1 <form action="deleteBurger.php" method="POST">
<?php
3     if(isset ($_POST["burger"])) {
        $burger = $_POST["burger"];
5         $query = "DELETE FROM 'Burger' WHERE BName='$burger'";

7         mysql_query($query) or die (mysql_error());
    }
9     $burgers = mysql_query("select * from 'Burger';") or die (mysql_error());
?>

11 <select name="burger">
<?php
13     while($burger = mysql_fetch_array($burgers)) {
15         echo ('<option value="' . $burger["BName"] . '">' .
                $burger["BName"] . '</option>');
17     }
?>
```

```
19 </select>
21 <input type="submit" value="delete">
22 </form>
23
24 <?php
25     if(isset ($_POST["burger"])) {
26         $burger = $_POST["burger"];
27
28         echo (" '$burger' has been deleted!");
29     }
30 ?>
```

Listing 2: deleteBurger.php

This file is used to delete an existing burger from the database.

3.3 Get burgers that are served by a given waiter to a specific table

```
1 <form action="getBurgers.php" method="POST">
2
3 <?php
4     $waiters = mysql_query("select * from 'Waiter'") or die(mysql_error());
5     $tables = mysql_query("SELECT * from 'Table'") or die(mysql_error());
6 ?>
7
8 <select name="waiter">
9     <?php
10         while($waiter = mysql_fetch_array($waiters)) {
11             if ($_POST["waiter"] == $waiter["WID"]) {
12                 echo '<option selected = yes value="' . $waiter["WID"] . '">' .
13                     $waiter["WLastName"] . " " . $waiter["WFirstName"] .
14                     '</option>';
15             } else {
16                 echo '<option value="' . $waiter["WID"] . '">' .
17                     $waiter["WLastName"] . " " . $waiter["WFirstName"] .
18                     '</option>';
19             }
20         }
21     ?>
22 </select>
23 <select name="table">
24     <?php
25         while($table = mysql_fetch_array($tables)) {
26             if ($_POST["table"] == $table["TID"]) {
27                 echo '<option selected = yes value="' . $table["TID"] . '">' .
28                     "Table Number: " . $table["TID"] . '</option>';
29             } else {
30                 echo '<option value="' . $table["TID"] . '">' . "Table Number: " .
31                     $table["TID"] . '</option>';
32             }
33         }
34     ?>
35 </select>
36
```

```

38 <?php
40     if (isset ($_POST["waiter"]) && isset ($_POST["table"])) {
42         $wid      = $_POST["waiter"];
43         $tid      = $_POST["table"];
44         $waiter = mysql_query("select * from Waiter where WID = $wid");
45         $name    = mysql_fetch_object($waiter);
46
47         $result = mysql_query("select BName from Burger where BID in
48                               (select BID from SalesSlip where
49                                WID = $wid and TID = $tid)");
50
51         while ($burger = mysql_fetch_array($result)) {
52             echo $burger["BName"] . " at table " . $tid . " by "
53                 . $name->WFirstName . " " . $name->WLastName;
54         }
55     }
56 ?>

```

Listing 3: getBurgers.php

Gets all burgers that are served by a given waiter to a specific table.

3.4 Get all burgers that have been served by a a specific waiter

```

<form action="waiterStatistic.php" method="POST">
2   <table width="600" border="0" cellpadding="0" cellspacing="0">
3       <tr>
4           <td width="125">Waiter:</td>
5           <td width="475"><?php
6               $waiters = mysql_query("select * from Waiter") or die (mysql_error());
7               ?>
8
9               <select name="waiter">
10                  <?php
11                      while ($waiter = mysql_fetch_array($waiters)) {
12                          if ($_POST["waiter"] == $waiter["WID"]) {
13                              echo '<option selected = yes value = "' . $waiter["WID"] . '">' .
14                                  $waiter["WLastName"] . " " . $waiter["WFirstName"] .
15                                  '</option>';
16                          } else {
17                              echo '<option value = "' . $waiter["WID"] . '">' .
18                                  $waiter["WLastName"] . " " . $waiter["WFirstName"] .
19                                  '</option>';
20                          }
21                      }
22                  ?>
23              </select>
24          </td>
25      </tr>
26      <tr>
27          <td>Smoker Table:</td>
28          <td>
29
30

```

```
32     <?php
    if (isset ($_POST["smokertable"]))
        echo '<input type="checkbox" name="smokertable" checked>';
34     else
        echo '<input type="checkbox" name="smokertable">';
36     ?>
    </td>
38 </tr>
</table>
40 </form>
42 <?php
    if (isset ($_POST["waiter"])) {
44         $wid = $_POST["waiter"];

46         if (isset ($_POST["smokertable"]))
            $smoker = 1;
48         else
            $smoker = 0;

50         $result =
52             mysql_query("SELECT * FROM Burger WHERE BID IN
                           (SELECT BID FROM 'SalesSlip' WHERE WID = $wid
54                           AND TID IN (SELECT TID FROM 'Table' WHERE
                           TSmoker = $smoker AND TID IN
56                           (SELECT TID FROM 'SalesSlip' WHERE WID = $wid))
                           )");
58
        while ($burger = mysql_fetch_array($result)) {
60             echo $burger["BName"]; ?><br><?php
        }
62     }
    ?>
```

Listing 4: waiterStatistic.php

You can choose a waiter to see the statistic of the Burgers that have been served by him.