Course Name: Cmpe 321 – Introduction to Database Systems
Semester: 2017/2018 Summer Term
Student Name: Burak Çuhadar
Student Number: 2015400153

Restaurant Management System

7 August 2018

Introduction

In this assignment we are expected to implement a restaurant management system. This system should provide a web-based user interface to enable the users to perform the operations. To implement these operations we should connect to a database system in our code.

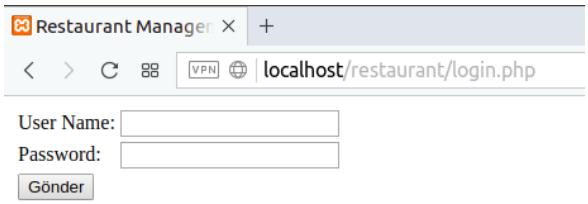
I have used xampp development environment for php to implement the project. This environment comes with a local web server and mysql dbms. I have chosen this environment because php can be used easily to implement the user interface with html.

To avoid SQL injection I have used prepared statements in php. To create a prepare statement in php we should have database connection. With calling connection -> prepare(sql) function statement is prepared. The parameter sql should be a string containing the sql query in which the places where the input should go are represented as question marks(?). The bindparam function for the statement binds the inputs taken from the user into these marks. However when executing the query, the query and the input data are sent to database server separately so that the SQL injection can be prevented.

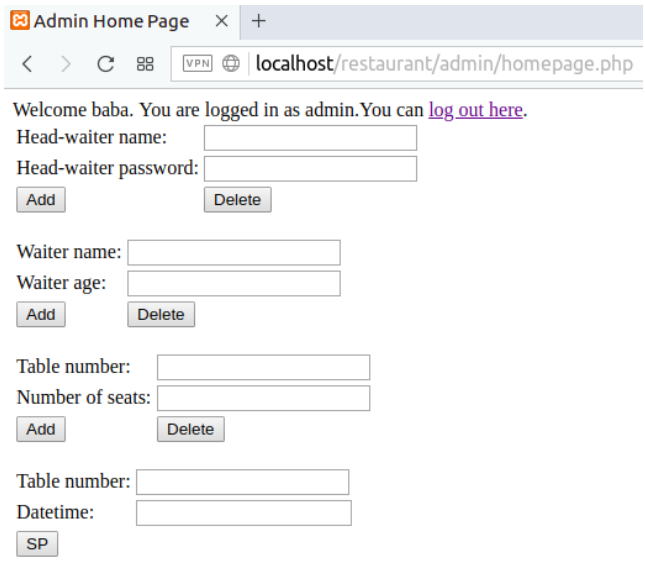
My login page is a simple html page containing fields for user name, password and a submit button. Clicking the button redirects the user to check.php whose job is to check the database if the user exists. If it exists it redirects the user to its appropriate homepage. There are two homepages which are for admins and head-waiters. These homepages look very similar to login page. Each button in the home pages redirects to the appropriate php code. These php codes look also very similar as every operation is add and delete. When I implemented one of these, I copied and pasted it and made the appropriate changes. Therefore this part of the project took little time. I have also wrote a helper php file called db_connection. Including this in every php file where database connection is required, establishes the database connection. The last php file called logout simply destroys the session and redirects the user to the login page. Lastly, my stored procedure and trigger can be found in database.sql file.

Interface

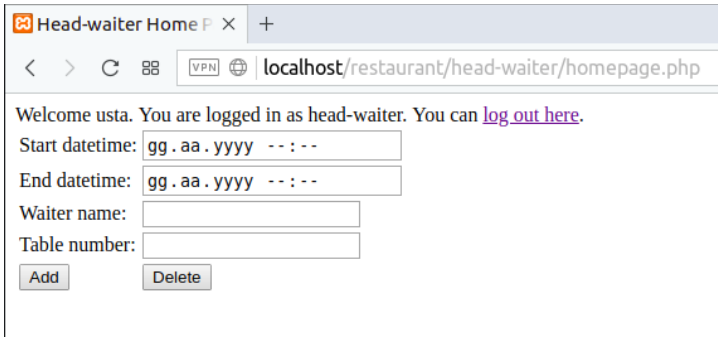
Login page of the management system:



Home page for the admin users:

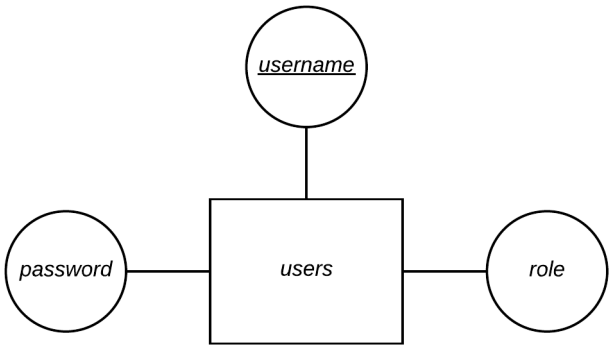


Home page for the head-waiter users

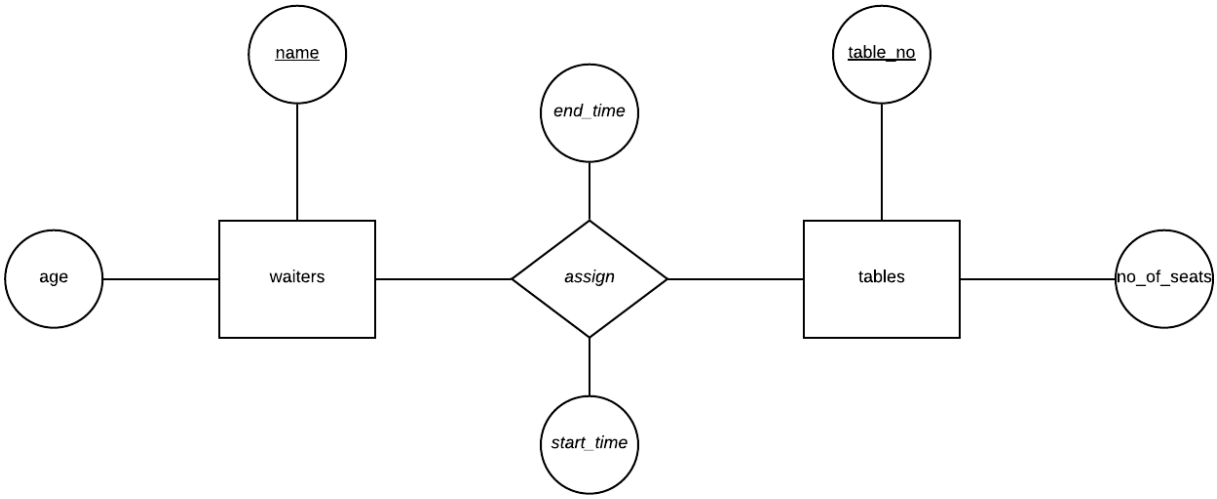


Database

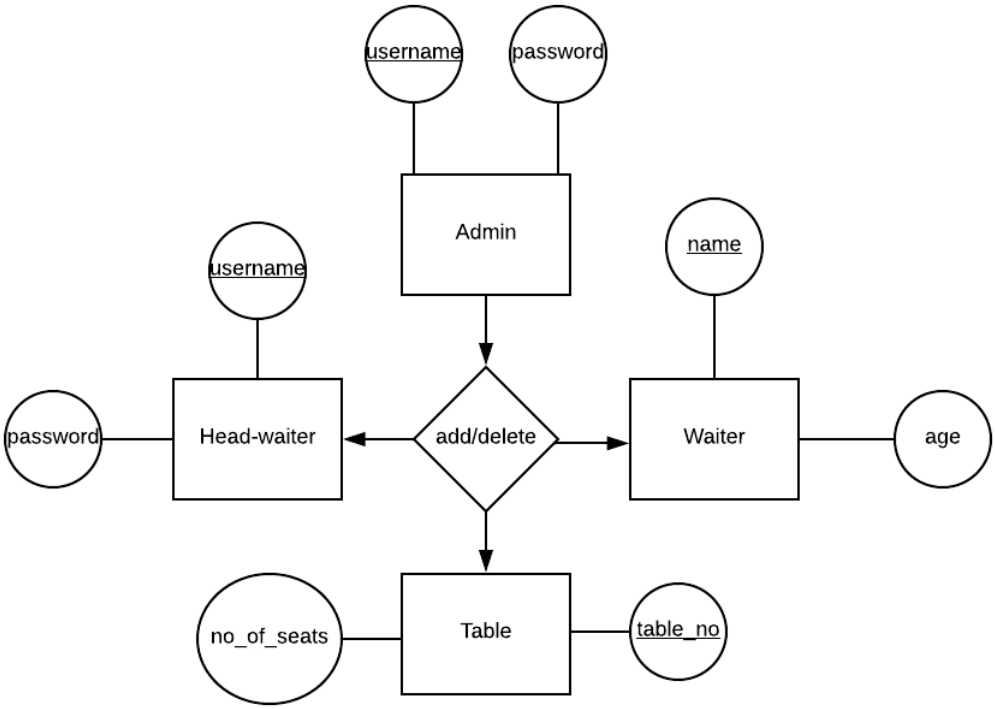
- E-R diagram for the users that can log in to the system: Username is key. Role is enum which can take two values ‘admin’ and ‘head-waiter’. Both head-waiters and admins are stored here.



- E-R diagram representing the assign relationship: End time and start time are in datetime format. A waiter is assigned to a table provided the end and start time of the assignment.



- E-R diagram representing add/delete operations of admin user: Admin can delete waiters, head-waiters and tables with providing their keys. To add them it has to provide all the fields.



Conclusion

In the end, I have successfully implemented the project. Using xampp environment and php has made it easy to implement. I have found the way that php and html work together intuitive and convenient. The database tools in xampp were also easy to learn and use.

The most difficult parts of the project was to learn php and adapting to web development which I was not familiar with before. Nevertheless it was not that difficult because of the easiness xampp and php. Writing SQL queries were easier as the project does not require complicated queries. Another thing I have learned doing this project was learning about SQL injection which is an interesting topic.