SOLENT UNIVERSITY

SCHOOL OF MEDIA ARTS AND TECHNOLOGY

MSc Computer Engineering

Academic year 2020

A report for Web Technology

Leila Najafi

STUDENT NUMBER: Q14095661

Tutor: **Prins Butt**

Contents

Introduction

There is no doubt using a website is essential for every business to be successful these days and it is true most for sellers. In this report I will explain a fundamental aspect of a simple online shop. This online shop is a dynamic website which interacts with user and implements user requests by connecting to server. The website created for this shop has 3 type of users, admin, author, and ordinary user(customer). This website allows user to register and update his/her account. Author can do the CRUD functions for products table which is containing products' details and finally admin can do these functions for products and users. Therefore, admin can change the user's role to be an author or admin. In following paragraphs, the database structure will be discussed and later the report shows use of different programming languages to make links to the database.

Database and data structure

The database used for this website containing tables:

Users, roles, orders, products, category as main table and orders_products table as a link table to make a relationship between two tables orders and products. The structure is as shown in the figure below:
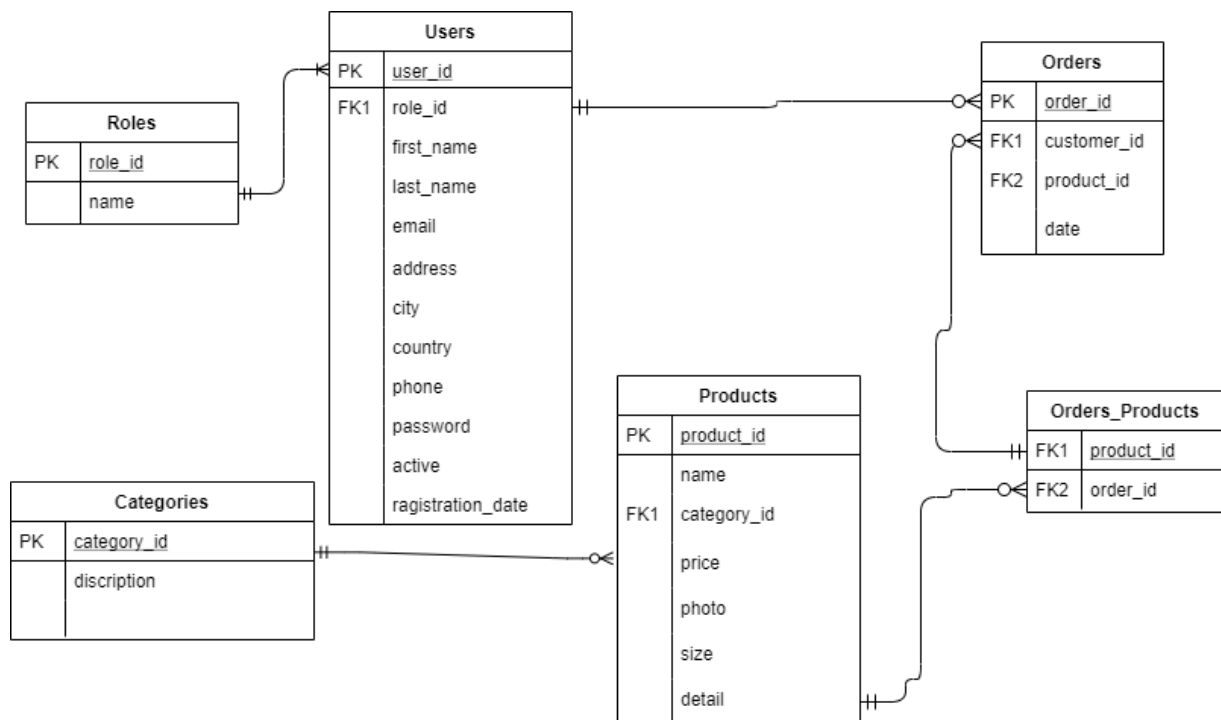
Normalisation

The structure shown earlier is in 3NF normalisation form because there are no dependencies on non-key attributes.

Back-End

Back-End refers to server side programming. PHP is one the server side programming languages which is scripting language. PHP works with DB and Front-end developing languages like HTML, CSS and JavaScript to implement a dynamic web applications.

To have access to DB and it's tables php needs to make a connection to the server. The following line used in PHP to make that connection:

```php
$dbc = @mysqli_connect (DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
```

That is a good practice to handle errors if the connection fails for any reason:

```php
// If no connection could be made, trigger an error:
if (!$dbc) {
    trigger_error('Could not connect to MySQL: ' . mysqli_connect_error() );
} else { // Otherwise, set the encoding:
    mysqli_set_charset($dbc, 'utf8');
}
```

Other error handling codes can be found in file "indlude/config.inc.php".

CRUD Functions

CRUD functions which are insert, delete, update and select can be executed for each table. However depends on each user privileges the application would let the user to do them.

For users table, every user can insert a record (a record for an email) by filing a registration form as shown below:

By clicking Register button a record will inserted for the new user. The two figures show the user table before and after it:

Before:

And after:

The code for inserting can be found in file register.php. The query for inserting is:

$q = "INSERT INTO users (first_name, last_name,email, password, active, registration_date) VALUES ('$fn', '$ln','$e', '$p', '$a', NOW())";

Before running this query all the user inputs validate by php to have correct values for variables, '$fn', '$ln','$e', '$p', and '$a'.

After registration user can login:

Login

Email Address: 4najal61@solent.ac.uk

Password: ••••••••••

[Login]

Front-End

Front-end is all about tools and languages used to develop pages for user interface. Using HTML,CSS,JavaScript as front end tools the application can interact with users. Html codes are within PHP codes and CSS used to implement the designing of pages. There are html files for including the header and footer for each page. These files are pure html5 codes with no php codes.

Header.html contains all the section to implement header and all the links needed to run the application such as links to css files and libraries like bootstrap. There is a menu able user to navigate the site.

Footer.html contains the tags used to have footer which contains links for dashboard and social medias.

Index.html is another index file which is the starting page for the website.

Other pages are php pages which included header.html at the first and footer.html at the end.

The website is mobile first

Validating the user inputs

User inputs can be validated using html, JavaScript and php. Using html by applying several attributes there will be limited inputs accepted. By applying type attribute inside input tag html restricts the user to input correct format of inputs. For example for email this line of html code would be usefule:

<p>Email Address: <input type="email" name="email" size="30" maxlength="60" value="<?php if (isset($trimmed['email'])) echo $trimmed['email']; ?>">

</p>

For password the following code can be applied:

<p>Password: <input type="password" name="password1" size="20" value="<?php if (isset($trimmed['password1'])) echo $trimmed['password1']; ?>"> <small>At least 10 characters long.</small></p>

Using JavaScript there would be more restrictions to lead user for correct and in some cases more secure inputs.

Conclusion

This website is a basic online shop which can improved by adding more features such as payment methods, filtering the products list and adding blog page to give more related information to site visitors. As well as design and implementation improvement the SEO principle needs to be considered by developer to have SEO friendly website.