Report: Web-Dev 306 Project

Steps to follow to understand the philosophy of the page:

1. Go to register page and enter your credentials and then click register.
2. Go to my account page and enter your credentials to enter in the products page. (Click LogIn)
3. Go to PHP my\_admin and import the excel file that is given to the folder to coffeeproducts inside the coffee\_chart database.
4. Go back to the products page, refresh it and click on each of then that you are willing to buy and then click proceed.
5. Then, you will be redirected to the page where all your products are displayed as well as their number and total price into separated boxes.
6. If you want, you can delete one or more and then refresh the page to update the results.
7. If you want, you can change password by entering the old one correctly and then inserting the new one by clicking on the Change Password button.

This project is an extension of the previous one where jQuery, PHP as well as ajax is added. Regarding the features that are added I am providing them in the form of the pages below:

Account\_Registration /dbconfig.php🡪This page contains functions that helps into connecting with SQL, creating a database for new users, connecting to that database, and performing operations to it such as creating the user\_register table, inserting the new users in the table and validation of the email and password (if the have been entered correctly). Also, a function for updating the password has been added in case the user wants to change it.

Account\_Registration /My\_Account\_Registration\_Page.php🡪This page has been updated from html to PHP and form validation has been added with jQuery as well as a redirection to the form to StoreUser.php to store what the user adds to each of the fields in the MySQL database.

Account\_Registration/ StoreUser.php🡪This page is the one that you will get redirected to after adding all the fields provided and clicking Register on the My\_Account\_Registration\_Page.php and then it will redirect you back to it after adding the data to the register table using the dbconfig.php.

The same procedure follows even the My\_Account\_Page/My\_Account\_Page.php. It has a form that will make the user Login to the hidden link of Coffee\_Page.php. This form undergoes form validation using jQuery as well and it needs to send the data of this form to the Authenticates.php, so PHP can see if the credentials of one of the users matches with one of the credentials on the table given in user\_register table on MySQL inside user\_login database.

After redirection to Authenticates.php page (Note this page has its own dbconfig.php file inside My\_Account\_Page where it will get specific functions). After authentication, this page will send the UserID and the Password for the user who entered the credentials to Coffee\_Page.php.

(Note: Please in this step, import the excel file given in this folder to coffeeproducts inside coffee\_chart database that got created). We need products table to retrieve some of the information for each of the products that we are going to put in our “trolley” on Coffee\_Page.php.  
Redirecting the user to Coffee\_Page.php the user will decide to immediately tap to the Checkout Button to see his previously added products to the “trolley” after logging out or he/she can add new or more products by clicking at the AddToChart Button at each of the products. Since we do not want the page to get redirected to another one after adding one product to the chart, I have used ajax form to send the data to Chart.php page. So, this page will get the Product ID index from a specific box (that is the same with the one that has the same information about a product on the Coffee Products table) as well as the UserID and will transmit data to Chart.php. On Chart.php, we call functions for creating a new table that is a one-to-many relationship between products and buyers (this page has its own dbconfig inside its folder) and then add the information we took to it. Insert\_Product\_Buyers function that is inside dbconfig.php will get information for a specific product using the index that we got from a specific row from coffeeproducts table such as product name and price and then it will concatenate it with the specific user id. After doing this, clicking on the checkout button will redirect us to ShowChart.php.

In ShowChart.php two dbconfig files where needed, the one that I took from Account Registration page since I needed it to retrieve data from the user\_register table as well as the one that was in this folder where ShowChart.php is located. Using information from both the tables of dbconfigs, in the first part of the page I show the user information such as name username as well as the email. On the other part I show all the information about the product that he selected as well as the number of products and total price in the end using the functions Calculate\_All\_Prodcuts and Count\_All\_Products that I created on dbconfig.php page on AddToChart folder. I also added delete options in all the products that I had in this page in case a user does not want to buy something. If user presses delete, an ajax form is going to send data to a Delete.php page. This data is the CardId that is hidden on each box for products and is the id for each of the rows on the product\_ buyers table. So, the specific row that the user selected is going to get deleted and when the page will get refreshed, it will not echo it since the data is not in the database, so the box with that product will not get shown and the box with the number of products and total price is going to get updated as well. I also created a section where user can update his password. In case he enters the old password incorrectly, an information is going to get prompted saying he entered the password wrong. In case he enters it correctly, and then enters the new password the new one is going to update the old one by sending the data to Account\_Registation/UpdateUserPassword.php with the help of ajax to update the password.

Finalizing I also added a new page with the best costumers with an anchor at Contact Page. It uses JSON parse to create Customers objects and then convert the information of these objects into strings in order to show the information for each customer into different boxes.