K00186813 Colin Maher Repeat Assignment

Username = admin password = password

Username = user password = password

PDO Prepared statements are used to increase security by preventing sql injection.

Most pages check if the user is logged in otherwise they are denied access.

I ran out of time and didn’t get a chance to get Ajax live bidding or messaging to work.

Javascript functions are stored in library/jscript/

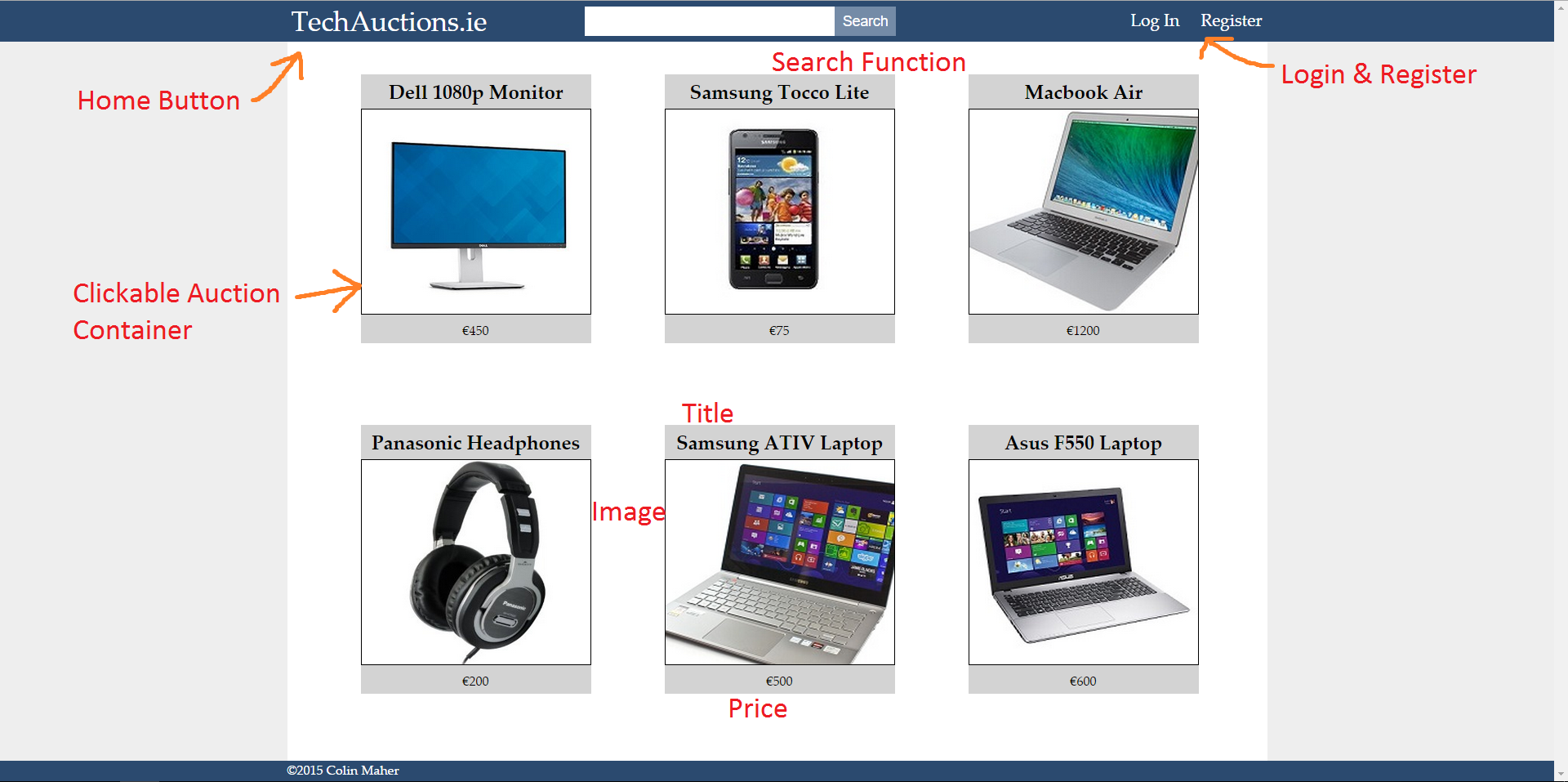
**Relational Notation**

**Users= {id, username, email, password, address1, address2, town, county, date}**

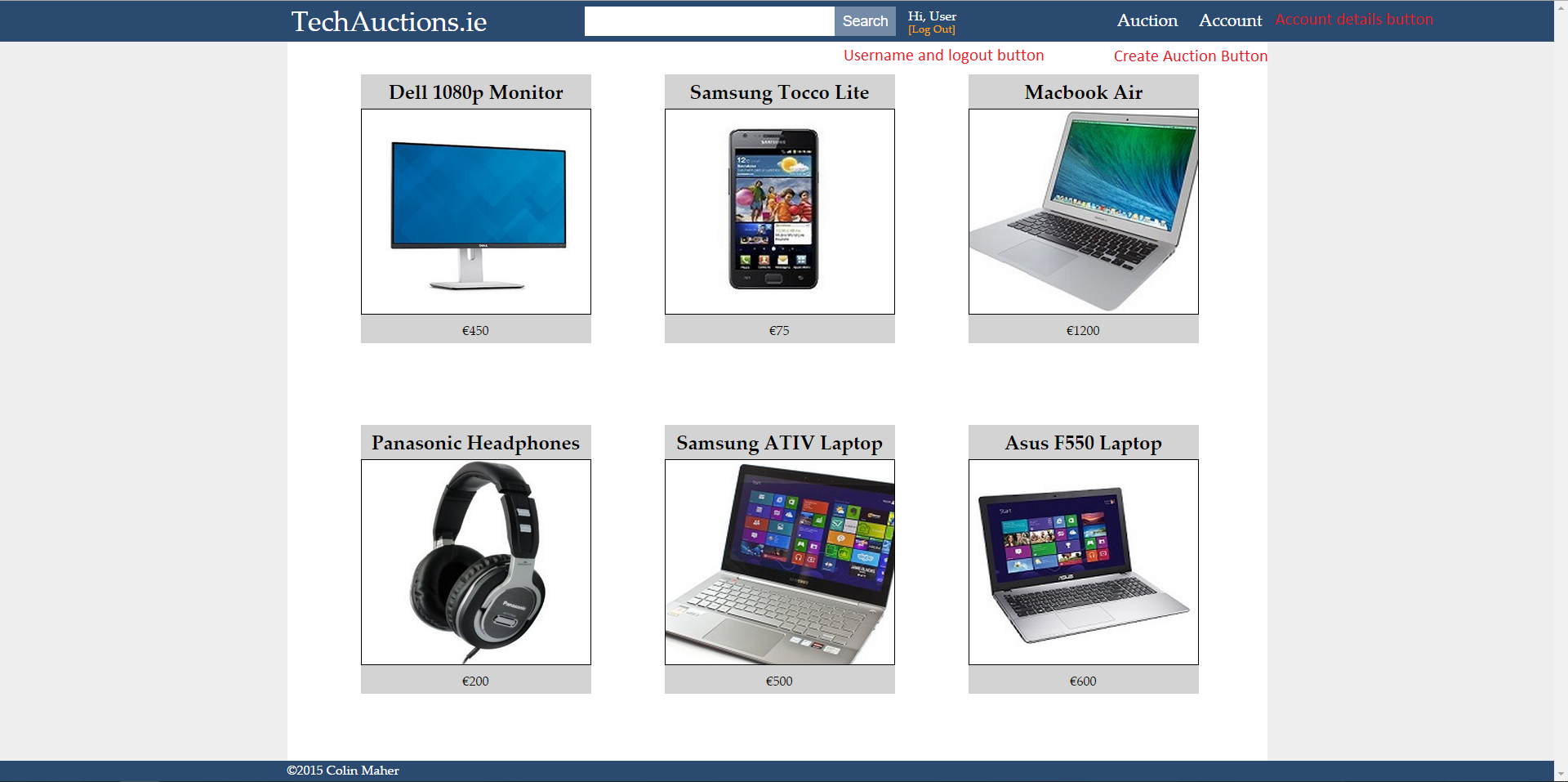
**Auctions={id, title, category, description, price, image, username}**

**Index.php**

The homepage is very straight forward.



When you log in the header changes. The code for the header is located in includes/header.php



**register.php**

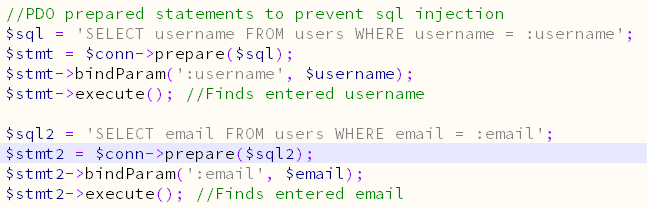
You can register here. The Javascript function validatePassword() is used to make sure password and confirm password are the same before submitting the form.

**registerScript.php**

The code in this file processes the information sent from register.php.

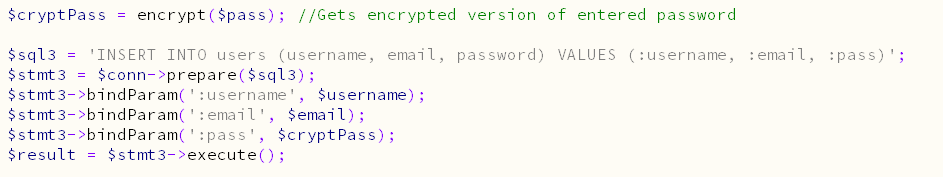
These prepared statements select rows from user table where the username or email is equal to entered username or email.

If there are any rows containing either this username or email the user is asked to enter another.



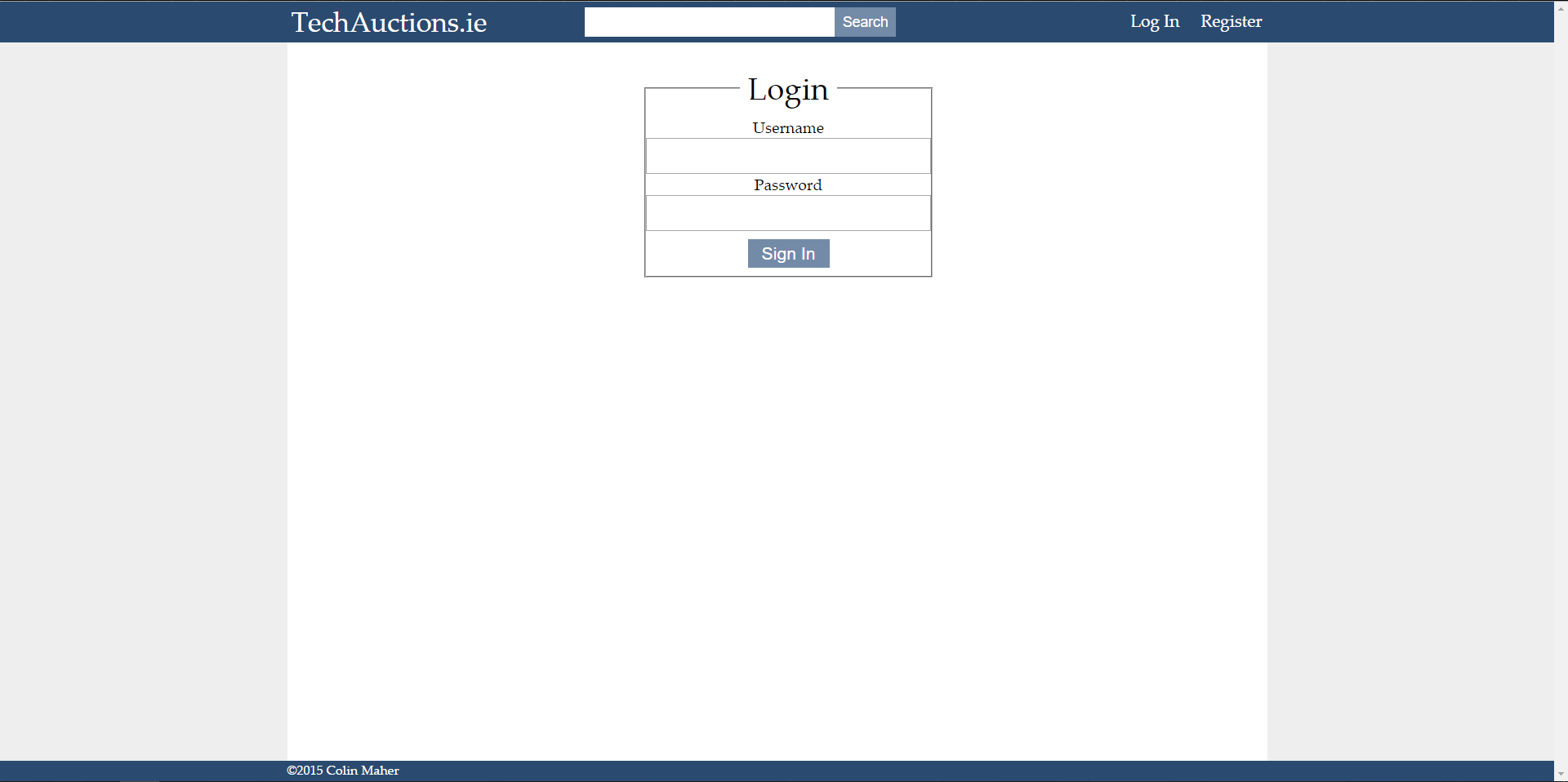
This encrypts the entered password using Salt. The the prepared statement enters the information into a new row in the users table.

The encrypted password is stored in the table so that if the database was hacked, the hackers still could not access any users accounts.



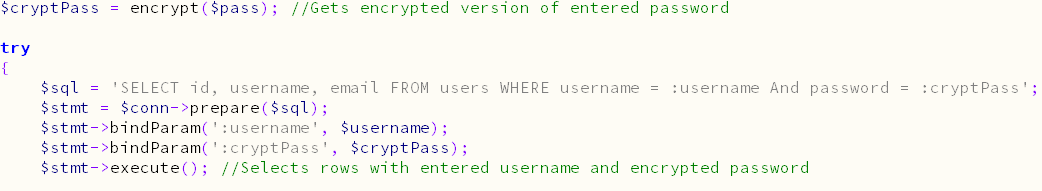
**login.php**

You can login here with your username and password.



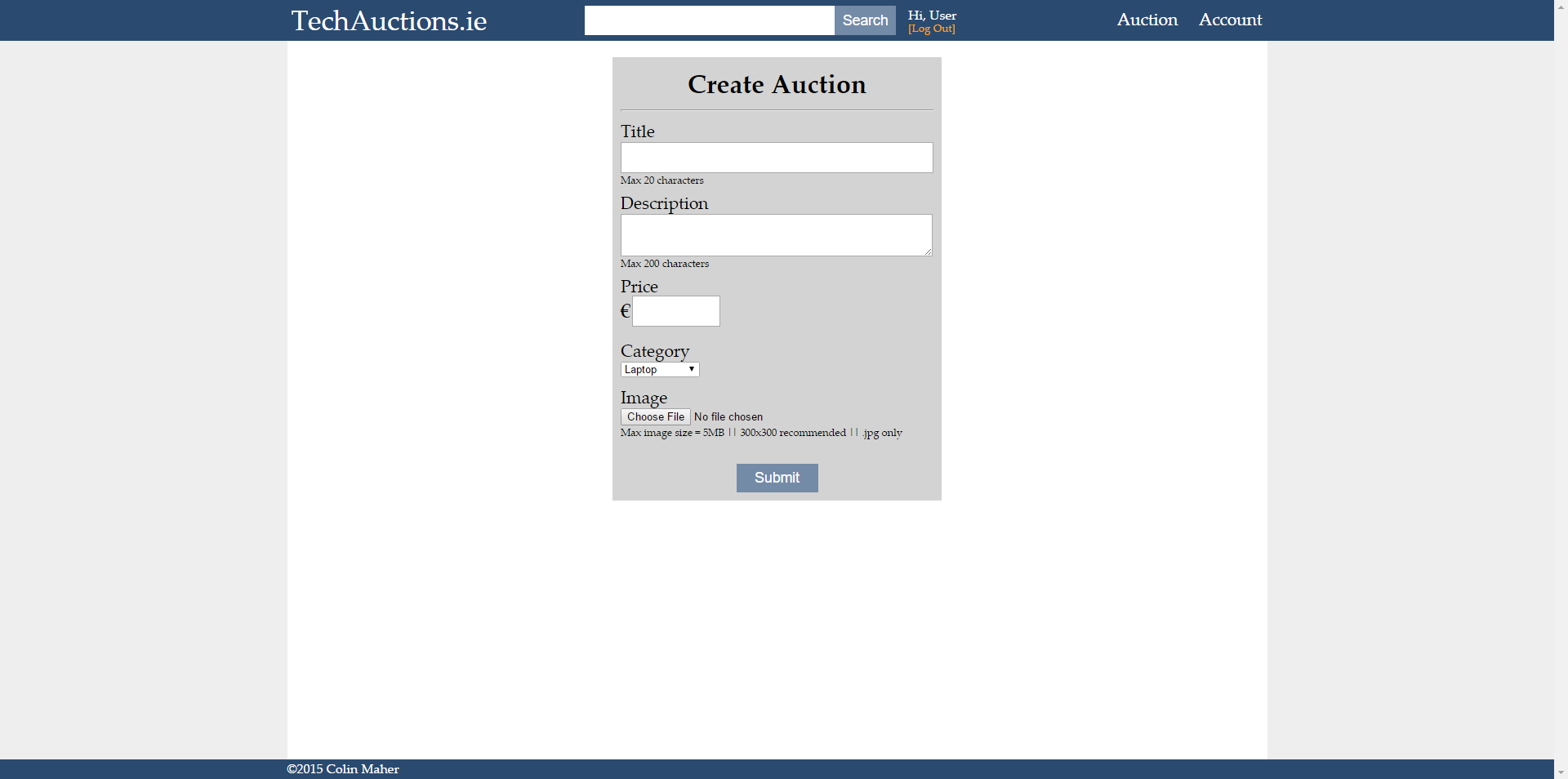
**loginScript.php**

When a user tries to login the entered password is encrypted and compared to the password in the database. This prepared statement selects id, username and email from whichever row contains both the entered username and encrypted password.



**auction.php**

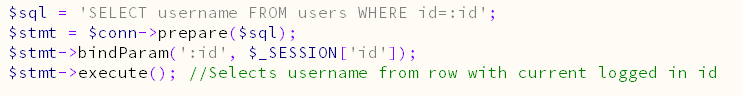
An auction can be created on the auction page.

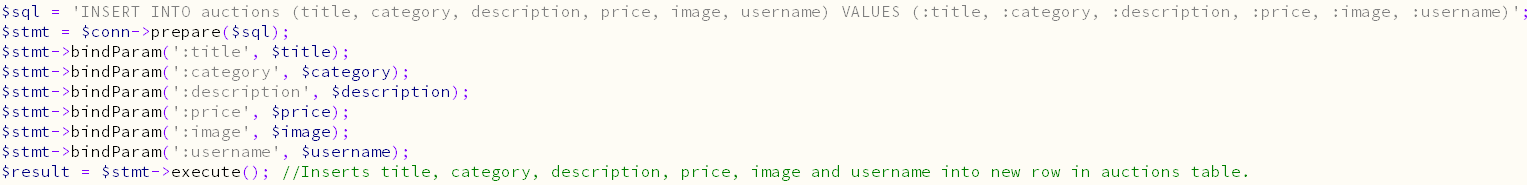


**auctionScript.php**

The first half of this file is code for evaluating the image. It sets the directory for saving, It makes sure it is under 5MB, it makes sure it’s extension is .jpg and it gives it a name including the id, title and date to prevent overwriting other images with the same name.

This prepared statement gets the the users username so the username can be added to auctions table.

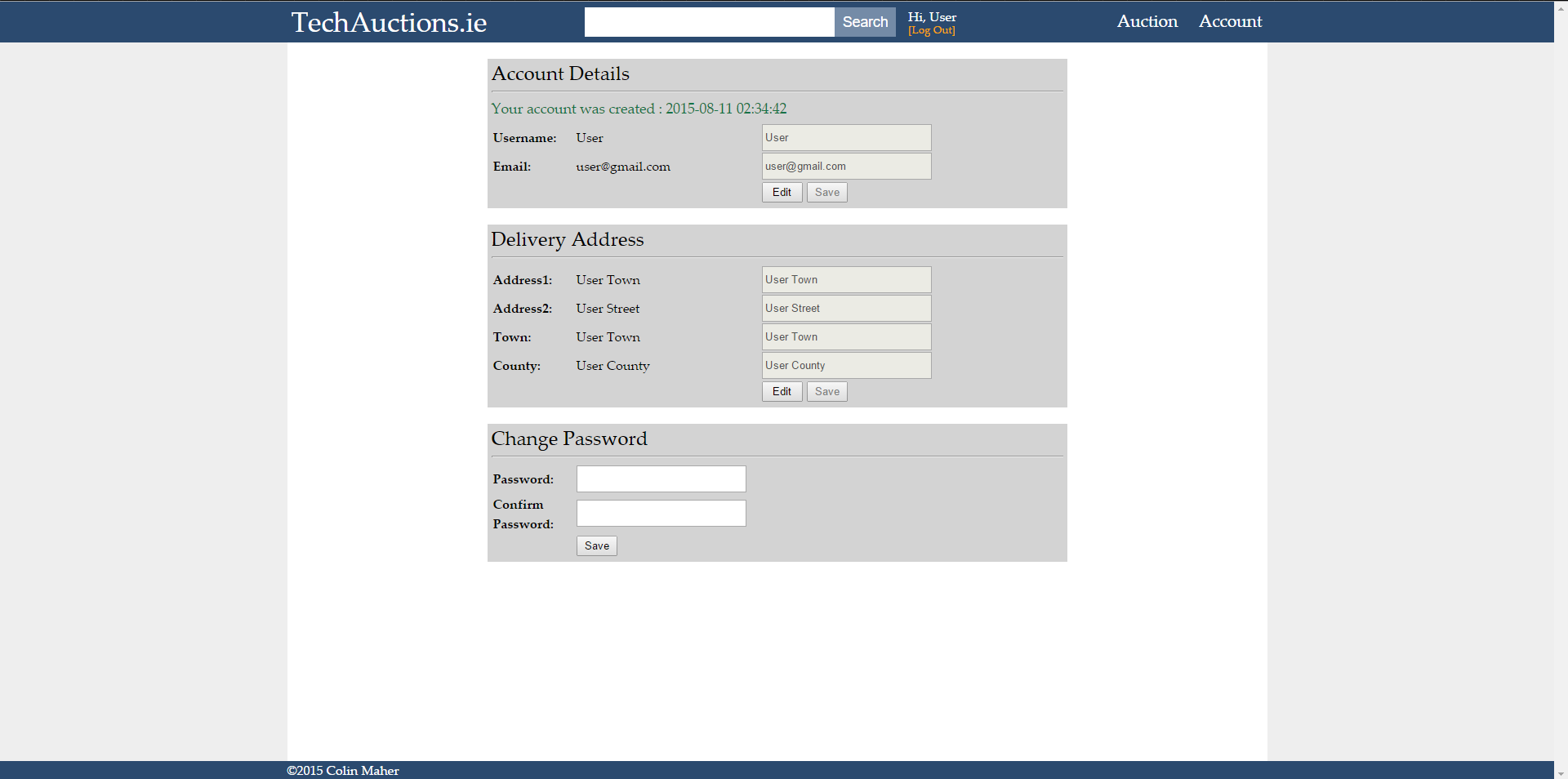
****

This prepared statement inserts all the information into the auctions table. The image url is saved so it can be concatenated into html. 

**account.php**

Your account info can be edited on the account page. Press the edit button to unlock the fields. The Javascript function validatePassword() is used to make sure password and confirm password are the same before submitting the form. This page uses while loops to display all the information from the $row arrays.

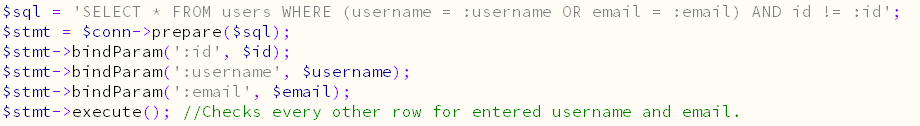
If you are logged in as admin this page changes to a control panel, where you can view all accounts.

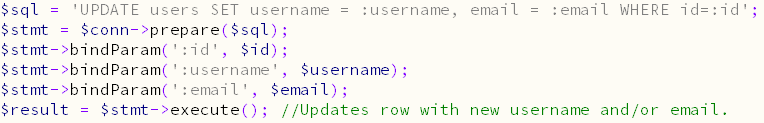


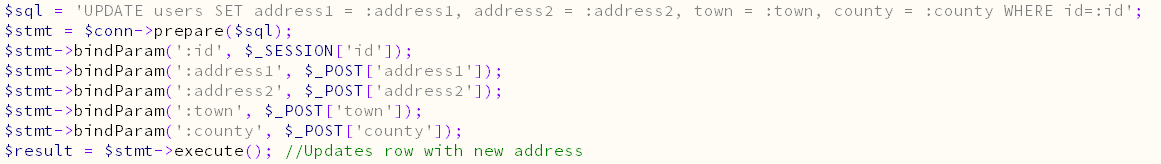
**accountScript.php**

This file processes information sent from account.php.

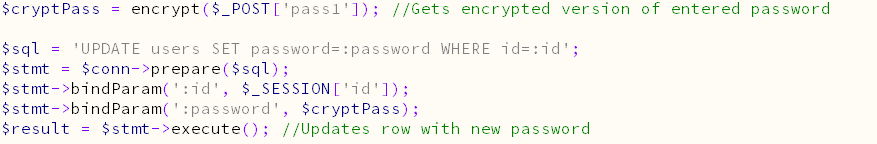
It checks which form has been posted. Either accDetails, accAddress or accPassword.

This prepared statement selects rows which contain entered username or email, except for the row of the currently logged in user. If it returns any rows then the entered username or email is in use and the user is alerted.****

This prepared statement updates the current users row with the new username and/or email. 

This prepared statement updates the current users row with the new address1, address2, town and county. 

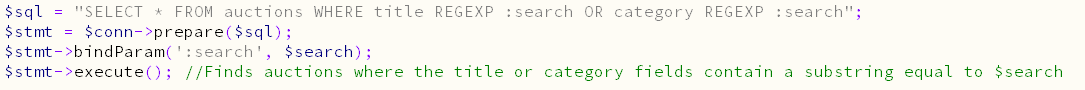
This prepared statement updates the currents users row with the new password.



**search.php**

This file contains code for searching through auctions.

This prepared statement selects rows where the entered string is a substring of either title or category. Eg Lap will return all auctions with the laptop category. Phone will return all auctions with titles of smartphone etc.



**viewAuction.php**

This page shows who created the auction, the price and the description.

**Functions.php**

This contains functions used in many different parts of the project. secureSession() has code which creates a much more secure session than just session\_start() on its own.

Encrypt() encrypts the password which is passed to it and returns the encrypted version.

Access() prevents access to pages to users who are not logged in.