

Internet Applications and Techniques – Individual Report

SAM LONGTON / Longtons

Summery

This website is optimised for use on Google Chrome on a 1920X1080 HD screen. To start with you are welcomed with the index page, this page displays the bare minimum information about each item and instructs the user to register or login in order to view more information, (login info below).

Once the user has registered or logged in the user can now see more information about each item available. The user has to enter all the information on the register page, including password validation, this makes sure the user is entering the password correct, my php makes sure the passwords match before allowing the user to login. The user can then select an item and enter proof that the item is theirs and click claim.

Once the item has been claimed the item is still available for other users to claim, the request is now visible in the Requests page. Once the admin approves the request the item is not available for other users to claim. If the admin chooses to decline the request, the item status will change to 'rejected'

The admin user can login and add a new item which will automatically be visible on the home page, any request that is made will be visible on the requests page, the admin can then either approve or decline a request. At this point the status will change. When the admin approves the request, the claimed field is changed to Y meaning the item is no longer available and can not be viewed on the home page any longer.

Website link

<https://sjlongton.co.uk/>

Users

<u>Email</u>	<u>Password</u>	<u>Usertype</u>
ADMIN@ADMIN.COM	ADMIN	<u>ADMIN</u>
USER@USER.CO.UK	USER	<u>USER</u>

Extra information

Throughout my website I have bene using sessions, this is to prevent access without users being logged in whilst also not allowing a normal user to access the admin page, if the user tries to access the admin page the user is greeting with the statement 'You do not have permission to view this page' when the admin user logs in it automatically heads to the admin page in order for the admin to add a new item, using sessions the admin user can view all the requests being made for every item made by every user.

All fields are required, and users are unable to register/ login and claim an item if the fields are not populated.

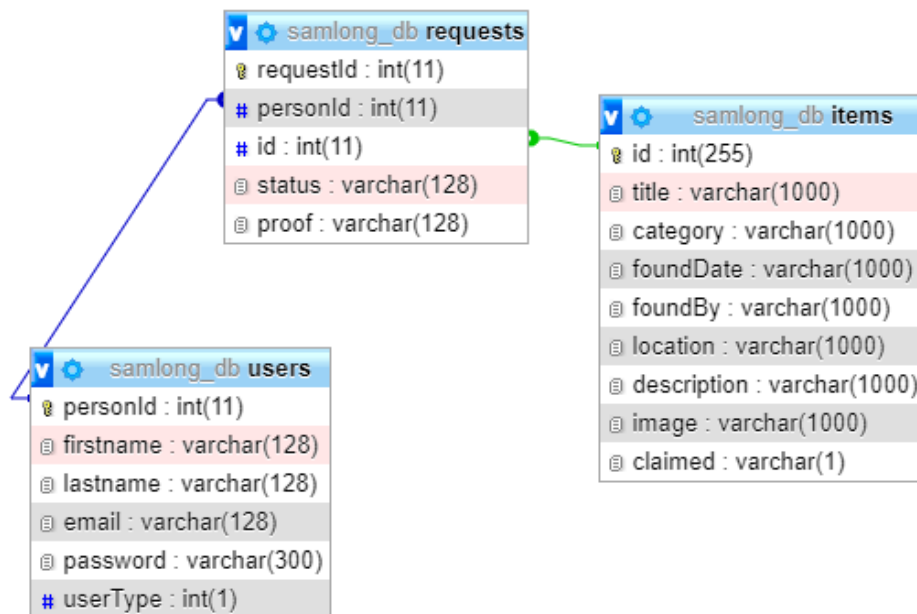
The password fields are hashed, meaning the passwords are not displayed in the database and have to be decrypted in order for the user to login, this is a important level of security. The password is displayed as a random sting value

'\$2y\$10\$L9xrUKUz1gkY9t5Np7AlaOhSlpGgC/zA6xWfV7jSP1sIrtKVAKUDK' and is then dehashed in order for the user to sign in this happens by using the 'hashedPwdCheck = password verify(\$password, \$row['password']);' statement which checks the password entered with the

Internet Applications and Techniques – Individual Report

password in the database and decrypts the password, if the password is correct the user is logged in else the user is told that the password is wrong.

Throughout my website I have required the database connection, meaning if the database cannot be connected to the website will not work.



Above is the relational diagram for my database, this shows that the requests page includes foreign keys from other tables in order to link them, my database consists of NOT NULL values for all fields meaning all the fields need to be populated and the database won't allow NULL values, the database also includes auto-incrementing values for all the IDs (personId, requestId, id) meaning every time a new item, request or person is added the value will always be the next number. I have set primary keys, which are all the IDs for each table. Within my SQL statements I have created joins in order to bring back information from 2 tables and display them together correctly for example line 70 on the requests.php page. (`$sql = "UPDATE items, requests SET items.claimed = 'Y', requests.status = 'Approved' WHERE items.id = $id AND requests.id = $id"`) this SQL statement is simply joining the items and requests table and updating both tables with one statement.

I decided to link my website to my own CSS page which I created in order to make the website a bit more appealing to the eye and easier to view and use, I also linked the website to Bootstrap again to help with the website being appealing to the eye and easy to use.

Whilst I have submitted this I have decided to continue to develop this in order to solidify my PHP, HTML and CSS knowledge such as developing an image upload function, I have started with this and my progress is visible within the admin page.

Internet Applications and Techniques – Individual Report
