

# Web Development Software Report

Rory McManus - 40132363

## Database Data

The data in my database was created based on the two key areas of my craft beer website: products & account information.

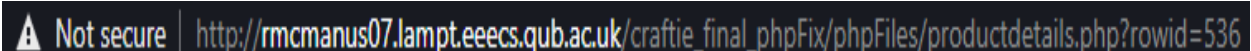
### Products -

The product data, which comes mainly from the craft beers dataset found on Kaggle.com. This included craft beer names, ABV%, beer styles and ratings relating to overall appeal, taste, appearance and aroma. The dataset also contains a description of each beer as well as information regarding the breweries each beer is produced by. I decided that for this assignment the inclusion of the breweries section was not necessary as it ultimately took away from the focus of the website which was the beers themselves. The images for each individual beer were taken from [www.beer52.com](http://www.beer52.com); a subscription-based craft beer e-commerce site which acted as a main inspiration for the design and functionality of the web application.

The full list of products can be found on the “Products” page. Here we see each product’s image, name, ABV%, beer style and its overall rating. Each “card” that displays this image is a link to the “Product Details” page which includes each individual items “row\_id” in the URL.

```
<a href='productdetails.php?rowid=$iddata'>
```

This is then received and displayed on the “Product Details” page using the “\$\_GET” superglobal within php.



This is the case for all instances of the website where the products are displayed on screen for the user. E.g. The “WishList” page & the “Cart” page.

### Accounts -

The account information, such as user information, account details and credentials and card payment details, all come from input from the user as they navigate through the web application.

All user account information was input into the database using the “\$\_POST” superglobal within php. Each piece of posted data was then complimented using the SQL library function “mysql\_real\_escape\_string”. This helps prevent SQL injection attacks which are often performed by using the ' character to append malicious code to an SQL query.

```

if (isset($_POST['carddetails'])) {
    $user_id = $_SESSION["userId"];
    $nameOnCard = mysqli_real_escape_string($conn, $_POST["nameOnCard"]);
    $cardtype = mysqli_real_escape_string($conn, $_POST["cardtype"]);
    $accountNum = mysqli_real_escape_string($conn, $_POST["accountNum"]);
    $CVC = mysqli_real_escape_string($conn, $_POST["CVC"]);
    $expiry = mysqli_real_escape_string($conn, $_POST["expiry"]);
    $expiry = date('Y-m-d', strtotime($expiry));
}

```

In addition to using the escape string function, various prepared statements were used as a means of implementing additional security measures against SQL Injections. This also had the effect of reducing parsing time as the preparation on the query is done only once, minimizing bandwidth to the server as only the parameters are sent each time.

```

function addCarddetails($conn, $user_id, $nameOnCard, $cardtype, $accountNum, $CVC, $expiry, $isMember)
{
    $hashedaccountNum = password_hash($accountNum, PASSWORD_DEFAULT);
    $hashedCVC = password_hash($CVC, PASSWORD_DEFAULT);
    $sql = "INSERT INTO payments (user_id, nameOnCard, cardtype, accountNum, CVC, expiry) VALUES (?, ?, ?, ?, ?, ?)";
    $stmt = mysqli_stmt_init($conn);
    if (!mysqli_stmt_prepare($stmt, $sql)) {
        header("location: ../account.php?error=stmtfail2");
        exit();
    }
    mysqli_stmt_bind_param($stmt, "ssssss", $user_id, $nameOnCard, $cardtype, $hashedaccountNum, $hashedCVC, $expiry);
    mysqli_stmt_execute($stmt);
    mysqli_stmt_close($stmt);
}

```

This particular method of using prepared statements was taken from Youtuber Dani Krossing's "How To Create A Login System In PHP For Beginners | Procedural MySQLi | PHP Tutorial"(Oct,11,2020) and was the applied to all the functions of the website where posting data was needed.

### Summary of DB Design

As stated, the database design is based on the idea that the website consists of two main areas: products & account information.

### Products -

For products, the key table is the "all beers" table which contains all the information relating to the products themselves. This table contains the most foreign key relationships in the entire database. In order to avoid repetition of data, various columns of the "allbeers" table have been normalised such as:

### Ratings -

Each of the different rating categories have are linked to the ratings table which contains html code relating to a "star" icon found on <https://fontawesome.com/icons/star>. When calling this table through a SQL query which joins the "allbeers" table and the "ratings" table, a series of stars is printed to screen consistent with the rating level. E.g., a rating of 1 will produce 1 star when called.

| appearance | aroma | images  |
|------------|-------|---|
| 5          | 3     | <a href="https://dth50iqs19w2y.cloudfront.net/">https://dth50iqs19w2y.cloudfront.net/</a> |
| 2          |       | <div class='rating1'>   |
| 4          |       | <div class='fa fa-star'>  |
| 3          |       | <div class='fa fa-star'>  |
| 1          |       | <div class='fa fa-star'>  |
| 2          |       | </div>  |



### Beer Style -

This website contains beers that vary across 16 different beer styles. Therefore, to prevent repetition of data across the 1700+ products, the beer style column was normalised. This, again, allows for the appropriate beer style to be called using a join SQL query and the “allbeers” table rows to remain as unique as possible.

| beerName      | beerstyle | beerABV | style_id | style      |
|---------------|-----------|---------|----------|------------|
| Lower De Boom | 1         | 7.40    |          |            |
|               |           |         | 1        | Hefeweizen |

Images -

The images column of the “allbeers” table was not normalised as in an authentic real-world e-commerce website each product’s image would be genuinely unique to that product.

ABV -

The ABV column was not normalised as this prevents confusion for an admin adding products without access to the database.

Crates & Wishlist -

The “item\_id” columns in both the “crates” and “wishlist” tables were linked to the “row\_id” of the “allbeers” table to ensure that the items chosen by the user is reflected appropriately on both pages of the web application.

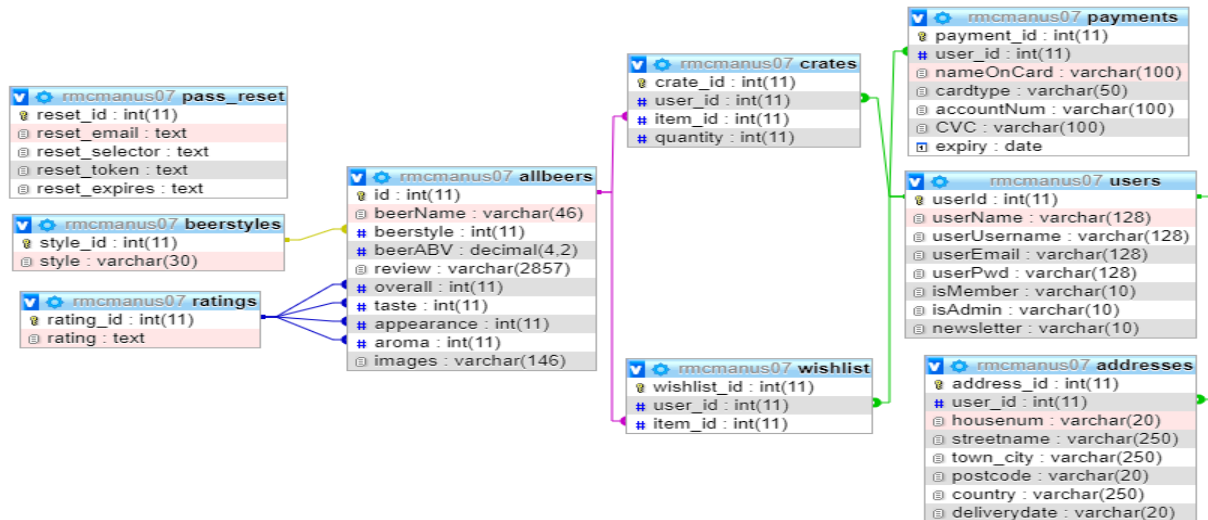
| crate_id | user_id | item_id | quantity |
|----------|---------|---------|----------|
| 9        | 25      | 4       | 12       |

Accounts -

Regarding the accounts section of the database, the main table of focus is the “users” table. This contains all the initial information a user provides when setting up a tier 1 account. Several of the accounts-related tables are linked to “user\_id” column in this table such as the : payments table , addresses table , crates table and wishlist table. This has the effect of ensuring that changes made to a users account are registered only to that account. As shown below:

| payment_id | user_id | nameOnCard   | crate_id | user_id | item_id      | qu | address_id | user_id | houenum      | wishlist_id | user_id | item_id |
|------------|---------|--------------|----------|---------|--------------|----|------------|---------|--------------|-------------|---------|---------|
| 6          | 25      | r k mcmanus  | 8        | 25      | 3            |    | 1          | 25      | 4            | 31          | 36      | 1664    |
| 7          | 33      | Rory McManus | 9        | 25      | Rory McManus |    | 2          | 38      | Rory McManus |             |         | brendan |

The full designer view of the entity relationships in my database is shown below:



## URL(s) to REST Application Programmable Interface (API) -

### Products -

- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/productdetails.php?rowid=966](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/productdetails.php?rowid=966) - product details page with product row\_id as endpoint

This is used by the following pages -

[http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/allproducts.php](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/allproducts.php)

[http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/allproductssorting.php/stylesort.php](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/allproductssorting.php/stylesort.php)

[http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/allproductssorting.php/ratingsort.php](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/allproductssorting.php/ratingsort.php)

[http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/allproductssorting.php/abvsort.php](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/allproductssorting.php/abvsort.php)

### Wish List -

- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/wishlist.php?error=addedWL](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/wishlist.php?error=addedWL) - "Wishlist" page with "added to Wishlist" message as endpoint
- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/wishlist.php?removed](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/wishlist.php?removed) - "Wishlist" page with "added to Wishlist" message as endpoint

### Error Messages -

- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/account.php?error=emptyinput](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/account.php?error=emptyinput) - "Account" page with error message for empty input as endpoint.
- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/account.php?error=passwordmismatch](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/account.php?error=passwordmismatch) - "Account" page with error message for password mismatch as endpoint.
- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/account.php?error=usernametaken](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/account.php?error=usernametaken) - "Account" page with error message for username taken as endpoint.
- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/account.php?error=incorrectpassword](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/account.php?error=incorrectpassword) - "Account" page with error message for incorrect password as endpoint.
- [http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/adminloginpage.php?error=nonadmin](http://rmcmamus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/adminloginpage.php?error=nonadmin) - "Adminlogin" page with error message for non-admin login attempt as endpoint.

Password-reset -

- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/passwordreset.php?emailsent](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/passwordreset.php?emailsent) - "passwordreset" page with message for reset email being sent attempt as endpoint.
- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/createnewpass.php?selector=4c5ee899f9da6739&validator=36da0f503dd33386a56cbc316ae101991f9681a4f3b5bccdd58d582752574a0b6](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/createnewpass.php?selector=4c5ee899f9da6739&validator=36da0f503dd33386a56cbc316ae101991f9681a4f3b5bccdd58d582752574a0b6) - "createnewpass" page with validator and selector tokens as endpoint (these have been set to expire after 15minutes after being generated).

Account -

- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/account.php?detailsadded](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/account.php?detailsadded) - "Account" page with message for successful registration of card payment and delivery address details as endpoint.
- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/index.php?memRemoved](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/index.php?memRemoved) - "index" page with message for success removal of membership status as endpoint.
- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/index.php?NLRemoved](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/index.php?NLRemoved) - "index" page with message for success removal of newsletter subscription as endpoint.
- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/newsletter.php?error=noneNLS](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/newsletter.php?error=noneNLS) - "index" page with message for success of newsletter subscription as endpoint.
- [http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/index.php?accountdeleted](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/index.php?accountdeleted) - "index" page with message for success of account deletion as endpoint.

## Authentication of credentials

In order to ensure sensitive data such as users password and card details are secure, these inputs were hashed using the "password\_hash" function which uses the "PASSWORD\_DEFAULT" hashing algorithm in php which stores sensitive data as a long sequence of random characters in the database.

| accountNum   | CVC   |
|--|---|
| \$2y\$10\$4/66Vrm7J4We2H6CCsmxpe7cmKvGPvCr./XHVChAW3k... | \$2y\$10\$NHOxsO4dSO3PjMjLtAwtO3kub/h4hYxkZpyIEDI2Q5... |

In regards to passwords, the "password\_verify" function was used to compare the password input by the user and the hashed password saved to the database.

```
$hashedpass = password_hash($password, PASSWORD_DEFAULT);  
$checkpwd = password_verify($password, $passwordHashed);
```

Admin Login -

An admin login has been added to allow for creating, updating and removal of products from the page without access to the database -

[http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie\\_final\\_phpFix/phpFiles/adminloginpage.php](http://rmcmanus07.lampt.eeecs.qub.ac.uk/craftie_final_phpFix/phpFiles/adminloginpage.php)

Login - rorym272      Password – 1234

## **Appendix**

### **References -**

#### **Login System -**

Dani Krossing's "How To Create A Login System In PHP For Beginners | Procedural MySQLi | PHP Tutorial"(Oct 11,2020) - <https://www.youtube.com/watch?v=gCo6JqGMi30>

#### **Password reset system -**

Dani Krossing's " How To Create A Forgotten Password System In PHP | Password Recovery By Email In PHP | PHP Tutorial"(Nov 28, 2018) <https://www.youtube.com/watch?v=wUkKCMEYj9M>

#### **Pagination System -**

Quickmaw's "Create Pagination in PHP and MySQL" (Nov 21, 2016)  
<https://www.youtube.com/watch?v=gdEpUPMh63s>

#### **PHP Mailer System -**

HowCode's "Send Emails with PHP & Gmail"(Mar 5, 2017)  
<https://www.youtube.com/watch?v=U13smZvdArI>

### **Images -**

Product images - <https://www.beer52.com/>

Craftie Logos - <https://www.freelogodesign.org/>

Staff/Testimonial images - <https://unsplash.com/>