

## Documentation of Hammer Boomerang Market Place

### 1. The Sql Queries you need to test your Database

#### a. Show All Hammer Results

```
SELECT * FROM hammer ORDER BY seeking ASC;
```

#### b. Show all User Results

```
SELECT * FROM user ORDER by id ASC;
```

#### c. Show all offer which is create by user correspond to hammers

```
SELECT o.*,h.*,u.* FROM offer as o INNER JOIN hammer as h ON o.hammer_id=h.id INNER JOIN user as u ON u.id=o.user_id ORDER BY o.submitted DESC;
```

#### d. Show all message sent by users to each other

```
SELECT m.*,from_user.email as from_email_id,from_user.username as from_username,to_user.email as to_email,to_user.username as to_username FROM message as m INNER JOIN user as from_user ON from_user.id=m.from_user_id INNER JOIN user as to_user ON to_user.id=m.to_user_id ORDER BY m.sent DESC;
```

#### e. Show their Recent hammer which is viewed by Users

```
SELECT r.*,h.*,u.* FROM recentlyview as r INNER JOIN hammer as h ON h.id=r.hammer_id INNER JOIN user as u ON u.id=r.user_id ORDER BY r.created_dt DESC
```

## 2. List of parts of the assignment you have completed or not completed.

Sri	Feature	Criteria	Status
1	Initial Data	Requirements satisfied	Done
2	Creating Database	Table structure, data types, field lengths, initial data entry	Done
3	User accounts	Account sign-up	Done
4	User accounts	Validation that password meets complexity requirements (at least 5 characters)	Done
5	User accounts	Log in and Log out	Done
6	Viewing items	List of all items	Done
7	Viewing items	Unavailable and accepted offers indicated	Done
8	Making offers	Users can make an offer on an item	Done
9	Making offers	Offers only on available item	Done
10	Making offers	Users can review offers made on their items	Done
11	Messages	Send a message to a user's	Done
12	Messages	List messages from other users	Done
13	Aggregate data	Total items on the site	Done
14	Aggregate data	Top three most-wanted items	Done
15	Bonus optional task - Moderation	Meets specification (user report, moderator list, reputation system) (no partial marks)	Not Done
16	Documentation	Initial data and test queries	Done

### 3. Anything interesting or cool you'd like to draw your marker's attention to

#### a. ReceiveInbox.php

```
include_once 'db.php';

if(isset($_POST['submit'])){

    $offer_id = $_POST['offer_id'];
    $from_user_id = $_POST['send_by'];
    $to_user_id = $_POST['send_to'];
    $message = $_POST['message'];

    $sql = "INSERT INTO message values(null,$from_user_id,$to_user_id,$offer_id,CURRENT_TIMESTAMP(),'$message')";

    $x = mysqli_query($con,$sql);

    $sql = "UPDATE offer set message_status=1 where id=$offer_id";

    $y = mysqli_query($con,$sql);

    if($x&&$y)
    {
        header('location:sendInbox.php?id='.$offer_id.'&status=success&message='.urlencode('Message Sent Successfully'));
    }
    else
    {
        header('location:sendInbox.php?id='.$offer_id.'&status=danger&message='.urlencode('Database Problem'));
    }
}

$offer_id = $_GET['id'];
$user_id = $_SESSION['id'];
$sql = "SELECT h.*,o.user_id,o.send_to,o.offer,o.accepted,o.submitted as offer_submitted,urs.email as sent_by,urr.email as send_to,email FROM offer as o INNER JOIN hammer as h ON h.id=o.hammer_id INNER JOIN user as urs ON urs.id=o.user_id INNER JOIN user as urr ON urr.id=o.send_to WHERE o.id=$offer_id";

$offer_data= mysqli_fetch_assoc(mysqli_query($con,$sql));
```

In this part, sending message concept is great. The logic which I created for send sms was very interesting for me. Using of Global variables, Session variables and Mysql Functions makes this part interesting for me.

#### b. received.php

```
<?php
$sql = "SELECT o.id,o.offer,o.accepted,o.submitted,o.message_status,h.title,h.image FROM offer as o INNER JOIN hammer as h on h.id=o.hammer_id WHERE o.send_to=$id ORDER BY o.submitted DESC";
$query = mysqli_query($con,$sql);
if(mysqli_num_rows($query)>0){
    $c=0;
    while($row=mysqli_fetch_assoc($query)){
        ?>
        <tr>
            <td><?=$c?></td>
            <td><?=ucwords($row['title']) ?></td>
            <td></td>
            <td><?=$row['offer']?></td>
            <td><?php
                $status = $row['accepted'];
                if($status==1){echo "Pending";}else if($status==2){echo "Accepted";}else if ($status==3){
                    echo "Rejected";}?></td>
            <td><?=$row['submitted']?></td>

            <?php if($status==1){?>

            <td><?php
                $message_status = $row['message_status'];
                if($message_status==0){echo "waiting of Buyer Replay";}else if($message_status==1){echo "
                    waiting of your Replay";}?></td>
            <?php }else if ($status==2){?>
                <td>Offer Accepted</td>
            <?php }else if ($status==3){?>
                <td>Offer Rejected</td>
            <?php }?>

            <td><a href="receiveInbox.php?id=<?=$row['id']?>" class="btn btn-success" onclick="confirm('
                Are you sure you want to do chat with Seller ? ');" >Make Chat</a></td>

        </tr>
    <?php
```

In this part, Creating Table with various conditions like offer acceptance , current status of offer so according to that my GUI(front end) of tables also changing which was very interesting code for me.