**Functionality 5: Commission Report**

Description: An administrator will be able to access this page, which will allow them to view all the sales made within the eSaint system. Information displayed will include the USER\_ID, USERNAME, FIRST\_NAME, LAST\_NAME, EMAIL, SELLER\_RATING, and the commission made on each item sold. A total of all commission made will be displayed at the bottom of the page.

SQL View:

CREATE VIEW COMMISSION\_REPORT(SELLER\_ID, USERNAME, FIRST\_NAME, LAST\_NAME, EMAIL, SELLER\_RATING, COMMISSION) AS

SELECT I.CREATOR\_ID, U.USERNAME, U.FIRST\_NAME, U.LAST\_NAME, U.EMAIL, AVG(S.SATISFACTION), (MAX(A.MAXIMUM\_BID) \*0.2) AS COMMISSION

FROM ITEM I, USERS U, SELLER\_FEEDBACK S, AUCTIONS A

WHERE I.CREATOR\_ID = U.USER\_ID AND I.ITEM\_ID = S.ITEM\_ID AND

I.ITEM\_ID = A.ITEM\_ID

GROUP BY I.CREATOR\_ID, U.USERNAME, U.FIRST\_NAME, U.LAST\_NAME, U.EMAIL;

Output:

mysql> SELECT \* FROM COMMISSION\_REPORT;

+-----------+------------+------------+-----------+----------------------+---------------+------------+

| SELLER\_ID | USERNAME | FIRST\_NAME | LAST\_NAME | EMAIL | SELLER\_RATING | COMMISSION |

+-----------+------------+------------+-----------+----------------------+---------------+------------+

| 1 | awzurn | Andrew | Zurn | awzurn@csbsju.edu | 1 | 1200.000 |

| 2 | williamiam | William | IAm | someemail@csbsju.edu | 5 | 8.000 |

+-----------+------------+------------+-----------+----------------------+---------------+------------+

2 rows in set (0.00 sec)

**Functionality 17: Items Bid on by a User**

Description: This page will retrieve the list of all items that a user currently has a bid on, and return the ITEM\_ID, ITEM\_NAME, CATEGORY, AUCTION\_START, AUCTION\_END, and START\_PRICE for each item.

SQL Query:

SELECT A.USER\_ID, I.ITEM\_ID, I.ITEM\_NAME, I.CATEGORY, I.AUCTION\_START, I.AUCTION\_END, I.START\_PRICE

FROM ITEM I, AUCTIONS A

WHERE I.ITEM\_ID = A.ITEM\_ID AND A.USER\_ID = **userId**;

Output:

mysql> SELECT A.USER\_ID, I.ITEM\_ID, I.ITEM\_NAME, I.CATEGORY, I.AUCTION\_START, I.AUCTION\_END, I.START\_PRICE FROM ITEM I, AUCTIONS A WHERE I.ITEM\_ID = A.ITEM\_ID AND A.USER\_ID = 2;

+---------+---------+--------------------+------------------+---------------------+---------------------+-------------+

| USER\_ID | ITEM\_ID | ITEM\_NAME | CATEGORY | AUCTION\_START | AUCTION\_END | START\_PRICE |

+---------+---------+--------------------+------------------+---------------------+---------------------+-------------+

| 2 | 3 | In Troubled Waters | Non-fiction book | 2013-11-07 15:34:43 | 2013-11-09 15:32:30 | 10.00 |

| 2 | 4 | 2005 Volvo S80 | Sports Car | 2013-11-07 15:34:43 | 2013-11-09 15:32:30 | 5000.00 |

+---------+---------+--------------------+------------------+---------------------+---------------------+-------------+

2 rows in set (0.00 sec)

**Functionality 22: Find All Items Won**

Description: This page will display all the items a user has successfully won in the auction system. It will show the ITEM\_ID, SELLER\_ID, the AUCTION\_END, and the MAXIMUM\_BID of the price that won the auction.

SQL Query:

SELECT I.ITEM\_ID, I.ITEM\_NAME, I.CREATOR\_ID AS SELLER\_ID, I.AUCTION\_END, currentBid(I.ITEM\_ID) AS WINNING\_BID

FROM ITEM I, AUCTIONS A

WHERE I.ITEM\_ID = A.ITEM\_ID AND A.USER\_ID = **userId** AND I.AUCTION\_END < NOW() AND

A.MAXIMUM\_BID = (SELECT MAX(MAXIMUM\_BID)

FROM AUCTIONS A2

WHERE A2.ITEM\_ID = I.ITEM\_ID)

GROUP BY I.ITEM\_ID, I.ITEM\_NAME, I.CREATOR\_ID, I.AUCTION\_END;

Output: where **userId = 90**

+---------+----------------+-----------+---------------------+-------------+

| ITEM\_ID | ITEM\_NAME | SELLER\_ID | AUCTION\_END | WINNING\_BID |

+---------+----------------+-----------+---------------------+-------------+

| 101 | B.S. Repellant | 1 | 2012-11-10 15:32:30 | 401.00 |

+---------+----------------+-----------+---------------------+-------------+

1 row in set (0.00 sec)

**Functionality 23: Leave Seller Feedback**

Description: This page will allow a user to leave feedback on a user for an item in which they bought from them. Feedback items include the overall satisfaction with the purchase process, the quality of the item delivery, the quality of the item, and any comments a user would like to share.

SQL Insert:

INSERT INTO SELLER\_FEEDBACK(ITEM\_ID, SATISFACTION, ITEM\_DELIVERY, ITEM\_QUALITY, COMMENTS)

VALUES(**itemId**, **satisfaction**, **itemDelivery**, **itemQuality**, **comments**);

Output: No query data present for this functionality.

**Functionality 27: Retrieve Buyer Feedback**

Description: This page will retrieve and display all the feedback given to a user as a buyer. The

Information that will be displayed will include the USERNAME of user that submitted the feedback, the ITEM\_ID, SATISFACTION, PAYMENT\_PROMPTNESS, and COMMENTS.

SQL Query:

SELECT DISTINCT I.CREATOR\_ID AS RATER\_ID, U.USERNAME, I.ITEM\_ID, B.SATISFACTION, B.PAYMENT\_PROMPTNESS, B.COMMENTS

FROM ITEM I, BUYER\_FEEDBACK B, AUCTIONS A, USERS U

WHERE U.USER\_ID = userId AND A.USER\_ID = **userId** AND B.ITEM\_ID = I.ITEM\_ID AND A.ITEM\_ID = B.ITEM\_ID;

Output:

+----------+---------+--------------+--------------------+------------------------+

| RATER\_ID | ITEM\_ID | SATISFACTION | PAYMENT\_PROMPTNESS | COMMENTS |

+----------+---------+--------------+--------------------+------------------------+

| 2 | 3 | 1 | 1 | Really great buyer! A+ |

| 1 | 4 | 6 | 6 | Hate this guy |

+----------+---------+--------------+--------------------+------------------------+

2 rows in set (0.00 sec)