Result Screen Shots:

Fig.6.1.1 Home page
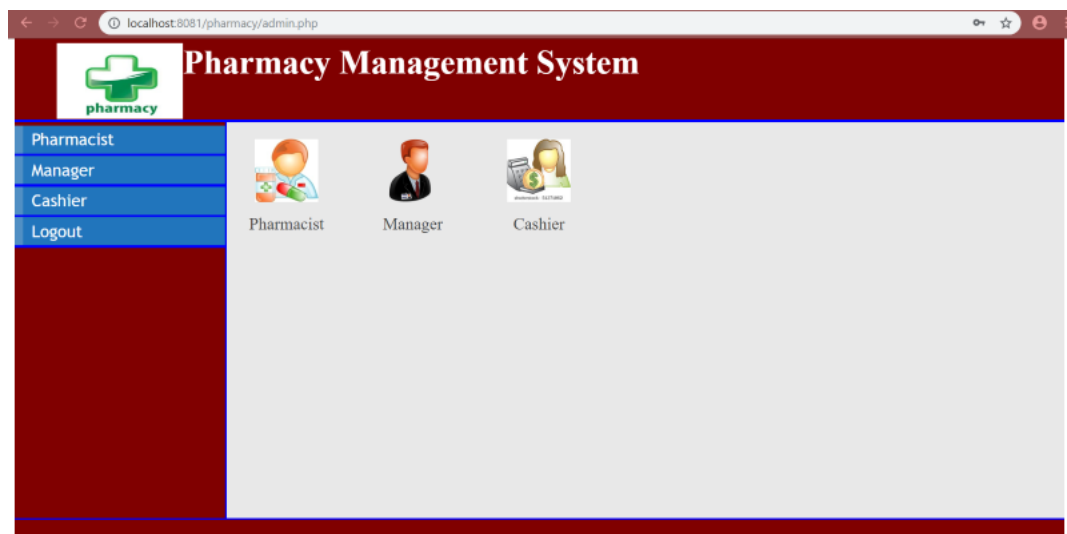

Fig.6.1.2 Admin page

Pharmacy Management System

Fig. 6.1.3 Admin managing manager details

Fig. 6.1.4 Admin view of manager details

Fig. **6.1.5 Admin managing pharmacist's details**

Fig. **6.1.6 Admin view of pharmacist's details**

Pharmacy Management System

Fig. 6.1.7 Admin managing cashier details

Fig. 6.1.8 Admin managing cashier details

Pharmacy Management System

Fig. 6.1.9 Pharmacist page
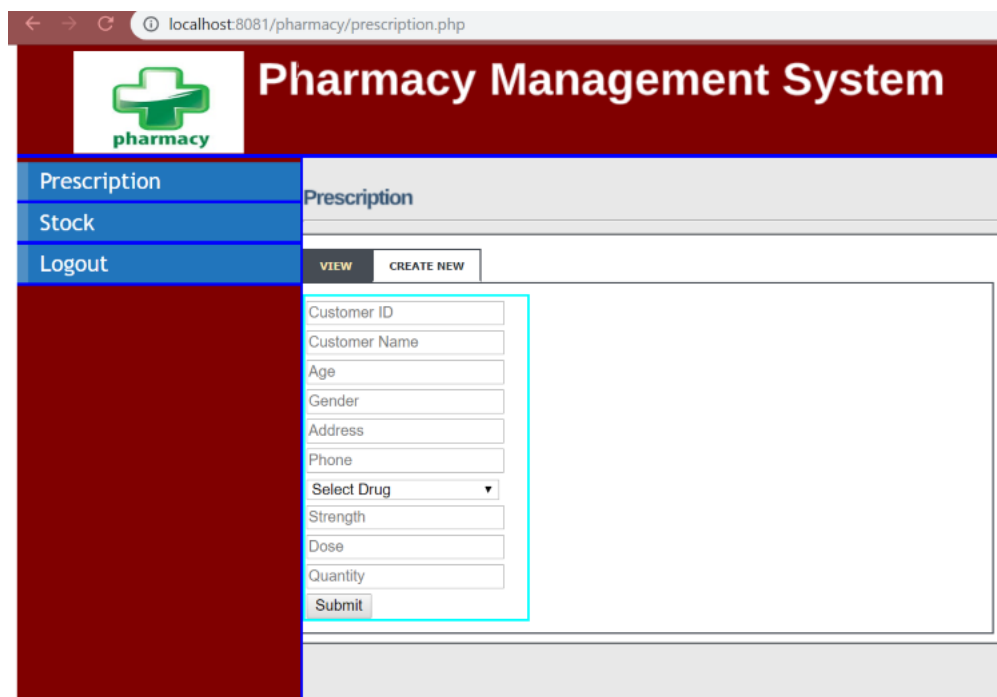
Fig. 6.2.0 Pharmacist managing prescription details
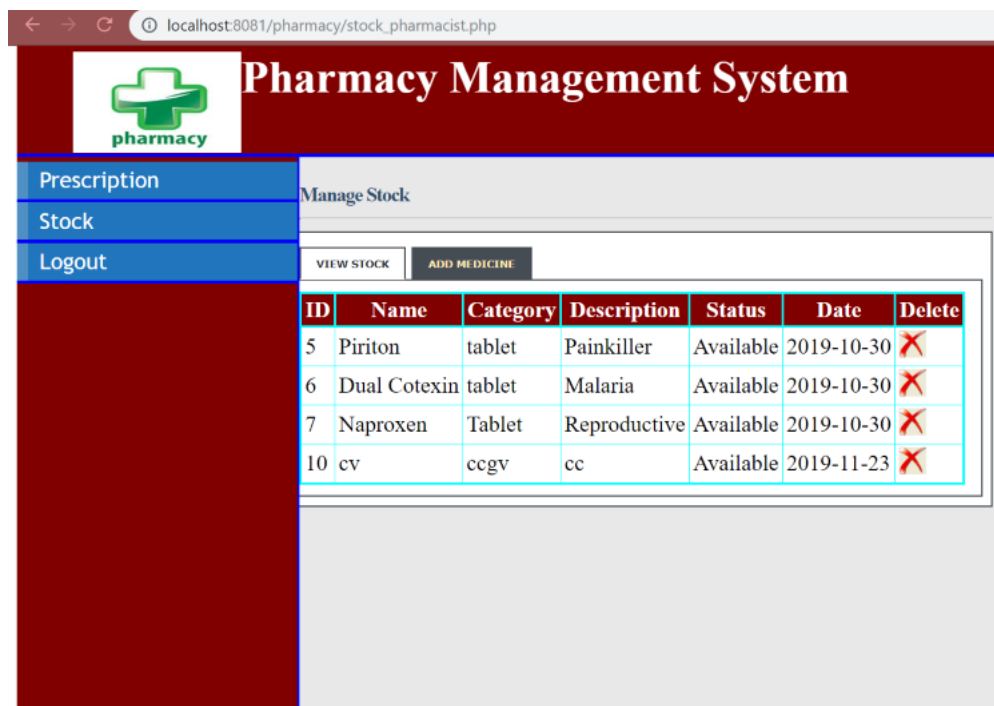
Fig. 6.2.1 Pharmacist managing stock details

Fig. 6.2.2 Pharmacist managing stock details

Pharmacy Management System

Fig. 6.2.3 Manager Page

Fig. 6.2.4 Manager managing pharmacists

Pharmacy Management System

Fig. 6.2.5 Manager managing prescriptions

Fig. 6.2.6 Manager managing stock

Pharmacy Management System

Fig. 6.2.7 Cashier page

Fig. 6.2.8 Cashier managing payment

6.4 Implementation of Triggers

In MySQL, a trigger is a stored program invoked automatically in response to an event such as insert, update and delete that occurs in the associated table. For example,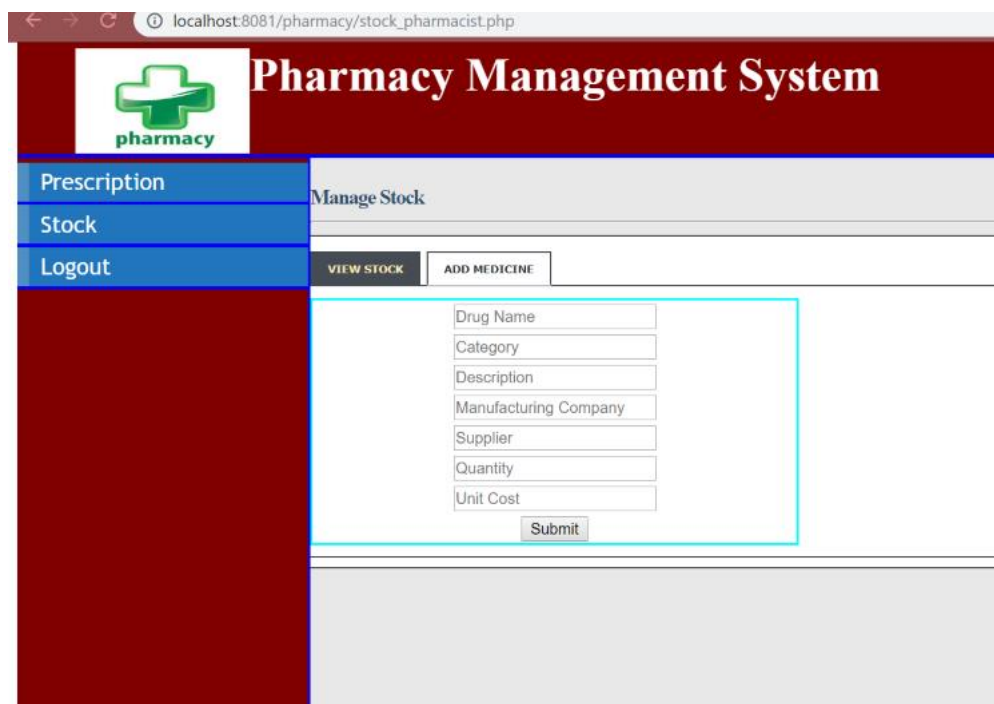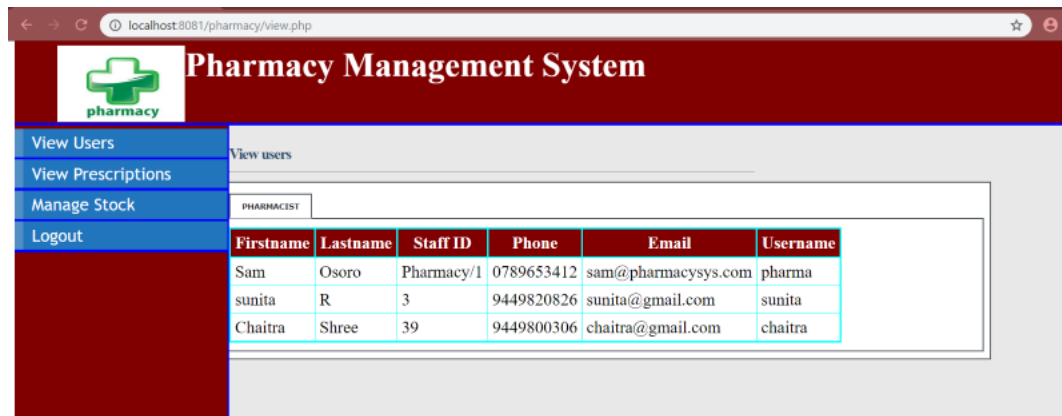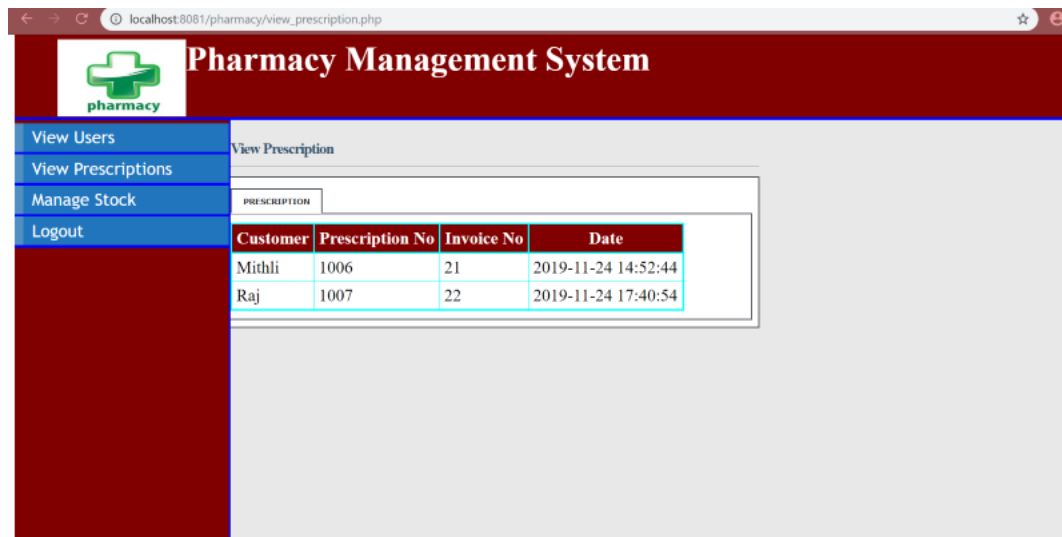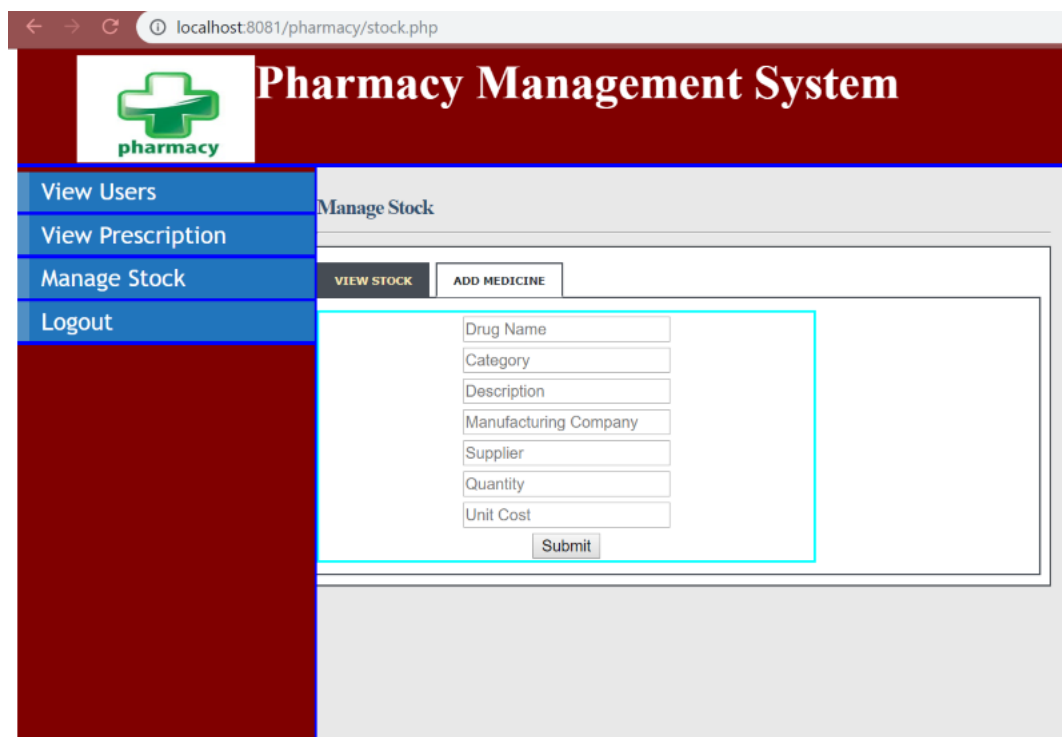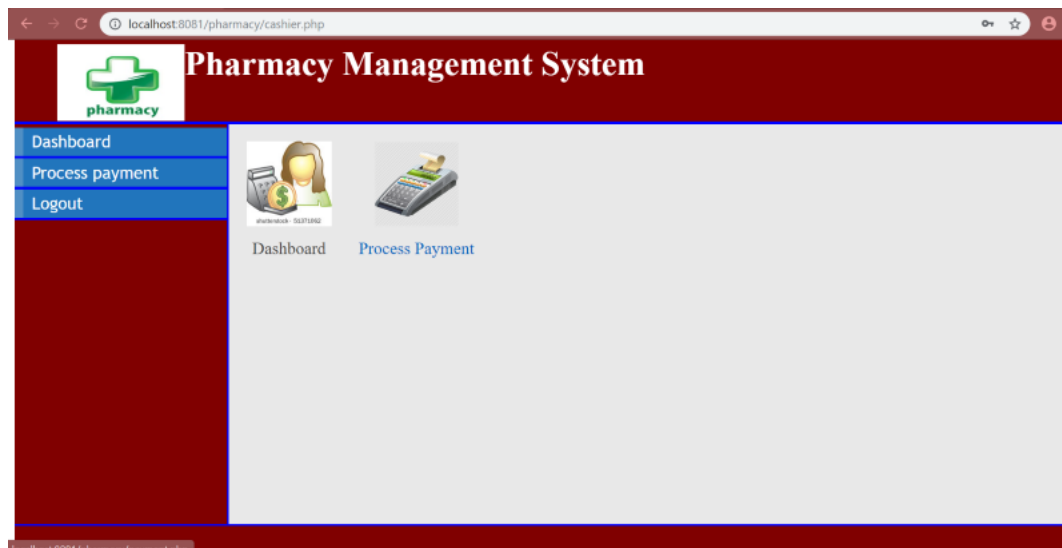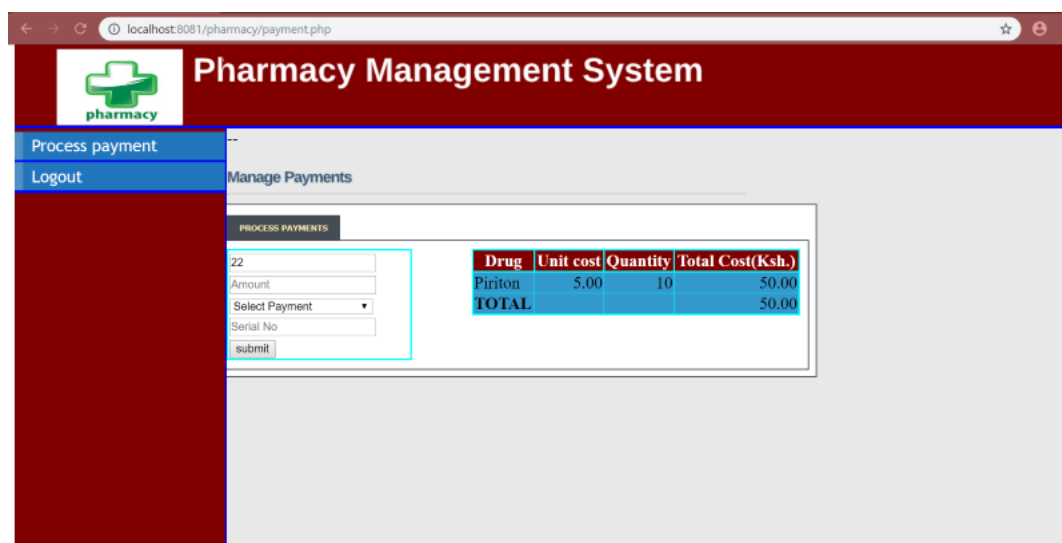 you can define a trigger that is invoked automatically before a new row is inserted into a table. MySQL supports triggers that are invoked in response to the insert, update or delete.

```
 1  CREATE TRIGGER `tr_prescription` AFTER INSERT ON
    `prescription`
 2   FOR EACH ROW BEGIN
 3  SET@date=NOW();
 4  END
 5  CREATE TRIGGER `tr_receipts` AFTER INSERT ON
    `receipts`
 6   FOR EACH ROW BEGIN
 7      SET @date=NOW();
 8  END
 9  CREATE TRIGGER `tr_invoice` AFTER INSERT ON
    `invoice`
10   FOR EACH ROW BEGIN
11      SET @date=NOW();
12  END
```
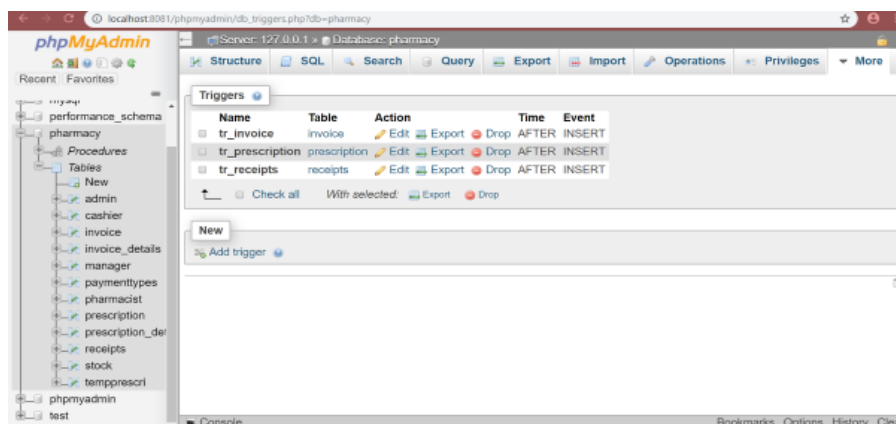

Fig. 6.4 Implementation of Triggers

Pharmacy Management System

6.5 Implementation of Stored Procedure

On giving input of pharmacist's email id as parameter for the stored procedure, we get

their details.

```
1  DELIMITER $$
2  CREATE DEFINER=`root`@`localhost` PROCEDURE
   `GetUserByEmail`(IN `gmail` VARCHAR(20))
3      NO SQL
4  BEGIN
5      SELECT * FROM pharmacist WHERE email=gmail;
6  END$$
7  DELIMITER ;
8
```
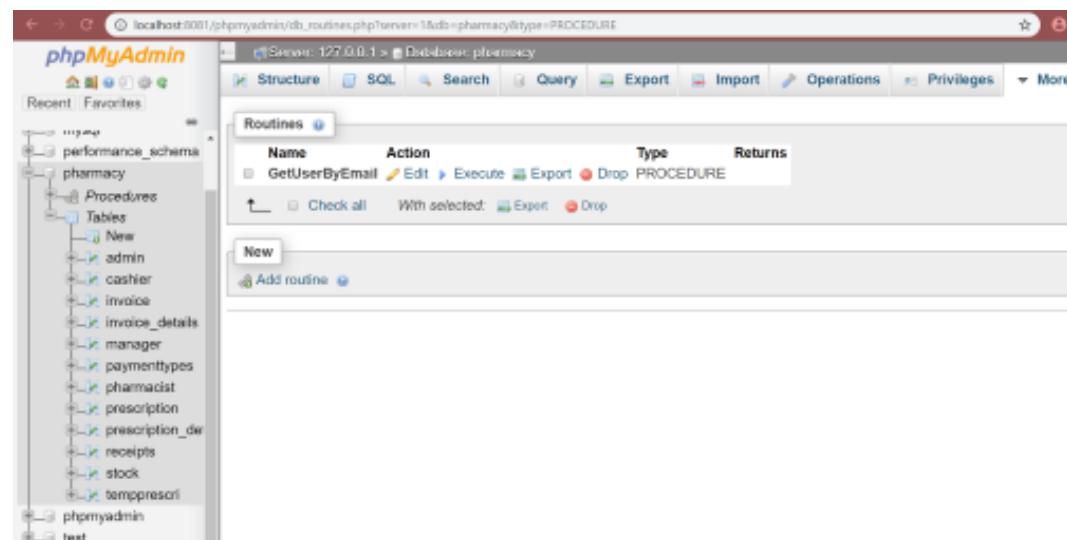

Fig. 6.5 Implementation of Stored Procedure