# Chapter-5 Testing

Testing is another phase in Software Development Life Cycle. It is done to ensure if software works as supposed to. Software testing is the process of assessing the quality of the software products and also evaluating an item in software for detecting differences between given input and expected output. The process of testing is done during the development process which is also known as verification and validation process.

There are numerous types of testing in software development such as unit testing, integration testing, system testing, black box testing, usability testing, Regression testing, Functional testing. Among them I have chosen black box testing and unit testing to ensure my system is error free and to ensure if the system works as supposed to.

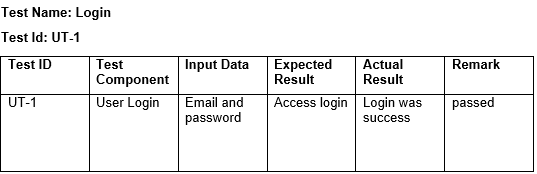
# 5.1 Unit Testing

As software is made of various units and Unit testing is the testing of an individual unit of related units. It is done to verify if it produces expected output against given input. This testing is usually performed by programmer themselves for validating whether every units of software performs as designed or not.

The reason for choosing unit testing in my project are:

* It allows the programmer to refactor the code and upgrade system libraries later and makes sure it still works finely.
* It helps to simplify the debugging process.
* It finds out software bugs at early stage which helps in resolving it without impacting the other piece of code.
* Since it identifies every defect before further implementation, it is considered to improves the quality of the code.
* It makes the code process more agile.

The unit testing of my project is given below





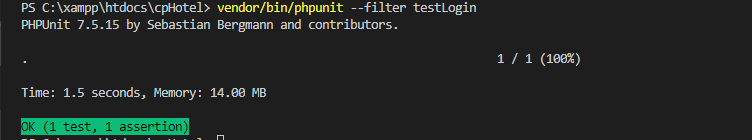
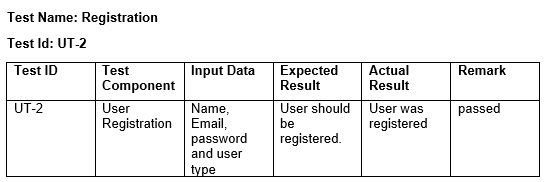


Figure : Unit testing of user Login



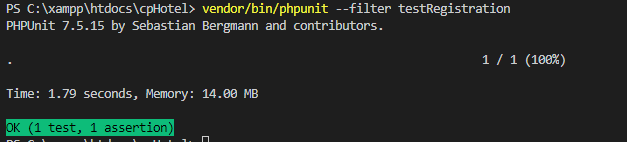
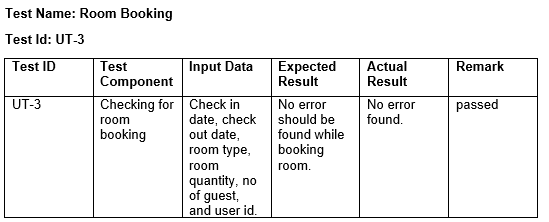


Figure : Unit Testing of Registration



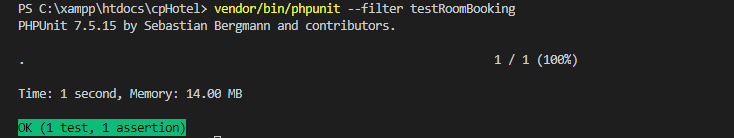
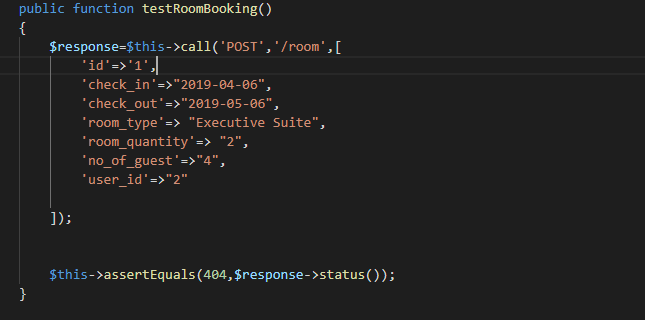
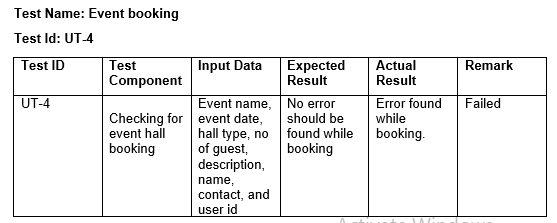


Figure :Unit Testing for Room Booking



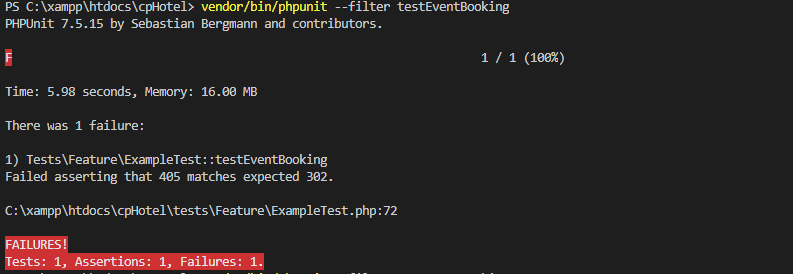
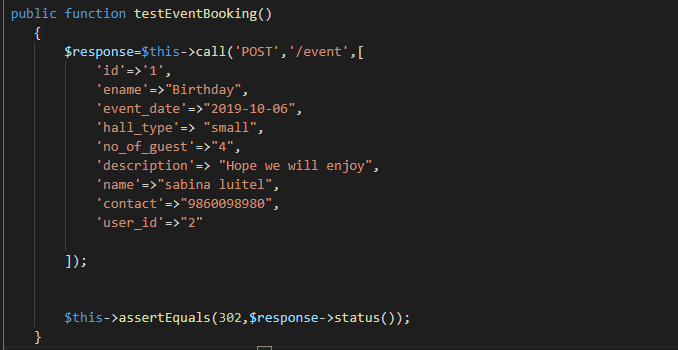


Figure :Unit Testing for Event Booking (Failed)

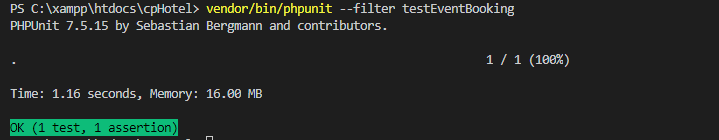
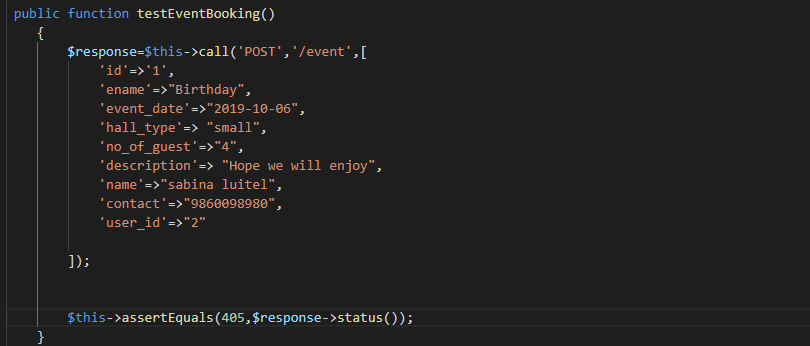
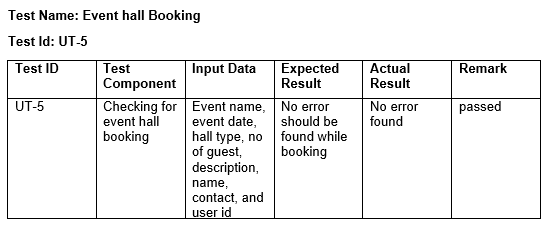
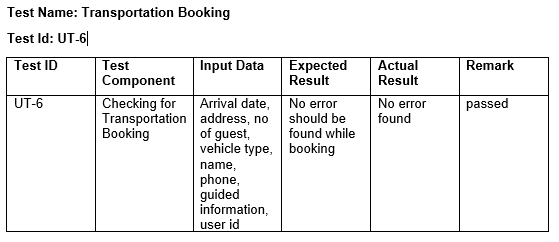


Figure : Unit Testing for Event Booking(Passed)



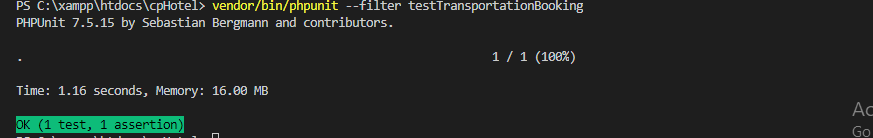
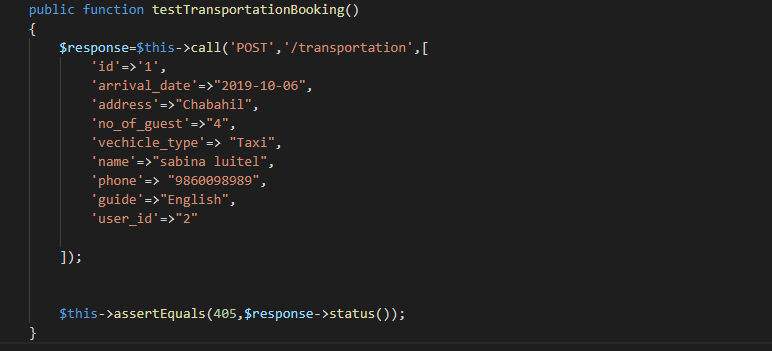
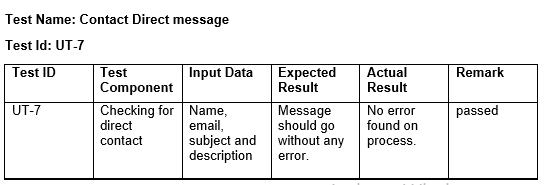


Figure : Unit Testing for Transportation Booking



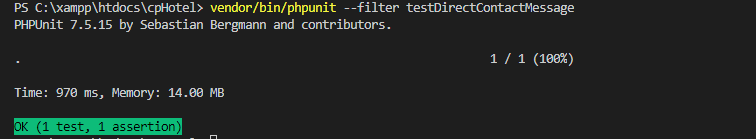
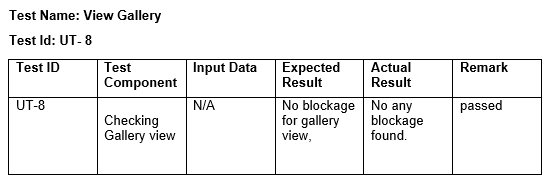


Figure :Unit Testing for Direct contact message



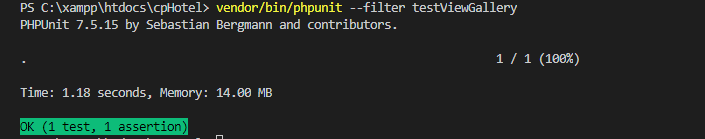
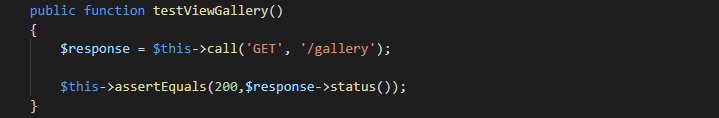
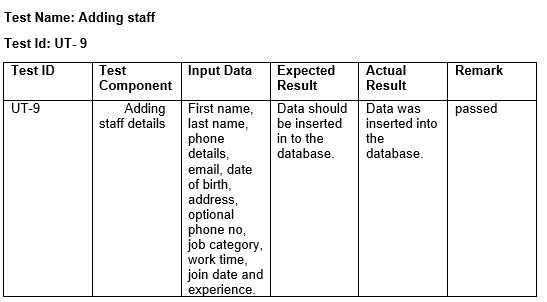


Figure :Unit Testing for gallery view



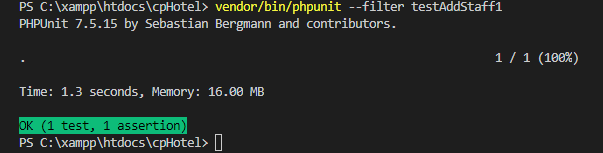
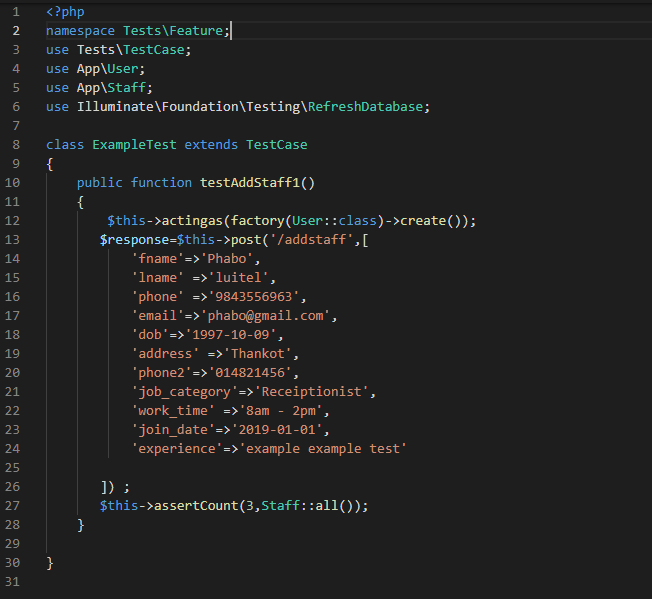
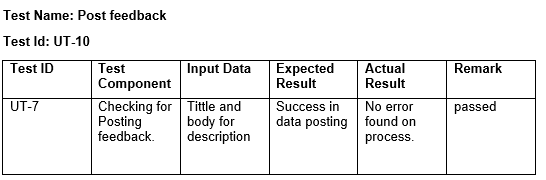
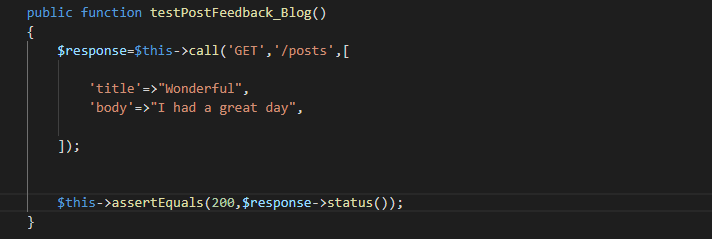


Figure :Unit Testing for Adding Staff Detail





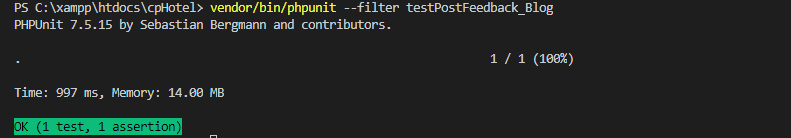


Figure :Unit Testing for Post

# 

# 5.2 Blackbox Testing

It is the software technique that focuses on the external mechanism generated against any input and execution of the system instead of internal mechanism of the system. It is also known as functional testing.

The reason for choosing black box testing are:

* It is efficient when used on large type of system.
* This testing helps to identify contradictions and vagueness in functional specifications.
* This test case can be design early after functional specification are completed.
* Testing id balanced and prejudiced since the tester and developer are independent to each other.

The black box testing of my project is given below.

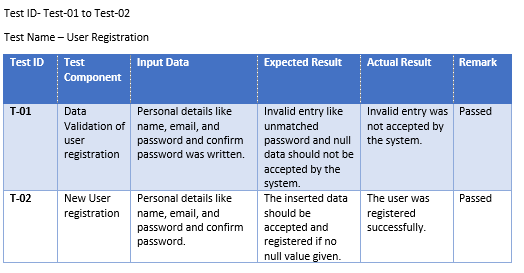


Figure 11:Blackbox Testing of User Registration

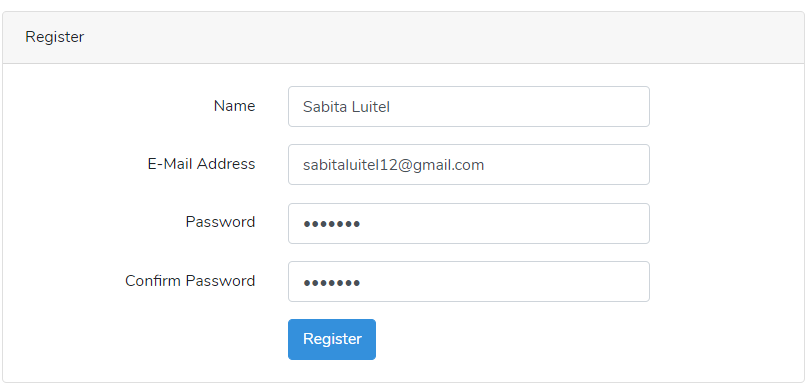


Figure 12:user registration

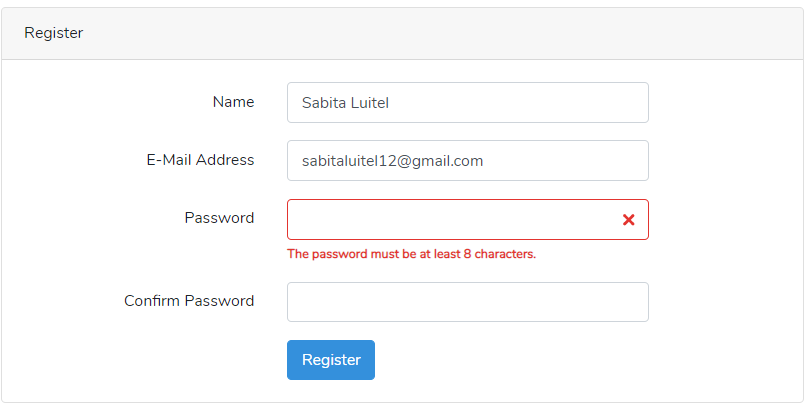


Figure 13:User password requirement for registration

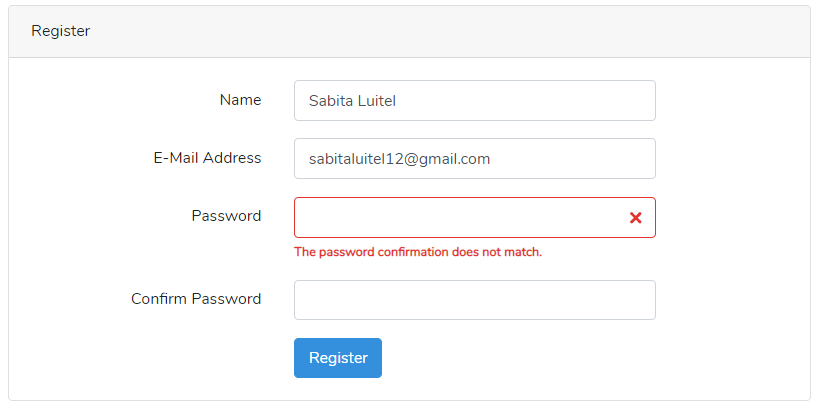


Figure 14::User password does not match

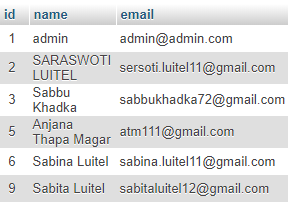
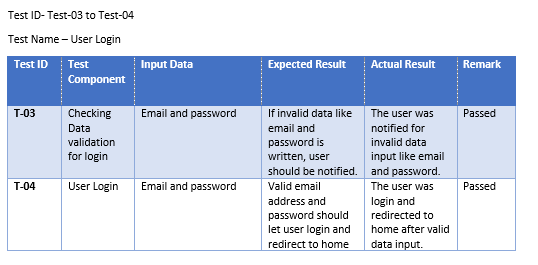


Figure 15:Database after successful registration

 Figure 16:Blackbox testing for User Login

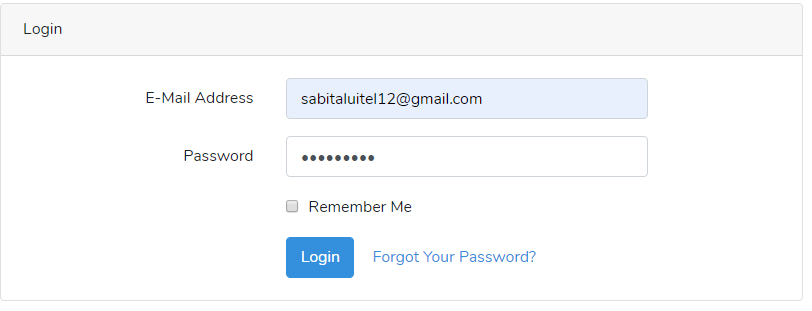


Figure 17:User Login

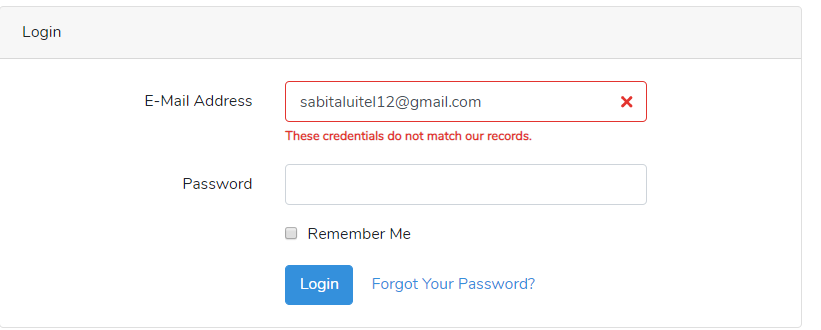


Figure 18: Invalid user details notification

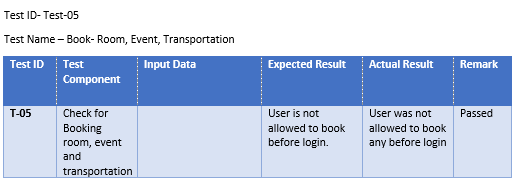


Figure 19:Blackbox for checking of booking

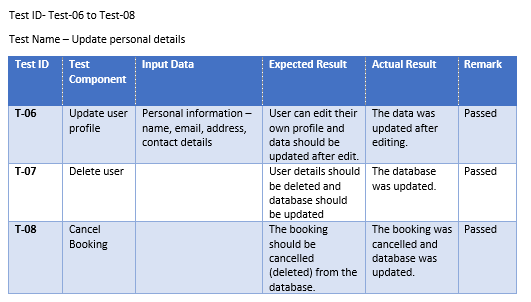


Figure 20:Blackbox testing for updating personal details

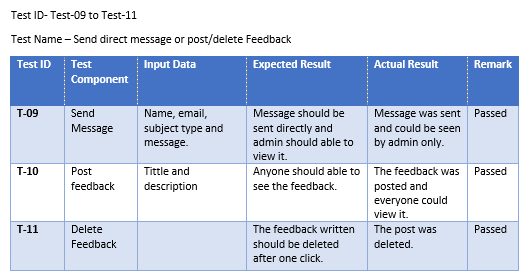


Figure 21:Blackbox testing of direct and post message

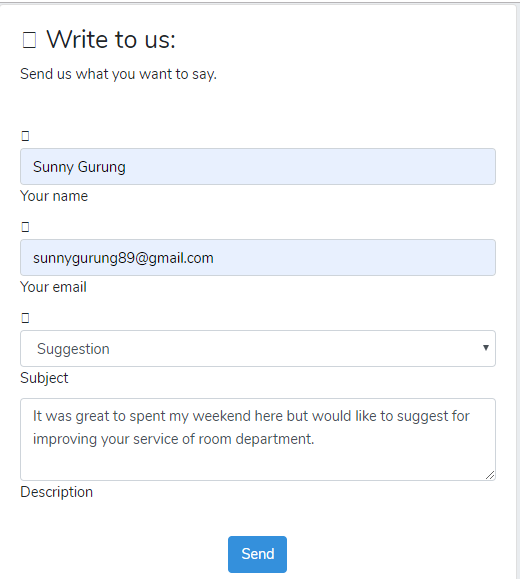


Figure 22: Inserting for direct message



Figure 23:Database after sending message

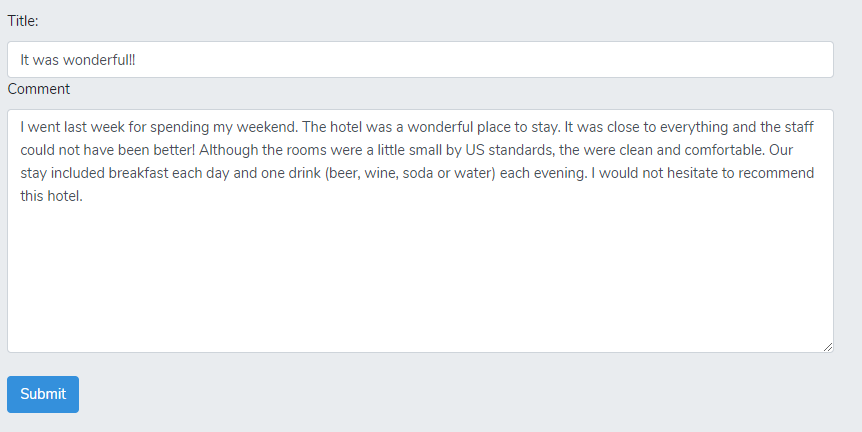


Figure 24:Inserting data for post



Figure 25:User Interface after posting

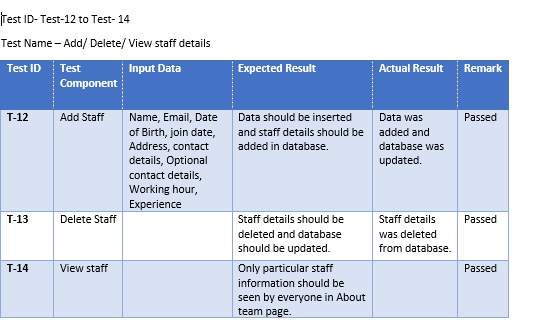


Figure 26: Blackbox for updating staff details

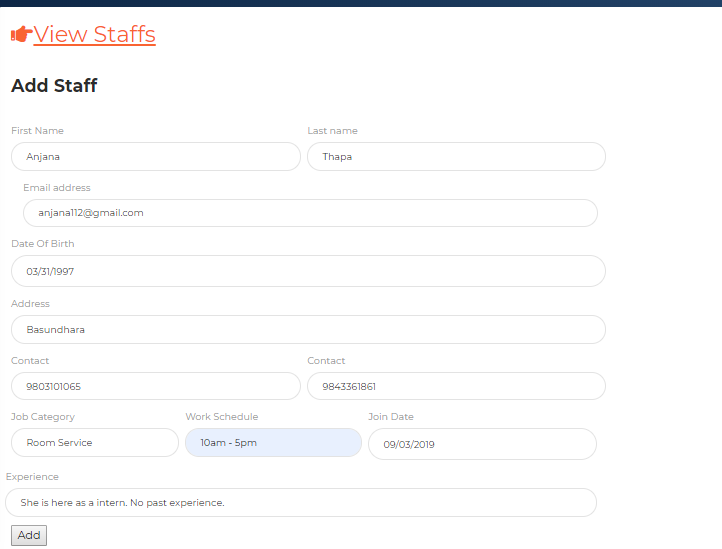


Figure 27: Inserting staff details



Figure 28: Database after adding staff

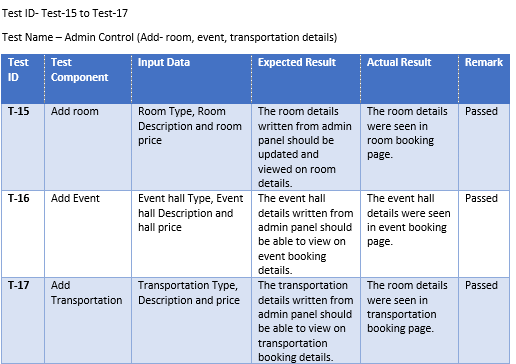


Figure 29: Blackbox testing for Adding room, event and transportation

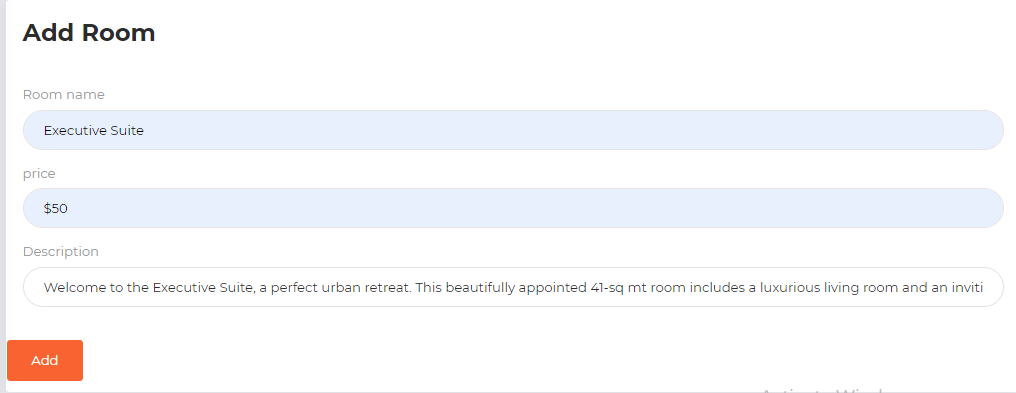


Figure 30: Adding room details

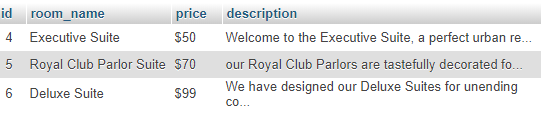


Figure 31:Database after adding room details

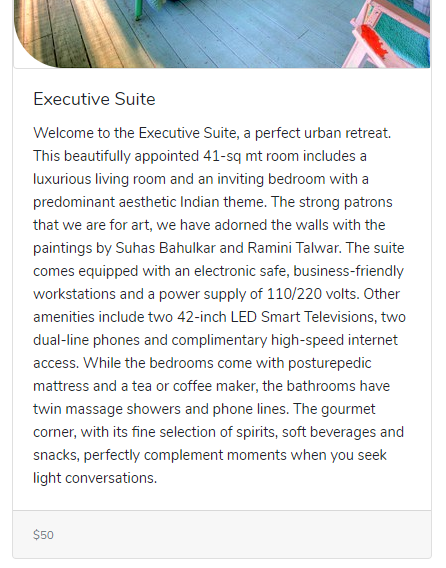


Figure 32:User interface after adding room details

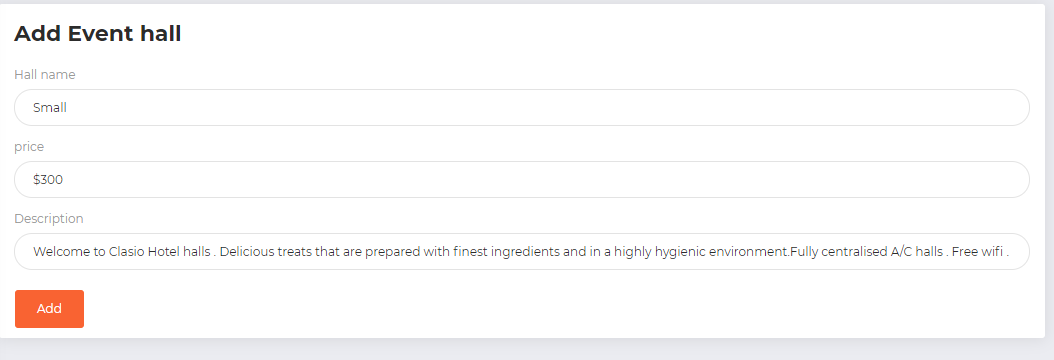


Figure 33: Inserting hall details

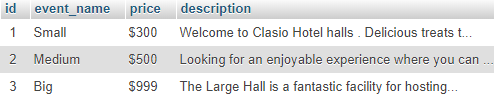


Figure 34:Database after inserting hall details

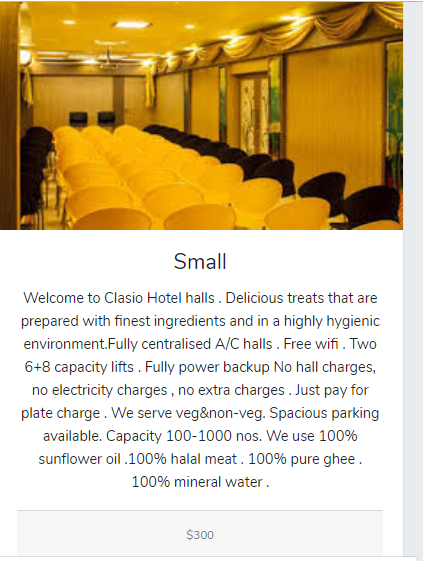


Figure 35: User Interface after inserting hall details



Figure 36: Adding transportation details

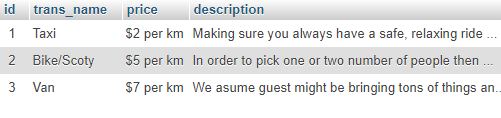


Figure 37:Database after adding transportation details

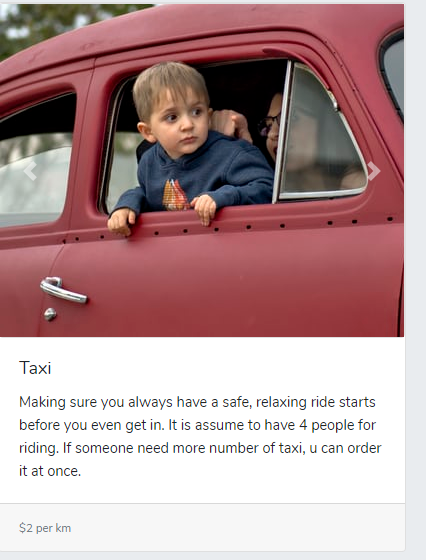


Figure 38: User interface after adding transportation details

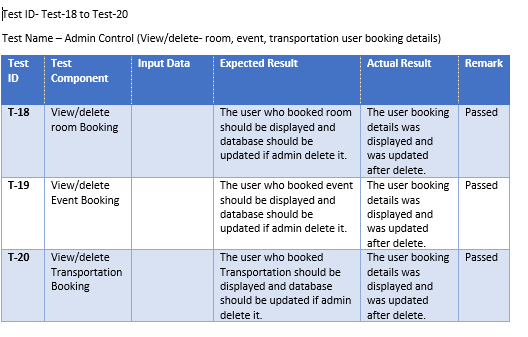


Figure 39: Admin control panel for room, event and transportation

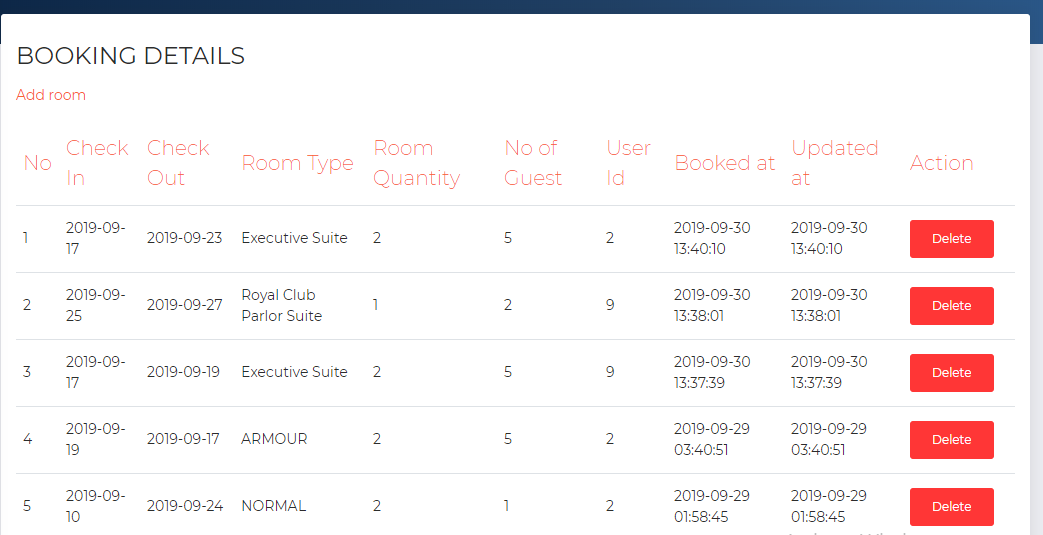


Figure 40: Admin panel of room booking details

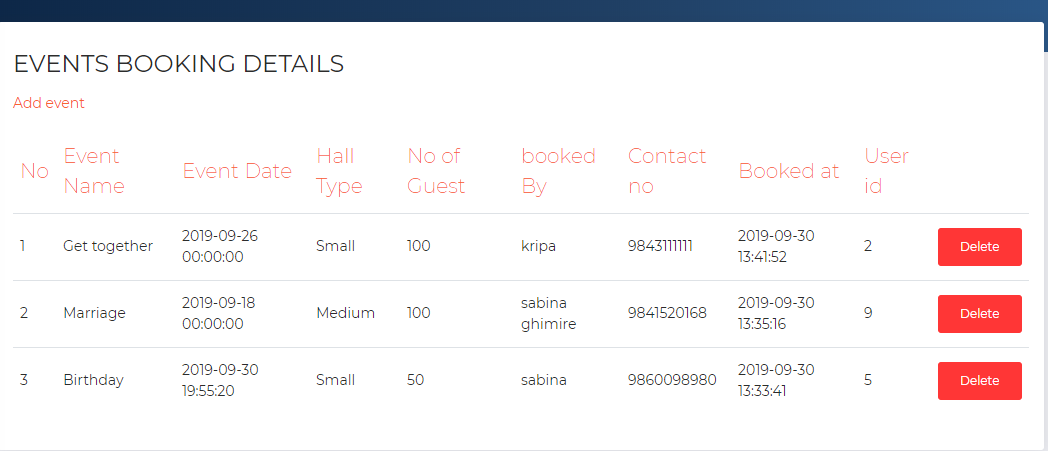


Figure 41: Admin panel of event hall booking details

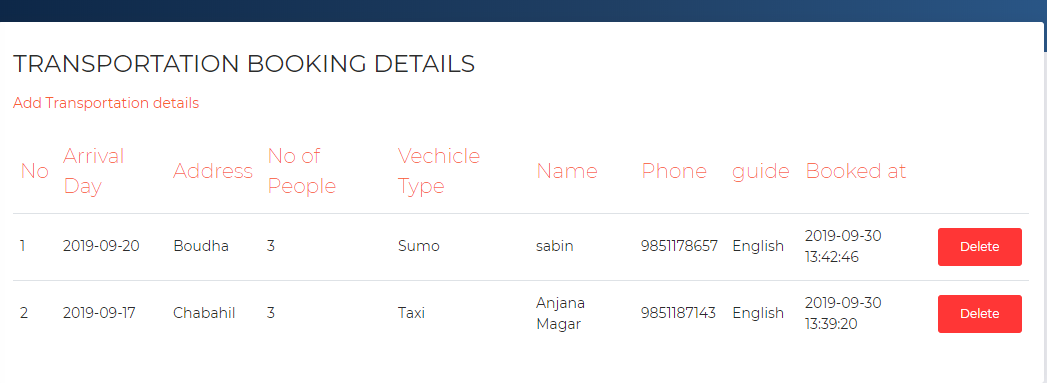


Figure 42: Admin panel of transportation booking details