Web Application for Mess

Test Document

Submitted by :
Abhash Yadav
Manish Kumar
Sumit Singh
Shikhar Jaiswal

Faculty Adviser:
Dr. Neeraj Goel
Dr. Mukesh Kumar Saini

February 20, 2017

Contents

Strategy for testing

Strategy for Testing : The basic strategy is that we would be giving custom inputs to the modules using a script/manually, and then comparing the output with expected output. This same procedure is applied to test all the modules.

0.1 User Module

0.1.1 Entering a new user

Here we will be entering a sequence of names, and then printing out the values that are actually stored in the server. If it worked correctly, then any information regarding the new user will be stored correctly(assuming auto accepting input by admin). Moreover we will also check if this is sending a request to admin or not.

0.1.2 Storing passwords

It is very important to encrypt the passwords before storing, and we will be using the built in sha1 OR md5 function of php to store the passwords on the database.To test the system, we will enter the passwords to the system, look at what is being stored and compare its value with the output of sha1 function given the original password string.

0.1.3 User Authentication

Strings of username and passwords will be entered and we will then check if using those two inputs, if we are able to log in or not.

0.1.4 Forgot Password Module

It will be checked if we are recieving the emails with the correct link to reset password page or not.

0.1.5 User Options

The options availiable to user looking at the day's menu and selecting whether he will go to the mess or not. To check weather the system is working or not we will be matching the values stored in the server , and then changing the value and see if the change is actually taking place or not. The weekly menu is a static option. Also the arrival info must be provided in an hour in advance, so we have to verify if the option to change arrival is closing at correct time or not. This will be verified by changing the values of mess opening time and then noting if the options are changed or not.

0.1.6 User Bills

The daily bills of the user are added to his/her monthly bill every time he/she goes to the mess, we will add arrivals of the user and then see if the corresponding values of monthly bill are correct or not.

0.2 Admin Module

0.2.1 Adding New User

We have already checked that 'is the new user request being sent or not', here we will check if the requests are recieved or not, and weather the user credentials are displayed correctly or not. This will be done by manually sending requests and seeing if the data arrived is correct or not.

0.2.2 Update Menu

Again we just have to verify that the menu details that are added are correctly stored in the server or not.

0.2.3 People Coming for meal

This is similar to checking of 'Adding new user' give above, just we have to see if the count is correct or not.The count is finalised only when 1 hour is remaining, and we have to see if the count is coming or not. We will add the arrival of some custom users, and then see if the corresponding details of arrival for each person is stored correctly or not. If it is stored correctly, the count will be checked for correctness correct.

0.2.4 The monthly bills

The monthly bills of all the users are added, and is shown. This is simple to verify, just we have to per verify if the data stored in the server about the user's arrival are correct or not, and that if the bills added are correct or not, which are already done, just we have to access the data and add the monthly fees of the users.

0.3 Overall Testing

After being assured that the modules are working correctly or not, an overall check is done on all the processes, and connections. We have to verify that the modules are correctly communicating with the server or not. We have to check the inter-modular linking as well.