# 5 Testing

**Testing** is the way of examining whether the program or software works as per the requirement or not. It helps to minimize or solve errors and also helps to find out if any.

There are different types of testing such as Black-box testing, Unit testing, Integration testing etc. For testing this system i.e. (Sport Events Management System) especially unit testing and black-box testing are performed.

**Black-box** testing is the way of checking user interface or external design without focusing on the internal functionality or structure of a program.

**Unit testing** is a type of testing that mainly focuses on the functionality and logical workings of the software or program. It is done to examine the program (by dividing it into units) whether it is functioning as required or not.

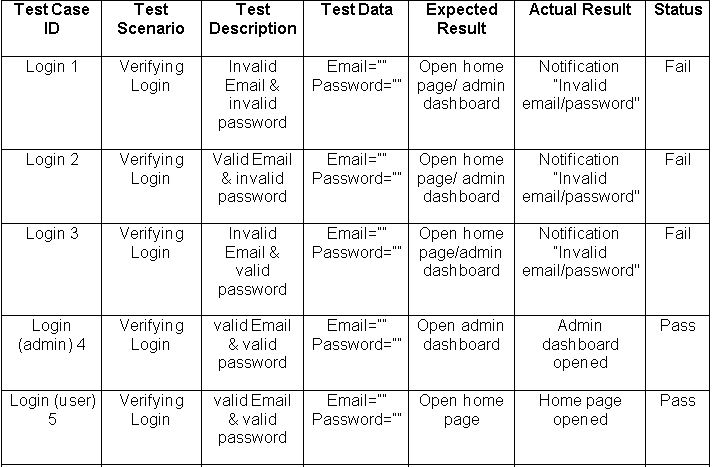
**Purpose of choosing Unit testing and Black-box testing**

* Black-box testing tests the interface/external design of this system i.e. (Sport Events Management System) which simultaneously makes better interface without complexities.
* Likewise Unit testing examines the functionality of the system by breaking it down into various units to make this system consistent while performing.
* Black-box tests external design/interface whereas Unit testing tests internal functions to check whether they are working as required or not and helps to make this system more precise.

## 5.1 Black-box Testing

Following is a black-box testing of Sport Events Management System.

**Login**



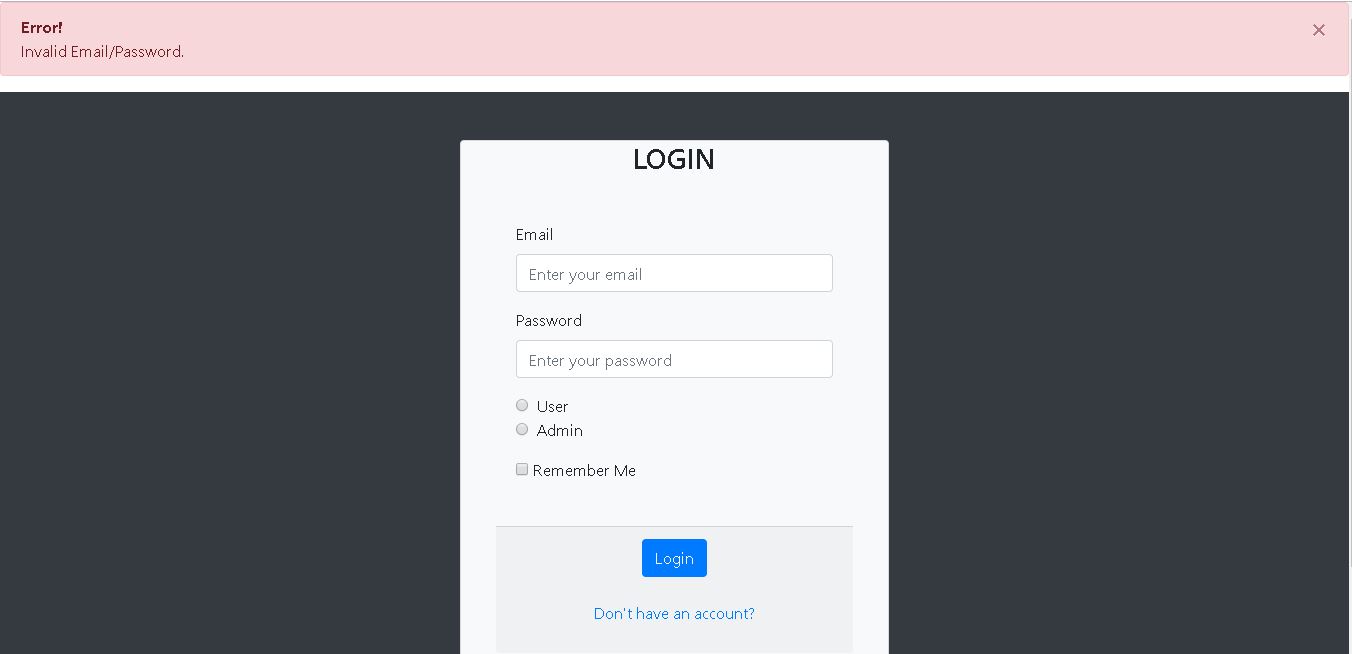


Figure 1: Notifying invalid login

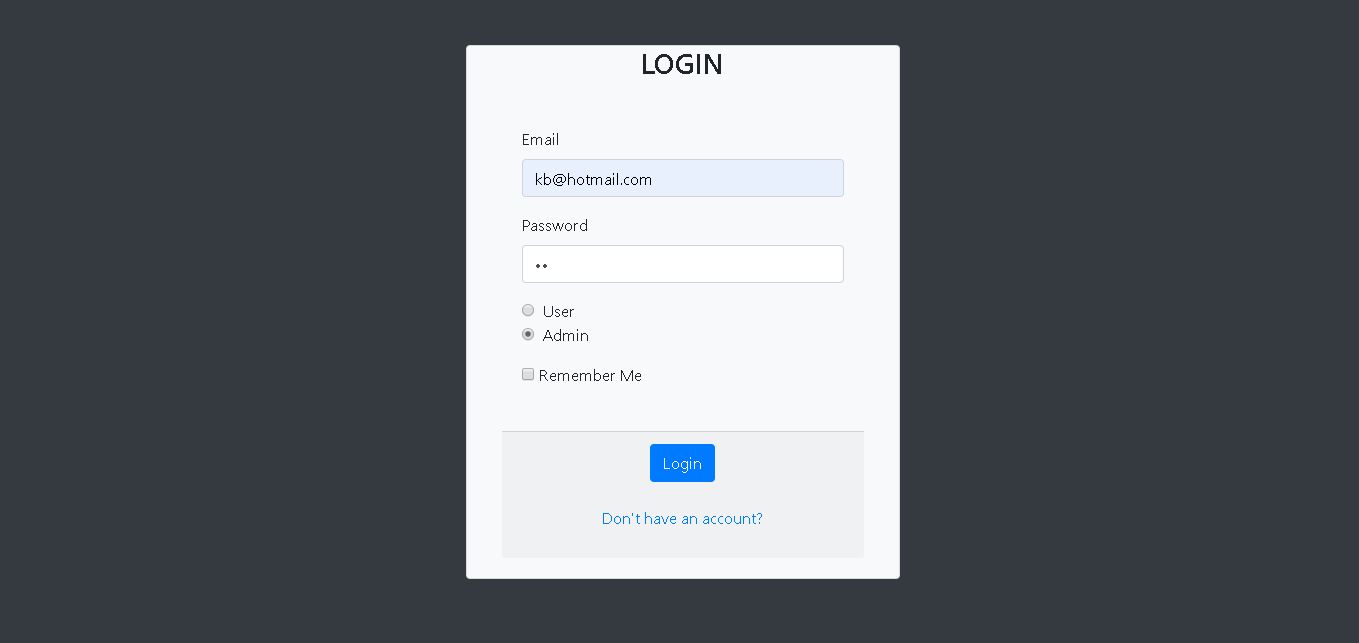


Figure 2: Logging as admin

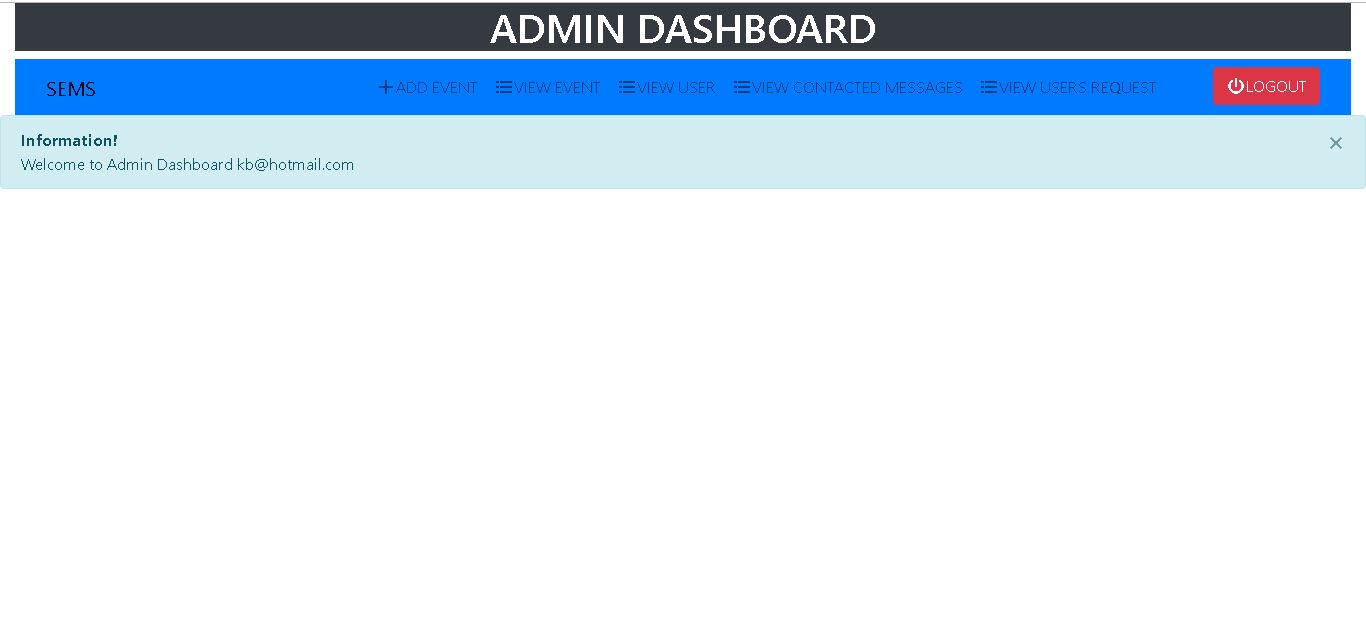


Figure 3: Admin dashboard after logging as admin

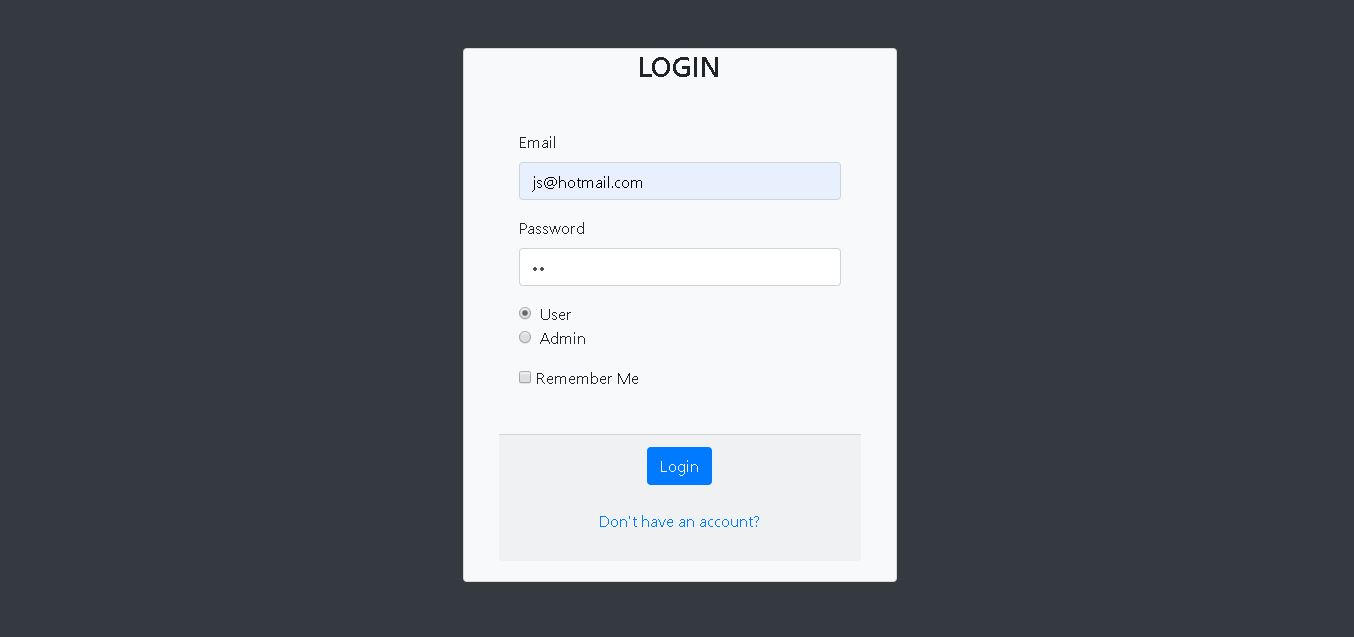


Figure 4: logging as user

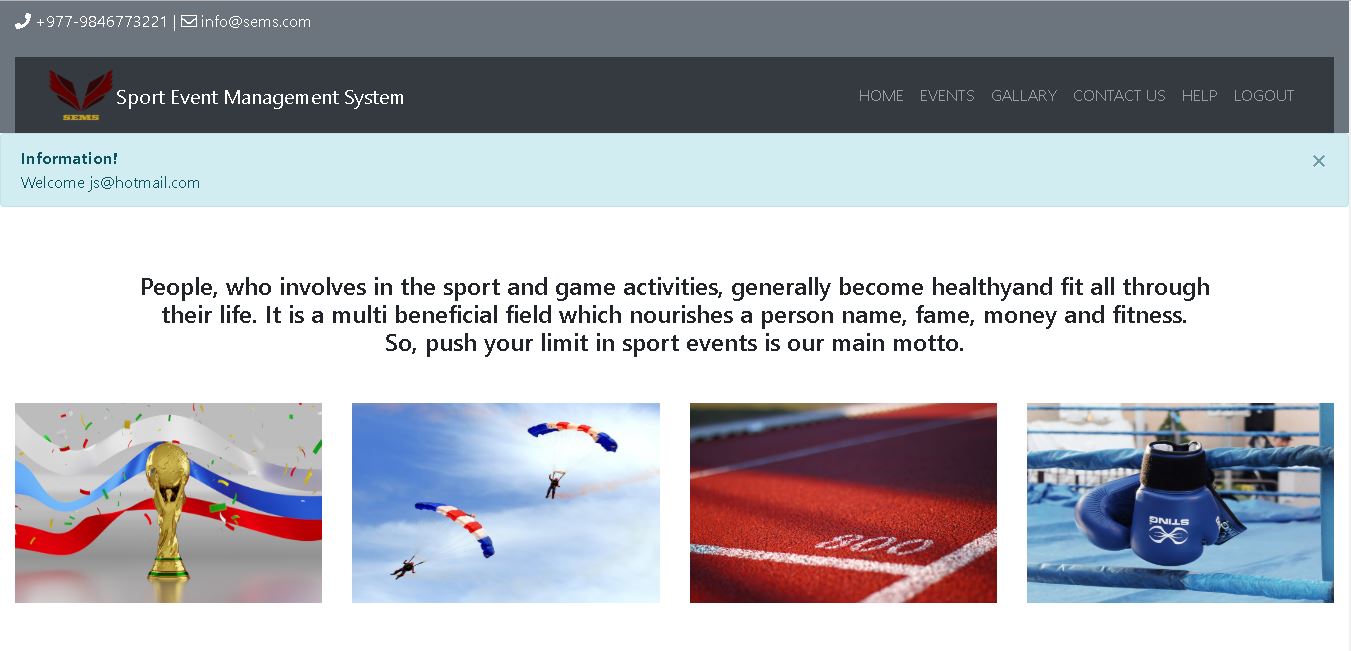
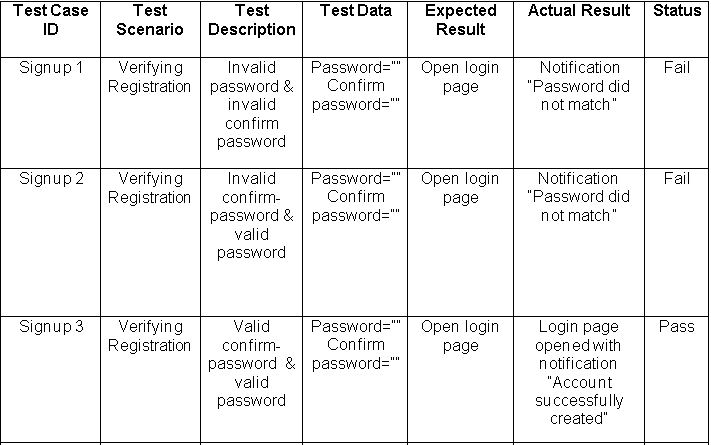


Figure 5: Home page after successful login as user

**Registration**

****

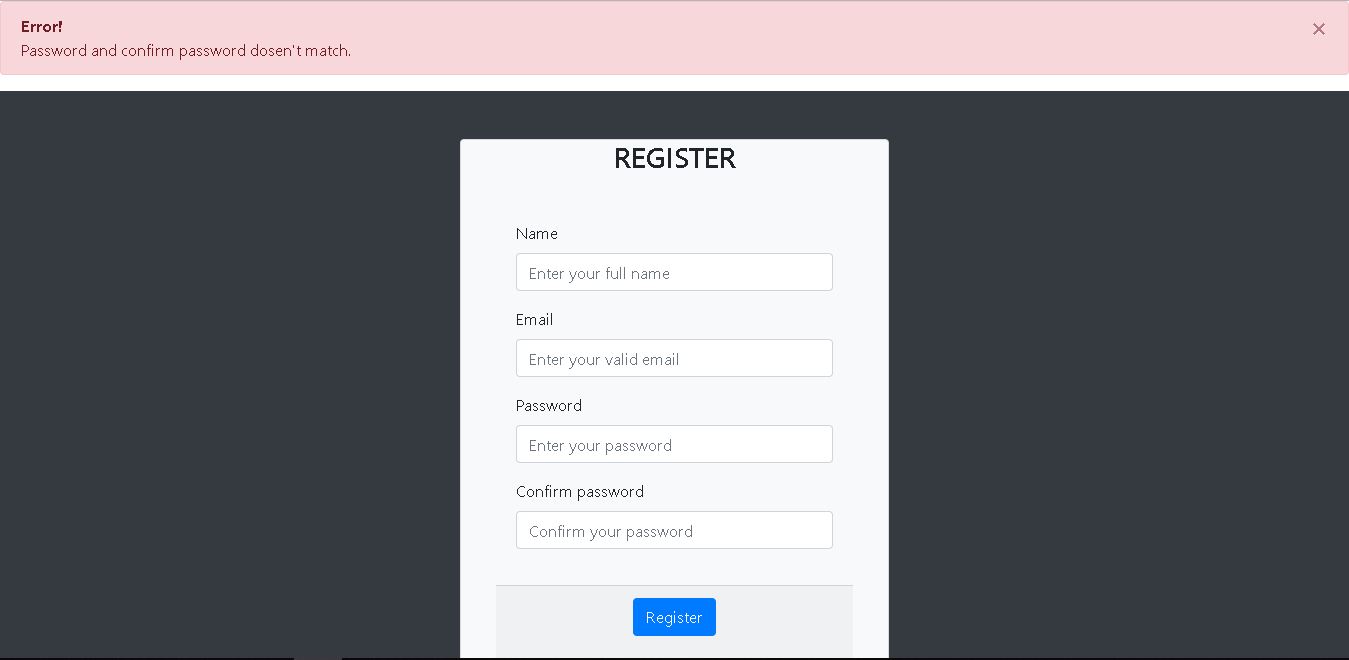
****

Figure 6: Notifying password and confirm password does not match

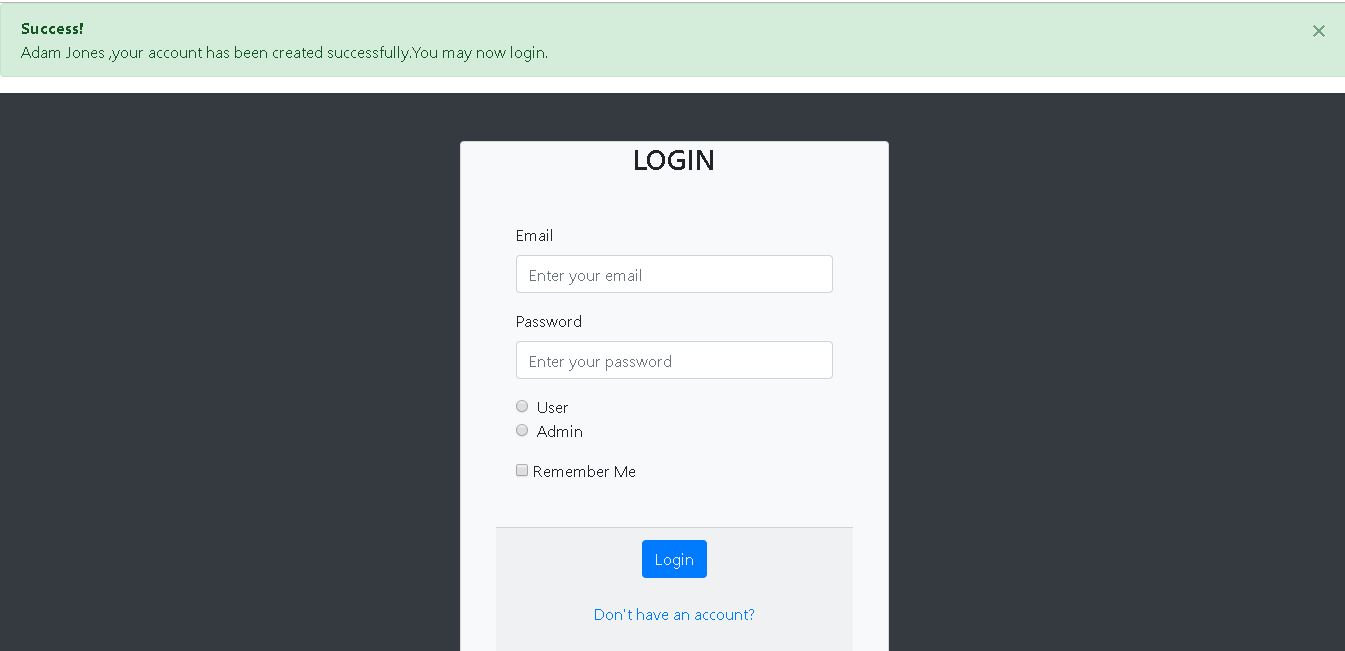
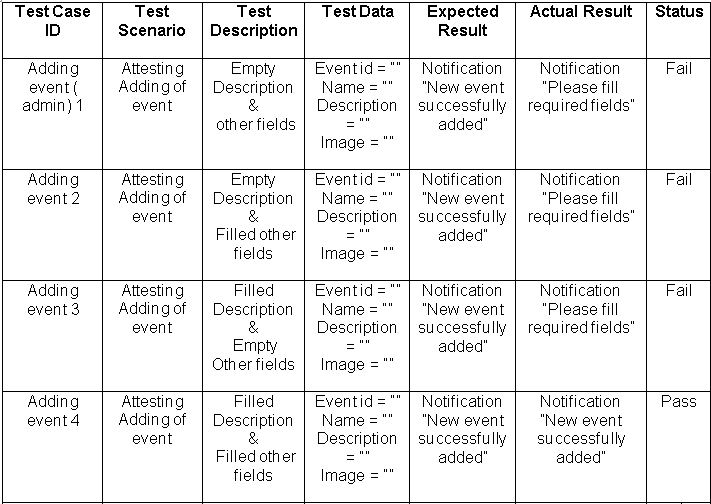


Figure 7: Login page after successful registration

**Adding Event**



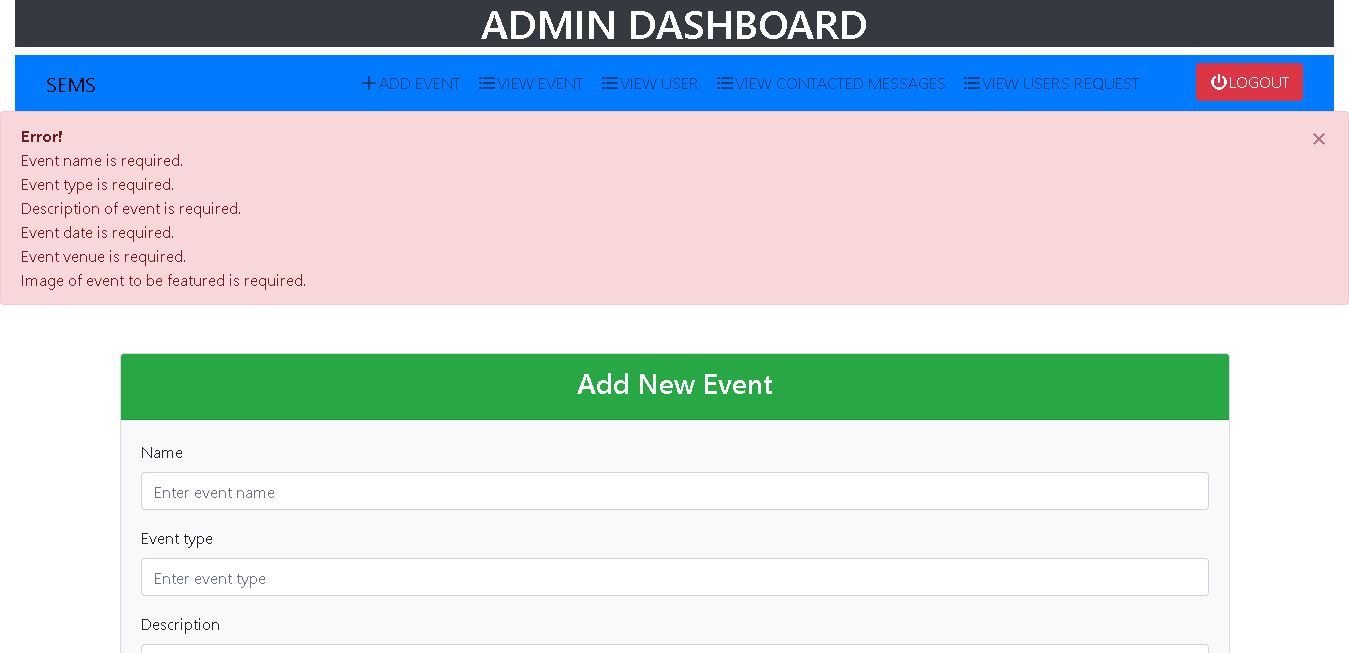
****

Figure 8: Error notification notifying to fill the required fields

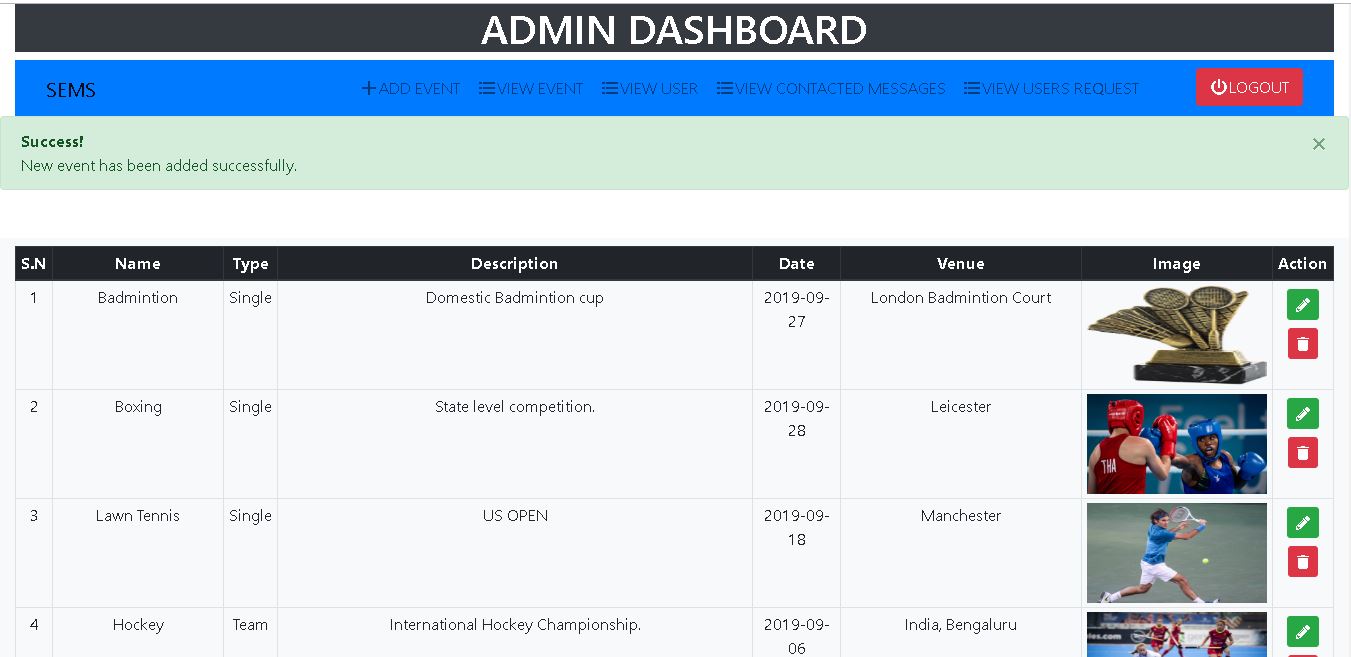
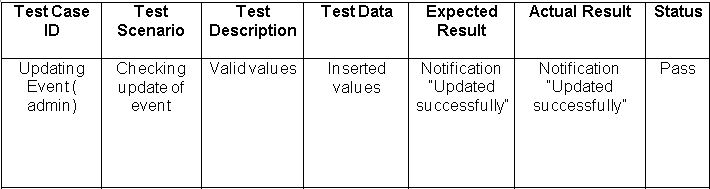


Figure 9: New event has been added successfully

**Updating Event**

****

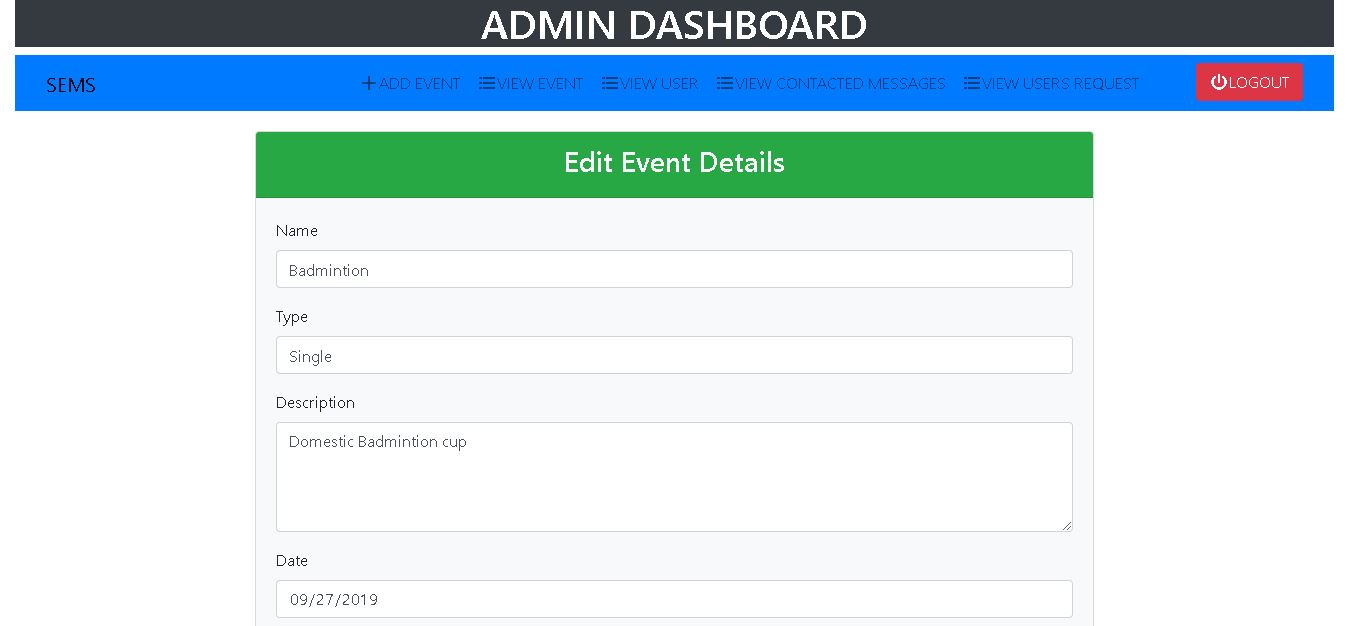


Figure 10: Updating event details

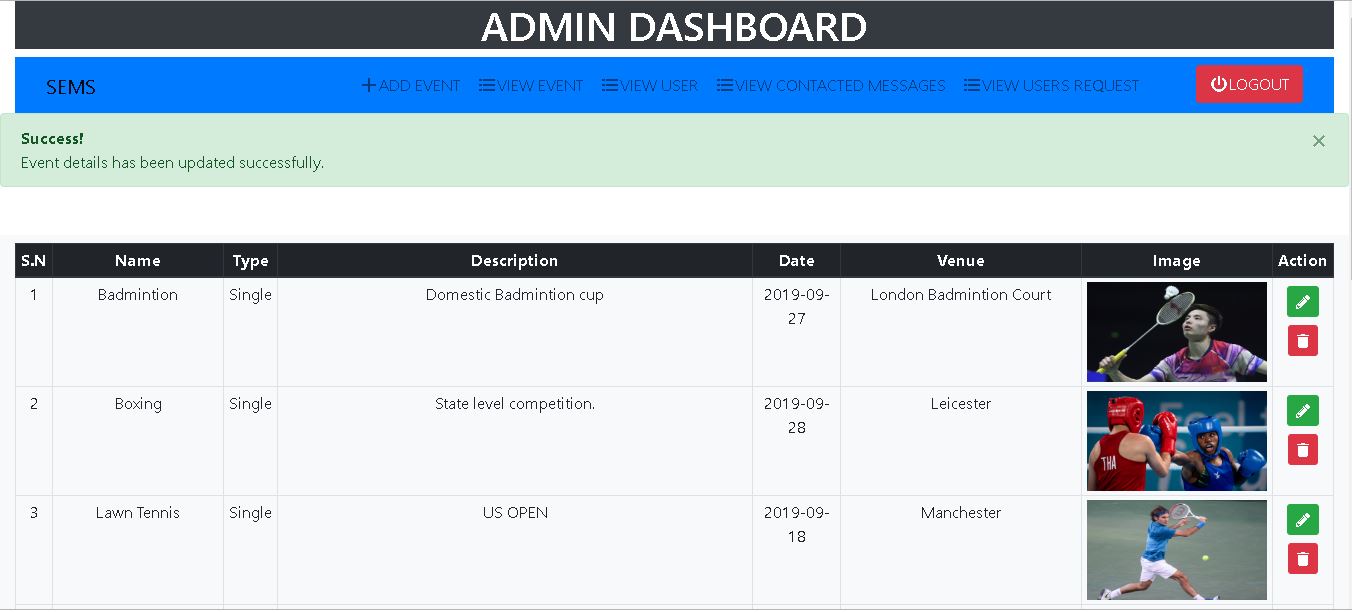
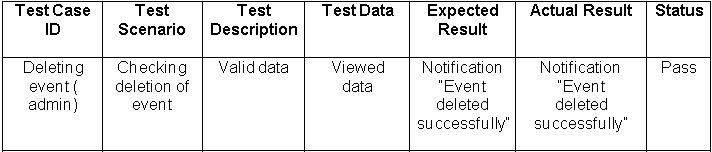


Figure 11: Badminton event's image has been updated successfully

**Deleting event**

****

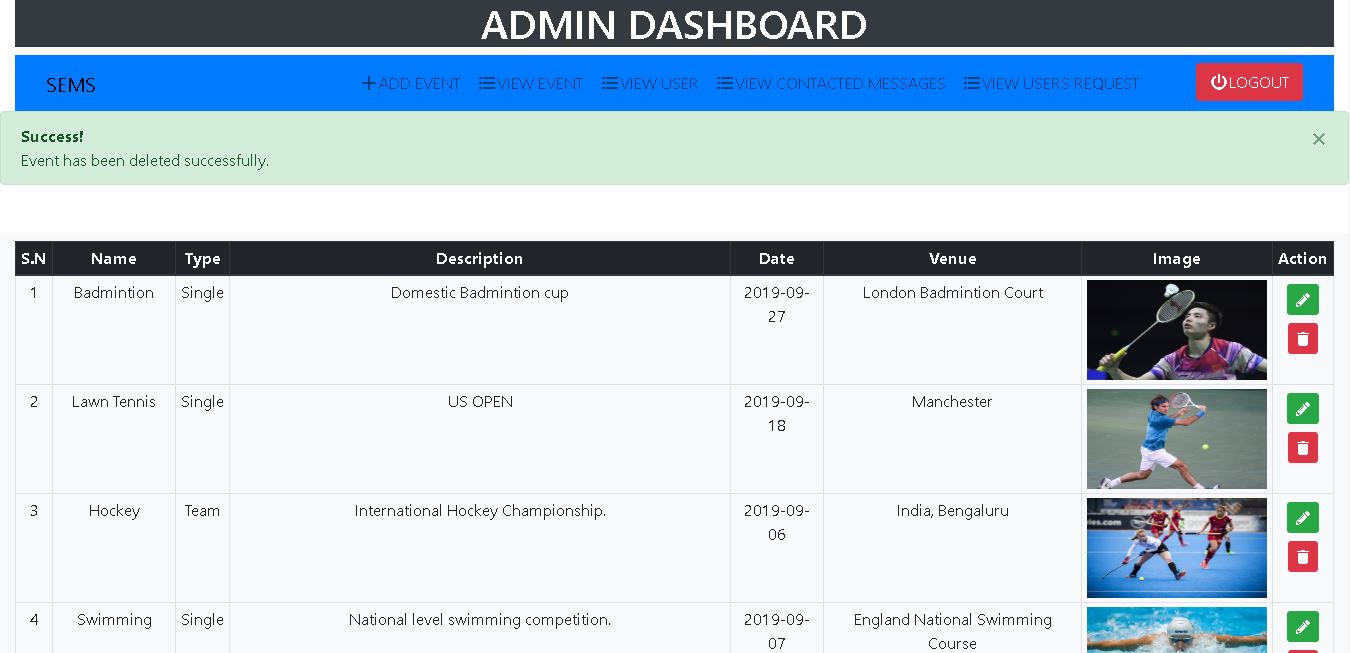
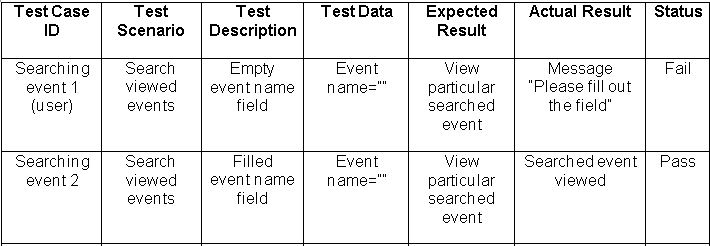
****

Figure 12: Boxing event has been deleted successfully

**Searching event**

****

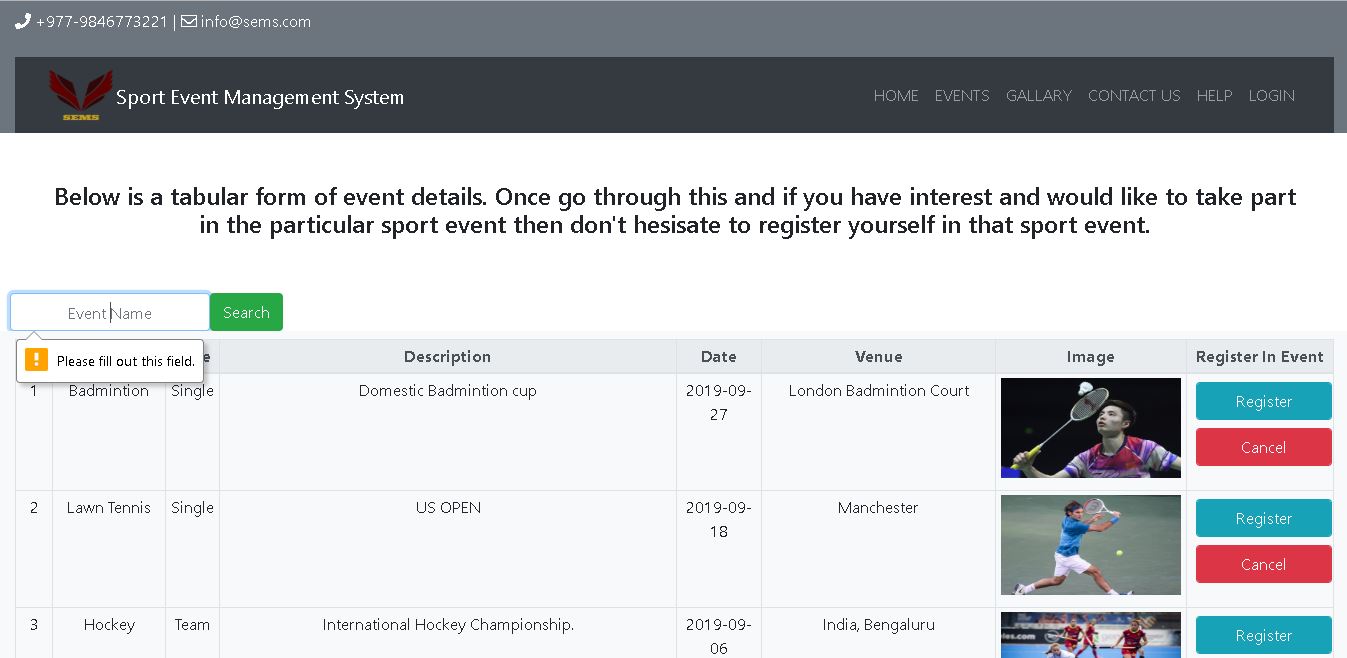


Figure 13: Notifying to fill the required event name

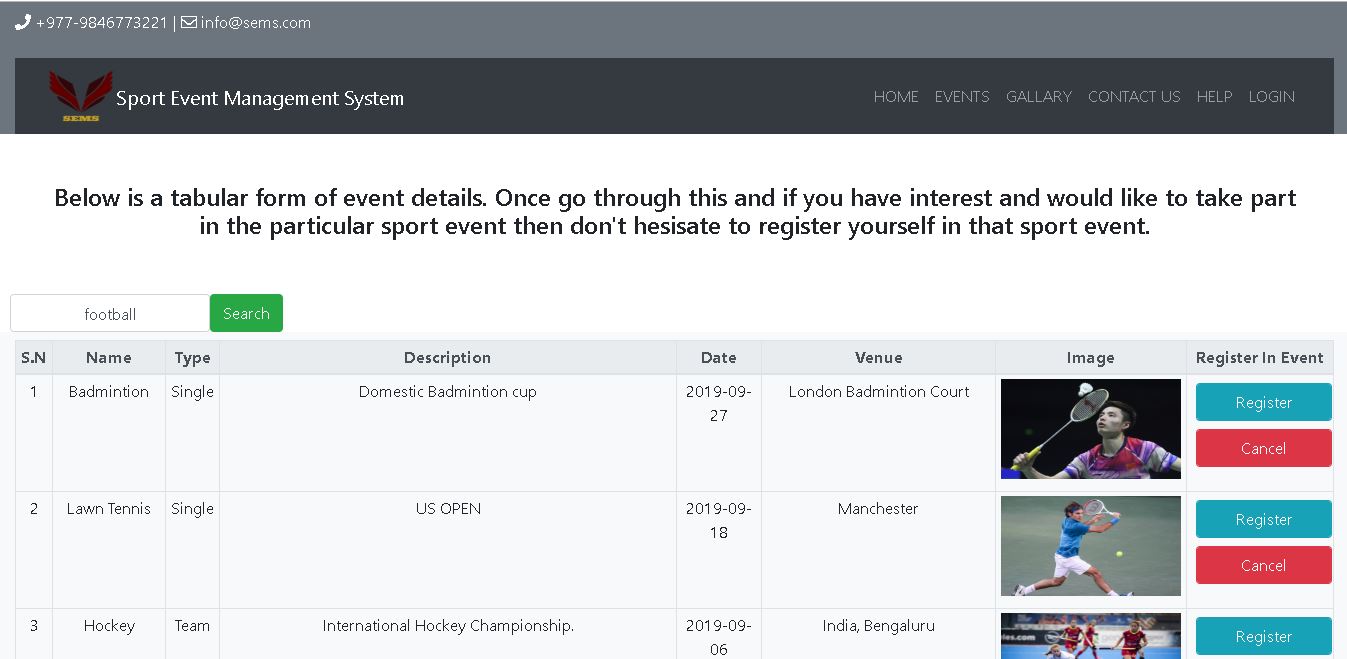


Figure 14: searching a particular event with event name

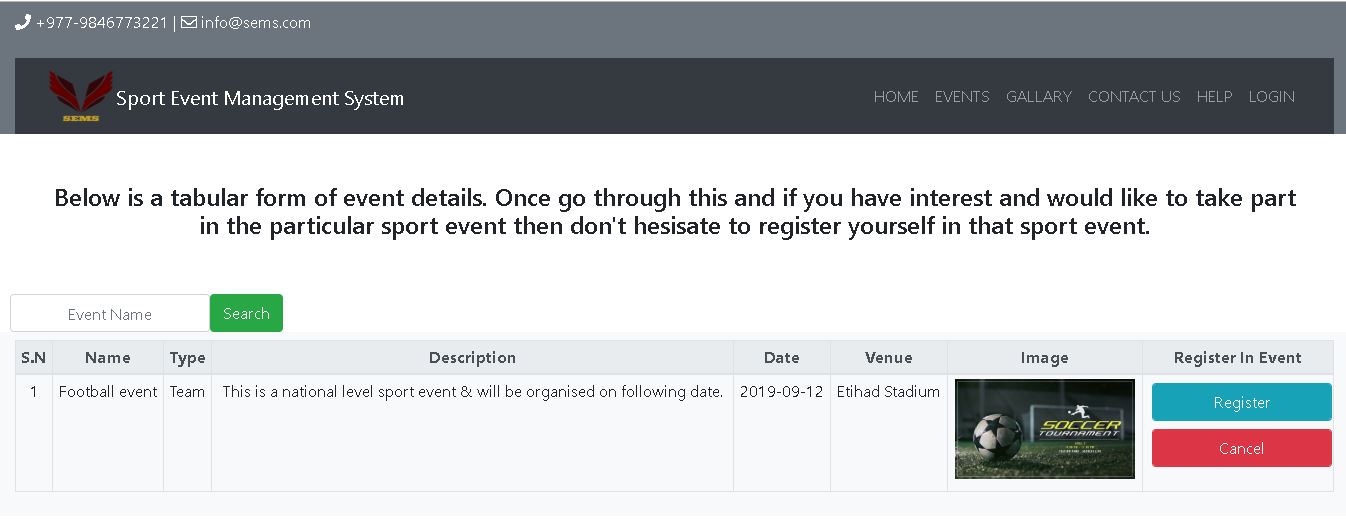
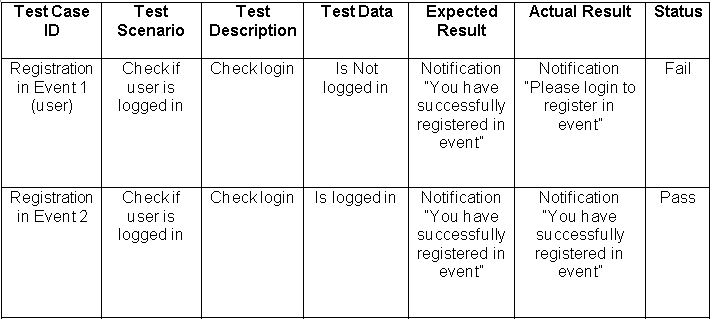


Figure 15: Result after searching with event name

**Registration in event**

****

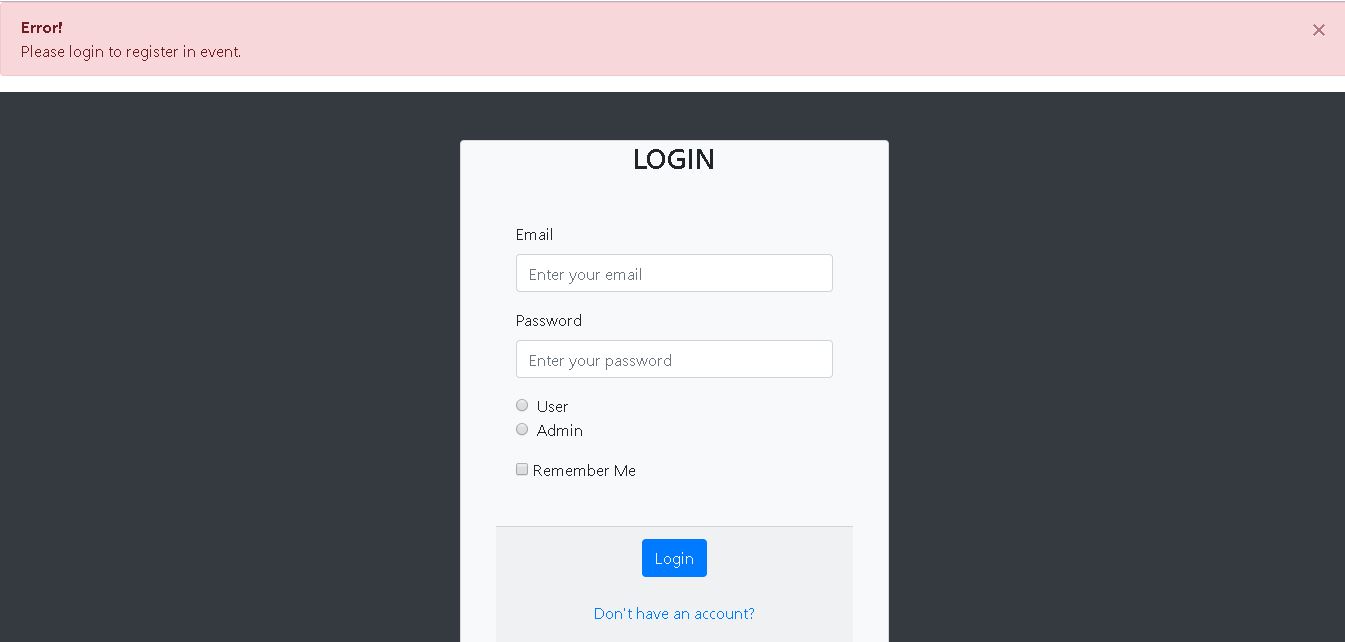


Figure 16: Notifying to login first for registration in event

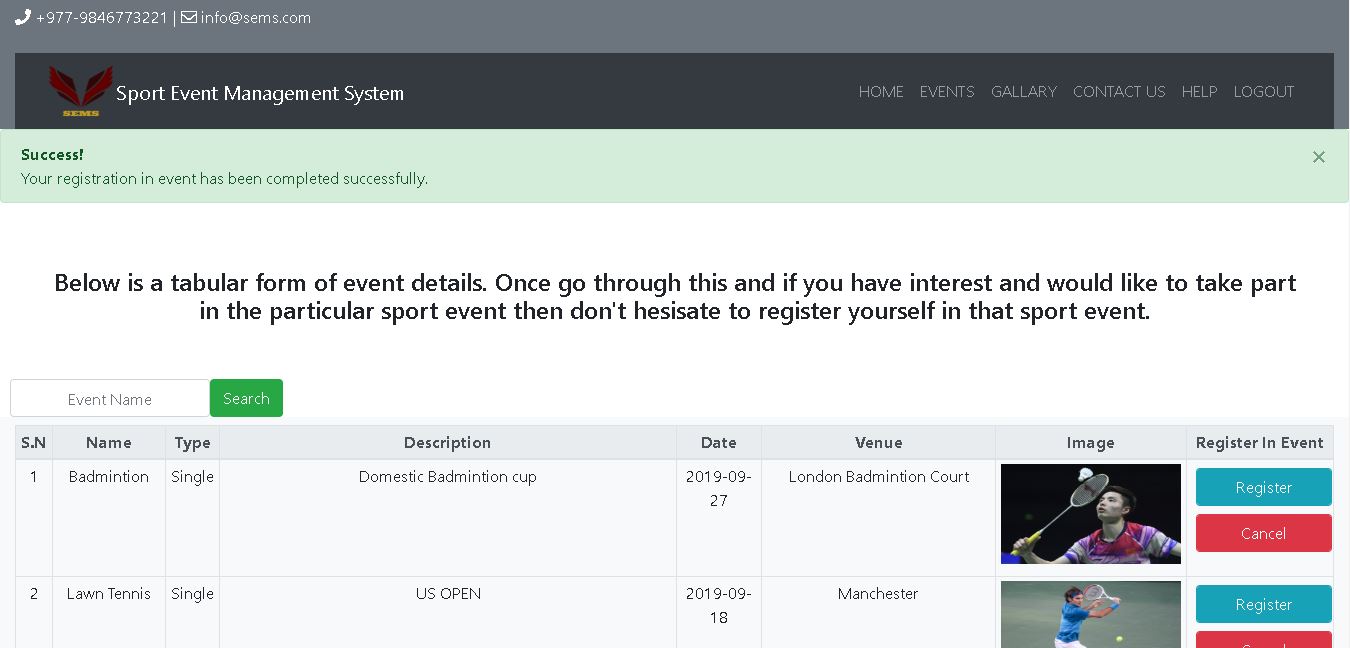
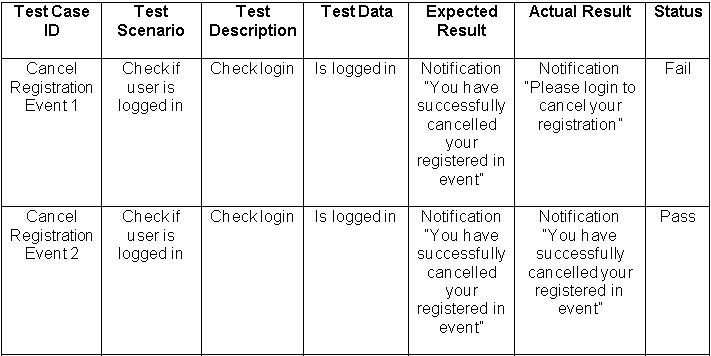


Figure 17: successfully registered in event after logging in

**Cancel registration in event**

****

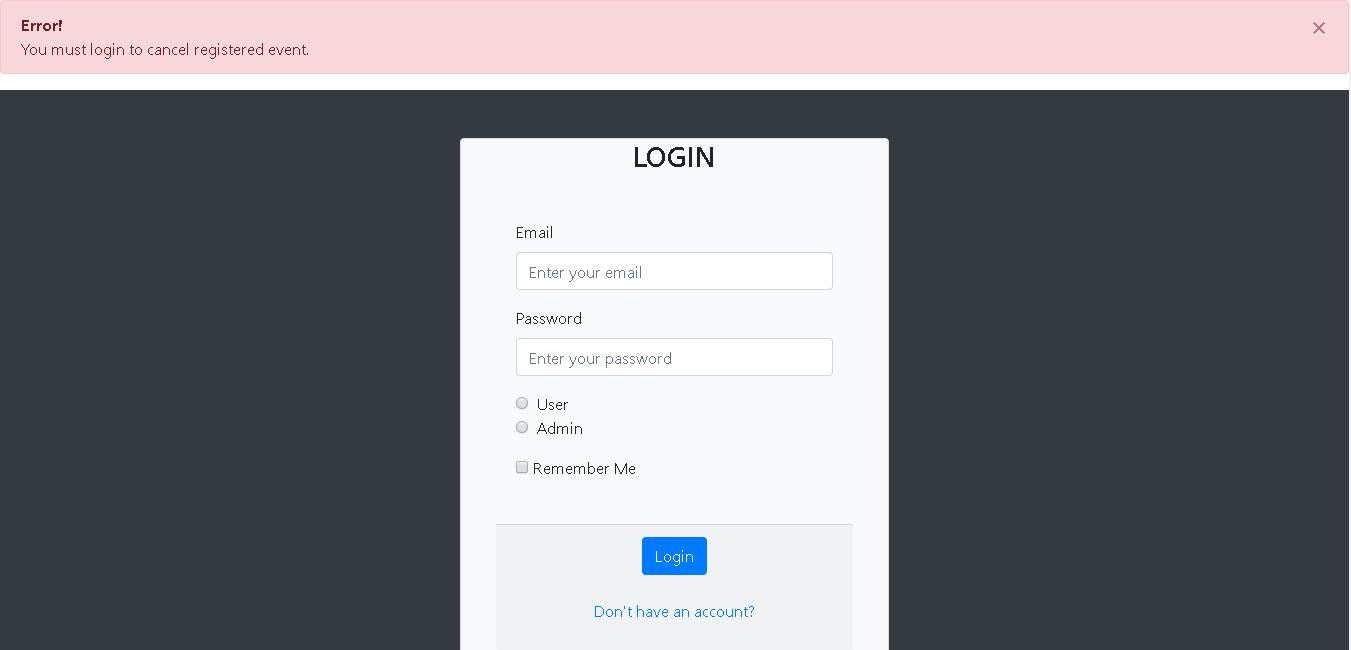
****

Figure 18: Notifying to login first for cancelling the registration in event

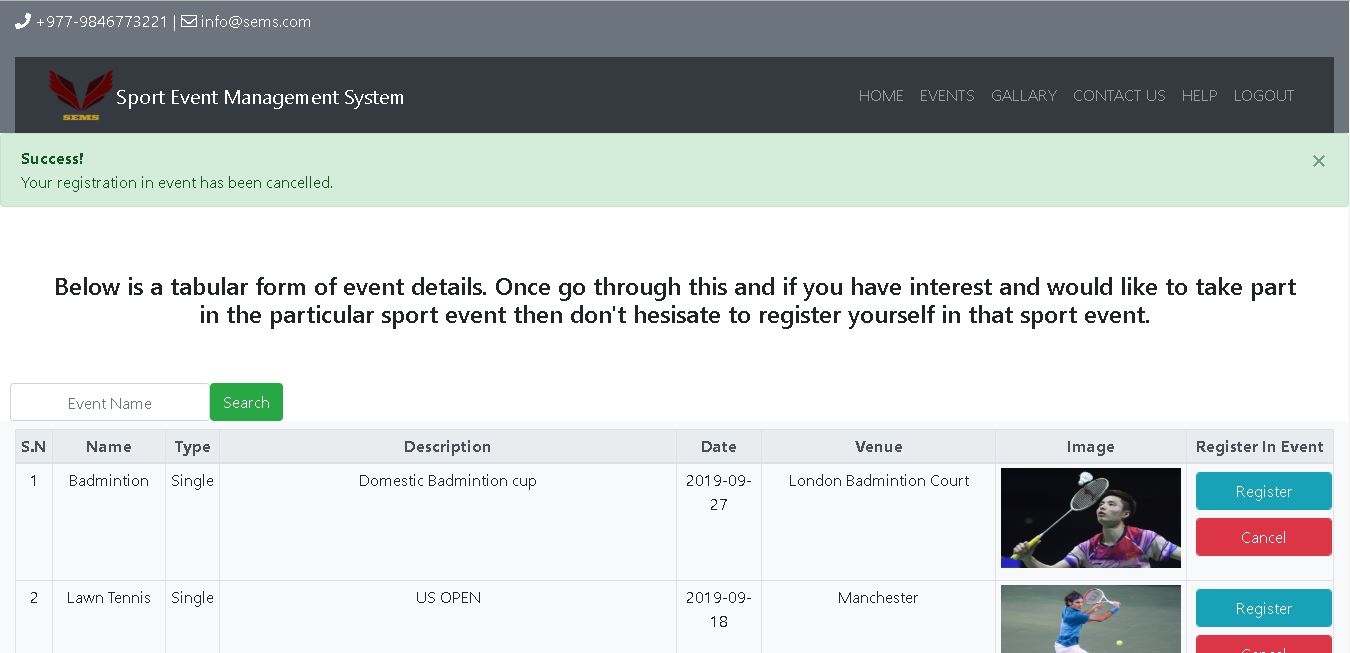
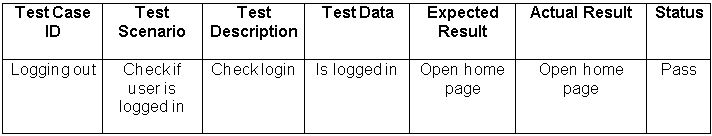


Figure 19: Registration in event cancelled successfully

**Logging out**

****

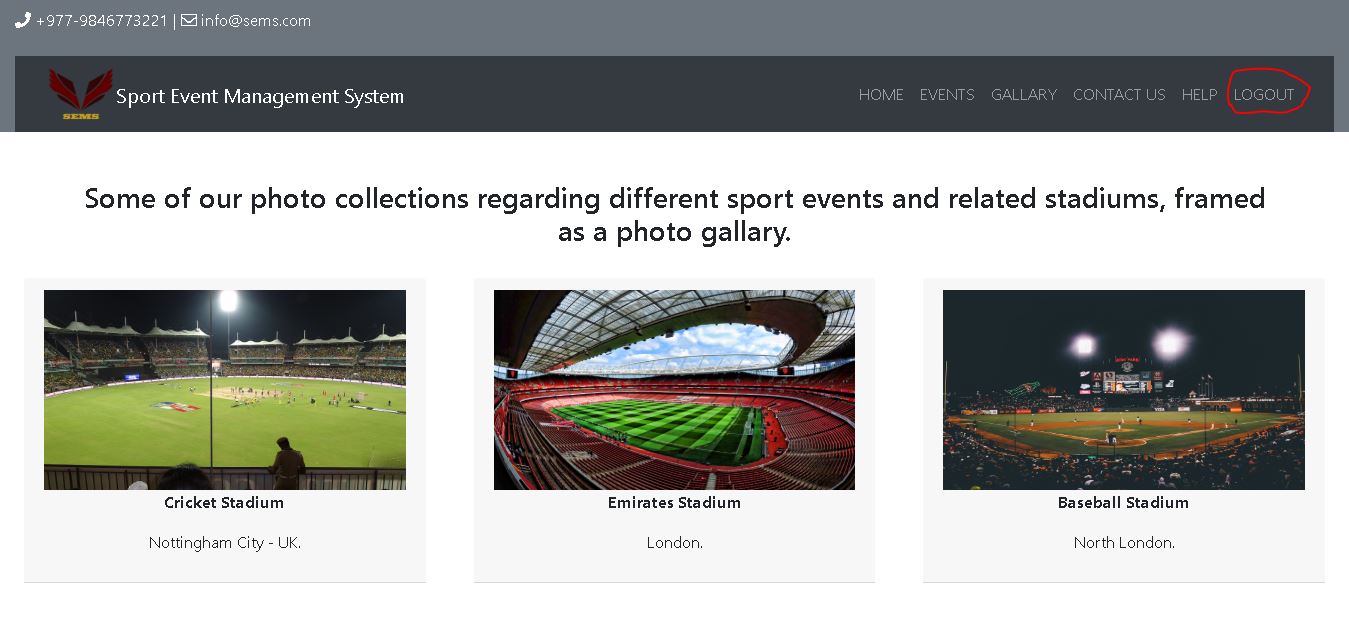


Figure 20: Page before logging out

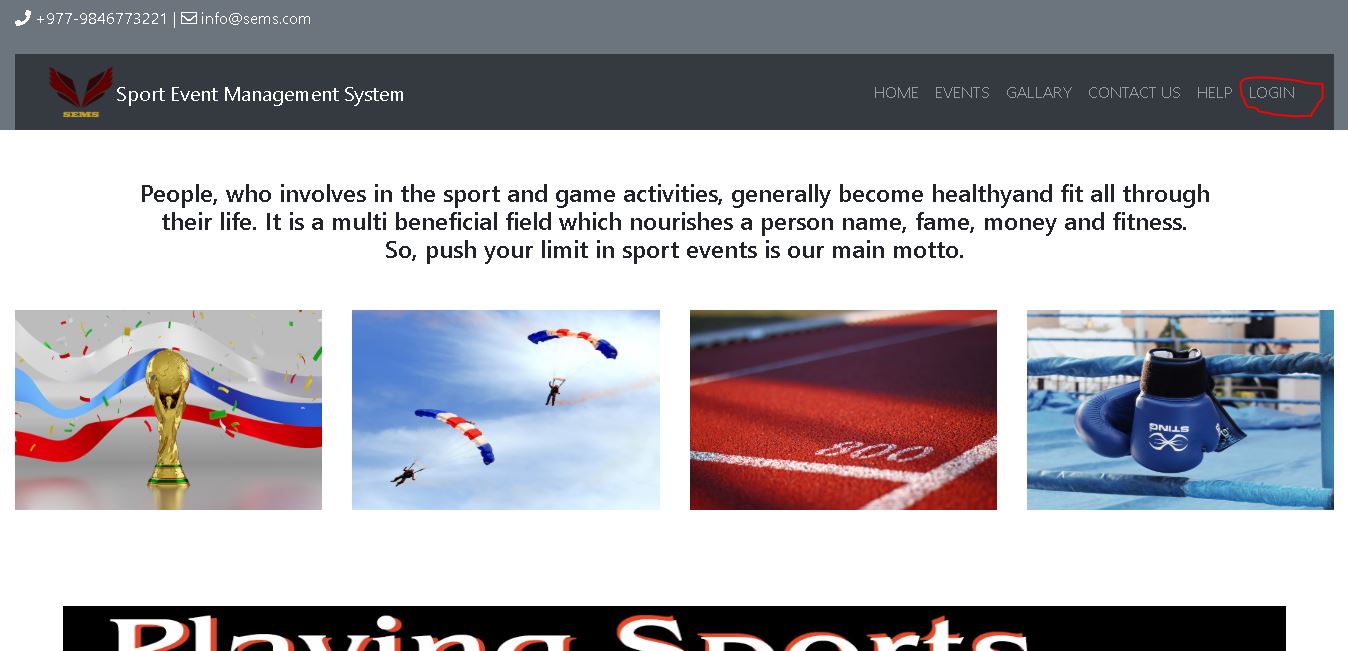


Figure 21: Home page after logging out

## 5.2 Unit-testing

Followings are the screenshots of the unit testing performed on Sport Events Management System

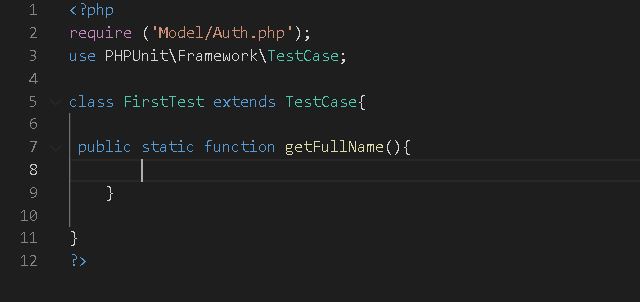


Figure : testing of empty method

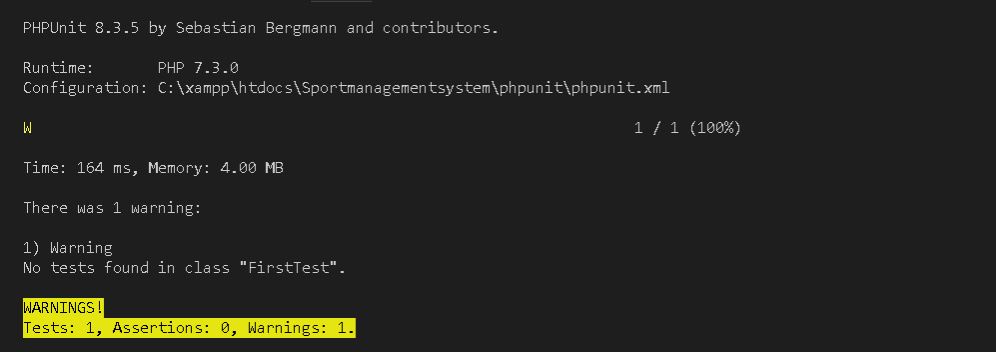


Figure : Warning 0 assertions



Figure : testing of setting and getting name

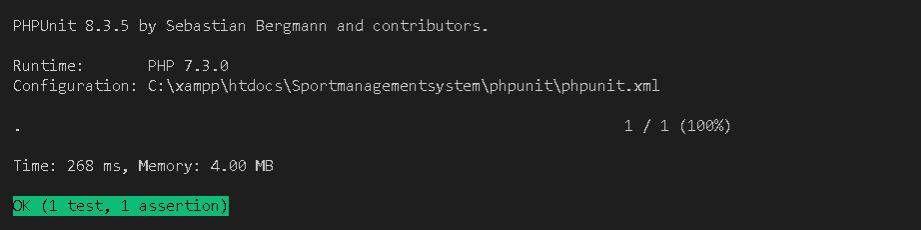


Figure : test successful

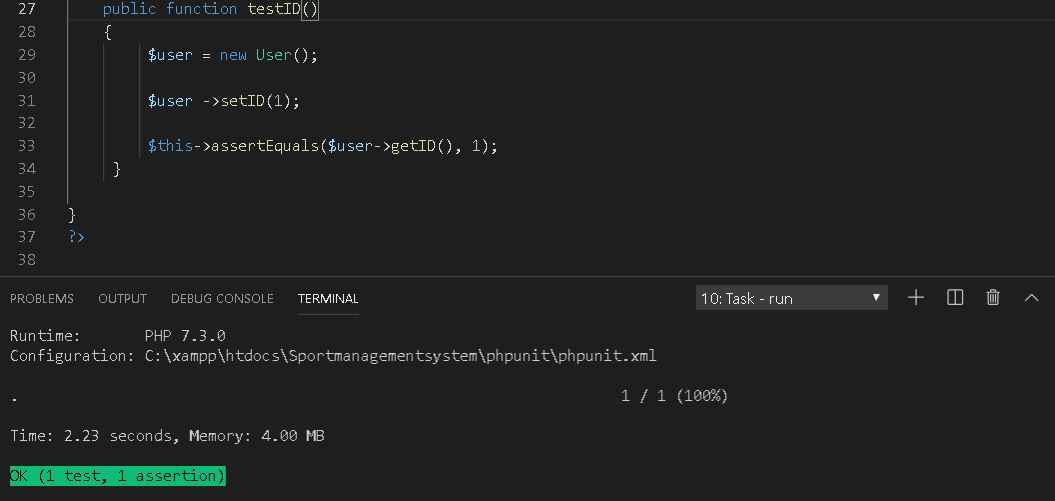


Figure : testing of setting and getting user id

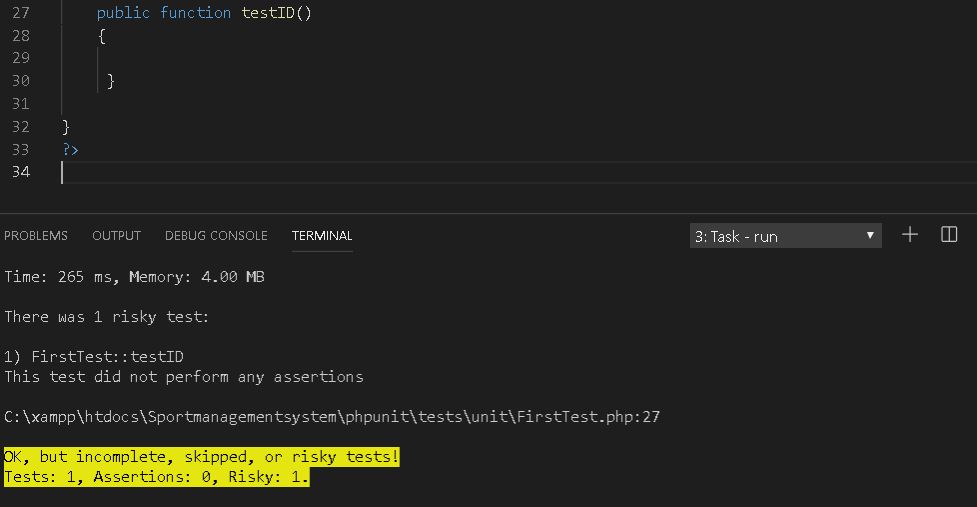


Figure : testing id fail

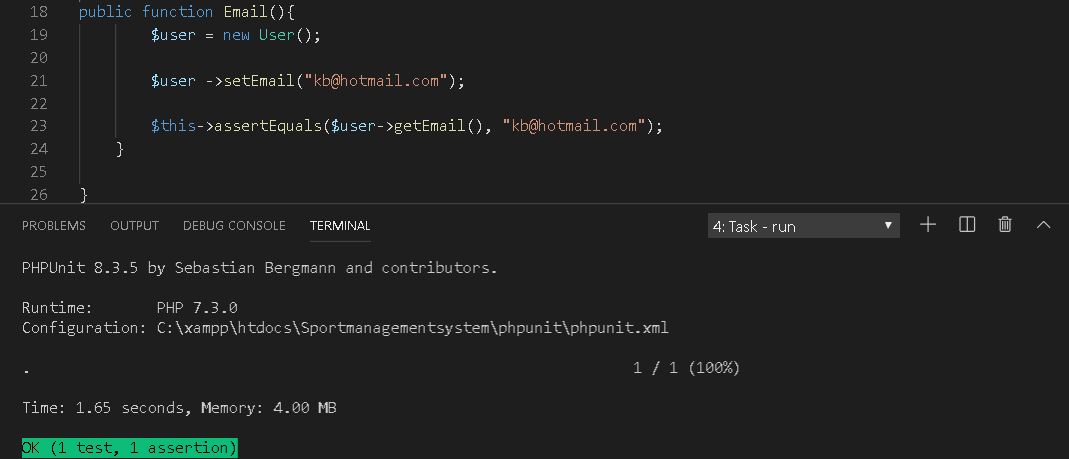


Figure : testing of setting and getting email

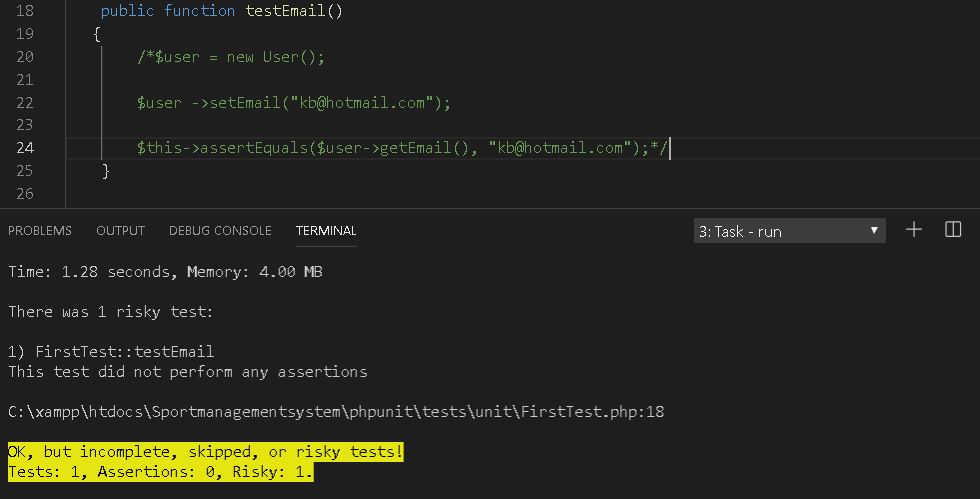


Figure : testing of email fail

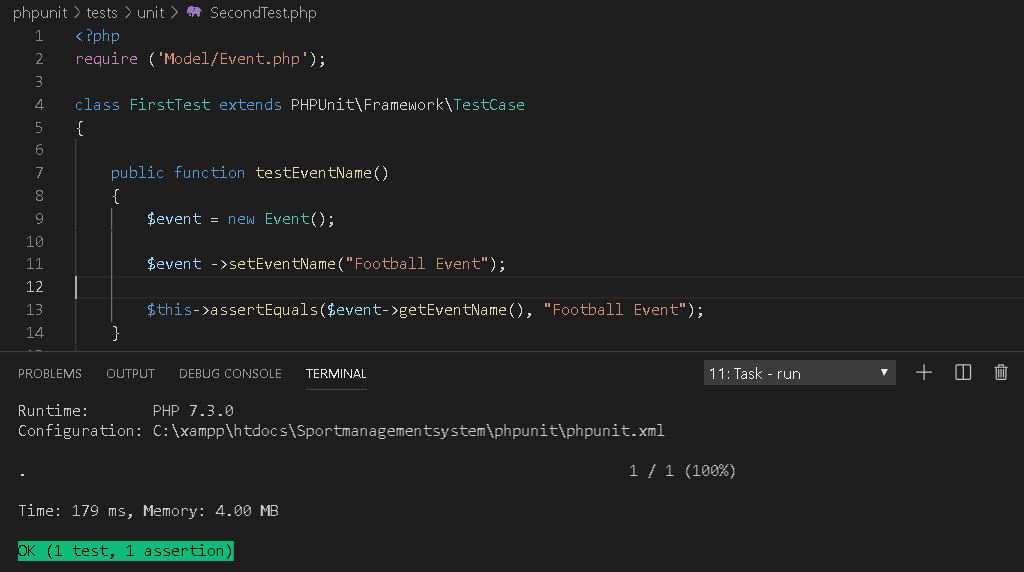


Figure : testing of setting and getting event name

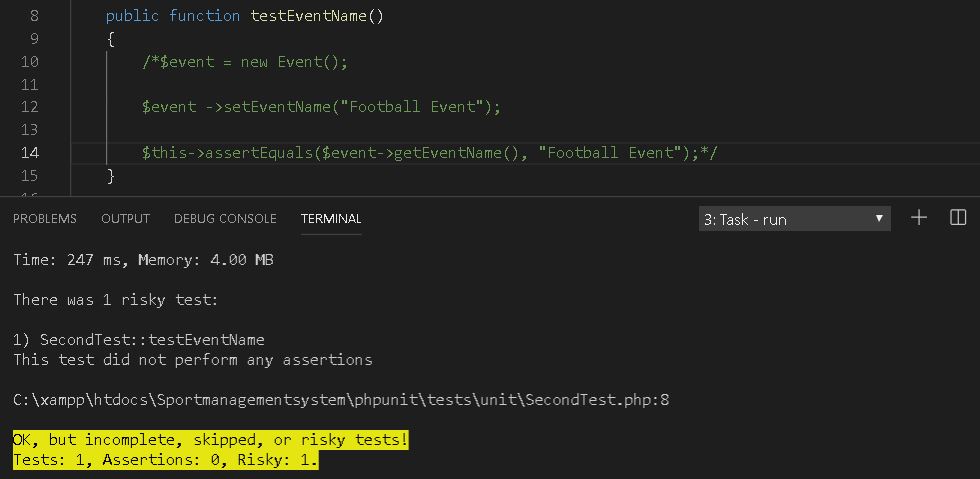


Figure : testing of event name fail