# **Chapter 5: Testing**

Testing is the process of evaluating the functionality of the software in order to check whether developed software met the actual requirement or not (Tutorial point, 2019). It identifies defects and errors made during development phase and ensures that the system is defect free. Also, it measures the quality of the product developed.

And testing is done in order to make sure customer’s reliability and their satisfaction with our developed software.

There are different types of testing to make sure it works as expected. Black box testing, Unit testing, White box testing, Acceptance testing etc. are types of testing. And among them I have performed **Unit testing** and **Black box testing** with test planning.

Test case or Test plan: It is the documentation of specific test with objectives, resource and process of testing.

## **Unit Testing**

Unit testing is the process of isolating a section of source code and executing each individual unit of source code to determine its correctness (Guru99, 2019). It is performed during the development of application.

Advantages of unit testing;

* Fix bugs in early development cycle.
* Improves the quality of code.
* Reduces defects and bugs.
* Less costly.

# **Black box testing**

Black box testing is the process of examining the functionality of the system through user interface without knowing the working of internal structure (Guru99, 2019). It only focuses on inputs and outputs of the software system.

Advantages of black box testing;

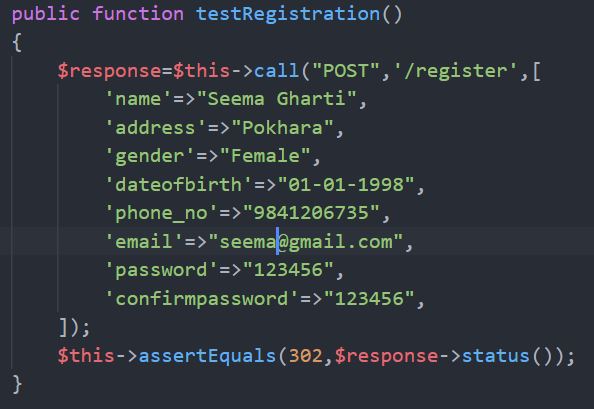
* Validates the functional requirement.
* Test can be done through end user’s point of view.
* No need of technical person for testing.

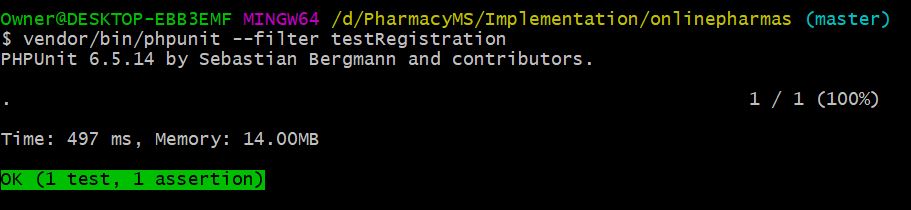
Here, 20 unit and black box testing with its test case for Medino pharmacy is show below;

1. User registration

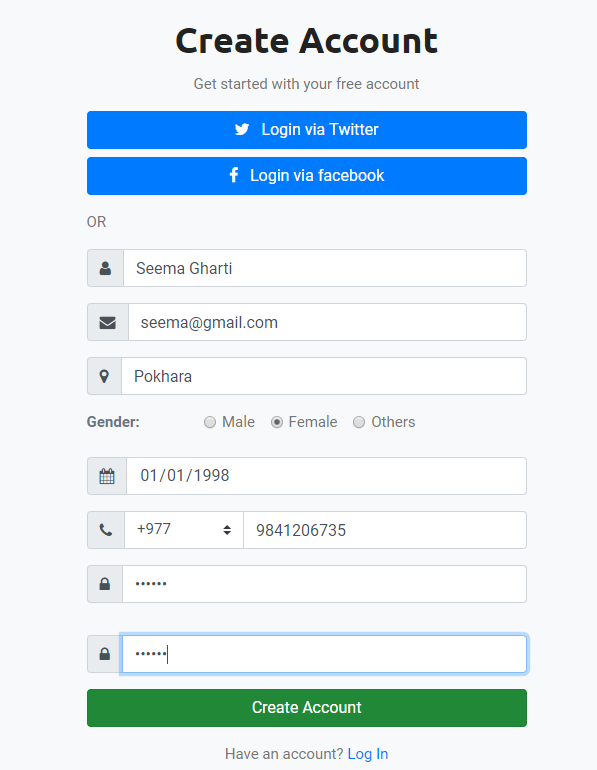
|  |  |
| --- | --- |
| **Test case no.** | **T1** |
| Purpose of the case | Checking whether user can register or not. |
| Test data | name: Seema Gharti  address: Pokhara  gender: Female  dateofbirth:01-01-1998  phone\_no: 9841206735  email: [seema@gmail.com](mailto:seema@gmail.com)  password: 123456  confirmpassword: 123456 |
| Class name | pharmacyTest |
| Function name | testRegistration |
| Expected result | Successful registration |
| Actual result | User added successfully |
| Conclusion | **Yes,** expected result matched with actual result. |

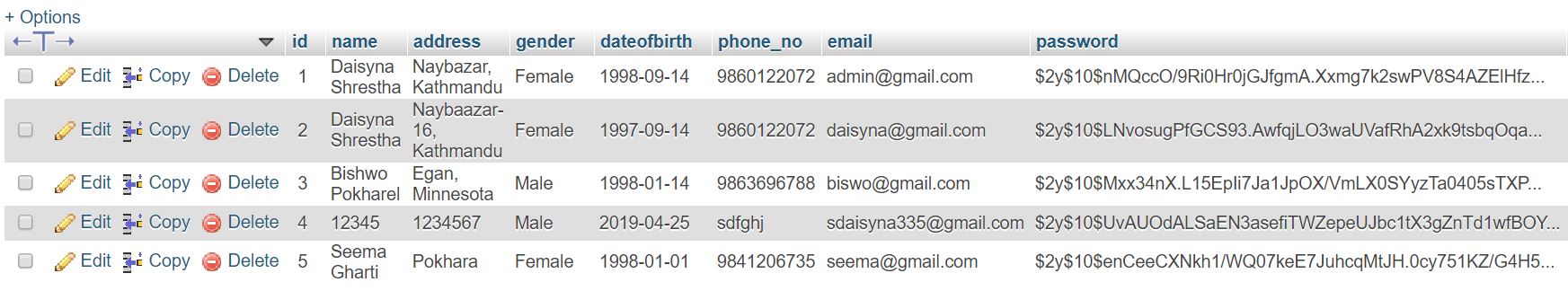
Unit testing:





Black box Testing:



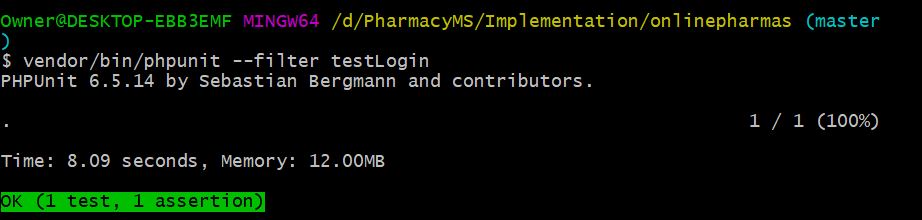


New user successfully added.

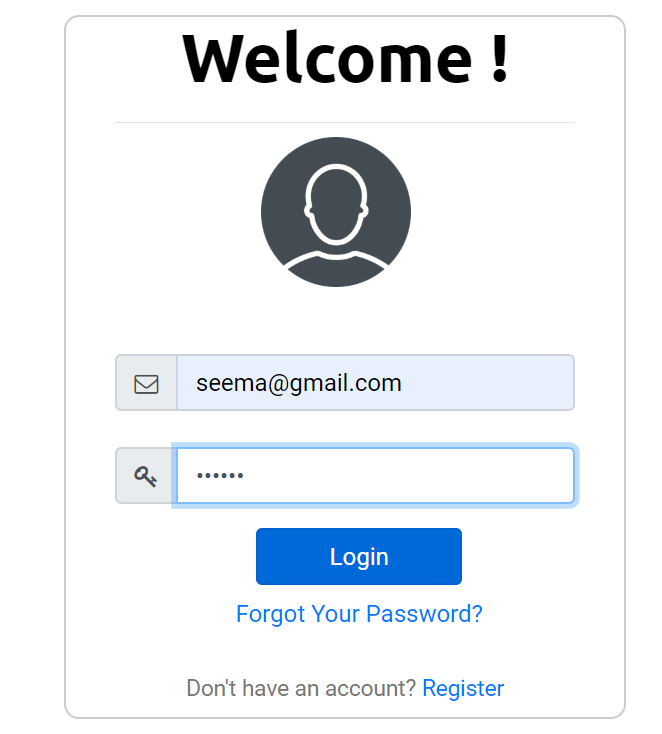
1. Login

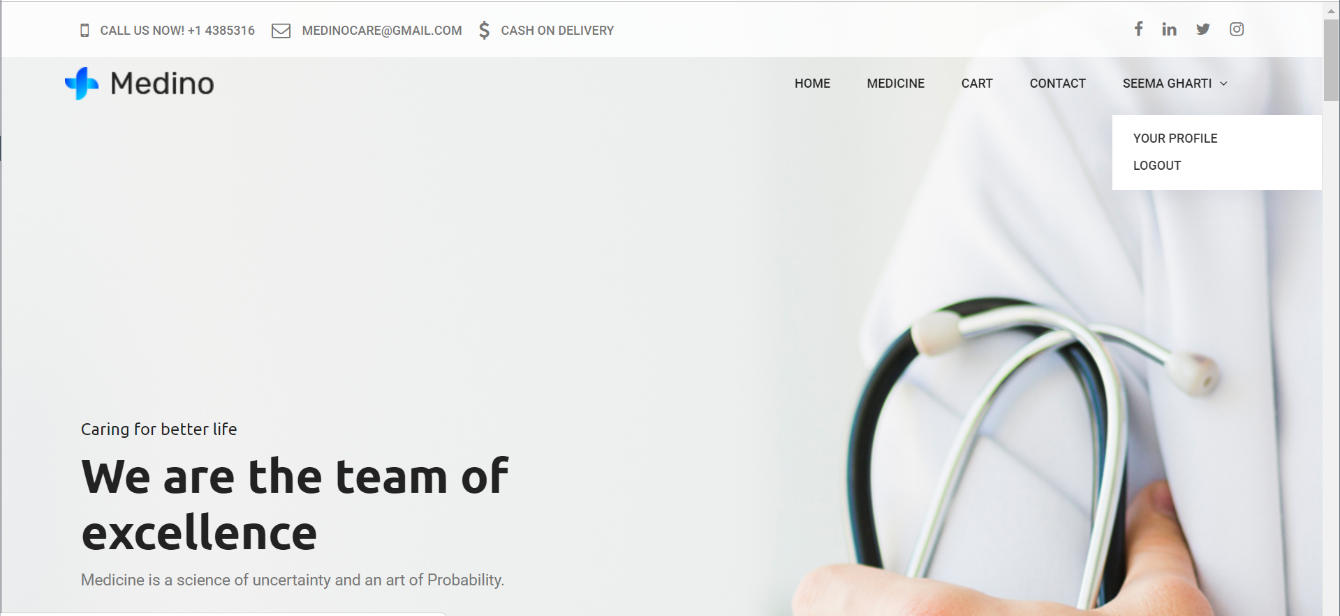
|  |  |
| --- | --- |
| **Test case** | **T2** |
| Purpose of the case | Login to the system by retrieving email and password. |
| Test data | email: seema@gmail.com  password: 123456 |
| Class name | pharmacyTest |
| Function name | testLogin |
| Expected result | User login to the system |
| Actual result | Successful login |
| Outcome as Expected | **Yes,** expected result matched with actual result. |

 Unit Testing:



Black box Testing:



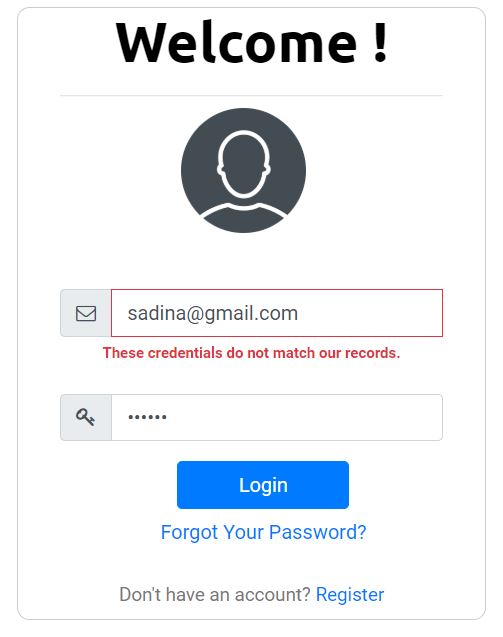


User logged in successfully.

1. Invalid user

|  |  |
| --- | --- |
| **Test case no.** | **T3** |
| Purpose of the case | To check whether invalid user can login or not. |
| Test data | email: sadina@gmail.com  password: sadina |
| Expected result | Throw error message. |
| Actual result | Error message shown successfully. |
| Conclusion | **Yes,** expected result matched with actual result. |

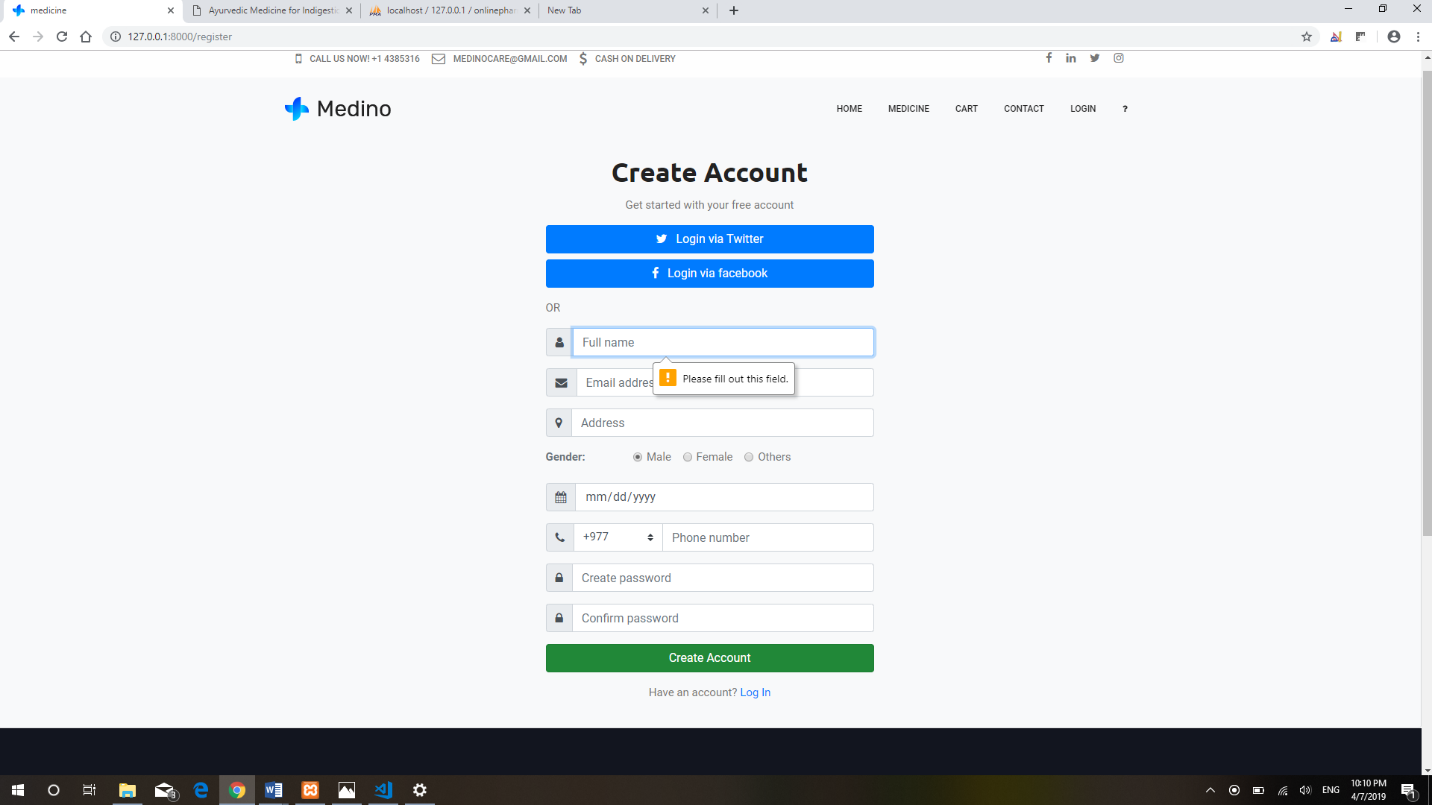
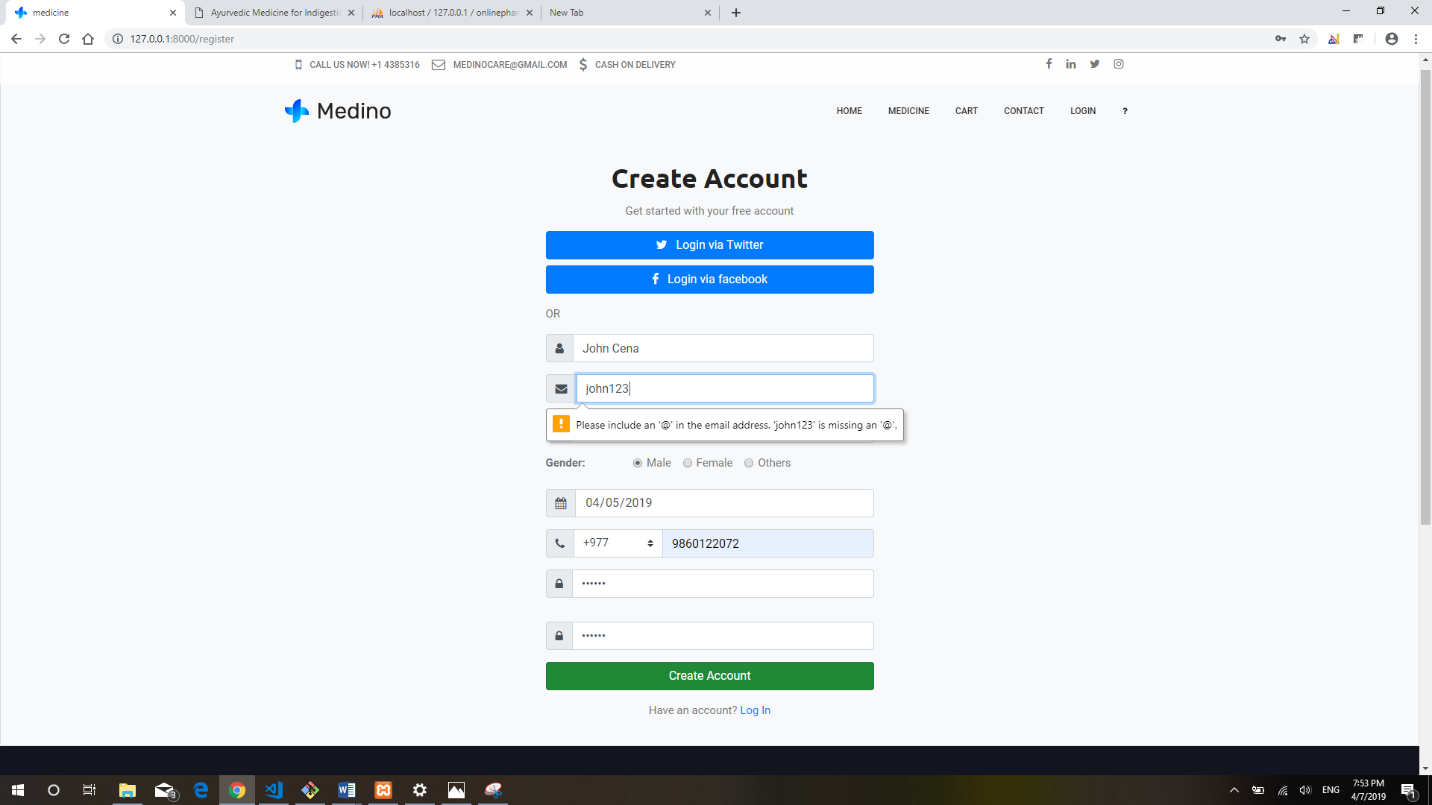
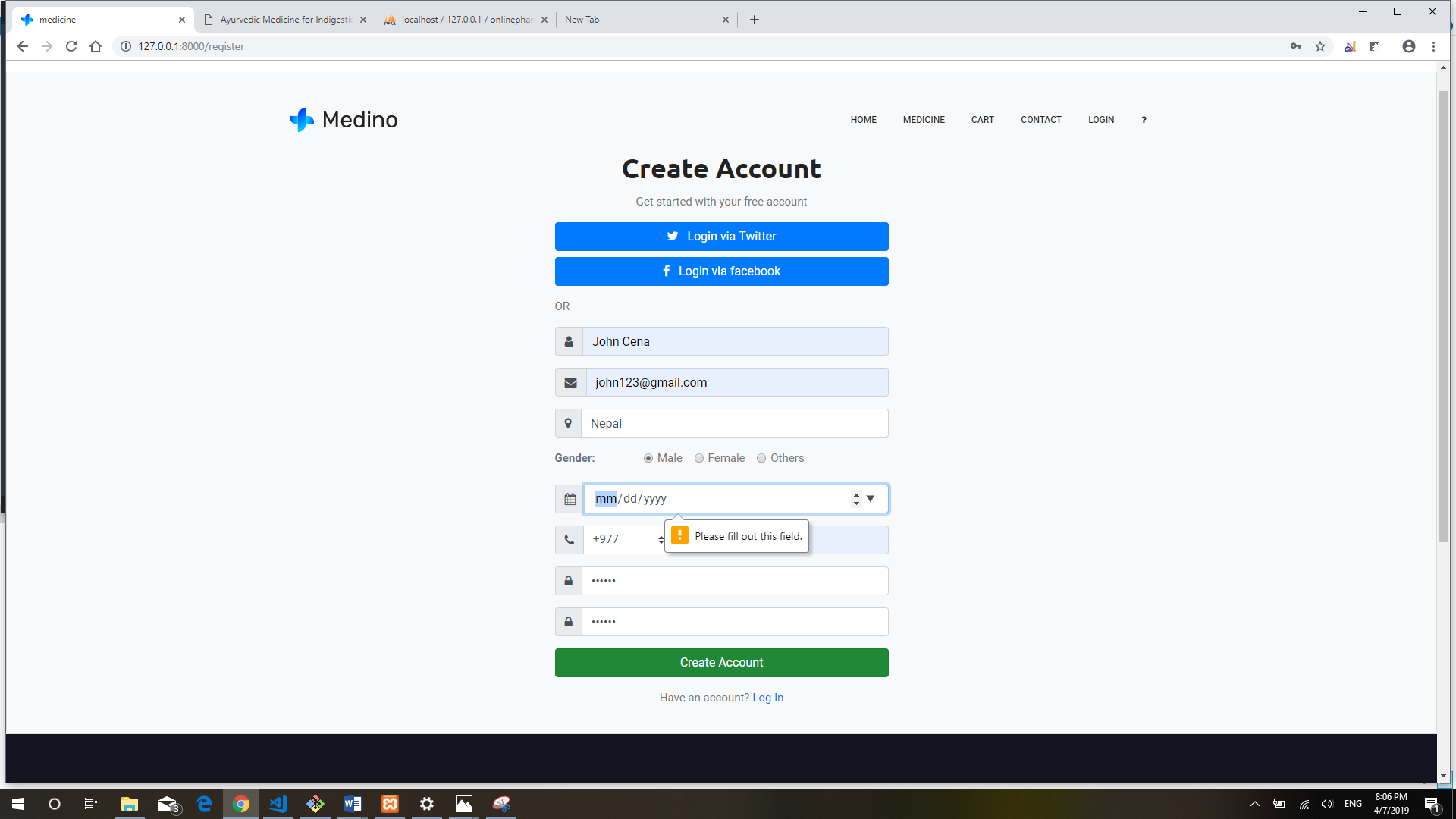
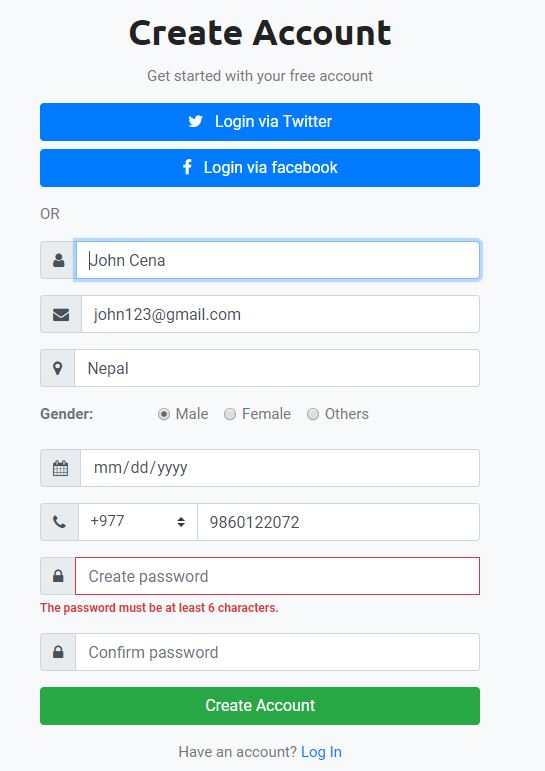
Black box Testing:

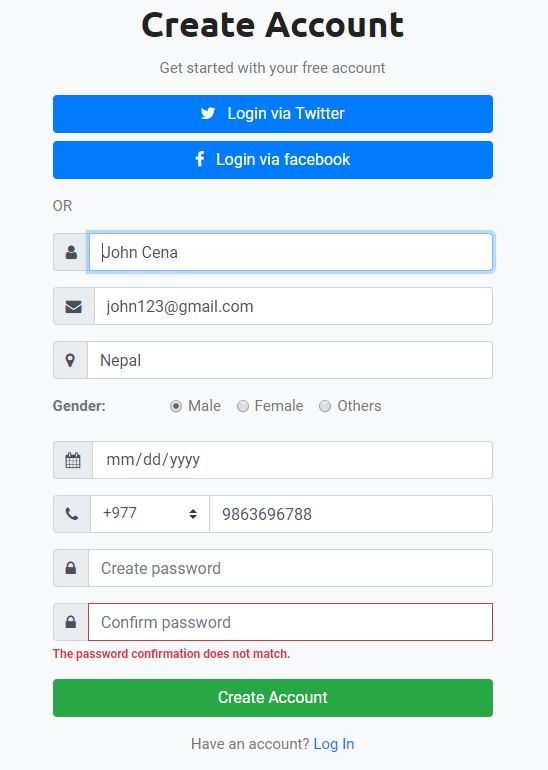


1. Validation of registration form

Black box Testing:

|  |  |
| --- | --- |
| **Test case no.** | **T4** |
| Purpose of the case | To test validation of registration page. |
| Test task | 1. User tries to register with empty field. 2. User doesn’t provide correct email format. 3. User tries to register with one field empty. 4. User enter less than 6-character password. 5. User enter different password and confirm password. |
| Expected result | 1. Throw error message “Please fill out this field”. 2. Throw error message “Please include an @ in email”. 3. Throw error message “Please fill out this field”. 4. Throw error message. 5. Throw error message. |
| Actual result | Error message shown successfully. |
| Conclusion | **Yes,** expected result matched with actual result. |

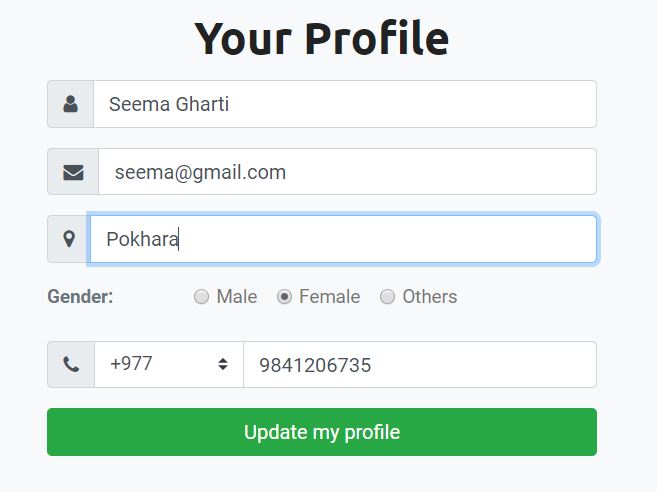


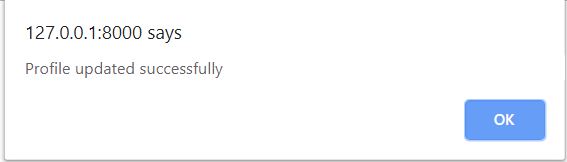


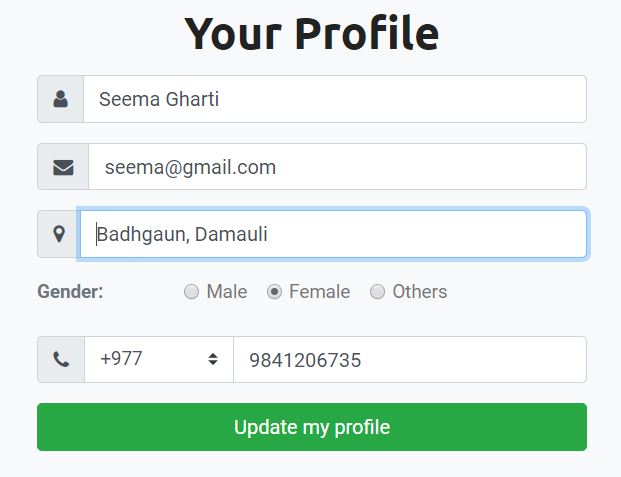
1. Edit profile

|  |  |
| --- | --- |
| **Test case no.** | **T5** |
| Purpose of test case | To check user can update their profile or not. |
| Test task | User change their profile details. |
| Expected result | Throw success message and update data in form. |
| Actual result | Successful message shown and profile updated. |
| Conclusion | **Yes,** expected result matched with actual result. |

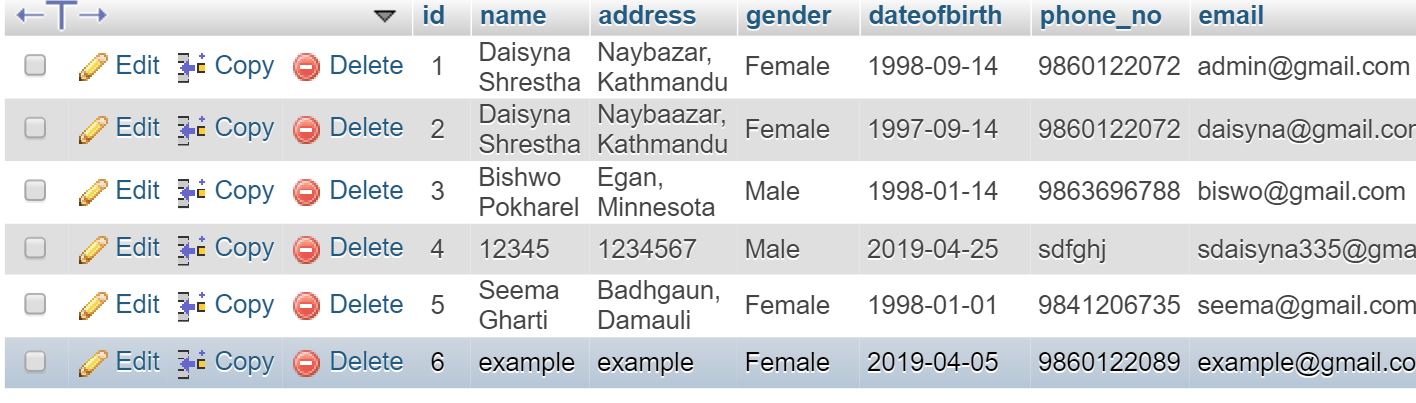
Black box testing:







Address updated.

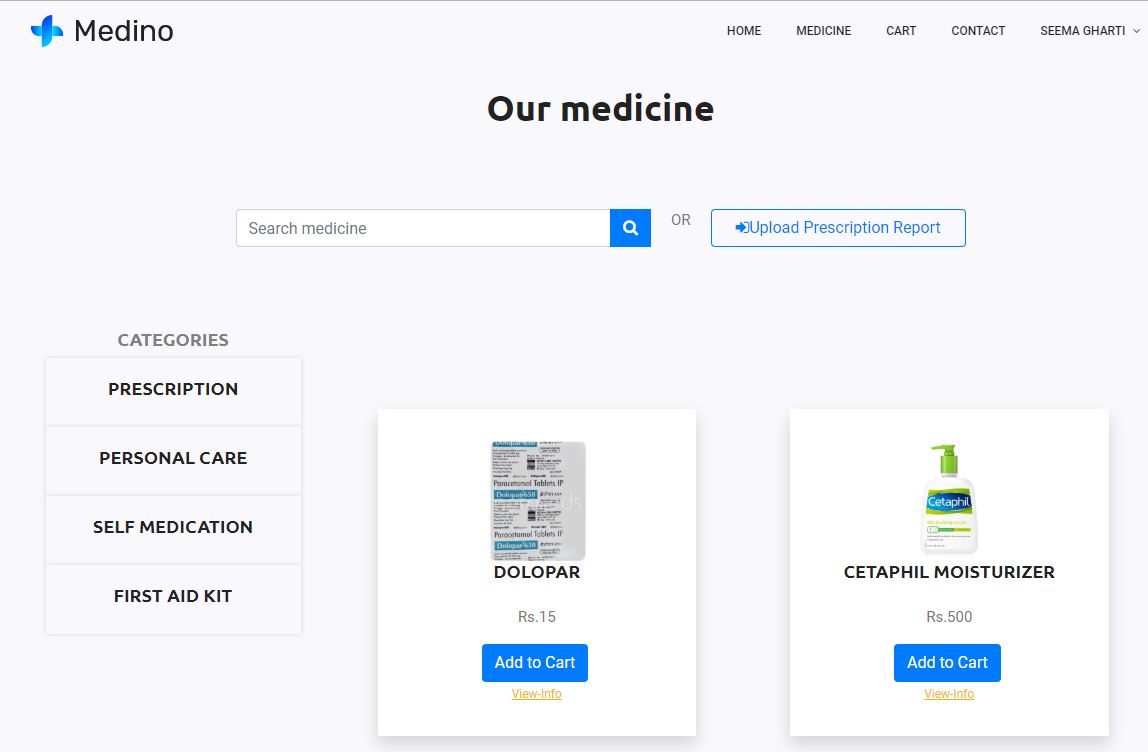


Data updated in database.

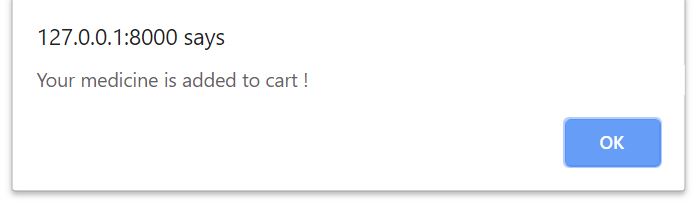
1. Add to cart

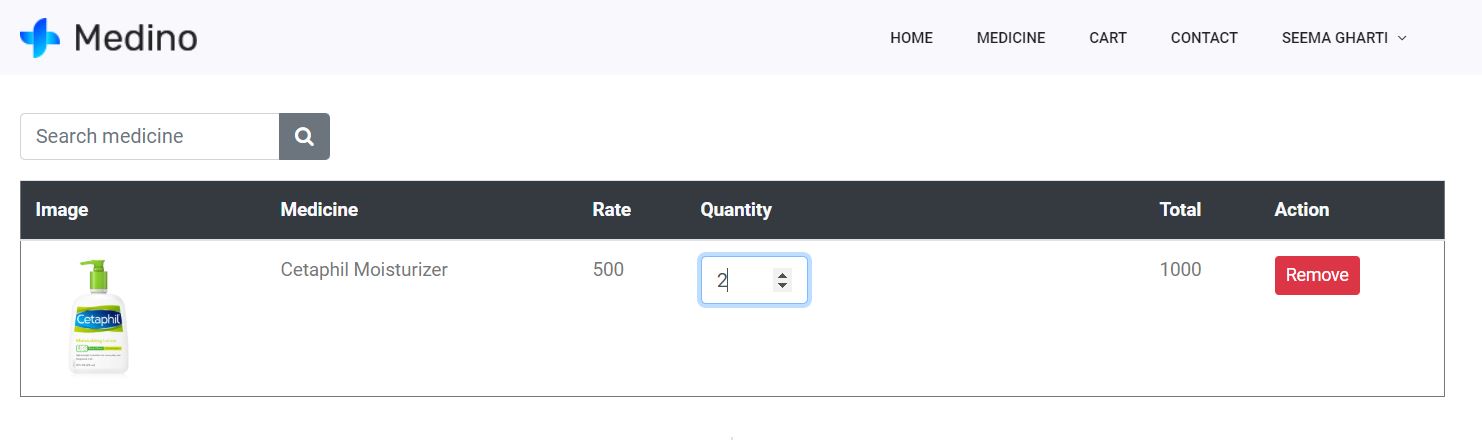
|  |  |
| --- | --- |
| **Test case no.** | **T6** |
| Purpose of test case | To check add to cart by user. |
| Test task | Addition of medicine in user’s respective cart. |
| Expected result | Throw success message and medicine must add to user cart. |
| Actual result | Successful message shown and medicine is successfully added in cart. |
| Conclusion | **Yes,** expected result matched with actual result. |

Black box Testing:



This product is added to Seema’s cart.

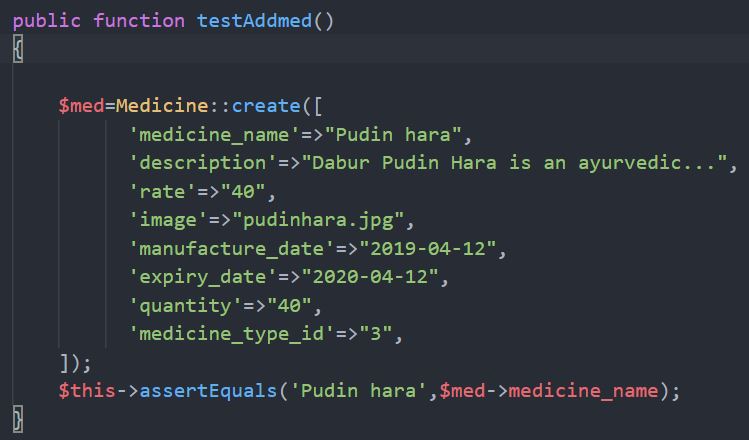


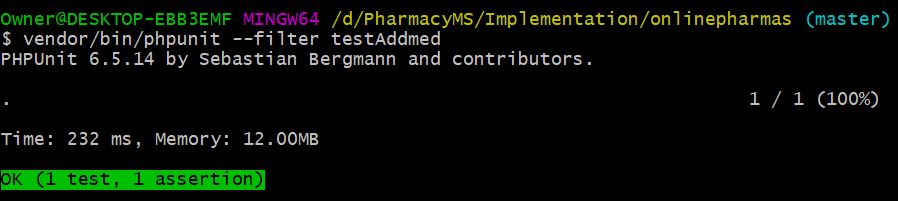


1. Add medicine

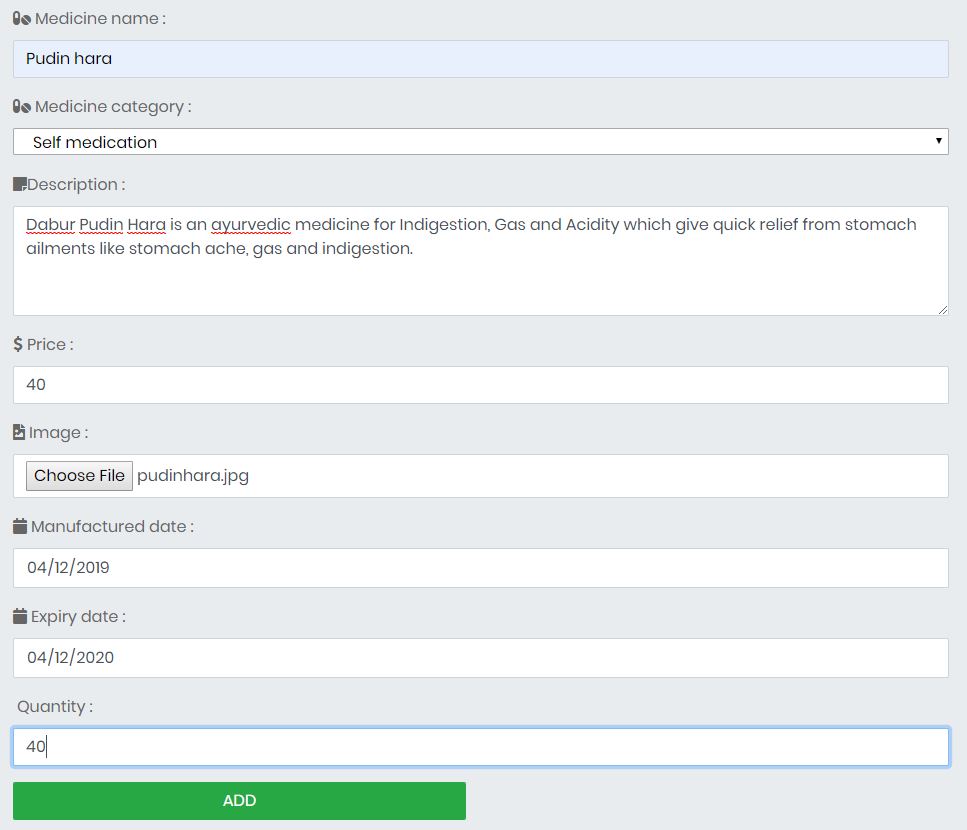
|  |  |
| --- | --- |
| **Test case no.** | **T7** |
| Purpose of test case | To check add medicine. |
| Test data | Medicine name: Pudin hara  Medicine category: Self medication  Description: Dabur Pudin hara is auyurvedic……  Price: 40  Image: pudinhara.jpg  Manufactured date: 04-12-2019  Expiry date: 04-12-2020  Quantity: 40 |
| Class name | pharmacyTest |
| Function name | testAddmed |
| Expected result | Throw successful message and medicine added in table. |
| Actual result | Successful message shown and medicine added. |
| Conclusion | **Yes,** expected result matched with actual result. |

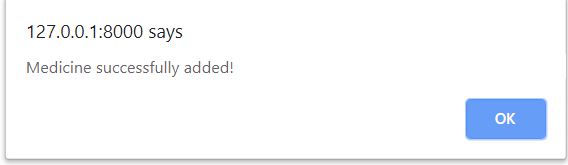
Unit Testing:

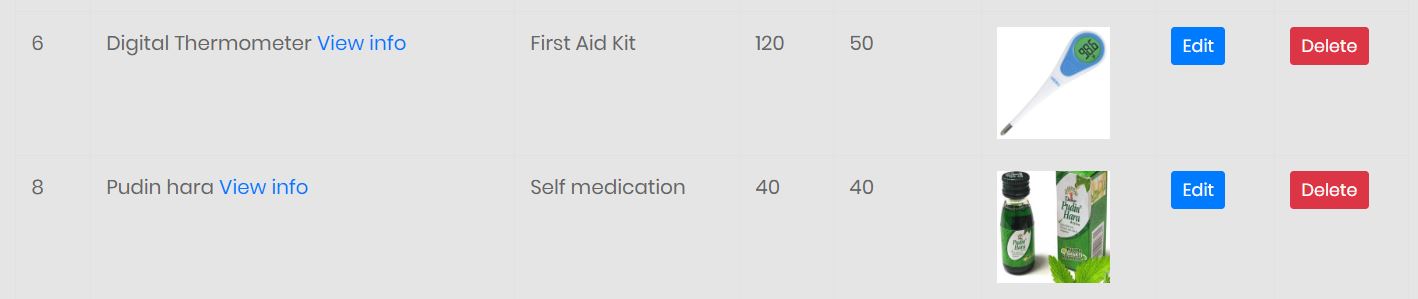




Black box Testing:



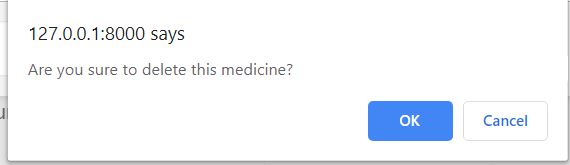
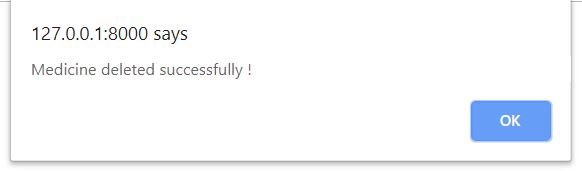


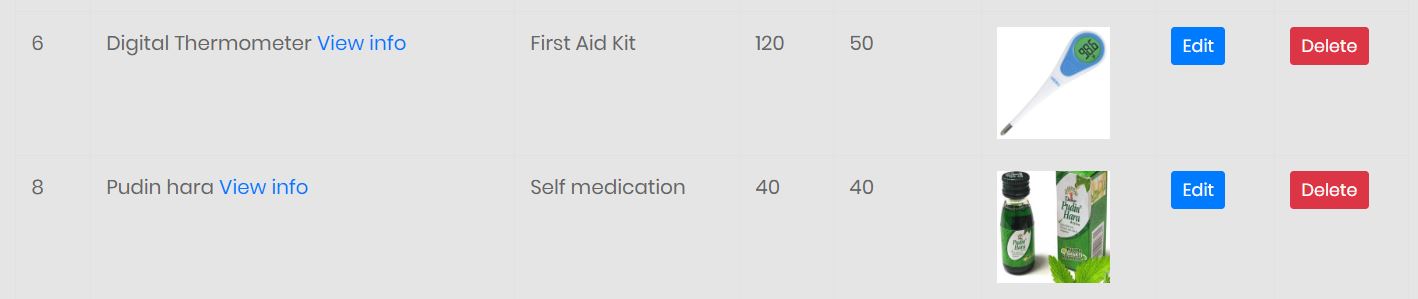


Pudin hara added successfully.

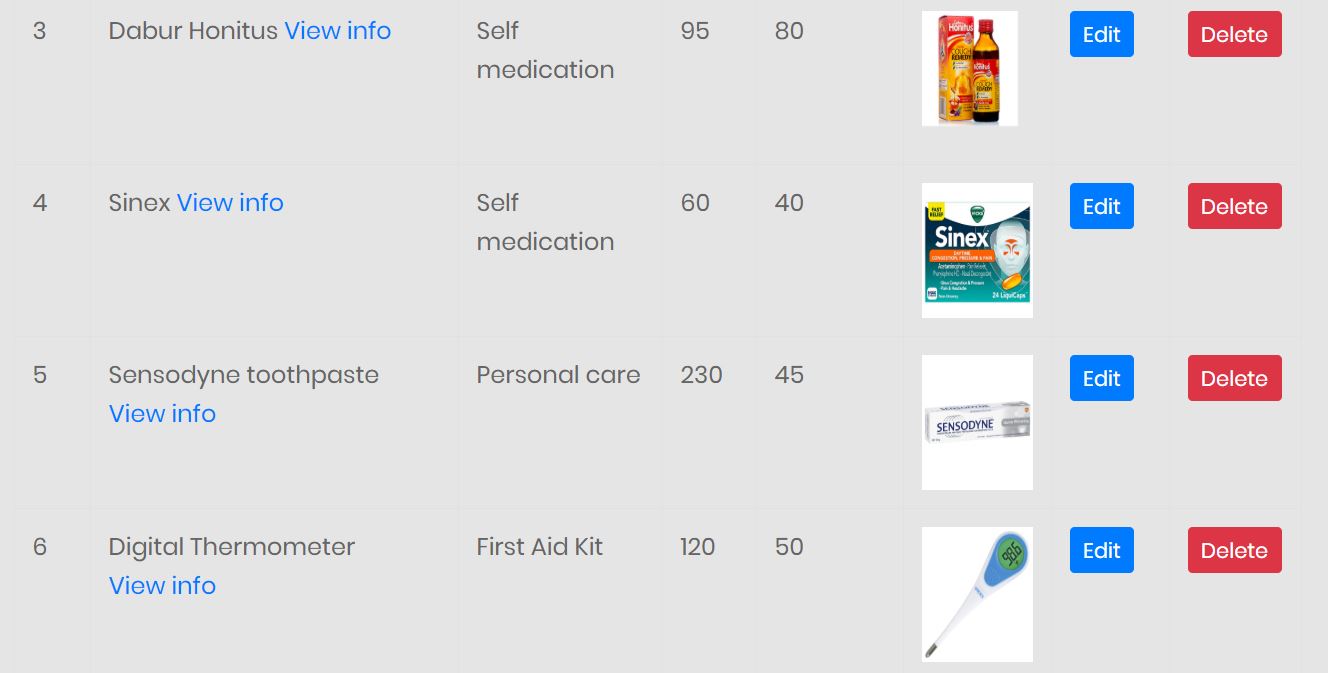
1. Delete medicine

|  |  |
| --- | --- |
| **Test case no.** | **T8** |
| Purpose of test case | To check medicine deletion |
| Test data | medicine\_id: 8 |
| Expected result | Throw confirmation message and success message. |
| Actual result | Successful message shown with deletion of medicine. |
| Conclusion | **Yes,** expected result matched with actual result. |





Click delete button to delete product

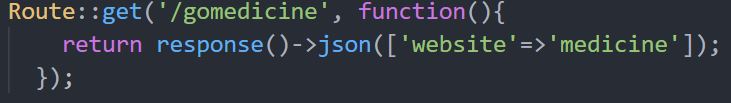


No data of Pudin hara

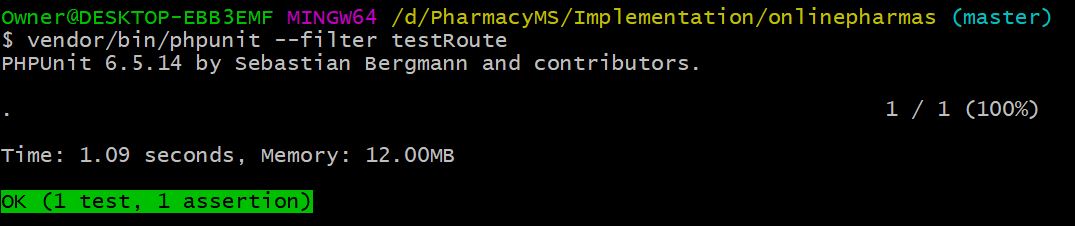
1. Route

|  |  |
| --- | --- |
| **Test case no.** | **T9** |
| Purpose of the case | To check response from the original route. |
| Test data | Route: ‘/gomedicine’ |
| Class name | pharmacyTest |
| Function name | testRoute |
| Expected result | Return same array response from origin route. |
| Actual result | Returning same array response. |
| Conclusion | **Yes,** expected result matched with actual result. |

Unit Testing:



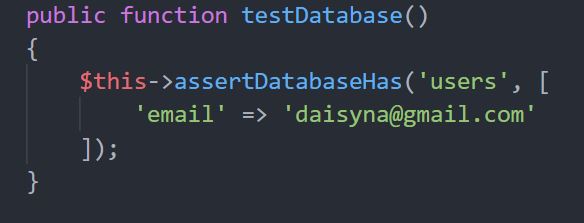


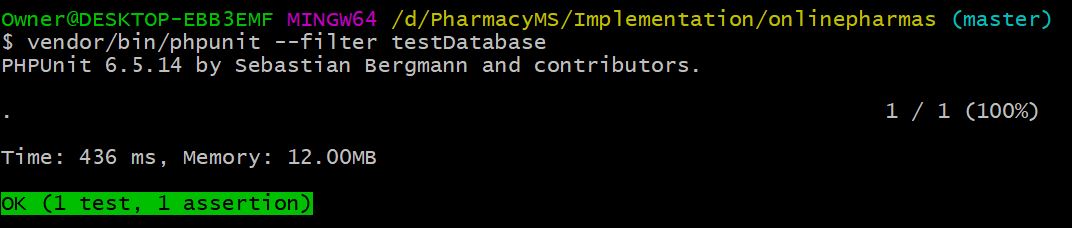


1. Database

|  |  |
| --- | --- |
| **Test case no.** | **T10** |
| Purpose of the case | To check whether data exists in database or not. |
| Test data | table: users  email: daisyna@gmail.com |
| Class name | pharmacyTest |
| Function name | testDatabase |
| Expected result | Data passed through function should match the data stored in database. |
| Actual result | Successful data matching. |
| Conclusion | **Yes,** expectedresult matched with actual result. |

Unit Testing:

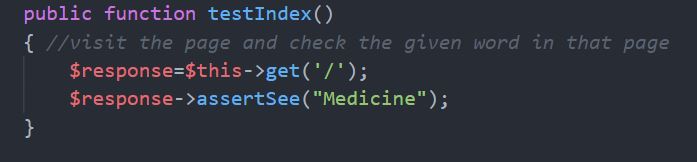


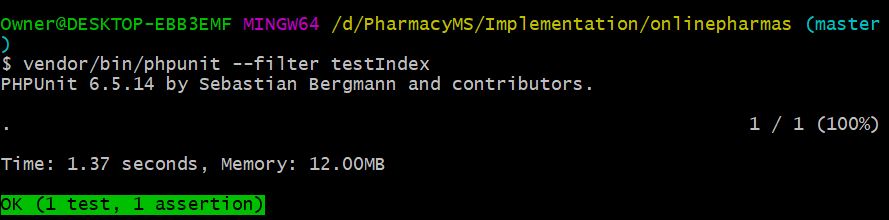


1. Visit page

|  |  |
| --- | --- |
| **Test case** | **T11** |
| Purpose of the case | To visit the page and check the given word in that page. |
| Test data | Route: /  Keyword: Medicine |
| Class name | pharmacyTest |
| Function name | testIndex |
| Expected result | Keyword passed through function should match the keyword in that page. |
| Actual result | Successful visiting and checking passed keyword. |
| Conclusion | **Yes**, expected result matched with actual result. |

Unit Testing:



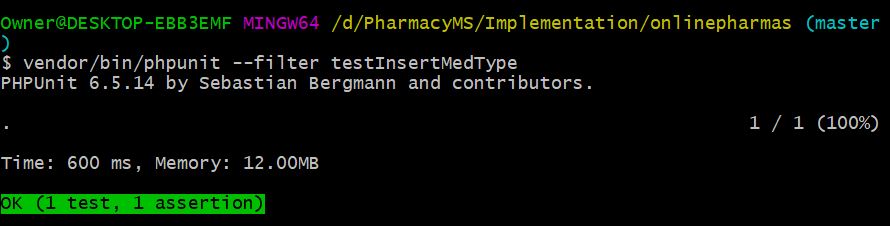


1. Add medicine type

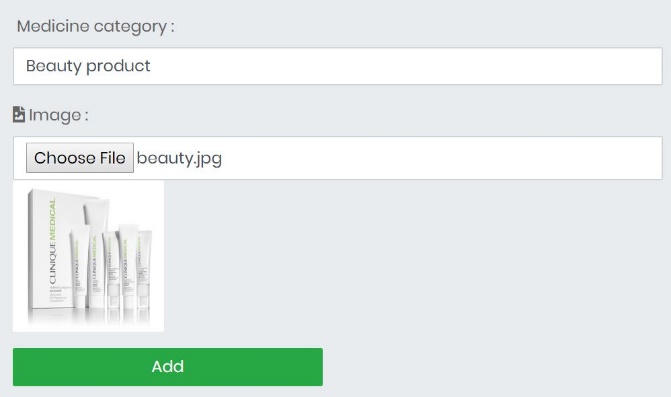
|  |  |
| --- | --- |
| **Test case no.** | **T12** |
| Purpose of the case | To check add medicine type. |
| Test data | Medicine\_type\_name: Beauty Product |
| Class name | pharmacyTest |
| Function name | testInsertMedType |
| Expected result | Throw success message and category should add in table. |
| Actual result | Successful message shown and added in table. |
| Conclusion | **Yes,** expected result matched with actual result. |

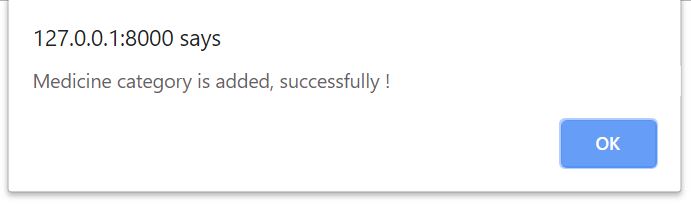
Unit Testing:

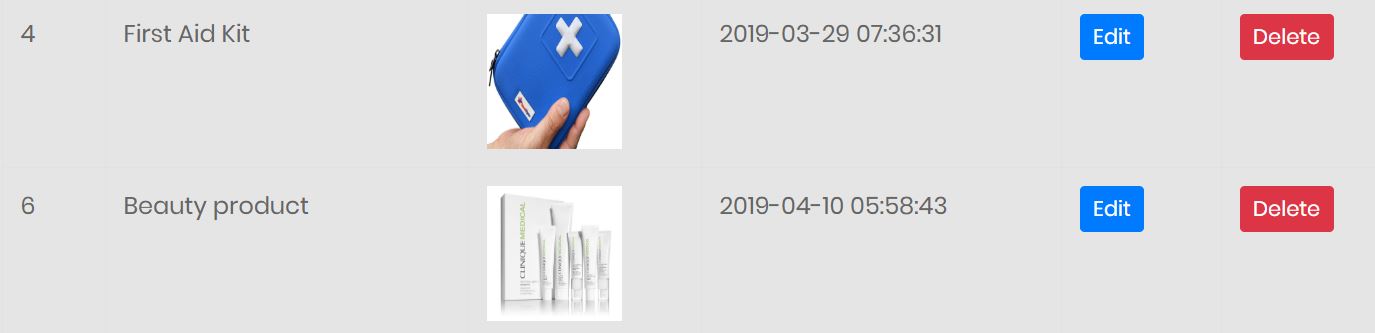




Black box Testing:





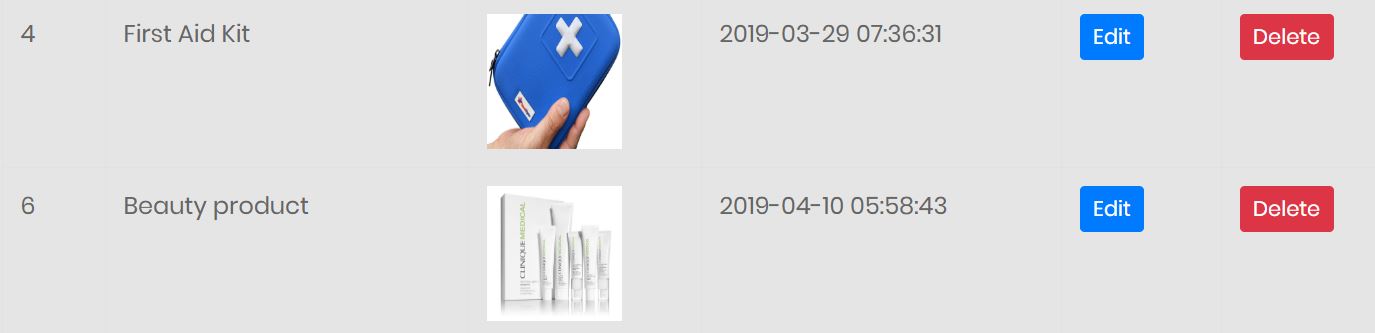


Medicine category is added successfully!

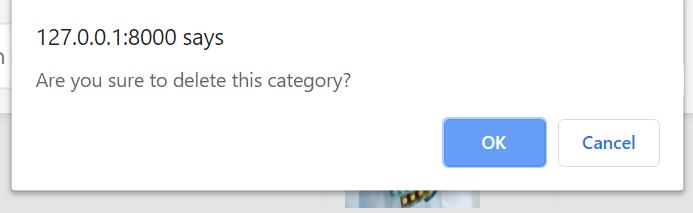
1. Delete medicine category

|  |  |
| --- | --- |
| **Test case no.** | **T13** |
| Purpose of the case | To check medicine type deletion. |
| Test data | medicine\_type\_id:6 |
| Class name | pharmacyTest |
| Function name |  |
| Expected result | Throw confirmation message and category should be deleted from database. |
| Actual result | Successful deletion. |
| Conclusion | **Yes,** expected result matched with actual result. |

Black box Testing:



Click delete button to delete category.



Click OK to confirm deletion



No data of Beauty product in database.