

**IT314 Software Engineering**

**Group 25**

|  |  |
| --- | --- |
| **Name** | **Student Id** |
| TEJABWALA MOHAMMAD JABIRBHAI | 202001406 |
| PRARTHEE BHAVINBHAI DESAI | 202001257 |
| AASTHA JAGDISH SHETTY | 202001260 |
| PATEL NISARG NAGINBHAI | 202001436 |
| DETROJA ARTH JITESHBHAI | 202001274 |
| BHUVA MEHULKUMAR VISABHAI | 202001437 |
| RUSHABH MAHESH PATEL | 202001419 |
| JAIMIN RATHWA (GR) | 202001423 |
| GANVIT GAURANG SHANKARBHAI | 202001247 |
| DAMOR ANIRUDDH RATANBHAI | 202001255 |

# Lab8

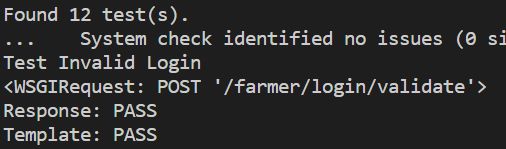
Methodology:

Here we have used the in-built Django testing methods and the following modules to test the code

Pymongo, requests, MongoClient, Client, TestCase, sys, os

If it is able to render the page or able to retrieve or save the information correctly and display appropriate information the test case is said to be passed else it is said to be failed.

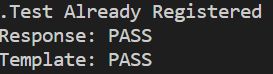
Testing for Farmer:



Cases:

1. Already\_registered
2. def testAlreadyRegistered(self):
3. print('Test Already Registered')
4. response = self.client.post('/farmer/register/entry', {
5. 'first\_name': 'test',
6. 'last\_name': 'test',
7. 'email': 'test1@gmail.com',
8. 'password': 'testPass',
9. 'phone\_number': '8488887253'
10. })
11. assertEqual(response.status\_code, 200)
12. assertTemplateUsed(response, 'f-register.html')

Output:



2. invalid\_password

    def testInvalidPassword1(self):

        print('Test Invalid Password 1')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'test',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword2(self):

        print('Test Invalid Password 2')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': '',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword3(self):

        print('Test Invalid Password 3')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'testPasssample',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword4(self):

        print('Test Invalid Password 4')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'afad',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword5(self):

        print('Test Invalid Password 5')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'tsafadf',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword6(self):

        print('Test Invalid Password 6')

        response = self.client.post('/farmer/register/entry',{

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'tefhgdh',

            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

    def testInvalidPassword7(self):

        print('Test Invalid Password 7')

        response = self.client.post('/farmer/register/entry', {

            'first\_name': 'test',

            'last\_name': 'test',

            'email': 'test1@gmail.com',

            'password': 'testPass141342312',

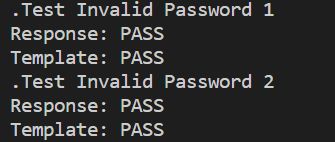
            'phone\_number': '8488887253'

        })

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-register.html')

Output:



3. unverified\_login

    def testUnverifiedLogin(self):

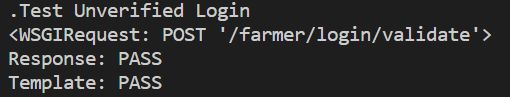
        print('Test Unverified Login')

        response = self.client.post('/farmer/login/validate', {'email': 'test2@gmail.com', 'password': 'testPass'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-login.html')

Output:



4. valid\_login

    def testValidlogin(self):

        print('Test Valid Login')

        response = self.client.post('/farmer/login/validate', {'email': 'test1@gmail.com', 'password': 'testPass'})

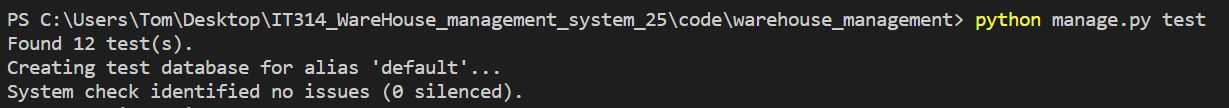
        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'f-home.html')

Output:

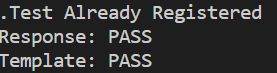


Testing for Warehouse:



1. Test\_already\_registered
2. def testAlreadyRegistered(self):
3. print('Test Already Registered')
4. response = self.client.post('/warehouse/register/entry', {
5. 'name': 'test',
6. 'latitude': '80',
7. 'longitude': '90',
8. 'storage\_capacity': '129',
9. 'email': 'test1@gmail.com',
10. 'password': '1234',
11. 'phoneNum': '8488887253'})

Output:



2. invalid\_login

    def testInvalidLogin(self):

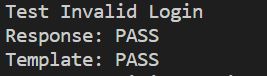
        print('Test Invalid Login')

        response = self.client.post('/warehouse/login/validate', {'email': 'test3@gmail.com', 'password': 'testPass'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-login.html')

Output:



3.invalid\_password

    def testInvalidPassword1(self):

        print('Test Invalid Password 1')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': '',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword2(self):

        print('Test Invalid Password 2')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'abcde123',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword3(self):

        print('Test Invalid Password 3')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'ABCDE123',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword4(self):

        print('Test Invalid Password 4')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'abcdEFGH1',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword5(self):

        print('Test Invalid Password 5')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'abcdefghi',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword6(self):

        print('Test Invalid Password 6')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'a@A1',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

    def testInvalidPassword7(self):

        print('Test Invalid Password 7')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

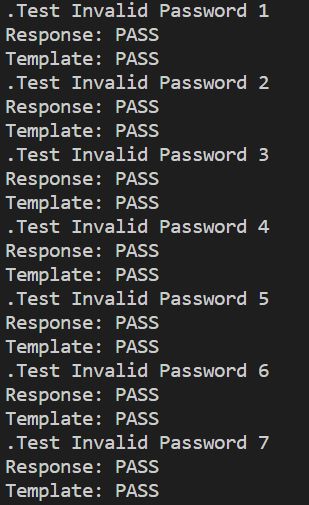
                                                        'password': 'abcde12345$678901',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-register.html')

Output:



4.unvarified\_login

    def testUnverifiedLogin(self):

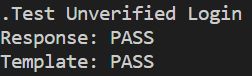
        print('Test Unverified Login')

        response = self.client.post('/warehouse/login/validate', {'email': 'test2@gmail.com', 'password': 'testPass'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-login.html')

Output:



5. valid\_login

    def testValidlogin(self):

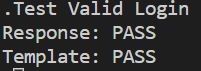
        print('Test Valid Login')

        response = self.client.post('/warehouse/login/validate', {'email': 'test1@gmail.com', 'password': 'testPass'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-home.html')

Output:



6. valid\_registration

    def testValidRegistration(self):

        print('Test Valid Registration')

        response = self.client.post('/warehouse/register/entry', {

                                                        'name': 'test',

                                                        'latitude': '80',

                                                        'longitude': '90',

                                                        'storage\_capacity': '129',

                                                        'email': 'test2@gmail.com',

                                                        'password': 'passTest1@',

                                                        'phoneNum': '8488887253'})

        assertEqual(response.status\_code, 200)

        assertTemplateUsed(response, 'w-login.html')

Output:

