

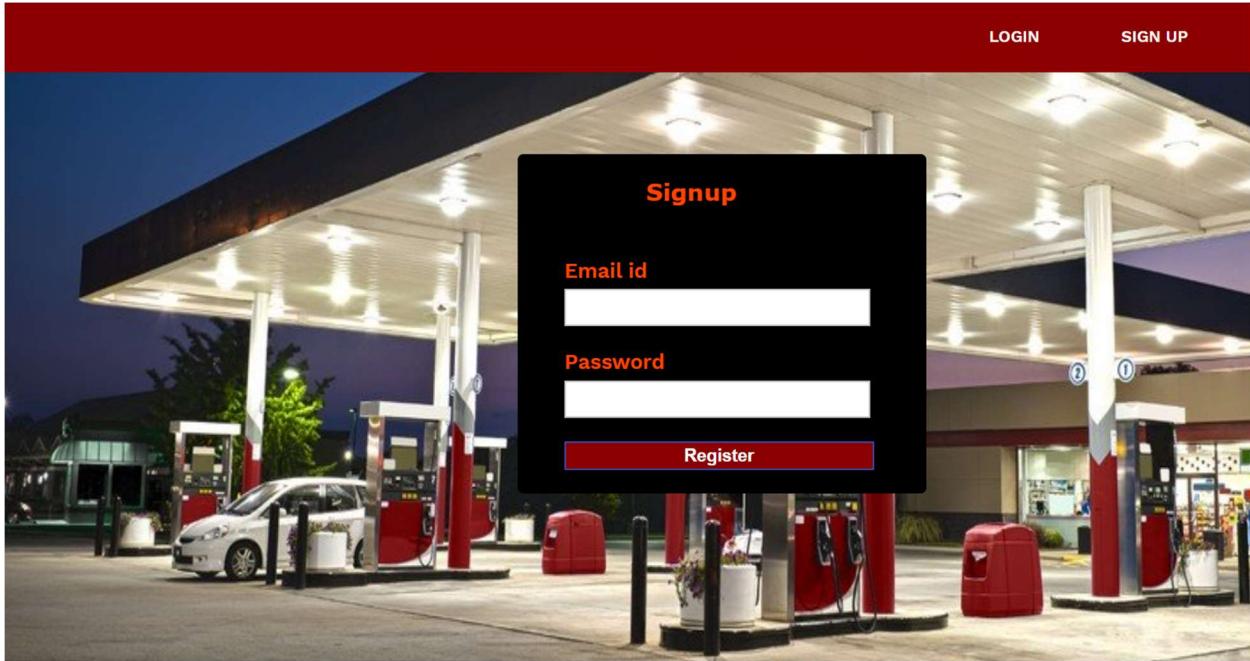
SOFTWARE DESIGN PROJECT REPORT

| | |
|---------------------------|---------|
| Sravanthi Pasumarthi | 1868536 |
| Sai Sri Varsha Gadde | 1885164 |
| Manisha Chowdary Paladugu | 1890737 |

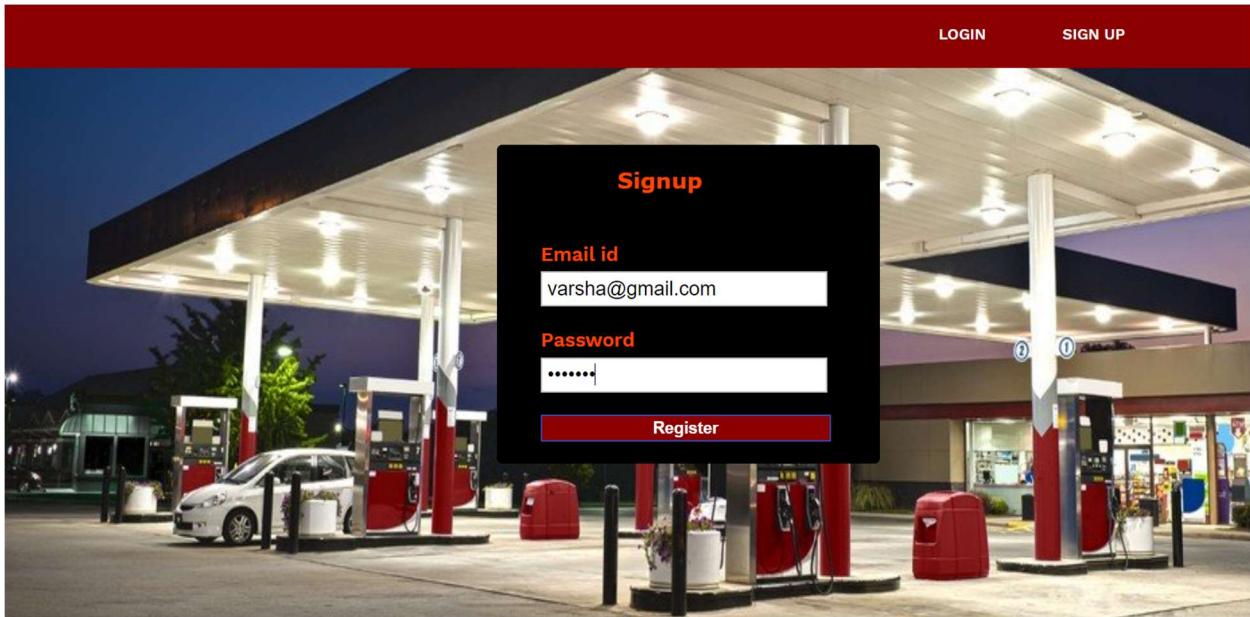
a) Fuel Quote Web Application end to end functionality.

When a new user first opens the Fuel quote web application, they need to first register/signup to access the application.

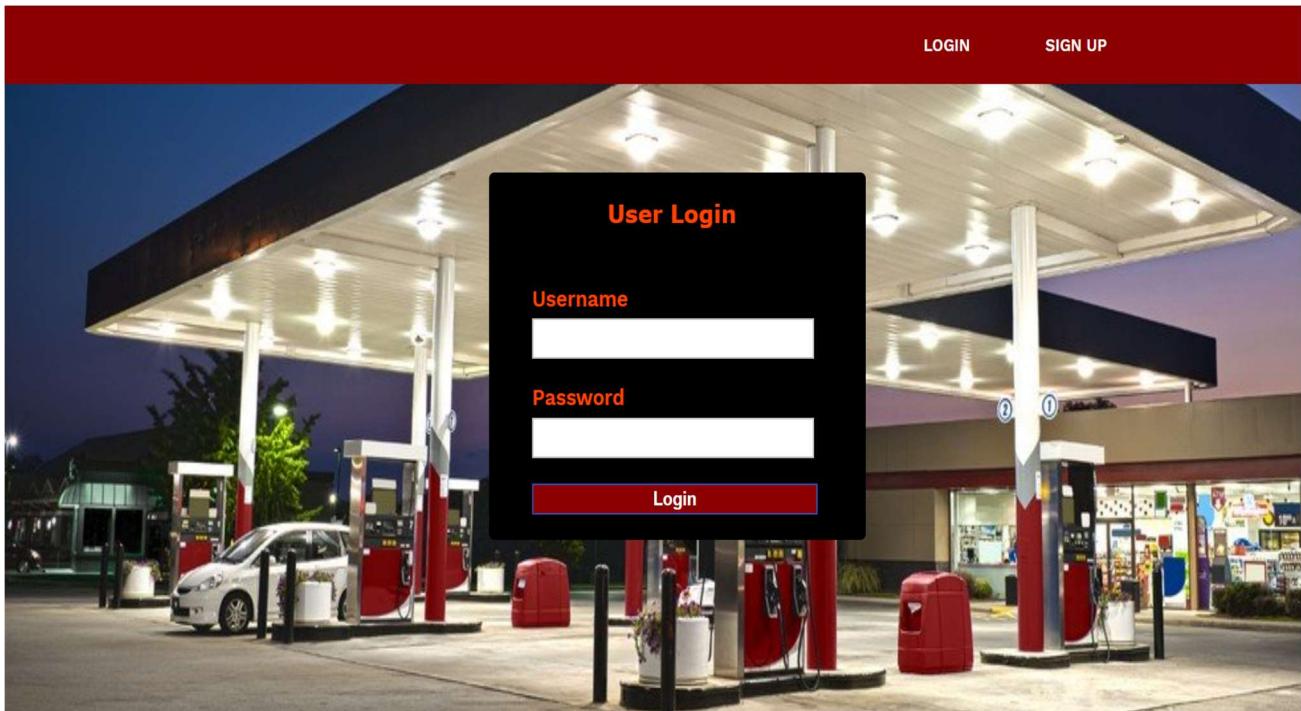
Signup Form:



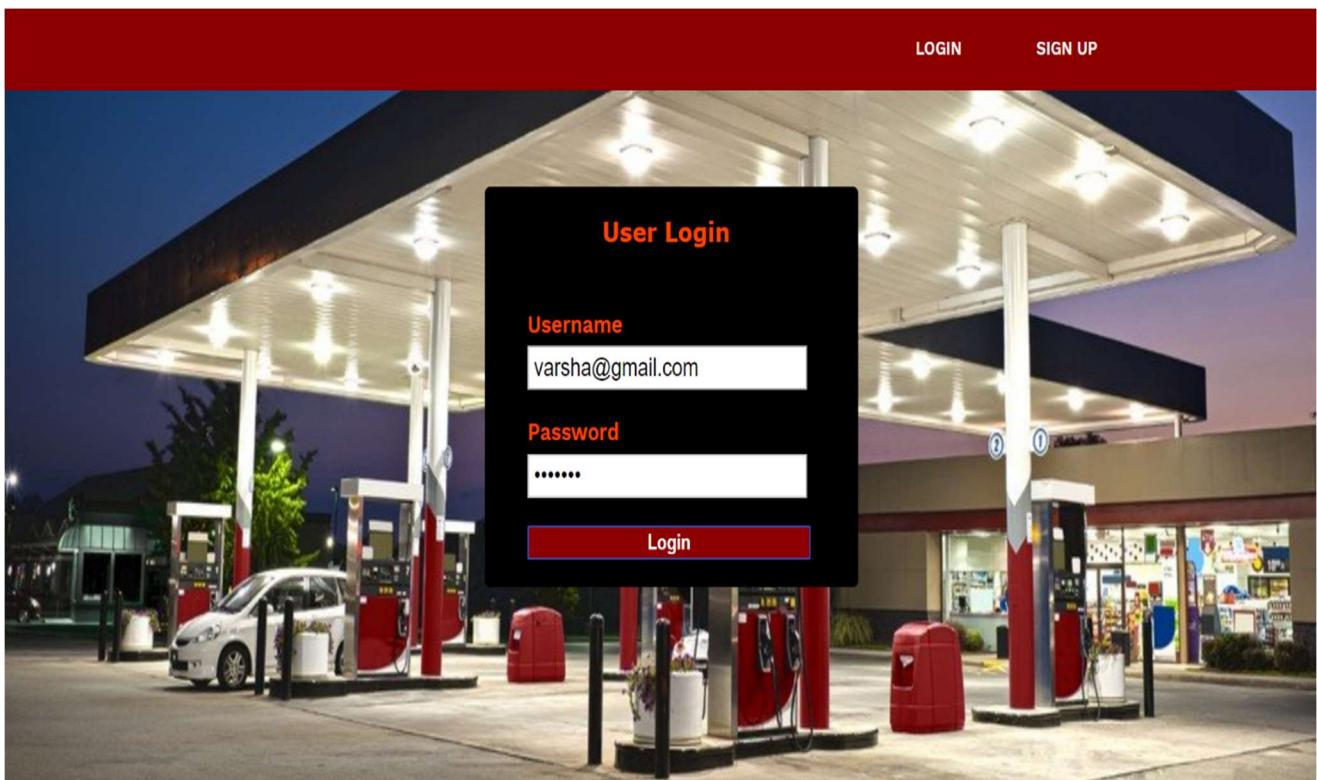
Then user gives both email id and password and user registers to the application by clicking on register.



After clicking on register, if all validations are passed, then navigates user to the Login Page.

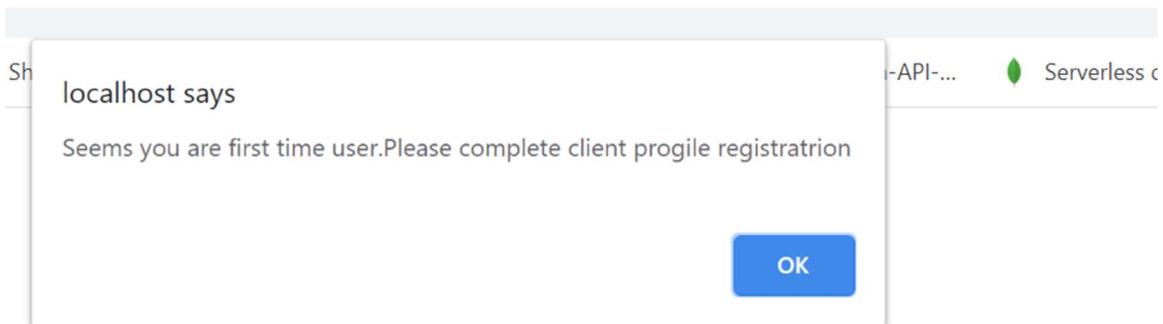


Now user login to the web application by giving his credentials and then clicks on login button.

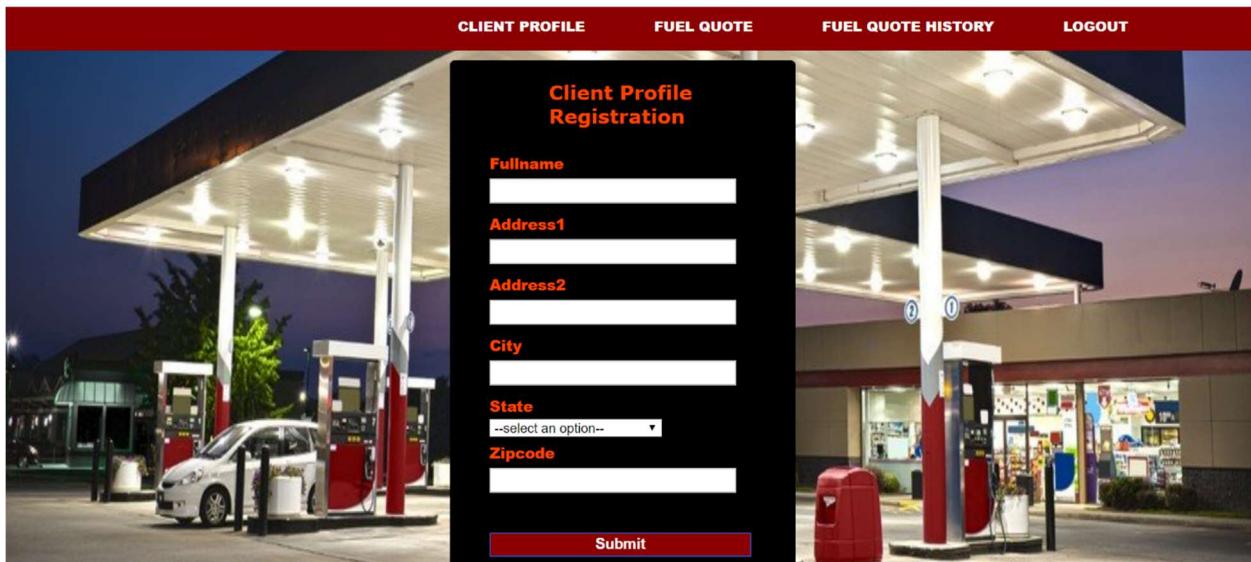


Then credentials are verified and redirects to an appropriate page depending upon whether user is new user or already existing user

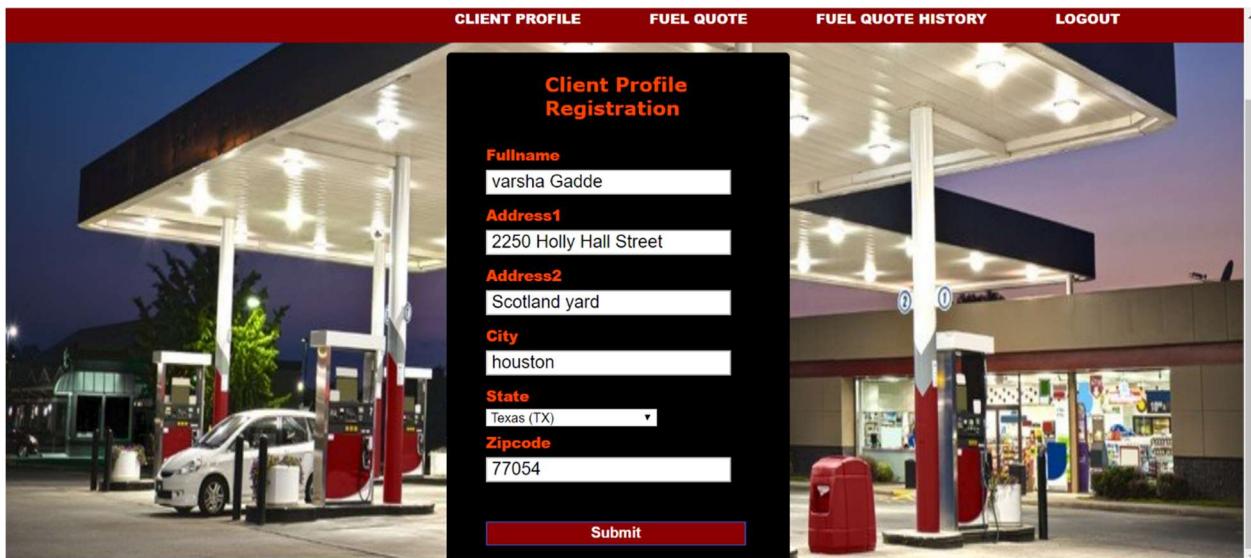
Case 1: If new user, then an alert box pops up mentioning that user must submit the client profile.



When user clicks on OK, it redirects to client Profile Page to complete the profile.



User fills all the details and then click on submit.

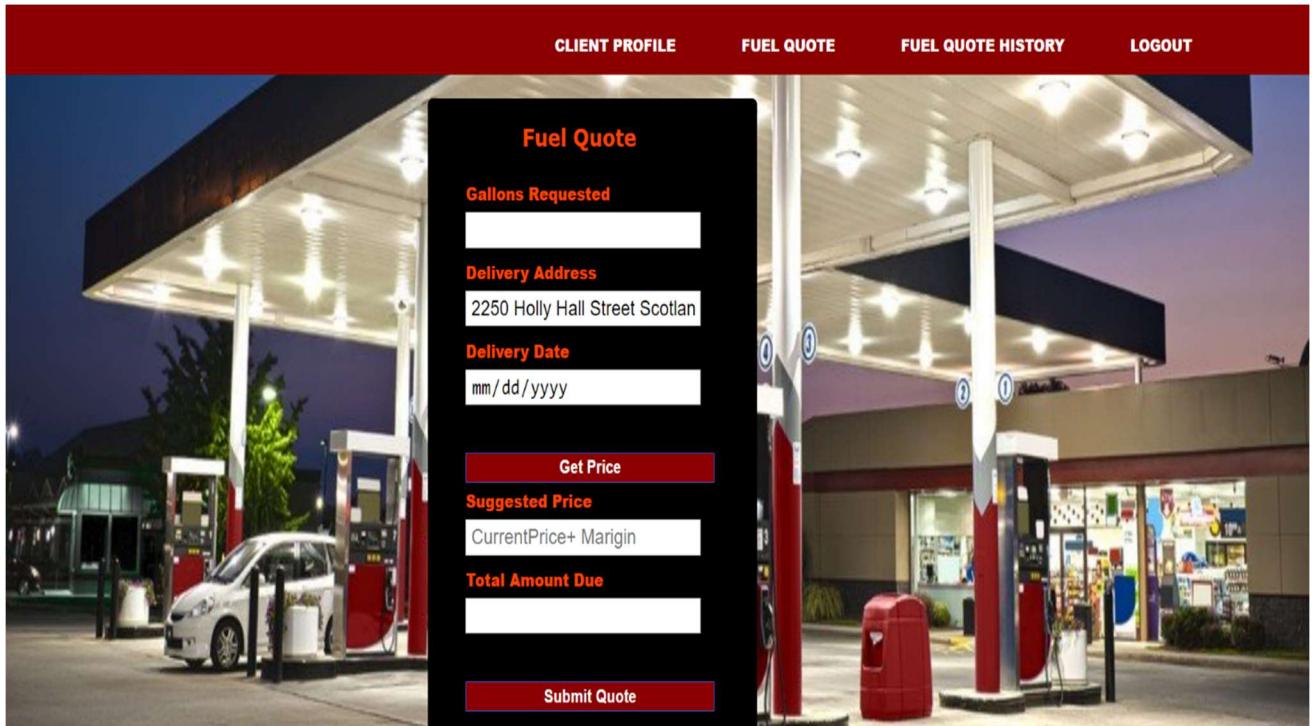


The screenshot shows a web application interface for 'Client Profile Registration'. At the top, there are navigation links: 'CLIENT PROFILE' (highlighted in red), 'FUEL QUOTE', 'FUEL QUOTE HISTORY', and 'LOGOUT'. The main content area features a dark overlay with form fields. On the left, there's a photograph of a gas station at night with a white car at a pump. On the right, there's another photograph of a gas station at night. The registration form includes fields for 'Fullname' (Varsha Gadde), 'Address1' (2250 Holly Hall Street), 'Address2' (Scotland yard), 'City' (Houston), 'State' (Texas (TX)), and 'Zipcode' (77054). A 'Submit' button is at the bottom of the form.

Once all the validations for these fields are passed, the submit button redirects user to Fuel Quote Page.

Case 2: If the user is a returning user, the login page directly takes the user to Fuel Quote Form.

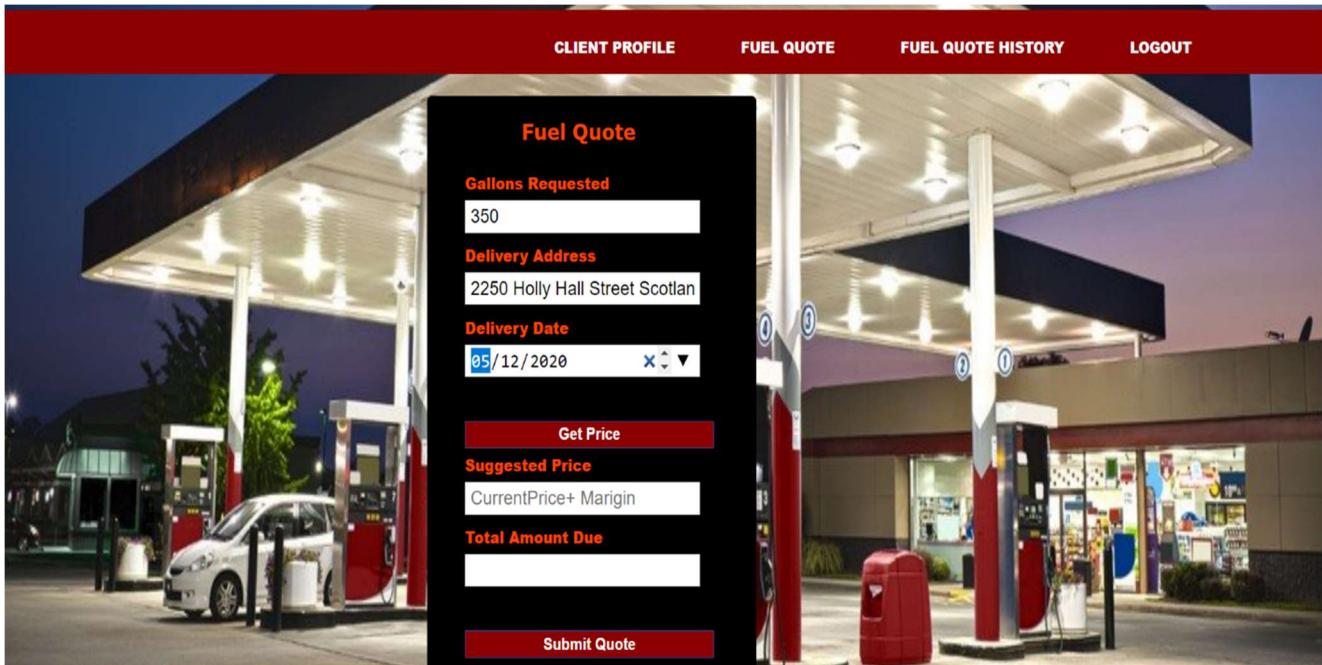
Fuel Quote:



The screenshot shows a web application interface for 'Fuel Quote'. At the top, there are navigation links: 'CLIENT PROFILE' (highlighted in red), 'FUEL QUOTE' (highlighted in red), 'FUEL QUOTE HISTORY', and 'LOGOUT'. The main content area features a dark overlay with form fields. On the left, there's a photograph of a gas station at night with a white car at a pump. On the right, there's another photograph of a gas station at night. The fuel quote form includes fields for 'Gallons Requested' (empty input field), 'Delivery Address' (2250 Holly Hall Street Scotland), 'Delivery Date' (mm/dd/yyyy input field), a 'Get Price' button, a 'Suggested Price' field (CurrentPrice+ Margin), 'Total Amount Due' (empty input field), and a 'Submit Quote' button.

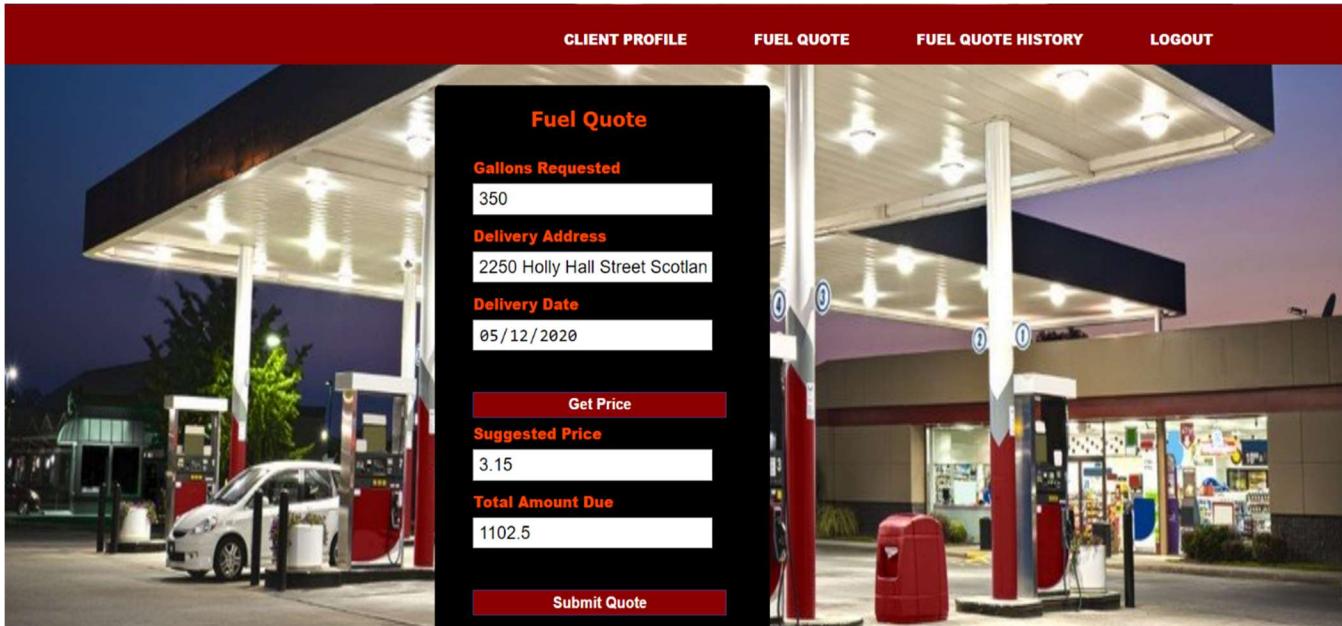
In above page, address will be populated already for the user from the client profile page.

Now user has to specify the **gallons requested** and **delivery date** and click on **Get Price**.



The screenshot shows a fuel station at night. A white car is at a fuel pump. In the foreground, a black modal window titled "Fuel Quote" is displayed. It contains fields for "Gallons Requested" (350), "Delivery Address" (2250 Holly Hall Street Scotlan), and "Delivery Date" (05/12/2020). Below these are buttons for "Get Price", "Suggested Price" (CurrentPrice+ Margin), and "Total Amount Due". At the bottom is a "Submit Quote" button.

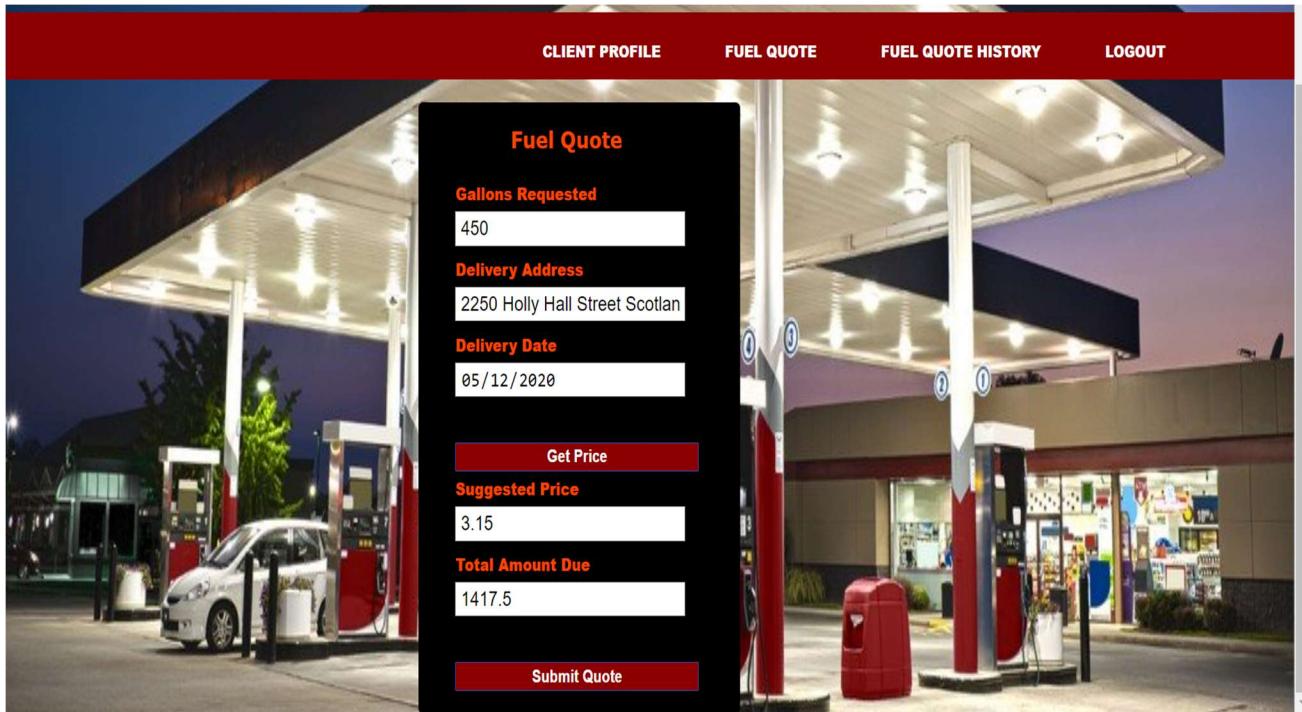
After clicking on Get Price button, suggested price and Total Amount Due will be displayed below.



The screenshot shows the same fuel station at night. The black modal window now displays the results of the "Get Price" button click. It shows "Suggested Price" (3.15) and "Total Amount Due" (1102.5). The other fields and buttons remain the same as in the previous step.

If user is ok with the Suggested Price and Total amount due, he can click on Submit Quote, Quote gets submitted.

If, not user change the values of gallons requested and delivery date and click on Get Price Again, as shown below.



CLIENT PROFILE **FUEL QUOTE** **FUEL QUOTE HISTORY** **LOGOUT**

Fuel Quote

Gallons Requested

Delivery Address

Delivery Date

Get Price

Suggested Price

Total Amount Due

Submit Quote

When the user is ok, and clicks on Submit Quote Button, QuoteHistory of the user will be displayed.



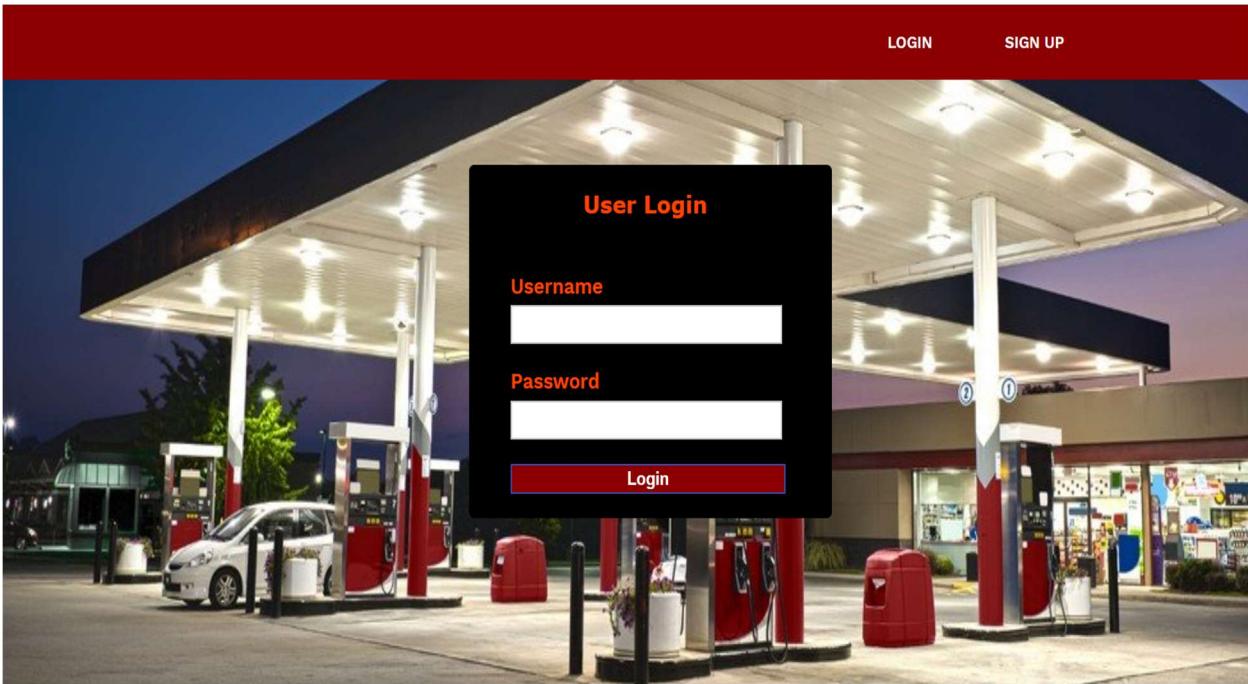
CLIENT PROFILE **FUEL QUOTE** **FUEL QUOTE HISTORY** **LOGOUT**

Fuel Quote History Details

| Email | Gallons | Delivery Address | Delivery Date | Suggested Price | Total Amount |
|------------------|---------|---|---------------|-----------------|--------------|
| varsha@gmail.com | 450 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-05-12 | 3.15 | 1417.5 |
| varsha@gmail.com | 45 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-01 | 3.15 | 141.75 |
| varsha@gmail.com | 34 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-21 | 3.15 | 107.1 |
| varsha@gmail.com | 45 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-07 | 3.15 | 141.75 |
| varsha@gmail.com | 56 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-14 | 3.15 | 176.4 |
| varsha@gmail.com | 56 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-14 | 3.15 | 176.4 |
| varsha@gmail.com | 125 | 2250 Holly Hall Street Scotland yard TX 77054 | 2020-04-15 | 3.15 | 393.75 |

The user has the flexibility to navigate to any page from the navbar above. User can update the his/her profile, get a new quote, check the fuel quote history or logout from the application.

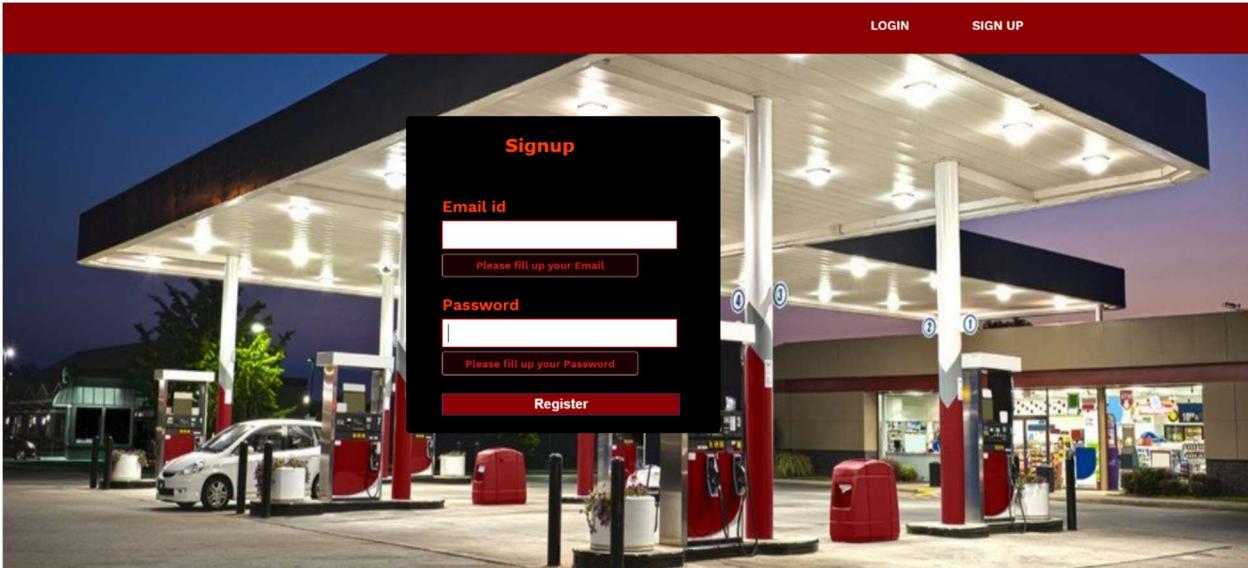
The logout button will logout user from all the sessions and take him/her to the login page.



b) Unit Tests and Validations:

Signup Form:

If any of the email or password fields are not specified it shows an error both are required. Without entering both the fields you cannot submit the form.



Unit test 2: After Submitting the form, checks whether the email format entered is correct, if not displays invalid email format.

Terminal Help signup-server.php - phpunit - Visual Studio Code

phpunit.xml FirstTest.php signup-server.php

```
tests > unit > signup-server.php
  2   session_start();
  3
  4   // initializing variables
  5   $username = "root";
  6   $email    = "";
  7   $errors   = array();
  8
  9   // connect to the database
 10  $db = mysqli_connect('localhost', 'root', '', 'test');
 11
 12  // REGISTER USER
 13  if (isset($_POST['reg_user'])) {
 14      // receive all input values from the form
 15
 16      $email = mysqli_real_escape_string($db, $_POST['email']);
 17      $password = mysqli_real_escape_string($db, $_POST['password']);
 18
 19      if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
 20          array_push($errors,"Invalid email format");
 21      }
 22  }
```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL

2: Task - run + □ ×

Test results may not be as expected.

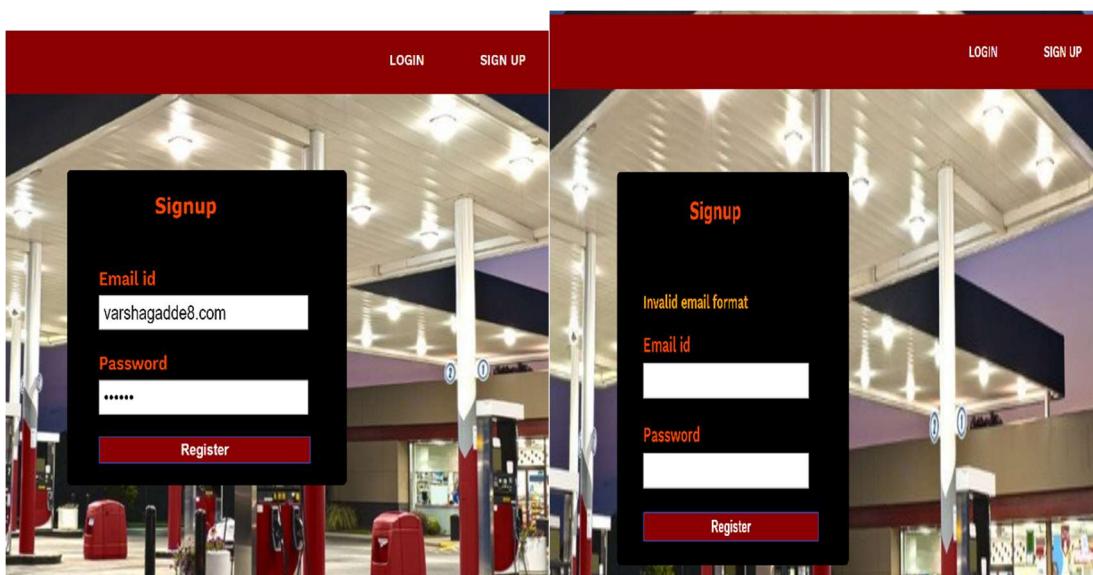
1 / 1 (100%)

Time: 67 ms, Memory: 4.00 MB

OK (1 test, 1 assertion)

Terminal will be reused by tasks, press any key to close it.

Ln 49, Col 1 Spaces: 4 UTF-8 CRLF PHP ⚙



Unit test 3: After Submitting the form checks length of the password if it is less than 6 characters then it will display an error message.

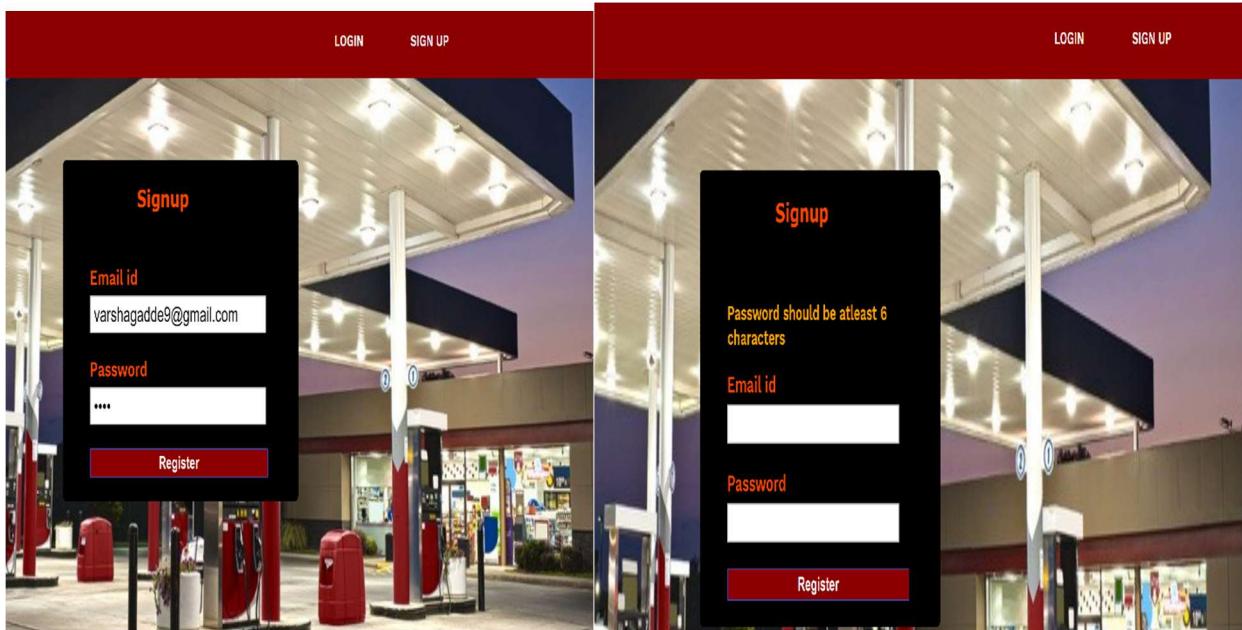
The screenshot shows a Visual Studio Code interface. The top bar has tabs for 'Terminal', 'Help', 'phpunit.xml', 'FirstTest.php', and 'signup-server.php'. The main editor area contains PHP code for a signup server. The code includes logic to check if the password length is less than 6 characters and add an error message to an array. Below the code, the terminal window shows the output of a PHPUnit run. It indicates 1 test and 1 assertion were run, both of which passed ('OK'). The message 'Test results may not be as expected.' is also visible.

```
tests > unit > SignupTest::testSignupWithShortPassword
3
4 // initializing variables
5 $username = "root";
6 $email    = "";
7 $errors   = array();
8
9 // connect to the database
10 $db = mysqli_connect('localhost', 'root', '', 'test');
11
12 // REGISTER USER
13 if (isset($_POST['reg_user'])) {
14     // receive all input values from the form
15
16     $email = mysqli_real_escape_string($db, $_POST['email']);
17     $password = mysqli_real_escape_string($db, $_POST['password']);
18
19     if(strlen($password)<6){
20         array_push($errors,"Password should be atleast 6 characters");
21     }
22 }

Test results may not be as expected.

.
.
.
Time: 67 ms, Memory: 4.00 MB
OK (1 test, 1 assertion)

Terminal will be reused by tasks, press any key to close it.
```



Unit test 4: when a new user tries to signup with the email id that already exists, error message is displayed that email exists.

```
31     if ($user) { // if user exists
32         if ($user['email'] === $email) {
33             array_push($errors, "email already exists");
34         }
35     }
36
37
38     // Finally, register user if there are no errors in the form
39     if (count($errors) == 0) {
40         $password1 = md5($password); // encrypt the password before saving in the database
41         $query = "INSERT INTO users_data (email, password)
42             VALUES('$email', '$password')";
43         mysqli_query($db, $query);
44         $_SESSION['email'] = $email;

```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL

Test results may not be as expected.

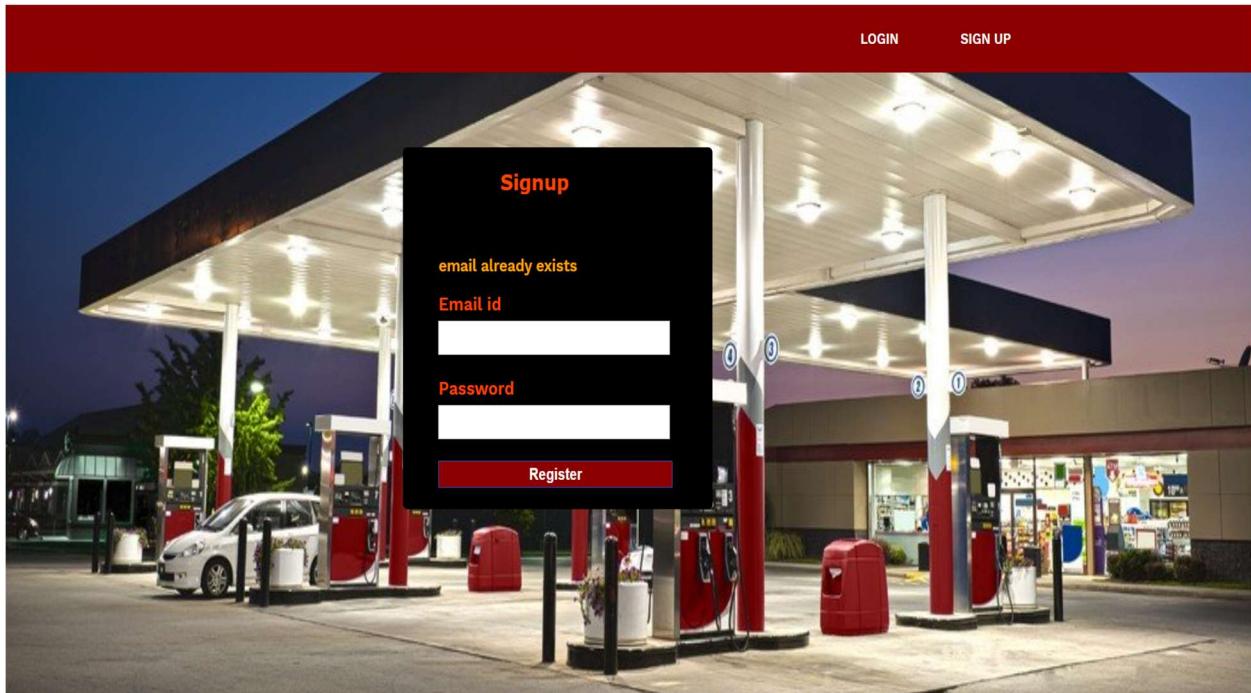
.

1 / 1 (100%)

Time: 67 ms, Memory: 4.00 MB

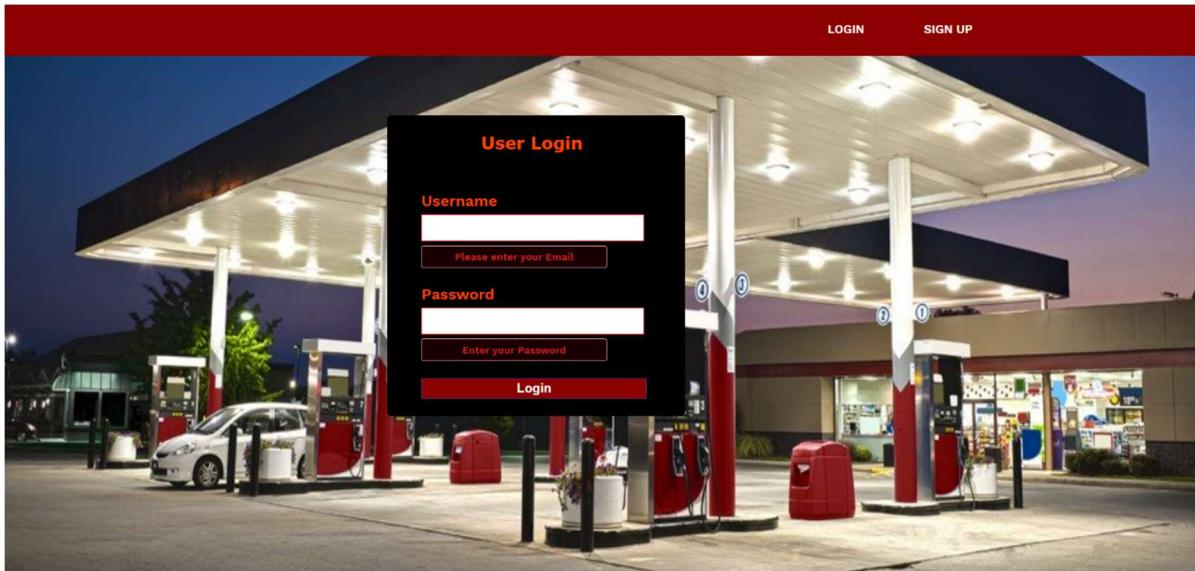
OK (1 test, 1 assertion)

Terminal will be reused by tasks, press any key to close it.



Login Form:

If any of the email or password fields are not specified it shows an error both are required. Without entering both of the fields you cannot submit the form.



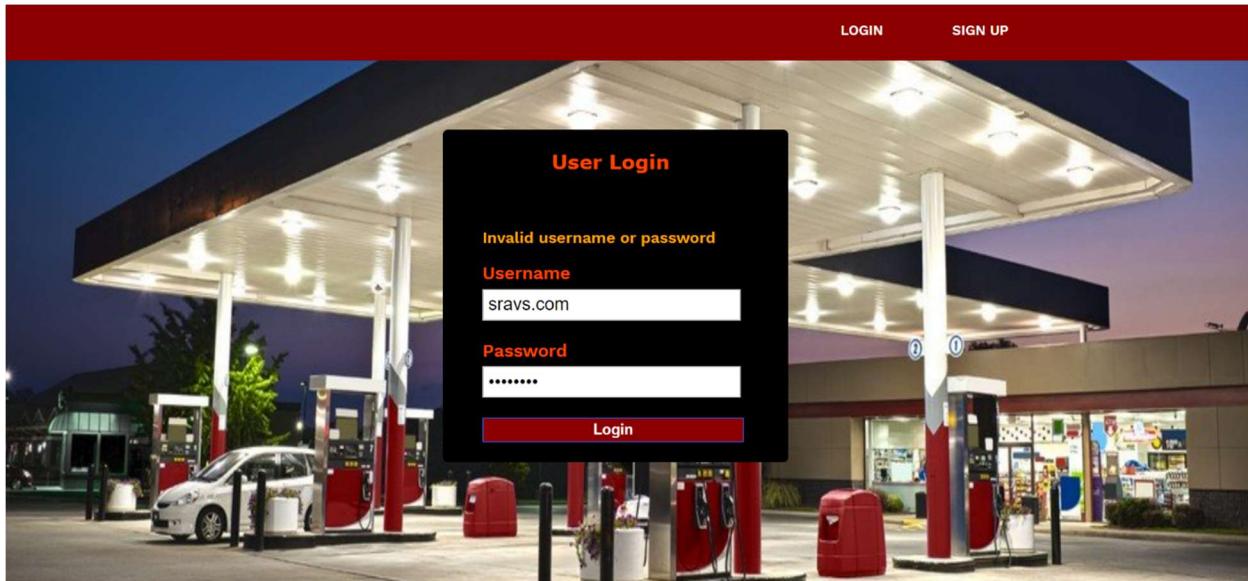
Unit test 2: After Submitting the form , checks whether the email and password are correct. If not display to the user wrong username password combination.

A screenshot of the Visual Studio Code interface. The top navigation bar shows "Terminal", "Help", and tabs for "phpunit.xml", "FirstTest.php", "signup-server.php", and "server.php". The main editor area displays a PHP file named "server.php" with code related to user login. The bottom status bar shows tabs for "DEBUG CONSOLE", "PROBLEMS", "OUTPUT", and "TERMINAL". The terminal tab is active and shows the output of a unit test run. The text in the terminal reads:

```
Test results may not be as expected.

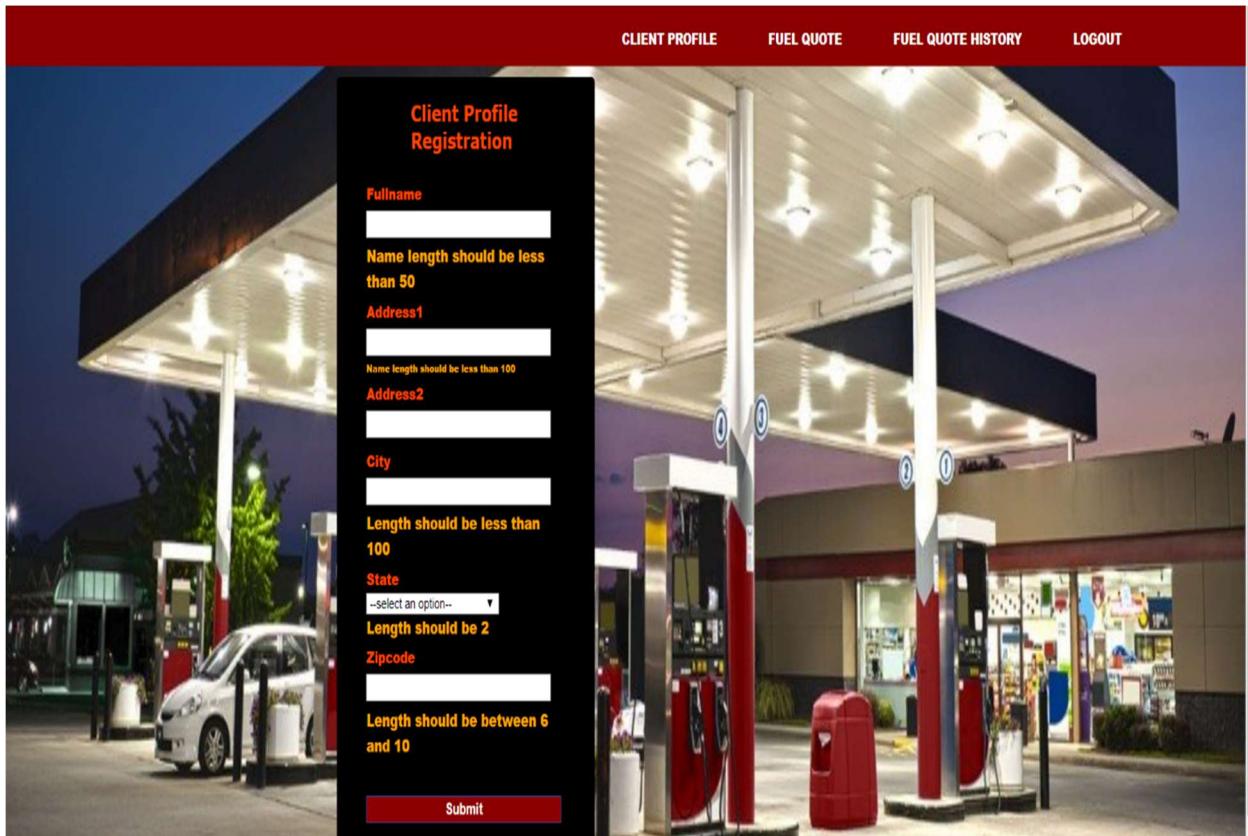
Time: 67 ms, Memory: 4.00 MB
OK (1 test, 1 assertion)

Terminal will be reused by tasks, press any key to close it.
```



Client Profile Form:

Unit test 1: All the required fields Fullname, Address1, City, State, Zipcode need to be filled. If any one of the field is missed out, it displays an error message.



Terminal Help • profile_server.php - phpunit - Visual Studio Code

phpunit.xml FirstTest.php signup-server.php server.php profile_server.php fuelQuote.php quoteServer.php

```

tests > unit > profile_server.php
18 // REGISTER USER
19 if(isset($_POST['profile_user'])){
20     // receive all input values from the form
21     array_push($results, "entered here");
22
23     $fullname = mysqli_real_escape_string($db, $_POST['fullname']);
24     if(strlen($fullname)>50 || $fullname==''){
25         array_push($fullname_error,"Name length should be less than 50");
26     }
27     $address1 = mysqli_real_escape_string($db, $_POST['address1']);
28     if(strlen($address1)>100 || $address1==''){
29         array_push($address1_error,"Name length should be less than 100");
30     }
31
32     $city = mysqli_real_escape_string($db, $_POST['city']);
33     if(strlen($city)>100 || $city==''){
34         array_push($city_error,'Length should be less than 100');
35     }
36     $state = mysqli_real_escape_string($db, $_POST['state']);
37     if($state==null){
38         array_push($state_error,'Length should be 2');
39     }
40
41     $zipcode = mysqli_real_escape_string($db, $_POST['zipcode']);
42     if(strlen($zipcode)>1 || $zipcode=='' || strlen($zipcode)<6){
43         array_push($zipcode_error,"Length should be between 6 and 10");
44     }
45
46 // $sql = "INSERT INTO clientprofile(FULLNAME, ADDRESS1, CITY, STATE, ZIPCODE) VALUES('new224', 'SAVANNAH', 'GA', '31408')";

```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL 2: Task - run + ☰ ^ x

Test results may not be as expected.

..... 5 / 5 (100%)

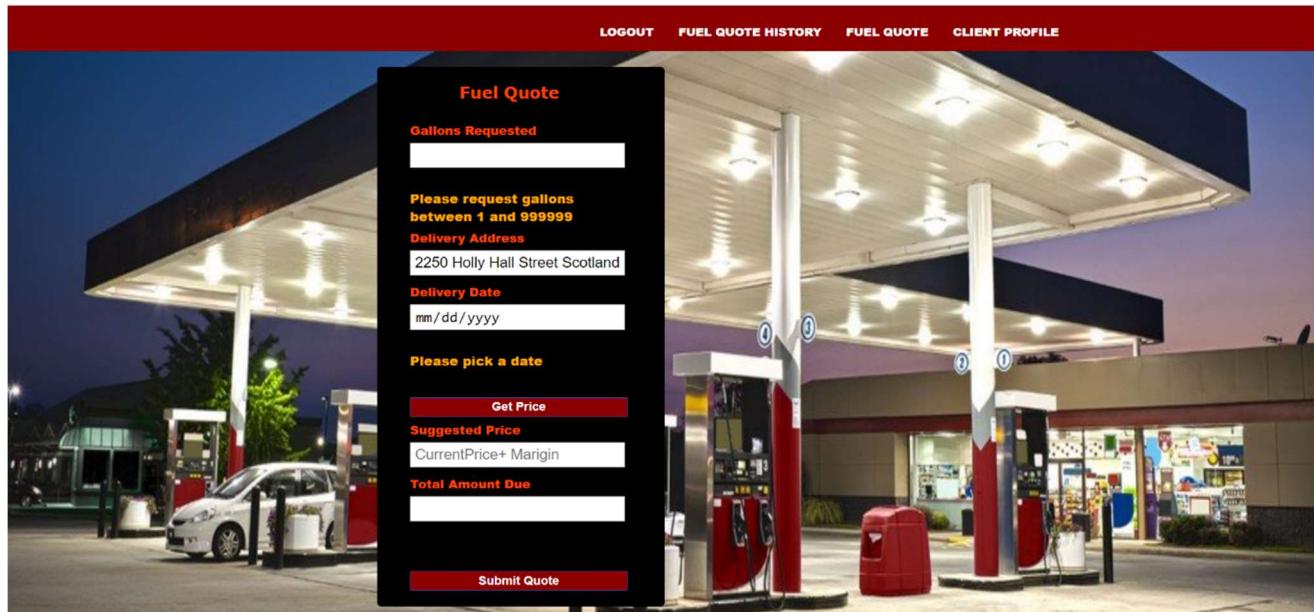
Time: 71 ms, Memory: 4.00 MB

OK (5 tests, 5 assertions)

Terminal will be reused by tasks, press any key to close it.

Fuel Quote: Unit Test code for front end validation

Unit test 1: All the required fields GallonsRequired, DeliveryDate need to be filled. Without filling these fields user clicks on Get Price button, then below validations occur.



QUOTE SERVERS.php quoteServer.php

```

100 |     $error=true;
101 //if ((filter_var($gallons, FILTER_VALIDATE_INT, array("options" => array("min_range"=>$min, "max_range"=>$max))) === false) || $gallons=
102 |     array_push($gallons_error, "Please request gallons between 1 and 99999");
103 |     //print("entering in error");
104 }
105 if ($deliveryDate==''){
106 |     $error=true;
107 // array_push($date_error,"Please pick a date");
108 }
109 if($error==false){
110 // print("entereing in the calc part");
111 $company_profit_factor=0.1;
112 $rate_fluctuation="";
113 $suggested_price="";
114 $total_price="";
115 $margin="";
116 $current_price=1.50;
117 $gallons_requested_factor="";
118 if($state=='TX')
119 |     $location_factor=0.02;
120 else {
121     $location_factor=0.04;
122 }

```

DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL

```

> Executing task: c:/Users/srava/phpunit/vendor/bin/phpunit.bat c:/Users/srava/phpunit/tests/unit/QUOTE SERVERS.php <
PHPUnit 9.1.1 by Sebastian Bergmann and contributors.

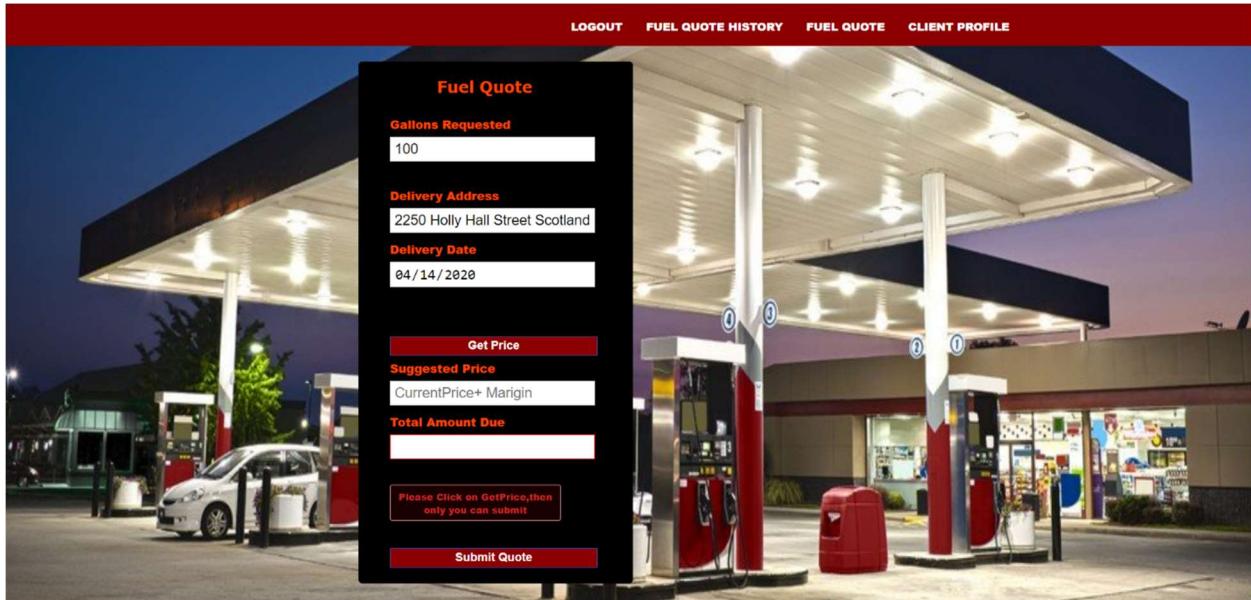
Runtime:      PHP 7.4.3
Configuration: C:/Users/srava/phpunit/phpunit.xml

..
Time: 63 ms, Memory: 4.00 MB
OK (2 tests, 2 assertions)

Terminal will be reused by tasks, press any key to close it.

```

without clicking on Get Price, if directly user clicks on submit quote button then it displays a following error message. (First get appropriate price then only you can submit quote)



Github Link For the code:

https://github.com/sravs1/Fuel_Quote_Prediction