----------

Name: Viswanadham Udaya Chitanya Krishna

Student number: 7083270

Email address: uckv235@uowmail.edu.au

Assignment number: 1

----------

**Files included in this assignment submission:**

**PHP Files**

1) index.php - the home page of our application

2) addbikelisting.php - page to add new interested seller + bike listing

3) managelisting.php - page to manage listing for existing buyers which have listed and for them to see their interested buyers that express interest to buy the bike

4) bikelisting.php - page to display all listings by sellers and interested buyers to express interest

**Text Files**

5) database/BikesForSales.txt - stores the user info + bike info

6) database/ExpInterest.txt - stores the interested users info

**Class File**

7) classes/bikelisting.php - stores the bikelisting class info

8) classes/interestuser.php - stores the interested user class info

**CSS File**

9) css/main.css - css styling common to all the pages

10) css/flex.css - css styling which helps to dynamically adjust sizing of boxes

11) css/tableui.css - css styling for tables used in the pages

12) css/addbikelisting.css - css styling for addbikelisting page only

**Description of pages for the assignment:**

**1. Index.php**The homepage of our application. We have the main 4 headers, this styling will be common throughout the whole app to keep a minimal code maintenance and easy intuitive access for the user. The main features of the app would be:  
- view listing: that will bring them to the bike listing page. When the user is there, user can search for bicycles which they are interested in buying and also express interest in the purchase.

- manage listing: this page can allow existing users that have listed their bike listing to be able to manage their bicycle listing which allowing them to view the people whom have express interest to purchase the bicycles

- add listing: this page allow buyers to be able to add new bikes to allow them post the bicycles which they would like to sell and their details on it

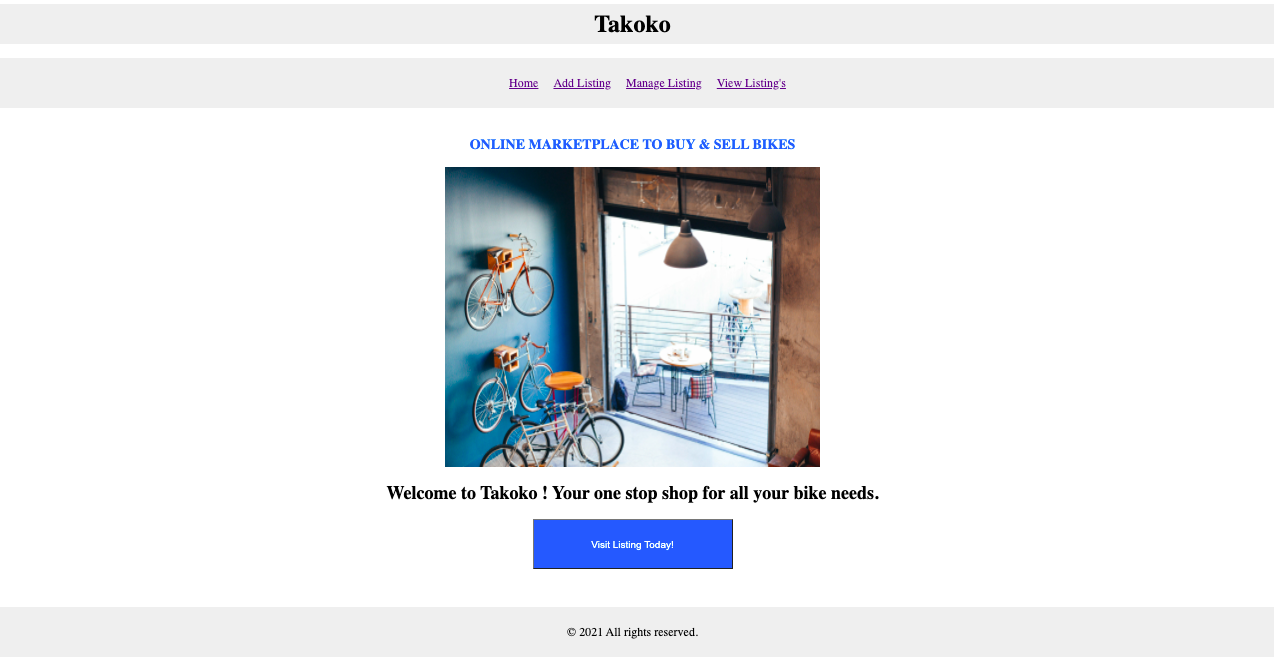


Image1.1 bike listing

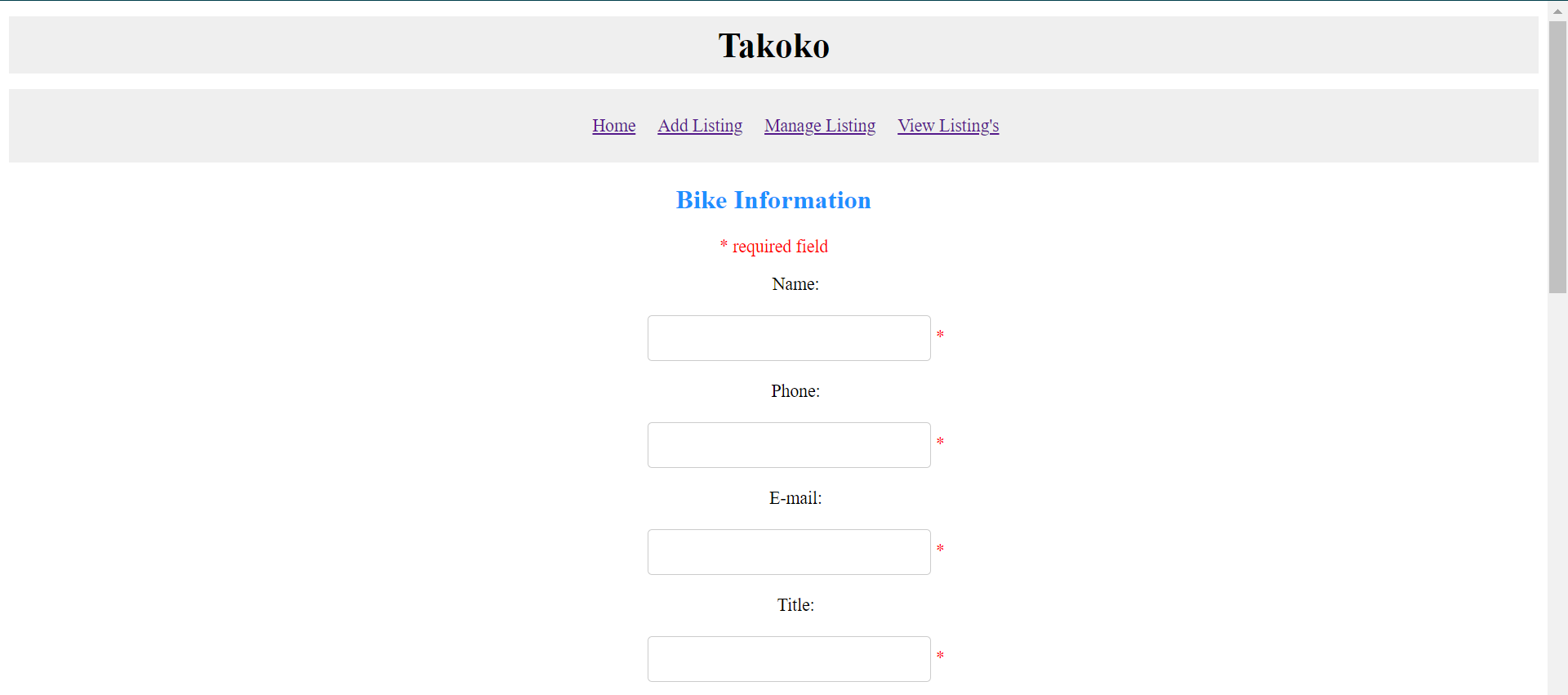
**2. addbikelisting.php**

The addbikelisting page allows users or sellers to sell their vehicles by adding in the relevant information about the Vehicle and updating it into the website to let buyers access the advertisement and show interest in it.

**PHP Form:**

The page itself comprises Php, HTML and CSS programming for the functionality, layout and on styling basis.The page consists of a form that has all the required fields to input the information from the user which is the vital functionality to maintain the website running in the business.

Most of the elements are using Php and HTML coding to satisfy the requirements and to look simple.



This form allows users to enter the details such as name, phone and email as personal information. Serial number, title, type and description for Vehicle details as well as additional information like year of manufacture, characteristics, condition about the vehicle to make buyers provide more information if they express their interest in it.

HTML & CSS Layout:

The layout is pretty simple and has a resizable form factor to make use of the extra screen in bigger displays and on point for smaller displays. To make use of the CSS codes HTML was useful and also helpful to provide different layouts. The CSS was used for padding and centering all the elements at one point and to provide them with the same shape and colour.

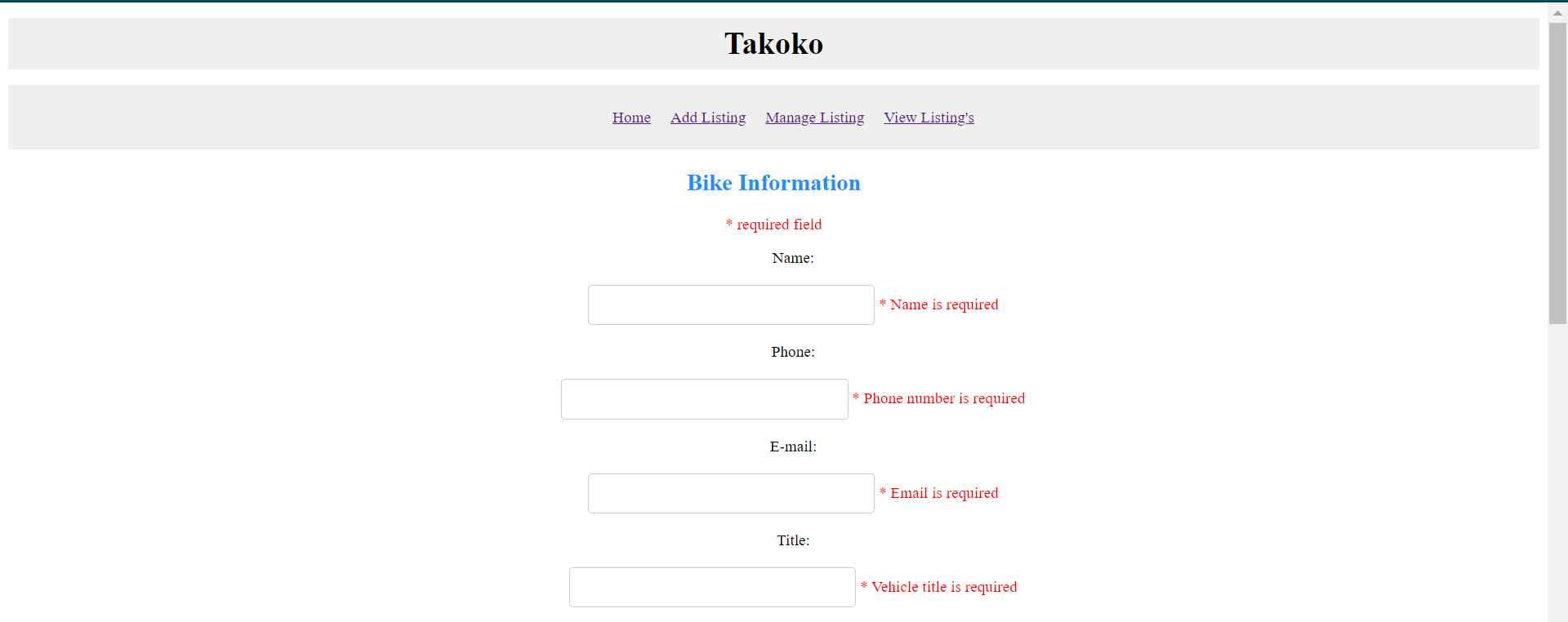
PHP Logic:

As per the requirement, the form is written in PHP with a functionality to capture the users data and store the data in a particular order when the user submit the form. It makes use of a text file to store the data and will be separated by a comma.

The page has a feature to refresh the page in 10 seconds after the user properly submitted the form. They are notified with a success message that lasts until the page refreshes to inform the user about the status of their form details. The condition field will be using the drop down menu of two options they can select.

This php file will be in contact with a txt file to store the data that is entered by the user and also store it in lowercase to ensure it will provide an optimized search when looking into the data file.

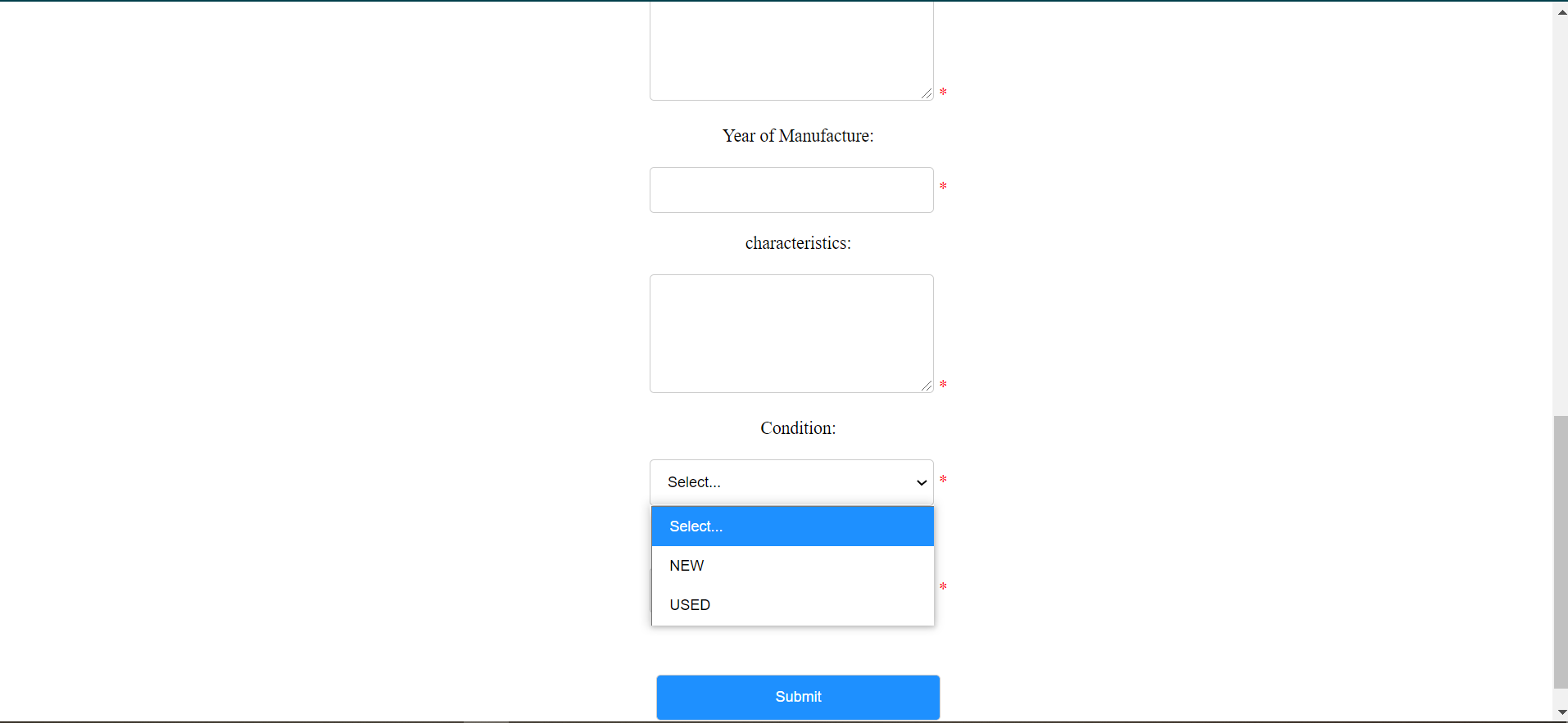
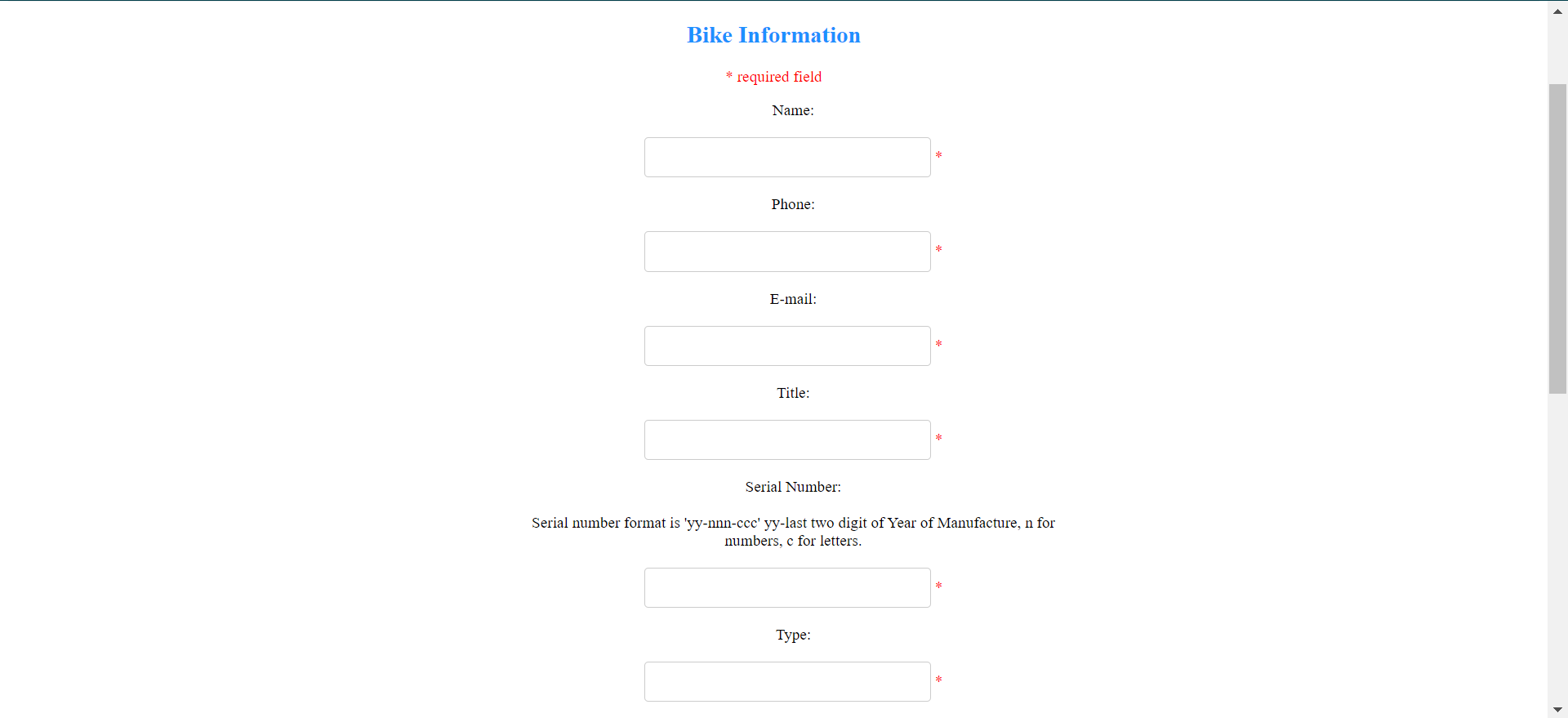
Whatever happens for all of the errors or issues, whether it is successful or fail, which field needs to be changed, what is the correct format all these things are well equipped and hope to provide the user a worry less experience in terms of functionality.



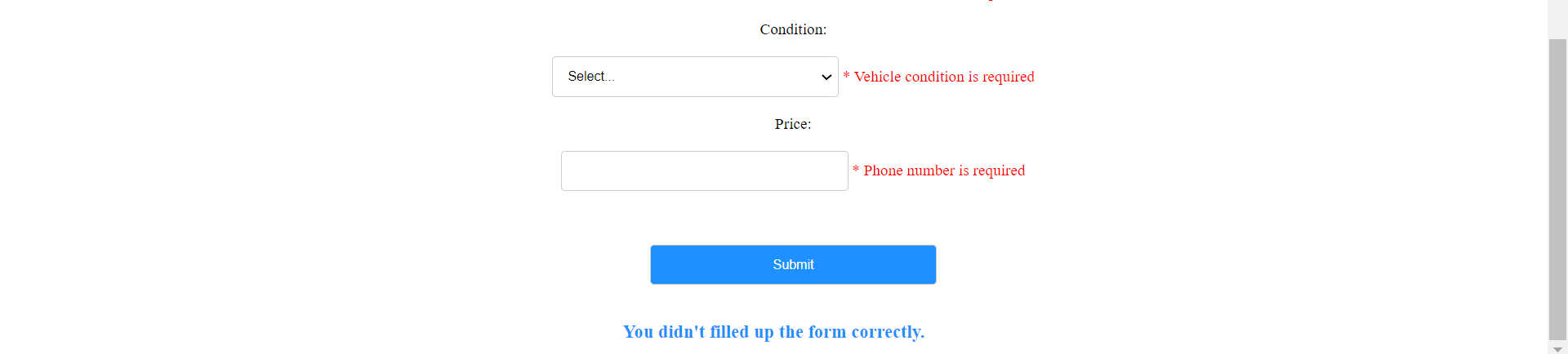
Validation:

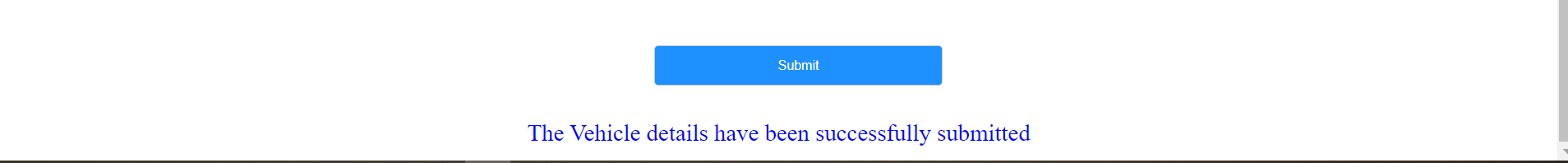
Almost all of the input fields come with certain validation rules to satisfy in order to get stored as valid data. All the fields will display an error message in order for the user to provide instructions and guide them to complete the form without any error before submitting. All of the fields are validated with a set of rules programmed whereas the Serial Number and the Year of Manufacture fields are related in their validation process. The goal here is to make the user use the provided pattern instructions when entering the serial number at the same time comparing it with the year of manufacturing as a double verification. Using preg\_match function all the verifications are performed with the given user input.

The user must satisfy all the requirements or must pass all the errors in order to successfully submit the form as most of the information is valuable and needs to be as accurate as possible.



Messages:





**3. managelisting.php**

The managelisting page allows the user to view their bicycle listings specific to what they have added before in the addbikelisting.php page.

3.1 Buyer require to login using name  
When a buyer login using name, the records are all stored in a list. There is a filter function which will filter through the list picking up only the user records that contain the person’s name. In the case of our application we have decided to use the person’s name as the person’s username. When the user is done with the session, the user can click on the logout button.

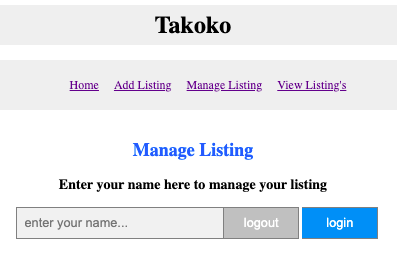


image 3.1 [before login]

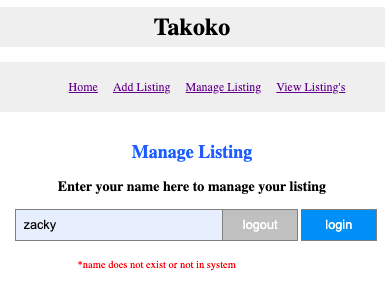


image 3.2 [error login] - name does not exist in BikesForSales.txt

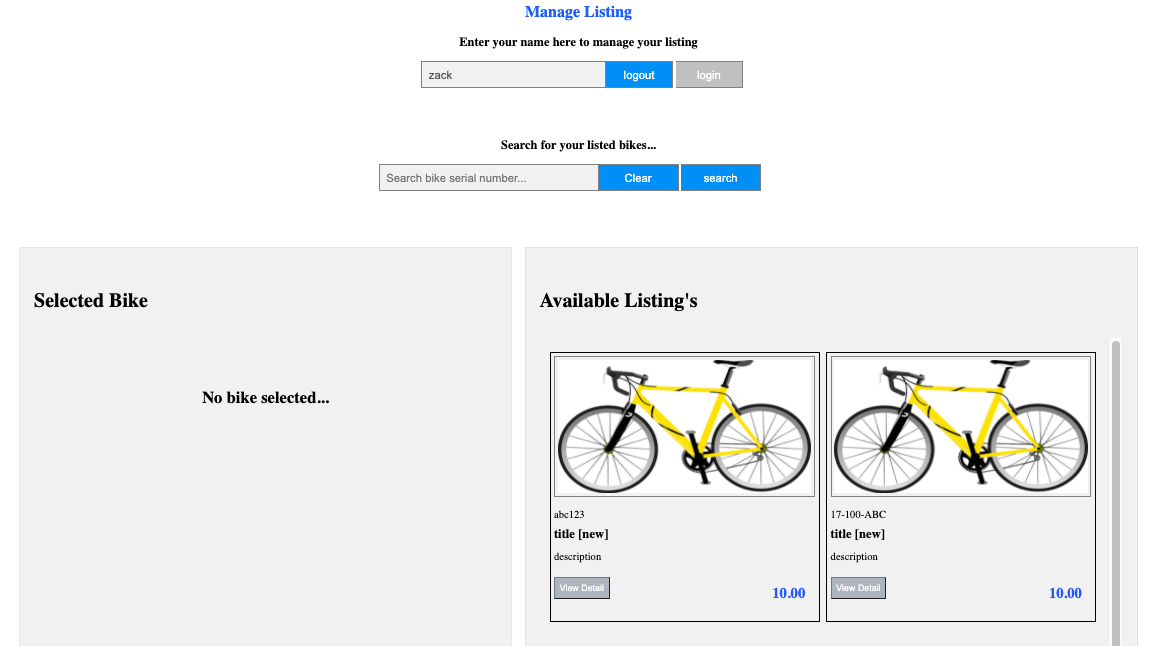


image 3.3 [Success login] - bike listing specific to the user is shown

3.2 Select Individual Listing

Buyer can select listing individually to view more info on it. It will present the detail info which is not possible in the 2 x 2 grid view.

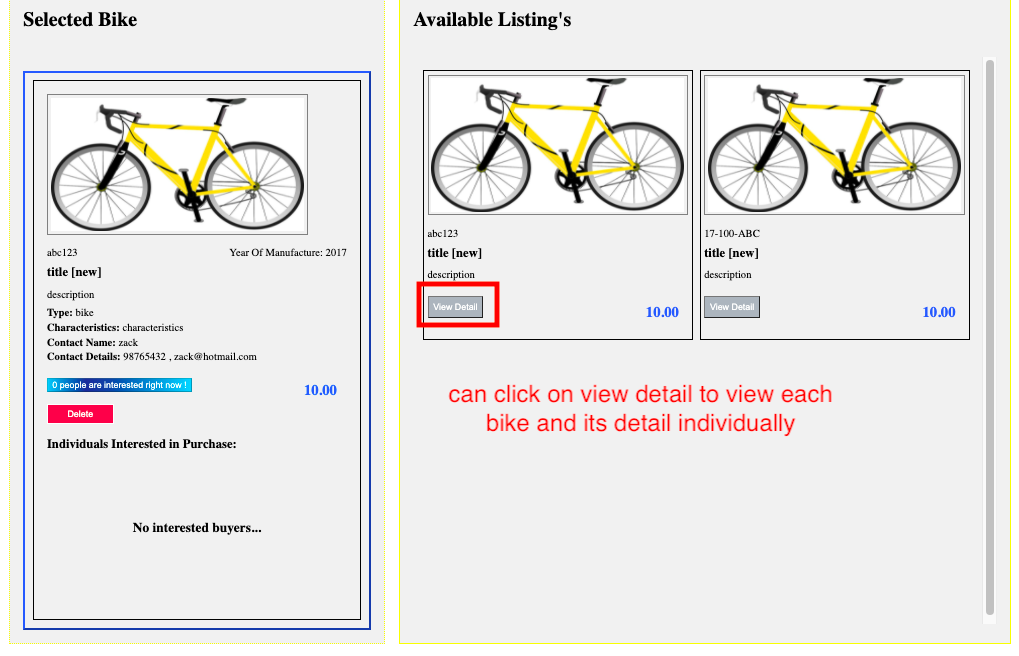


image 3.4 [select individual listing] - bike info successfully shown

3.3 User can view interested buyers

After clicking “view detail” to view individual listings, now the user can view interested buyers. If there are no interested buyers, then the web application will show as “no interested buyers” and if there are interested buyers it will be shown in a table format entailing the details of the interested buyer. There is also a highlighted box showing the interested number of buyers currently who would like to buy.

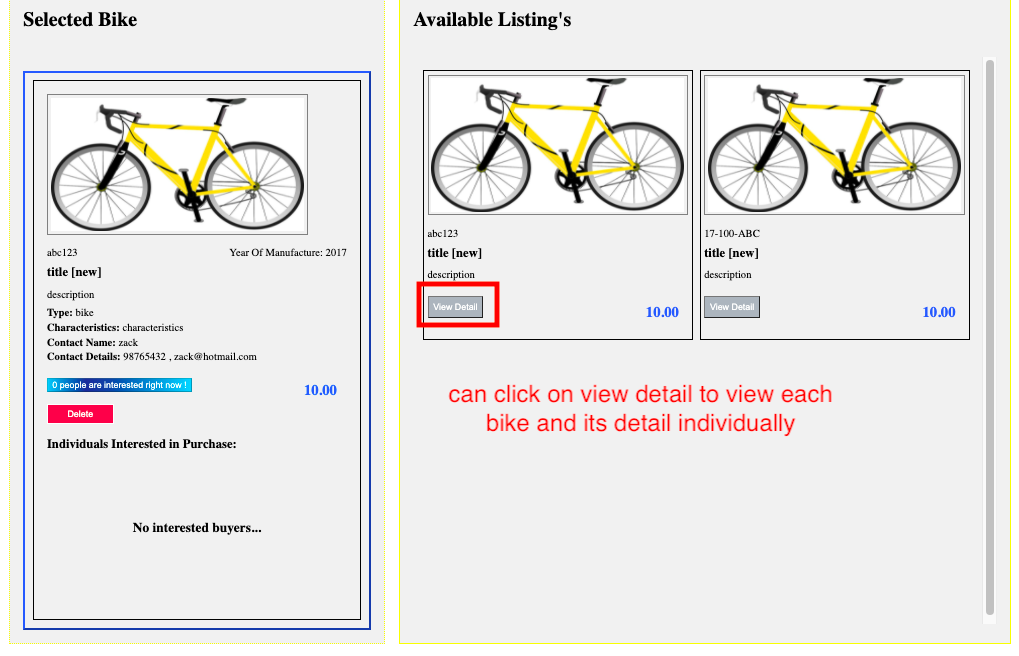


image 3.4 [No interested buyer] - when nobody express interest

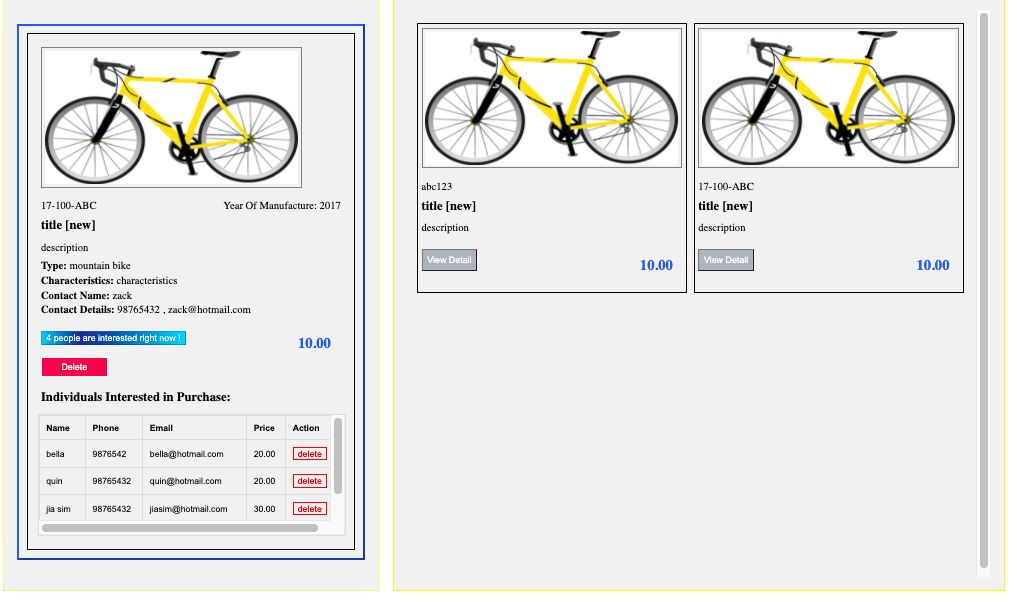


image 3.5 [interested buyer] - shown in a table format

3.4 User can delete unwanted records

Sellers can delete unwanted records when it is no longer relevant. Buyer can delete an entire bike listing and the page will automatically re-render and update the records accordingly while also allowing sellers to delete interested buyer records.

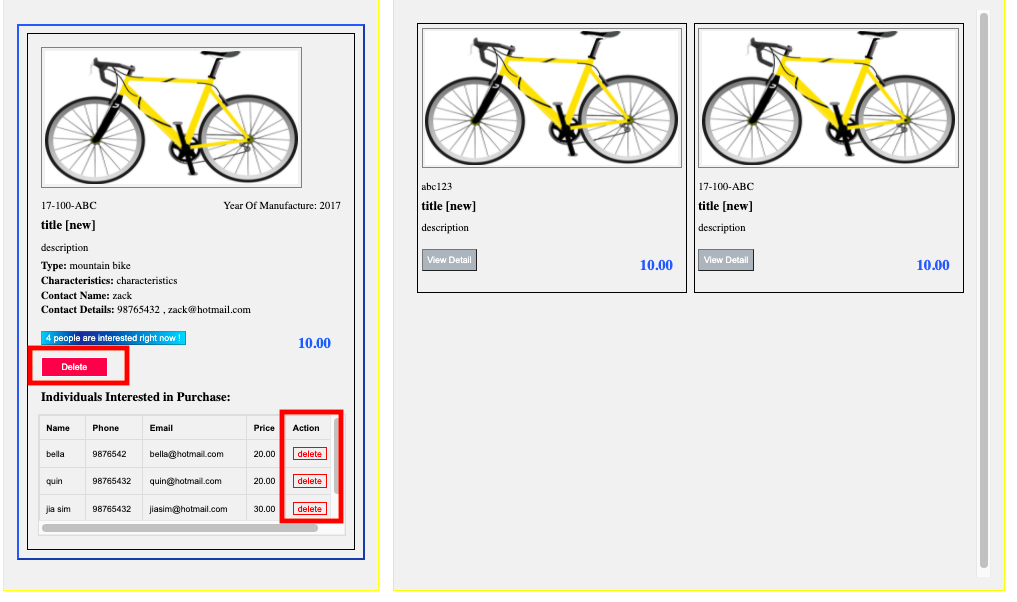


image 3.6 delete records when no longer required

**4. bike listing.php**

4.1 View bike listing in grid view

Allows buyers to scan through multiple bicycle listings and find the bicycle which they would like. The grid listing is set in a 2 column by a 2 row format. The grid listing scrollable feature is managed using CSS while the data is extracted from the text file and later stored in an array to be rendered later in the format shown as below.

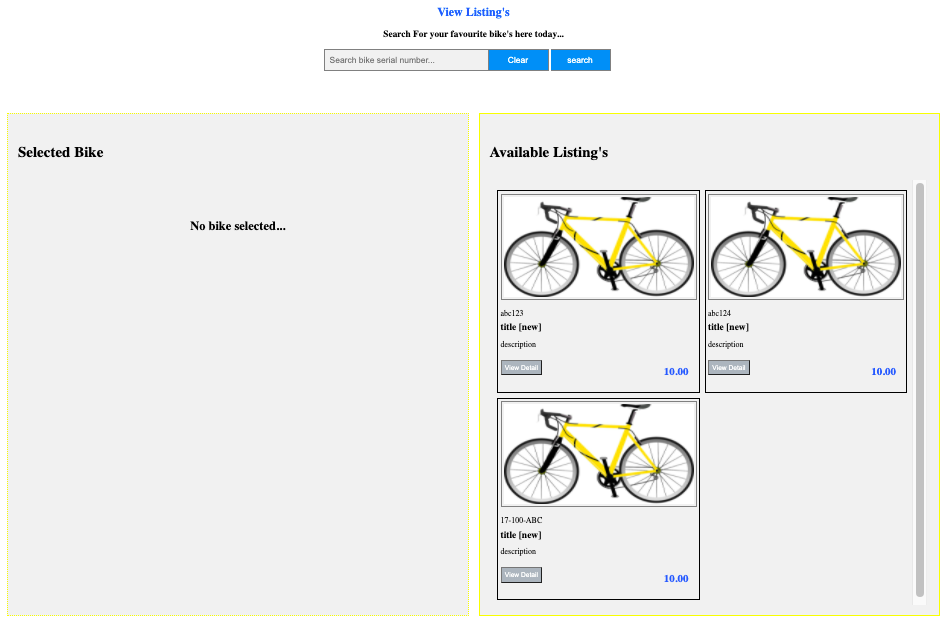


Image 4.1 [bike listing] - no search query performed

4.2 Search Query

The search query works on both managelisting and bikelisting page and using the same logic. Using regex we are able to perform search queries. We can *match exact query* & *query with a like clause feature in SQL*.

What was done is basically we loop through the array, if these are the records that we would like to have, we would put it into a new array and then update the old array with the new array allowing the results to be filtered.

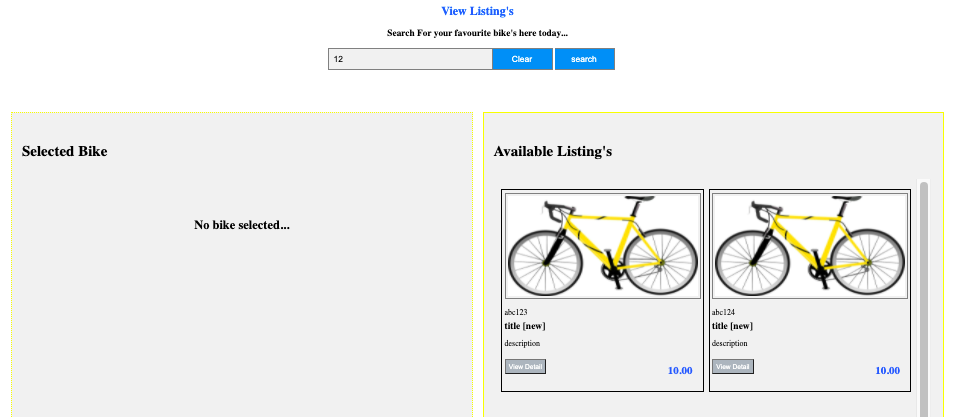


Image 4.2 [bike listing] - search query performed with a like clause

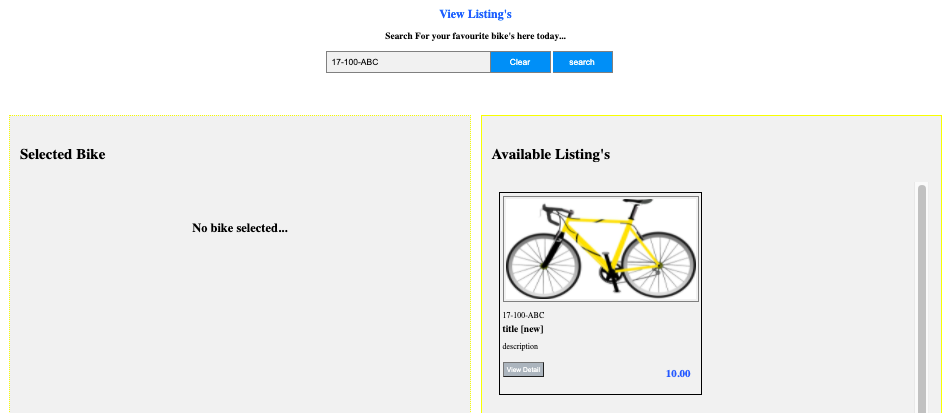


Image 4.3 [bike listing] - search query performed with a exact match clause

4.3 Selecting individual listing to view more info

By selecting individual listings, we can view more information about an individual bicycle. More details which were not shown earlier are now shown. At the same time, buyers can view the interested buyers while at the same time, at the bottom of the page there is a section for them to express interest to buy this bicycle should they be interested to purchase this bicycle.

For our application the only possible combination for the condition for the bicycle is only NEW or USED. it will be shown beside the title.

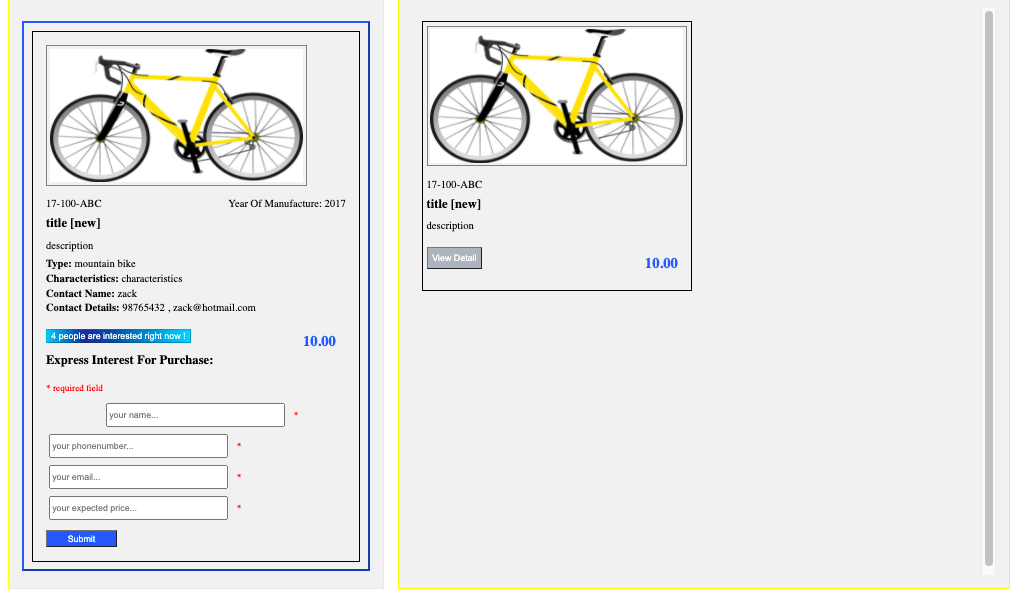


Image 4.4 [view bike detail]

4.4 Express interest to buy bicycle

Buyers can now express interest in buying the bicycle. By entering the name, phone number, email and interested price, the user will be able to update the buyer that he is interested in purchasing the bike. Since the user has selected the bicycle, the serial number of the bicycle has been stored ahead of time using global variables and in the url parameters. So at the point of saving the data is retrieved from these locations to obtain the serial number of the bicycle.

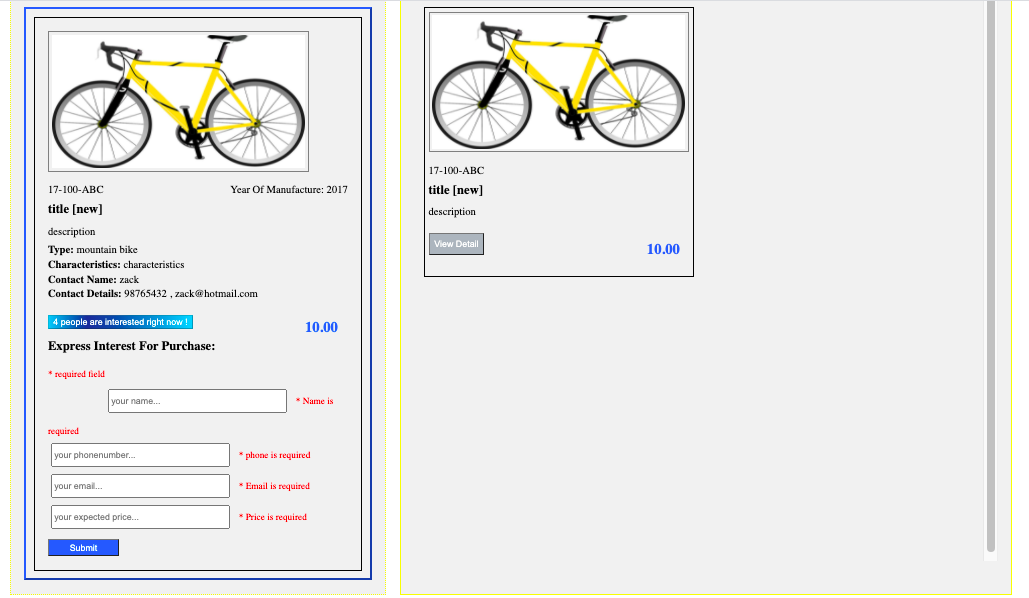


Image 4.5 [validation due to incomplete submission]

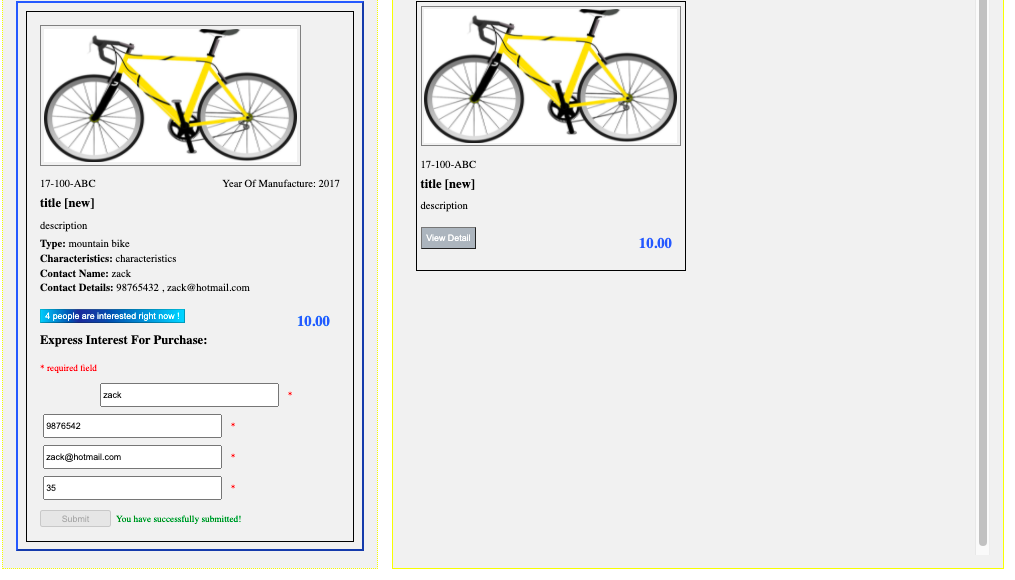


Image 4.6 [validation success] - page will refresh and update interested counter