Recommender System

Recommender systems are actually one of the most successful and widespread application of machine learning technologies in business nowadays.

There are two major ways most of recommendation engines work.

- **content-based filtering method:**
  They can either rely on the properties of the items that each user likes, discovering what else the user may like.

Figure 1: content based filtering.

- **collaborative filtering method:**

  Recommendation engines can rely on likes and desires of other users in order to compute a similarity index between users and recommend items to them accordingly.
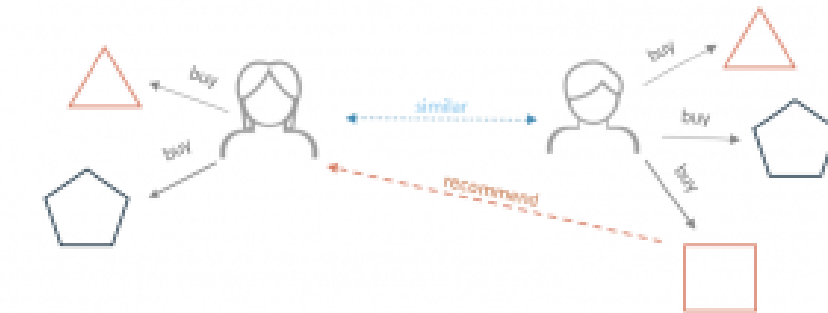

Figure 2: Collaborative filtering

For our e-commerce recommender system we used the collaborative filtering method to implement our recommender system. All the steps to a successful recommender system is shown below.

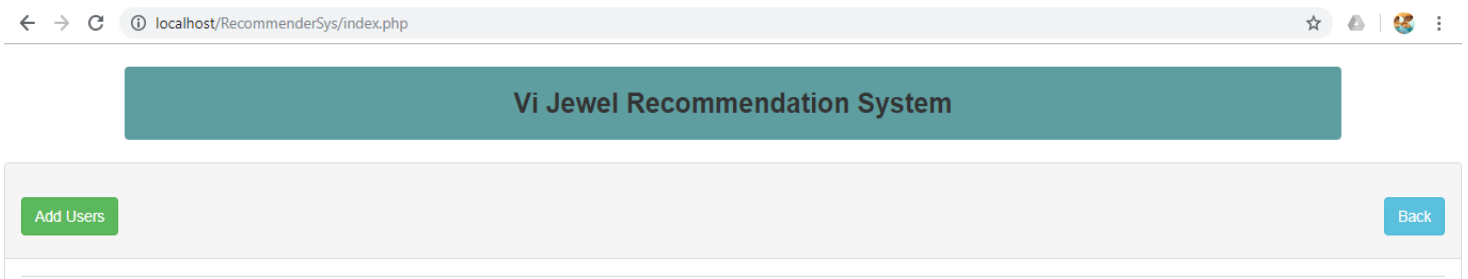The front page of our recommendation system:

Figure 3:Recommendation Front page

To add user to the recommendation system, we have to enter the name of the user in the TextField and click on the 'Add User' button.
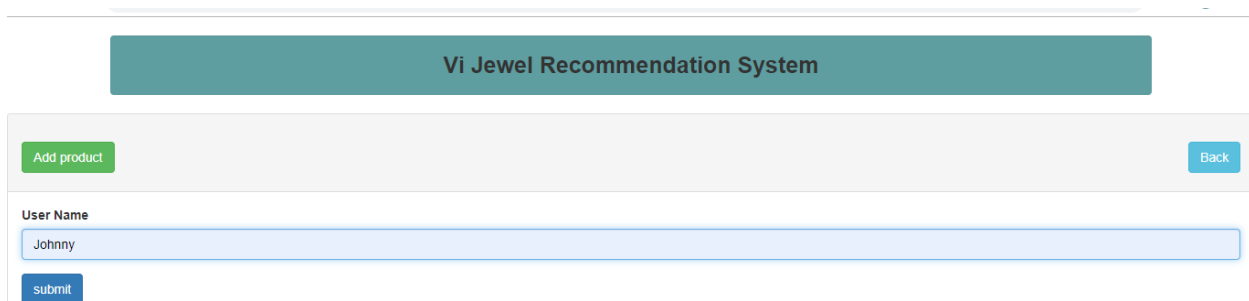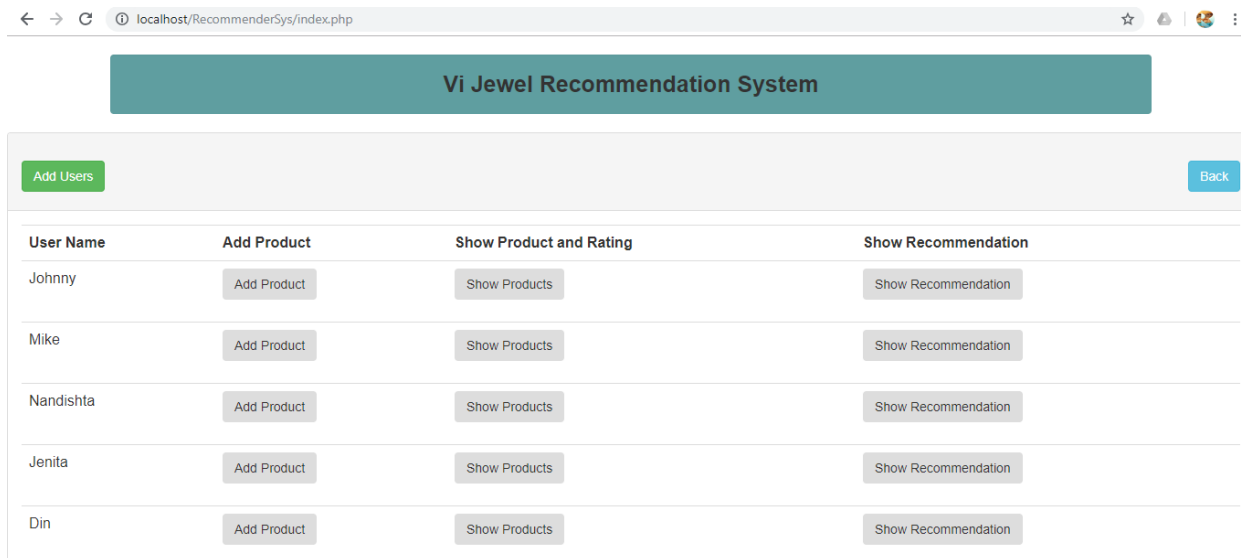

Figure 4: Adding user page

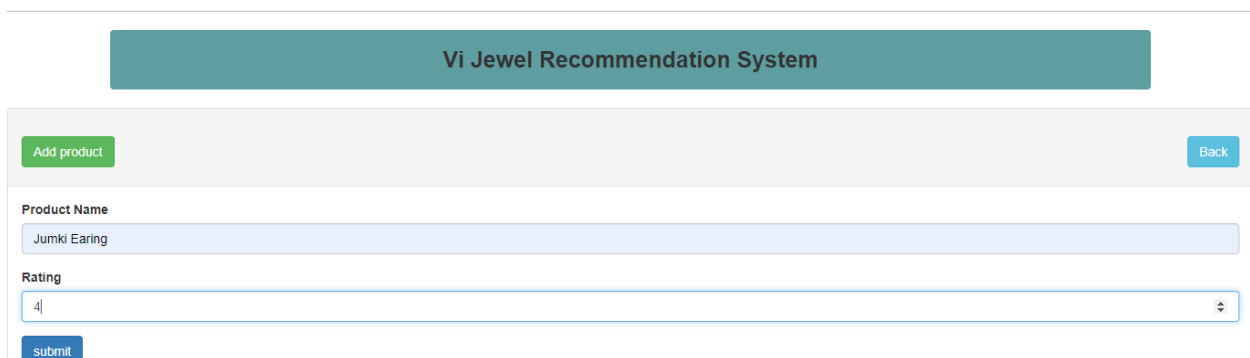The diagram below shows several users have been added to the system.

Figure 5:several users have been added.

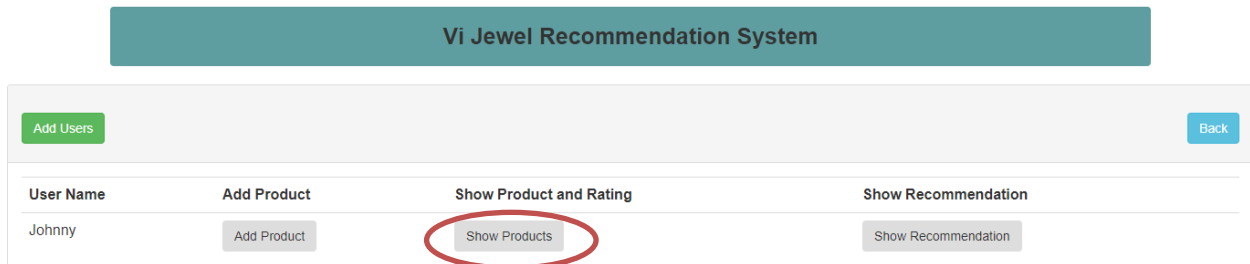The user then clicks on 'Add Product' button next to the user name to add products along with the ratings.

Figure 6:Add a product and insert its rating
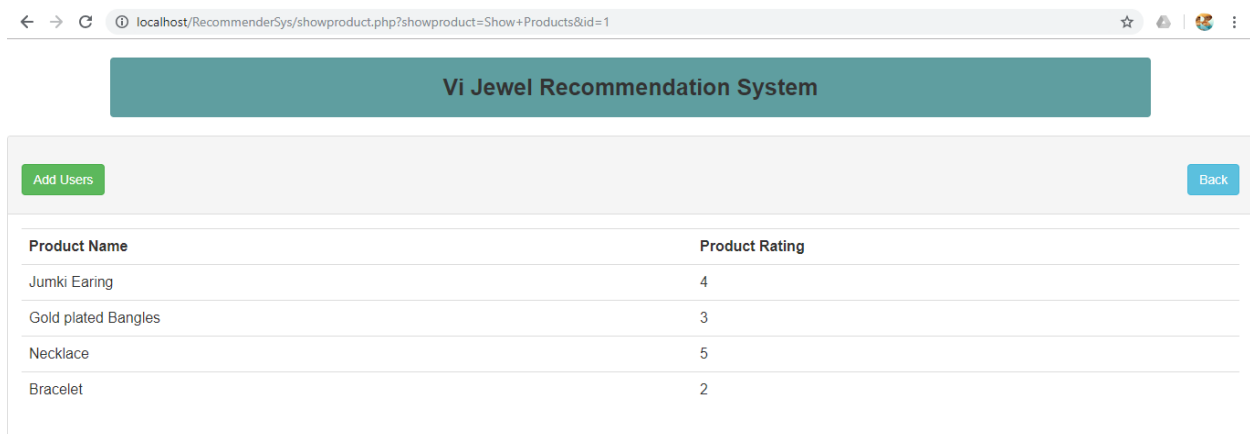
The user clicks on the 'Show Product' button to view the product and its rating given by each user.

Figure 7:Show product with ratings

The products along with its respective ratings:

Figure 8: Product Names with its respective ratings

A **matrix** is created to generate the product name and rating for each user so as the data can be access and compare later to calculate its similarity.

```
Array
(
    [Johnny] => Array
        (
            [Jumki Earing] => 4
            [Gold plated Bangles] => 3
            [Necklace] => 5
            [Bracelet] => 2
        )

    [Mike] => Array
        (
            [Gold plated Bangles] => 2
            [Ring] => 5
            [Necklace] => 3
            [Belly Chain] => 2
            [Anklet] => 4
        )

    [Nandishta] => Array
        (
            [Jumki Earing] => 4
            [Gold plated Bangles] => 2
            [Necklace] => 4
            [Ring] => 4
            [Anklet] => 1
            [Belly Chain] => 5
            [Toe Ring] => 1
        )

    [Jenita] => Array
        (
            [Belly Chain] => 4
            [Gold plated Bangles] => 2
            [Jumki Earing] => 5
            [Ring] => 4
            [Anklet] => 2
            [Toe Ring] => 2
        )

    [Din] => Array
        (
            [Jumki Earing] => 2
            [Gold plated Bangles] => 3
            [Necklace] => 4
            [Ring] => 3
            [Anklet] => 3
            [Belly Chain] => 4
        )
```

We have to calculate similarity between users using the:

Euclidean Distance Formula

$$d(x,y) = \sqrt{\sum_{i}^{n} (x_i - y_i)^2}$$

To calculate the similarity of 'Johnny' to other user, the following codes are used:

```
showproduct.php ☒ | recommendation.php ☒ | recommend.php ☒

 1    <?php
 2
 3      function similarity_distance($matrix,$person1,$person2)
 4      {
 5          $similar=array();
 6          $sum=0;
 7
 8          foreach($matrix[$person1] as $key=> $value)
 9          {
10              if(array_key_exists($key,$matrix[$person2]))
11              {
12                  $similar[$key]=1;
13              }
14          }
15          if($similar==0)
16          {
17              return 0;
18          }
19          foreach($matrix[$person1] as $key=> $value)
20          {
21              if(array_key_exists($key,$matrix[$person2]))
22              {
23                  $sum=$sum+pow($value-$matrix[$person2][$key],2);
24              }
25          }
26          return 1/(1+sqrt($sum));
27      }
28
29
```

```
32
33
34      function getRecommendation($matrix,$person)
35      {
36          foreach($matrix as $otherperson=>$value)
37          {
38              if($otherperson!=$person)
39              {
40                  $sim=similarity_distance($matrix,$person,$otherperson);
41
42                  var_dump($sim);
43              }
44          }
45      }
46
47
48    ?>
```

```php
<?php

include("db.php");
include("recommend.php");

$product=mysqli_query($con, "select * from user_products");

$matrix=array();

while($p=mysqli_fetch_array($product))
{
    $users=mysqli_query($con,"select username from users where id=$p[user_id]");
    $username=mysqli_fetch_array($users);

    $matrix[$username['username']][$p['product_name']]=$p['product_rating'];

}

//echo "<pre>";
//print_r($matrix);
//echo "</pre>";

getRecommendation($matrix,"Johnny");
```

The output for the similarity of 'Johnny' to other user is shown below:

← → C   ⓘ localhost/RecommenderSys/recommendation.php

float(0.30901699437495) float(0.4142135623731) float(0.4142135623731) float(0.30901699437495)

	Mike	Nandishta	Jenita	Din
Johnny	0.3090	0.4142	0.4142	0.3090

To use collaborative filtering method, we made use of the following formula:

$$p(u,i) = \frac{\sum_{N \in similarTo(i)} (S_{i,N} \times R_{u,N})}{\sum_{N \in similarTo(i)} (|S_{i,N}|)}$$

p(u,i) means we are going to predict the rating user u will give item i.

Ru,N is the rating user u gave item N

Si,N is the similarity between i and N

```php
33
34   function getRecommendation($matrix,$person)
35   {
36       $total=array();
37       $simsum=array();
38       $ranks=array();
39
40       foreach($matrix as $otherperson=>$value)
41       {
42           if($otherperson!=$person)
43           {
44               $sim=similarity_distance($matrix,$person,$otherperson);
45
46               foreach($matrix[$otherperson] as $key=>$value)
47               {
48                   if(!array_key_exists($key,$matrix[$person]))
49                   {
50                       if(!array_key_exists($key,$total))
51                       {
52                           $total[$key]=0;
53                       }
54                       $total[$key]+=$matrix[$otherperson][$key]*$sim;
55
56                       if(!array_key_exists($key,$simsum))
57                       {
58                           $simsum[$key]=0;
59                       }
60                       $simsum[$key]+=$sim;
61                   }
62               }
63           }
64       }
65
66       foreach($total as $key=>$value)
67       {
68           $ranks[$key]=$value/$simsum[$key];
69           array_multisort($ranks,SORT_DESC);
70           return $ranks;
71       }
72   }
```

Recommended product for 'Johnny' is calculated as shown below:

```php
23   var_dump(getRecommendation($matrix,"Johnny"));
24
```

The output for the calculated recommended product is:

← → C ⓘ localhost/RecommenderSys/recommendation.php

array(4) { ["Ring"]=> float(4) ["Belly Chain"]=> float(3.8590902828461) ["Anklet"]=> float(2.3545462895385) ["Toe Ring"]=> float(1.5) }

Calculating recommendation for all users is as follows:

```html
<div class="panel-body">
    <table class="table table-stripped">
        <th> Product Name</th>
        <th> Product Rating</th>


        <?php
        $recommendation=array();
        $recommendation=getRecommendation($matrix,$username['username']);

        foreach($recommendation as $product=>$rating)
        {
        ?>
        <tr>
            <td><?php echo $product; ?> </td>
            <td><?php echo $rating; ?> </td>


        </tr>
        <?php } ?>
    </table>

</div>
```
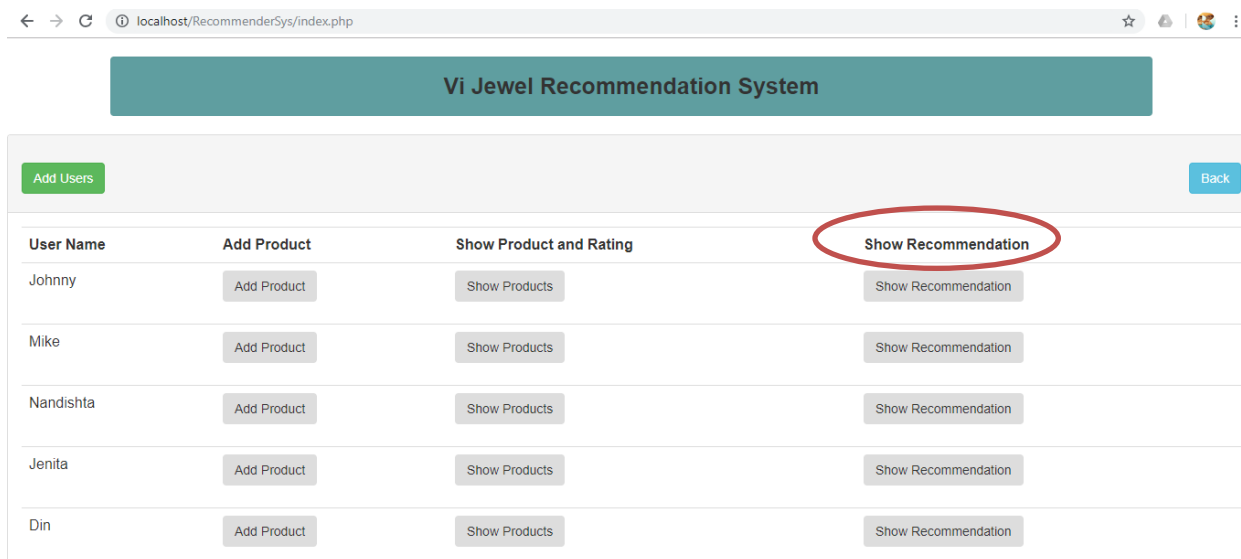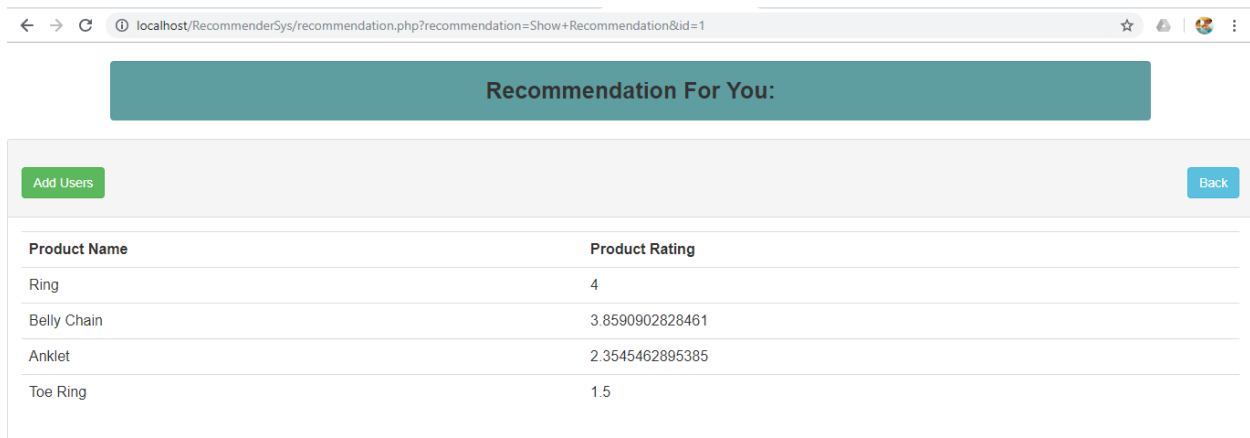
All the recommendations calculated is shown under the 'Show Recommendation' column and the user has to click the 'Show Recommendation' button to be able to view which product is recommended for him/her.

When the user has clicked on the 'Show Recommendation' button, the following page is open and all the recommended products along with their respective rating is shown.

Figure 9:Recommendation list for one user