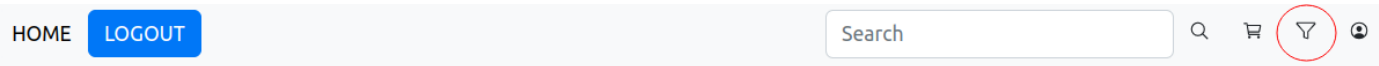


Description of the web application

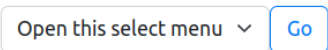
1. Features in the original proposal

- **Recommend and range categories to users based on different attributes in the filter, such as size and color.**

You can click the `filter` button (showed on the upper right corner of the image) to open the `filter page` and select specific attribute and click go.



Then you can see the result of the filter showed below.



Product Color	Number of Purchases
Blue	37
Pink	35
Black	32
Red	30
Green	16
Brown	6

- **Show the range of the age and the proportion of gender of customers who ordered some specific categories.**

Similarly, you can also select the age and the gender attribute in the filter.

- **Search for the reviews for a specific supplier**

The `filter` button also allow the search of some specific suppliers.

- **The user can collect or like one product.**

If you want to collect some products, you can click the `star` button on the left of every product.

HOME	LOGOUT	<input type="text" value="Search"/> 🔍 🛒 🏠 👤				
Product	Description	Color	Size	Price	Rating	
Organic Cotton Sheet Set	Organic Cotton Sheet Set: Our organic cotton sheets are soft, breathable, and free of harmful chemicals for a restful and healthy sleep.	Pink	M	49.99	3 stars. It is an okay product, but I would not necessarily recommend it.	☆

If the product has been added into the cart, the web application will notice you when you try to repeatedly add the same product.

- **Search for ratings of the specific product**

Similarly, you can see the rating while searching different products in the **search page**.

2. New features

- **Search bar**

HOME	LOGOUT	<input type="text" value="Search"/> 🔍 🛒 🏠 👤
------	--------	---

This is the **home page** of the web application, you can type in the name of the product you want to buy, and click the **search** button. You can see its information on the **search page** if it exists in our database.

If you don't type in anything and click the **search** button directly, you can see all products in our database.

Product	Description	Color	Size	Price	Rating	
Natural Lavender Body Wash	Natural Lavender Body Wash: Our all-natural body wash is infused with lavender and chamomile for a calming and relaxing shower experience.	Red	S	12.99	5 stars. Excellent product! I would definitely buy it again.	☆
Natural Lavender Body Wash	Natural Lavender Body Wash: Our all-natural body wash is infused with lavender and chamomile for a calming and relaxing shower experience.	Red	S	12.99	5 stars. I absolutely love this product! It exceeded my expectations.	☆
High-Performance Blender	High-Performance Blender: This powerful blender features a 1500-watt motor and a durable glass pitcher for smooth and efficient blending.	Blue	S	89.99	4 stars. This product is great, but there is still room for improvement.	☆
Organic Cotton Sheet Set	Organic Cotton Sheet Set: Our organic cotton sheets are soft, breathable, and free of harmful chemicals for a restful and healthy sleep.	Pink	M	49.99	3 stars. It is an okay product, but I would not necessarily recommend it.	☆
Portable Bluetooth Speaker	Portable Bluetooth Speaker: This compact and portable speaker delivers crisp and clear sound with Bluetooth connectivity for easy streaming.	Red	M	39.99	2 stars. I was disappointed with this product. It did not live up to my expectations.	☆
Premium Coffee Beans	Premium Coffee Beans: Our premium coffee beans are sourced from sustainable farms and roasted to perfection for a rich and flavorful cup.	Blue	L	16.99	1 star. This product was a complete waste of money. I regret buying it.	☆

- User management and title bar

At the first time you open the web application, you can see the **login page**.

LOGIN

SIGN UP

Login

Email Address

Enter your email

Password

Enter your password

Login

Don't have an account? [Sign up here](#)

If you don't have the account of this web application, you should click the **Sign up here** button to registrate.

[LOGIN](#)[SIGN UP](#)

Sign Up

Please fill in this form to create an account.

Email Address

Username

Password

Your password must be 8-20 characters long, contain letters and numbers, and must not contain spaces, special characters, or emoji.

Password Confirm

☐ I agree all statements in [Terms of service](#)

[Sign Up](#)

Have already an account? [Login here](#)

After registration, you can login to the [home page](#). There are several buttons on the title bar, whose features will be described later.

[HOME](#)[LOGOUT](#)

Logged in successfully!

[search](#)





- **Cart management**




Continuing from the collection function stated above, the [cart](#) button on the upper right of the [home page](#) can open the [cart page](#).

[search](#)

In this page, you can see all products added in the cart and choose whether to delete it by clicking the `delete` button on the right of the product.

HOME [LOGOUT](#)



Product Name	Description	Color	Size	Price	
Natural Lavender Body Wash	Natural Lavender Body Wash: Our all-natural body wash is infused with lavender and chamomile for a calming and relaxing shower experience.	Red	S	12.99	
Portable Bluetooth Speaker	Portable Bluetooth Speaker: This compact and portable speaker delivers crisp and clear sound with Bluetooth connectivity for easy streaming.	Red	M	39.99	
Organic Cotton Sheet Set	Organic Cotton Sheet Set: Our organic cotton sheets are soft, breathable, and free of harmful chemicals for a restful and healthy sleep.	Pink	M	49.99	

- **User profile**

You can click the `profile` button to visit the `profile page`. All the information in this page is related to the email you input when logging in. If you are a user of the shopping website, `profile page` will show your personal information, shipping address, and pass orders. If you are not a user of the shopping website, `profile page` will only show your username and email address of this website.

Matthew

matthewjackson@bp.com

ACCOUNT INFORMATION

First Name	Last Name	Gender	Age
Matthew	Jackson	F	48
Primary Phone Number	Secondary Phone Number		
363-430-1813	559-181-3744		

SHIPPING ADDRESS

Shipping Address 1

Street Address		
538 Cedar Ave		
City	State	Zip Code
Katy	TX	77449

Shipping Address 2

Street Address		
5 Spohn Circle		
City	State	Zip Code
Arlington	TX	62137

PAST ORDERS

ORDER O0023

Product	Shipped By	Order Status
All-Natural Shampoo	United Parcel Service	Delivered

ORDER O0025

Product	Shipped By	Order Status
All-Natural Shampoo	Federal Express	Delivered

For testing purpose, we have already create a few accounts:

1. Users of the shopping website -- email: matthewjackson@bp.com | password: database
2. Users not from the shopping website -- email: reianna@gmail.com | password: database

3. Missing part

- **Shipping speed of every order**

We replace it with the **order status** attribute.

Two interesting pages

1. Filter page

In `filter.html` file, we use this code to receive input values from the web page.

```

<form action="{{ url_for('views.filter') }}"class="form-inline" method="POST">
  <div class="form-group">
    <div class="input-group">
      <select name= "select-filter" class="form-select" aria-label="Default select example">
        <option selected value="0">Open this select menu</option>
        <option value="1">Number of Purchases</option>
        <option value="2">Size</option>
        <option value="3">Color</option>
        <option value="4">Gender</option>
        <option value="5">Age</option>
        <option value="6">Supplier</option>
      </select>
    </div>
    <input
      type="hidden"
      name="tbn-pressed"
      value="select-filter"
    />
    <button type="submit" class="btn btn-outline-primary">Go</button>
  </div>
</form>

```

As you can see in the code showed above, every option in the `select` menu has its unique value, when you click the button in this `form`, the `views.py` file located in the backend can get the value of the option by using this code

```
filter_parameter = request.values.get('select-filter')
```

After that, the program can execute postgresql query and post the result back to `filter.html` file based on different value.

```

if filter_parameter == "1":
    with engine.connect() as g.conn:
        cursor = g.conn.execute(text("""SELECT name, SUM(quantity_ordered) as Quantity_Ordered
                                         FROM order_contains
                                         NATURAL JOIN product
                                         GROUP BY name
                                         ORDER BY Quantity_Ordered DESC
                                         LIMIT 10"""))

        rowResults = []
        for result in cursor:
            rowResults.append({'product_name': result[0],
                              'quantity': result[1]})
        cursor.close()

    return render_template("filter.html", recentRecords=rowResults)

```

Eventually, you can see different results when you choose different option on this page. We think this is an ingenious design.

2. Collect page

The design of this page can be divided into two sections. The first is to display the products in a specific user's shopping cart. The second is to delete a specific product in the shopping cart as his wish.

- **Filter products**

We use session module in flask library to ensure the web application can remember the user's login email address when he enter the **collect page**. Then we use postgresql query language in `views.py` file to get information about the products in the shopping cart corresponding to the email address, and send it to `collect.html` file.

```
with engine.connect() as g.conn:

    params = {}
    params['email'] = userp

    cursor = g.conn.execute(text("""SELECT *
                                   FROM collects C
                                   NATURAL JOIN product P
                                   WHERE C.email = :email
                                   """), params)

    rowResults = []
    for result in cursor:
        print(result)
        rowResults.append({'product_id': result[0],
                           'product': result[5],
                           'description': result[2],
                           'color': result[6],
                           'size': result[7],
                           'unit_price': result[4]})

    cursor.close()
    return render_template("collect.html", recentRecords=rowResults)
```

In `collect.html` file, we use a table to display information from backend program.


```

<table class="table table-hover" cellspacing="0">
  <thead>
    <tr>
      <th scope="col">Product Name</th>
      <th scope="col">Description</th>
      <th scope="col">Color</th>
      <th scope="col">Size</th>
      <th scope="col">Price</th>
    </tr>
  </thead>
  <tbody>
    {% if recentRecords == [] %}
    <p>You have not collected any product!</p>
    {% else %} {% for rows in recentRecords %}
    <tr>
      <td>{{ rows.product }}</td>
      <td>{{ rows.description }}</td>
      <td>{{ rows.color }}</td>
      <td>{{ rows.size }}</td>
      <td>{{ rows.unit_price }}</td>
      <td>
        <form action="{{ url_for('views.delete_collect')}}" method="POST">
          <input
            type="hidden"
            name="btn-pressed"
            value="{{rows.product_id}}"
          />
          <button type="submit" class="btn btn-outline-default waves-effect">
            <ion-icon name="trash-outline"></ion-icon>
          </button>
        </form>
      </td>
    </tr>
    {% endfor %} {% endif %}
  </tbody>
</table>

```

• Delete products

It's worth nothing that there is a `delete` button in such table. When you click on it, the code will redirect to the `delete_collect` function in `views.py` file. The backend file will also get the specific ID of the product you want to delete at the same time. Based on this ID, the postgresql query can delete the specific information in the collect table created before and notice you with a flash on the web application.

```

product_id = request.form.get("btn-pressed")
with engine.connect() as g.conn:

    params = {}
    params['email'] = userp
    params['product_id'] = product_id

    g.conn.execute(text("DELETE FROM collects WHERE email = :email AND product_id = :product_id"), params)
    g.conn.commit()

    flash('Deleted! Search the website for more product you like.', category='success')

```