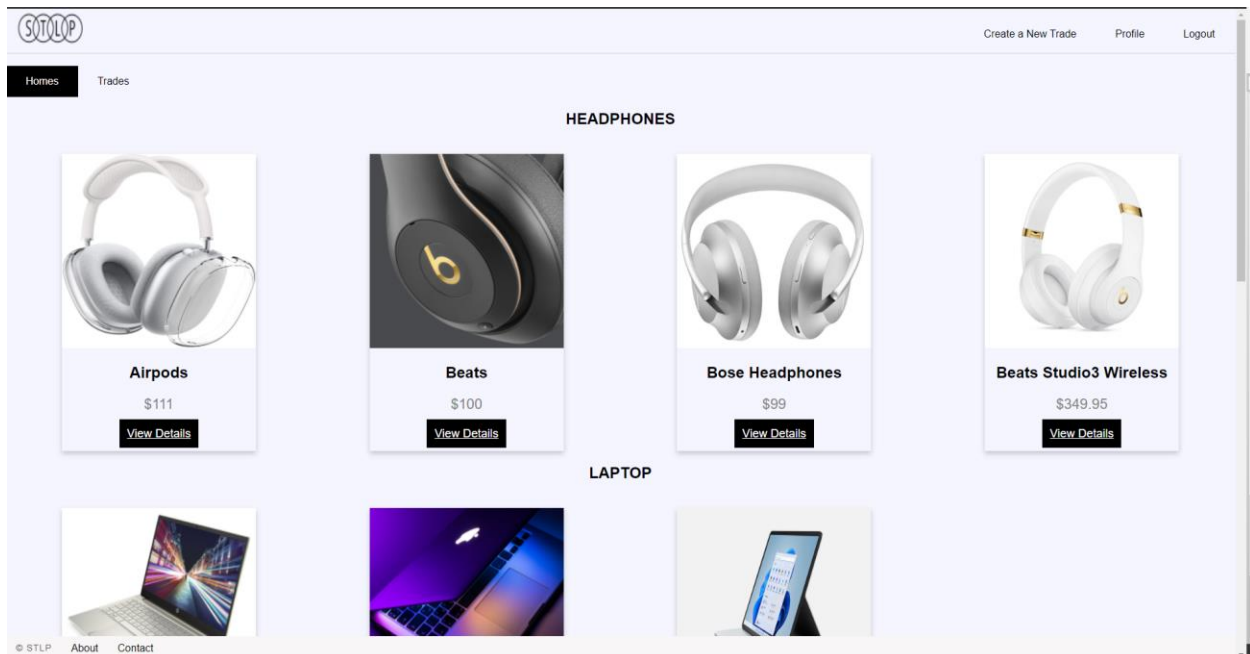


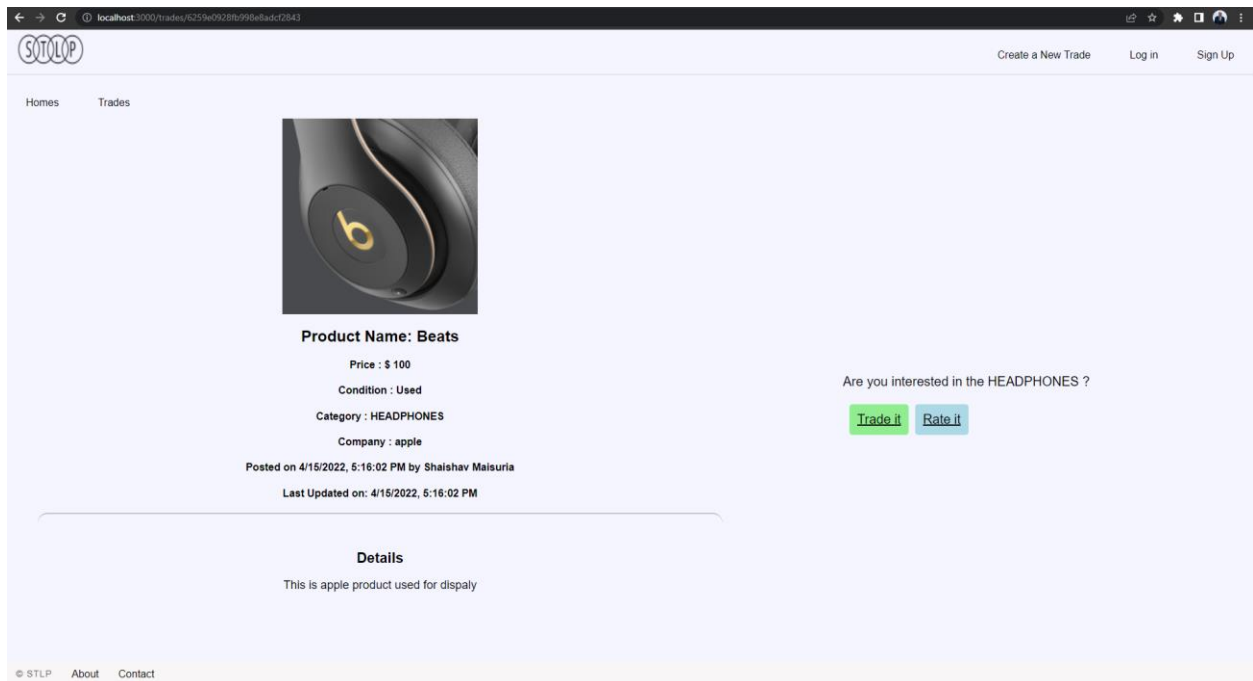
Index page / home page



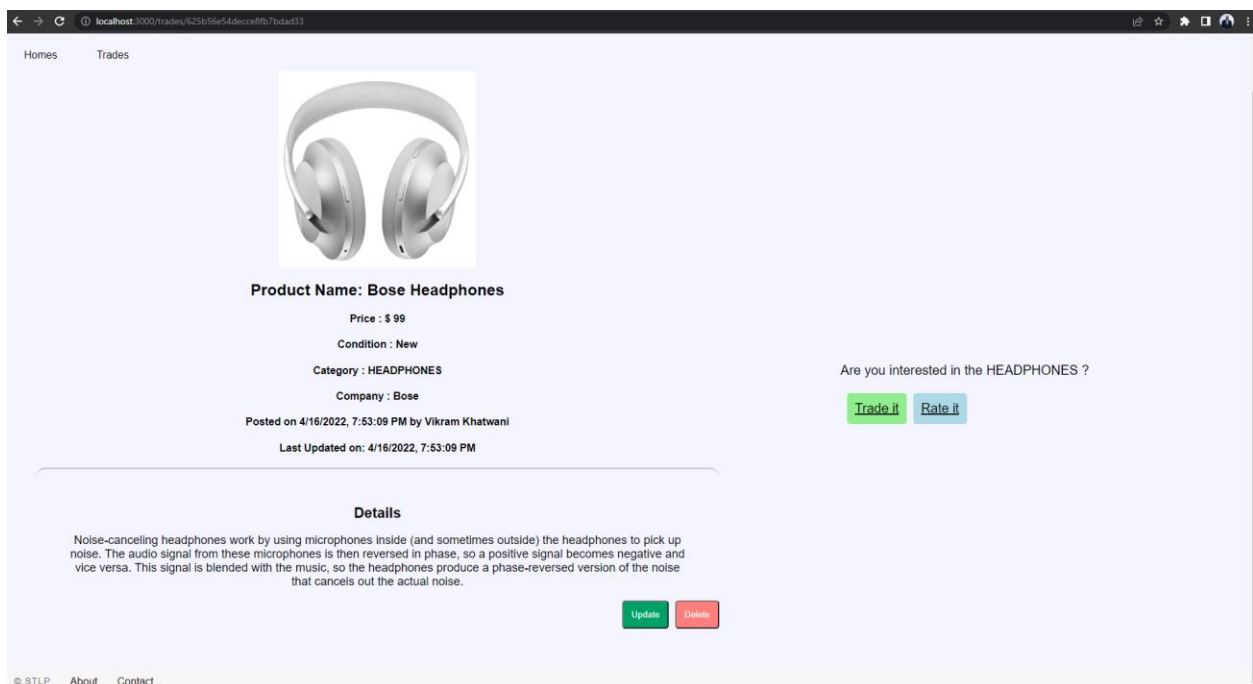
Trades page with all list of categories and products in sorted manner



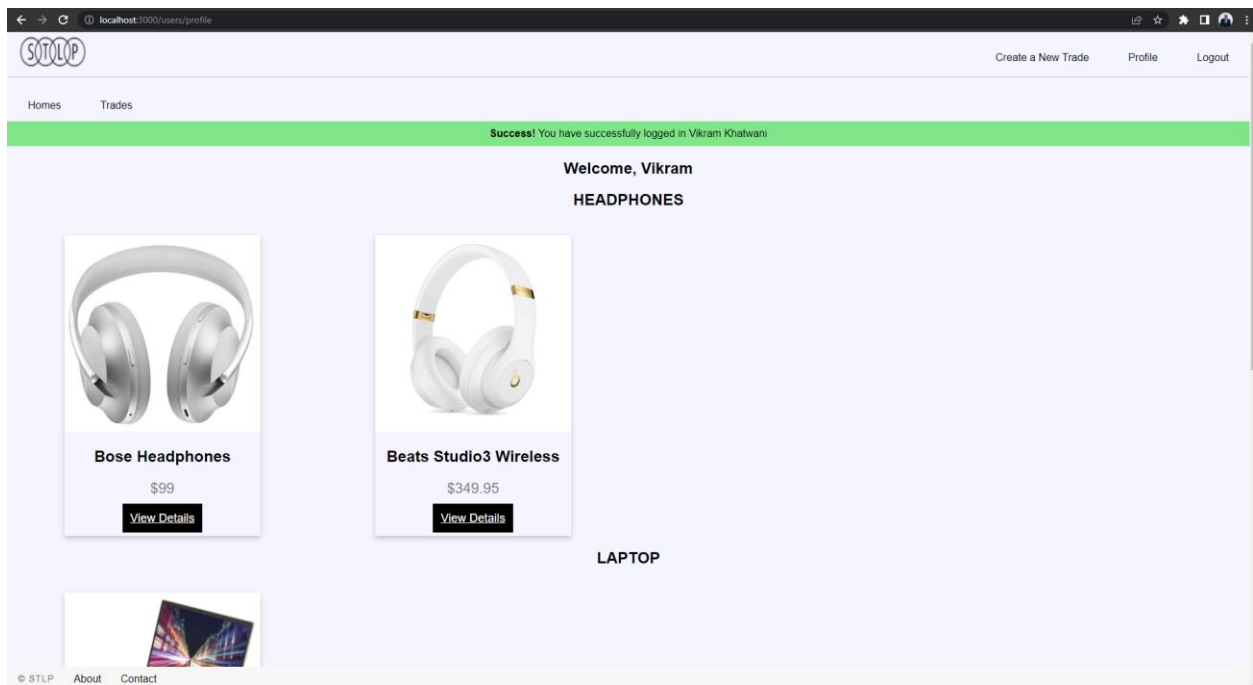
If user is not the author of the story update and delete option Is not provided to the current user



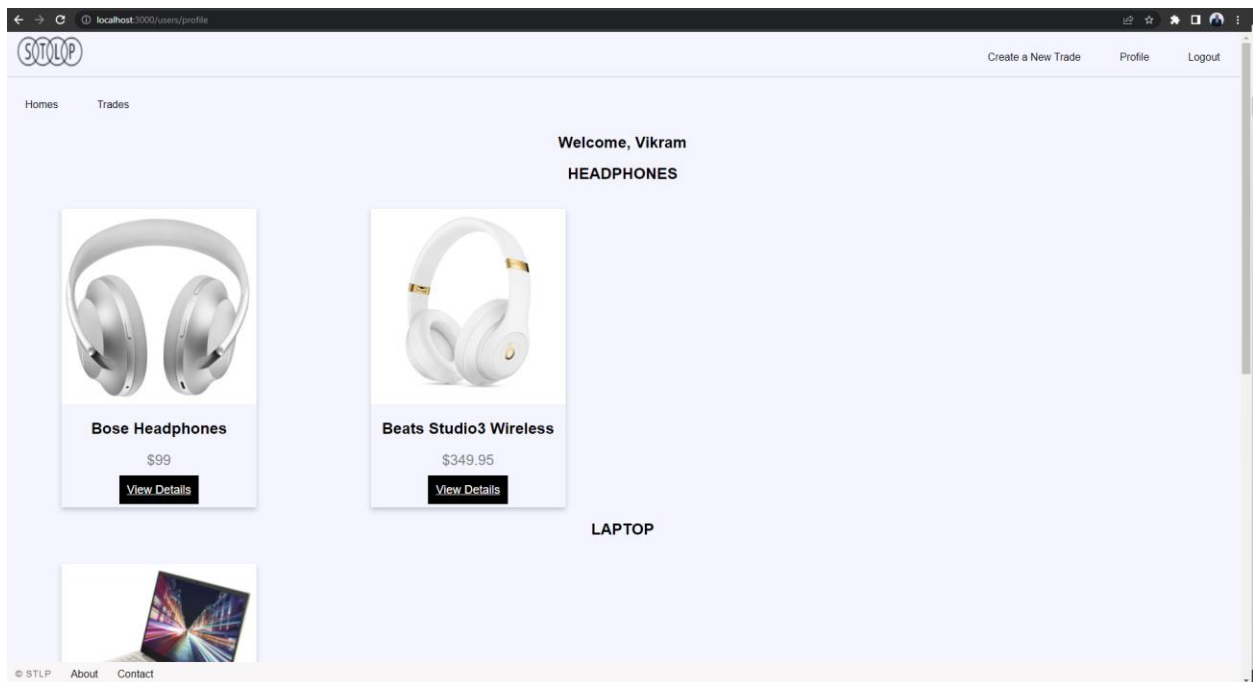
When user is author of the story we show update and delete option



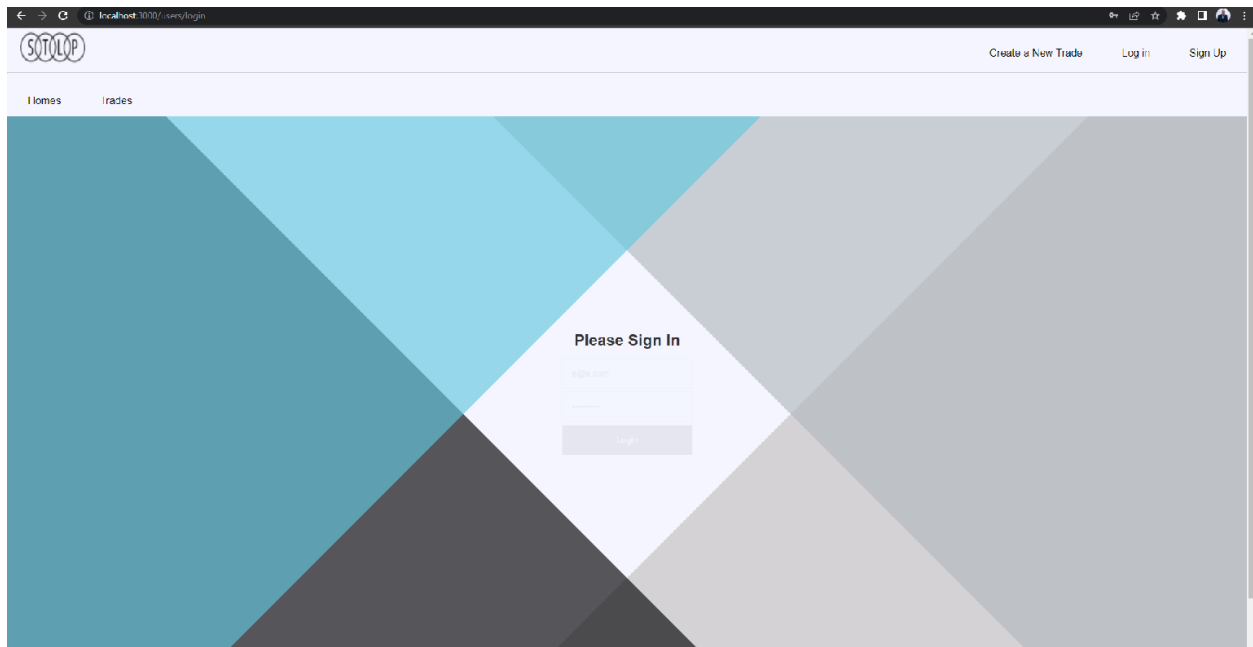
User flash messages are shown as expected



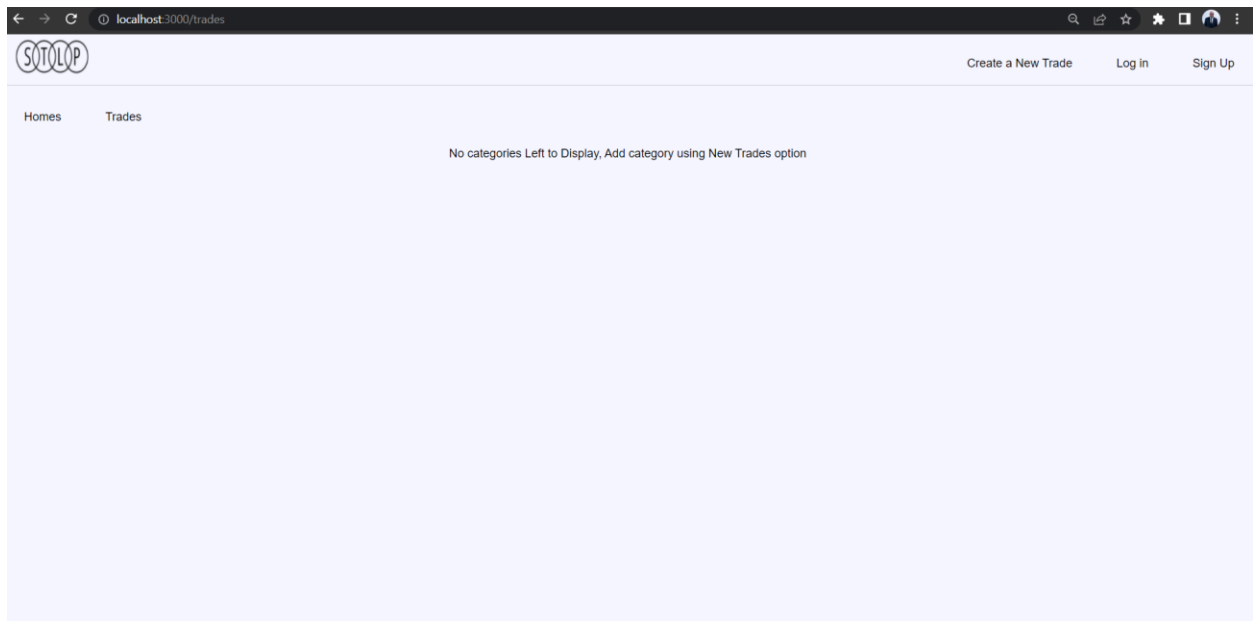
User created trades are displayed to the user in sorted category order



Hovering over the login and singup page display animation and allows user to input details



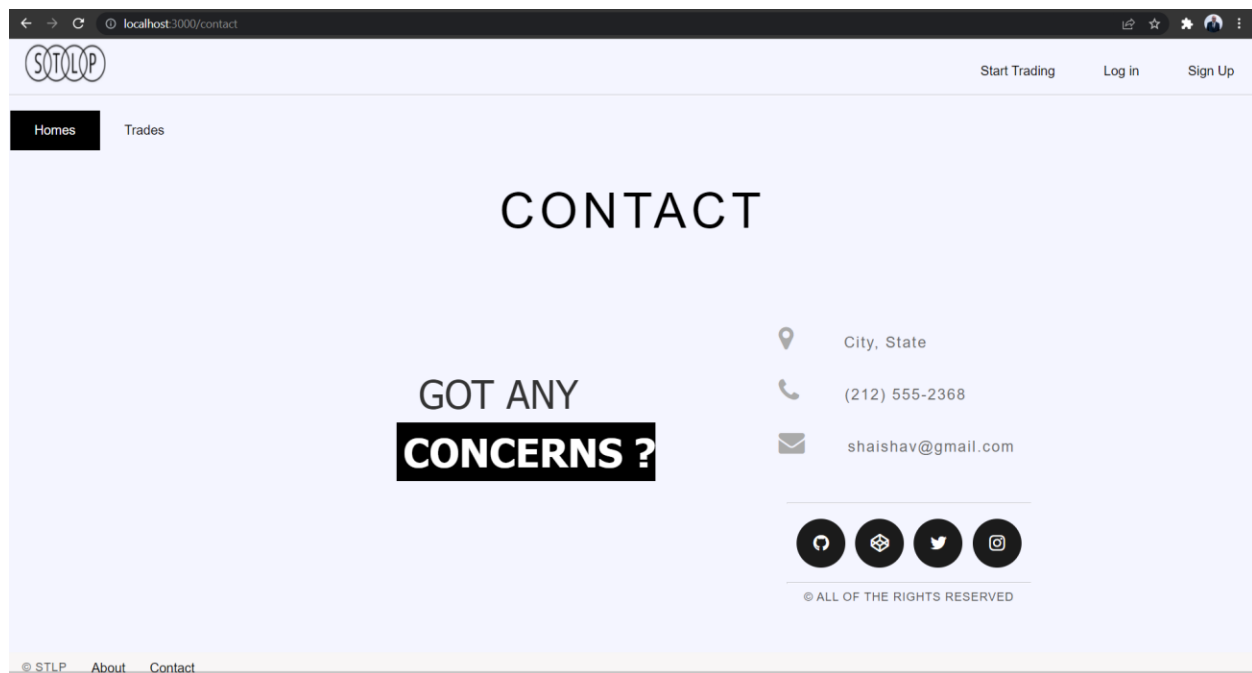
Routes: <http://localhost:3000/trades>



Create about page and added svg animation in the footer therefore created all extra designerFooter.ejs just for about page



Contact page added with animation



All errors are handled properly

## Invalid id



## Wrong id



MVC pattern followed

No errors and all working code

```

<td colspan="2" class="align-right"><button login/button></td>
</tr>
</table>
</form>

<h2 class="middle"> Please Sign In</h2>
<div class="container" onclick="onclick">

  <div class="top"></div>
  <div class="bottom"></div>
  <div class="center">

    <!-- <h2>Please Sign In</h2> -->
    <form action = '/users/login' method = 'POST'>
      <table>
        <tr>
          <td><input type="email" id="email" name="email" class = "fixed-width" required placeholder="email"></td>
        </tr>
        <tr>
          <td><input type="password" id="password" name="password" class = "fixed-width" required placeholder="password"></td>
        </tr>
        <tr>
          <td colspan="2"><input type="submit" value="login"> </td>
        </tr>
      </table>
    </form>

    <h2><br/></h2>
  </div>
</div>

```

```

{
  "createdAt": "2022-04-17T08:00:23.530Z",
  "updatedAt": "2022-04-17T08:00:23.530Z",
  "v": 0,
  "_id": "new ObjectId('625859614dece0fb7bdad58')",
  "category": "Laptop",
  "name": "Surface Laptop Studio",
  "cost": "1422.00",
  "image": "https://c.s-microsoft.com/en-us/OSImages/Surface_F21_DeviceFamily_SG3_V1.jpg?version=daebebe-72a2-17e3-4f25-1c17a7208787",
  "condition": "new",
  "company": "Microsoft",
  "details": "Set your imagination free on the most powerful Surface Laptop, designed to light up the best of Windows 11. Boundary-pushing design lets you flex your creative muscle on the sleek 14.4\"

```

Note the model structure I have contains following format

Contains schema,

Category, name, author, cost, image, condition, company all these field are added and have required as validator. In addition, we details component is also added with required validator and minLength validator

- ➔ Each page was validated
- ➔ Trades page showing list of trades is sorted based on categories.
- ➔ Each page is connected apps followed to routes followed to controllers
- ➔ Mongoose connection is formed and all errors are handled
- ➔ Errors tested using postman and site testing
- ➔ User Validations implemented
- ➔ User error validation checked and working with all 400 errors given
- ➔ User created new trade are displayed in sorted order to the user
- ➔ Flash messages are added as per requirement
- ➔ Logout implemented properly and having no issue.

## 1. Explanation of additional features, if any.

Added the user functionality such as models, routes, controllers, and view templates related to user (login, singup user, profile)

## 2. Explanation of status, stopping point, and issues if incomplete.

**Status:** everything is working as per expectation, all requirements are met

**Stopping point:** CSS animation issues was where I got stucked, but now everything is resolved.

**Issues:** No issues all complete, Everything solved and working,

## 3. What was the most challenging parts of the assignment. How did you overcome all or some of those challenges?

Most challenging part was that, I had to understand how session works and how we created, delete and change sessions when someone get on the website to login in the website at the user.

## 4. What was the most exciting/interesting concepts you encountered while completing this project?

Creating the new user are likely most fun part where I was able to create, authenticate and use see product details based on the user item creation.

## 5. Description of the RESTful routes for requests associated with trading items. (you can combine this with the screenshots in the previous point)

/trades > contains all the list of trades

/newtrade > contains form to create a newtrade when user is logs in

/about > html page for about page

/contact > is used for contact page for html

/trades//id is used to display the each trade

/trades/id/edit to edit the product with id

/users/profile > contains user profile when user logs in

/users/login > contains view templated for user to login

/users/new > contains view templated for user to signup as new account.