1. **Project Objective**

Kitchen Story is an e-commerce portal that lets people shop basic food items on their website. The website needs to have the following features:

* A search form in the home page to allow entry of the food items to be purchased by the customer.
* Based on item details entered, it will show available food items with price.
* Once a person selects an item to purchase, they will be redirected to the list of available items. In the next page, they are shown the complete breakout of the order and details of the payment to be made in the payment gateway. When payment is done, they are shown a confirmation page with details of the order.

**For the above features to work, there will be an admin backend with the following features:**

* Admin login page where admin can change password after login if he wants to
* A master list of food items available for purchase
* A functionality to add or remove food items

1. **Application Name**

Kitchen Story, an e-commerce portal.

1. **Developer Details**

Anurag Pal, 216 Java SL (Evening Batch), Phase-4 Final Assessment

1. **Product’s Capabilities**

Kitchen Story is an e-commerce portal where users can search for products and purchase them through a payment gateway.

On the admin side, the admin has the capability to add or remove products. The admin can also see the list of products as well. On the user side, the user can search for products and purchase them.

Be it the admin or the user, they both have to login using the given username and password. If the credentials are wrong, an error message is displayed, otherwise, the admin or the user is then navigated to the dashboard.

For the admin, the dashboard displays four links, add product, list products, change password and logout. To add a product, a form is displayed. The admin has to provide the product ID, product name, price and the product’s brief description. To display all the products, the admin has to click on list products. The products are then displayed in a table format. A button is given beside every product. By clicking on this button, the admin can remove one or more products. To change the password, a form is displayed asking the admin for existing username and new password. If the username entered is incorrect, then an error message is displayed, or else a success message is displayed and the admin is asked to login again. Finally, if the admin clicks on logout, then a success message is displayed and the admin is logged out.

For the user, the dashboard displays three links, search product, change password and logout. To search for a product, the users have to provide the product name. The list of product names matching the name provided by the user is then displayed below. Beside every product, a button named Purchase is given. Clicking on that button will navigate the user to the purchase section. Here, the details of the product that the user wants to purchase is displayed in a table format along with a text box asking the user for the number of items. After the user provides the number of items, the user is then navigated to the payment gateway where the total price is displayed along with a success message. To change the password, a form is displayed asking the user for existing username and new password. If the username entered is incorrect, then an error message is displayed, or else a success message is displayed and the user is asked to login again. Finally, if the user clicks on logout, then a success message is displayed and the user is logged out.

1. **GitHub Link**

The github link for this product is as follows:

1. **Core Concepts Used**
2. Different components are used for different aspects of Kitchen Story.
3. Three classes have been created for Admin, Product and User. In these classes, Admin and User contains username and password, and Product contains id, name, price and description.
4. Three services have also been created, admin service, user service and product service.
5. The different components used are as follows:
   1. Add product component for adding a product in admin side. A form is displayed and it is validated. After this, the addproduct method in product service is called. In this method, the product details are pushed into the Product array.
   2. Admin login component for logging in the admin with correct username and password. A form is displayed and it is validated. After this, the login method in admin service is called. In this method, the username and password is validated.
   3. Admin dashboard component to display the admin dashboard.
   4. Admin logout component to logout the admin.
   5. Change password component, in which the admin can change their password. A form is displayed and it is validated. After this, the changepwd method of admin service is called. In this method, the new password is overwritten into the Admin array.
   6. Change password user component, in which the user can change their password. A form is displayed and it is validated. After this, the changeupwd method of user service is called. In this method, the new password is overwritten into the User array.
   7. List component to display all the products of the Product array. In this component, getAllProducts method of product service is called.
   8. Order details component serves as a payment gateway. In this component, the total price is displayed along with a success message.
   9. Purchase component to display the list of the products which the user wants to purchase, along with a text box asking user for the number of items required.
   10. Remove product component to remove a product from the Product array using splice method.
   11. Search product component to display all the products corresponding to the product name provided by the user.
   12. User login component for logging in the user with correct username and password. A form is displayed and it is validated. After this, the login method in user service is called. In this method, the username and password is validated.
   13. User dashboard component to display the user dashboard.
   14. User logout component to logout the user.
6. **Generic Features**
7. Admin login with correct username and password.
8. Admin dashboard displaying menu.
9. Add product through a form in which id, name, price and description is provided by the admin.
10. List all the products present.
11. Remove any product.
12. Change password of the admin.
13. Admin logout.
14. User login with correct username and password.
15. User dashboard displaying menu.
16. Search product by providing the product name. Products corresponding to the product name are displayed below.
17. Purchase product by clicking on the Purchase button. User also needs to provide the number of items required.
18. Payment gateway displaying the total price.
19. Change password of the user.
20. User logout.
21. **Conclusion**

This application is mainly designed using Angular, JavaScript and CSS. The concept of angular components is used to create different aspects of the application while JavaScript and CSS are used together to create the front-end of the application.

**Unique Selling Points (USPs)**

1. Lots of different features helps the admin to carry out the daily operations smoothly.
2. Adding and removing products is a lot more user friendly. Also, listing all the products helps in identifying any misinformation that was wrongly provided.
3. Changing password is simple while also effective as the valid username has to be provided as well.
4. Searching for products is a smooth process as the user only needs to give the product name.
5. Error and Success messages have also been provided to guide the user or admin while accessing this application.