

CPSC 471 Functional Design

Zhijie Xia, Yuan Liu, Zixing Wei, G-69

March 27, 2022

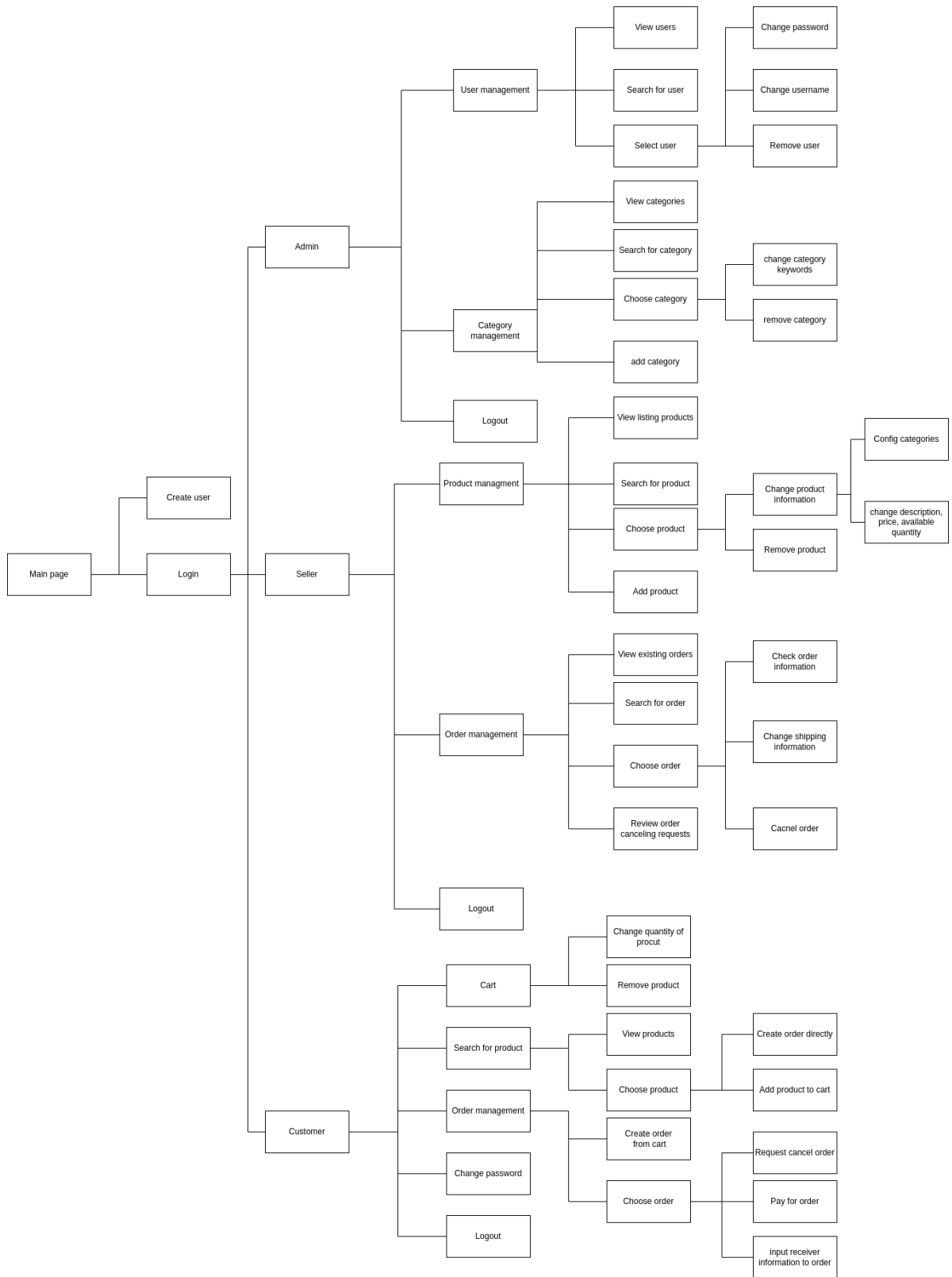
Notes:

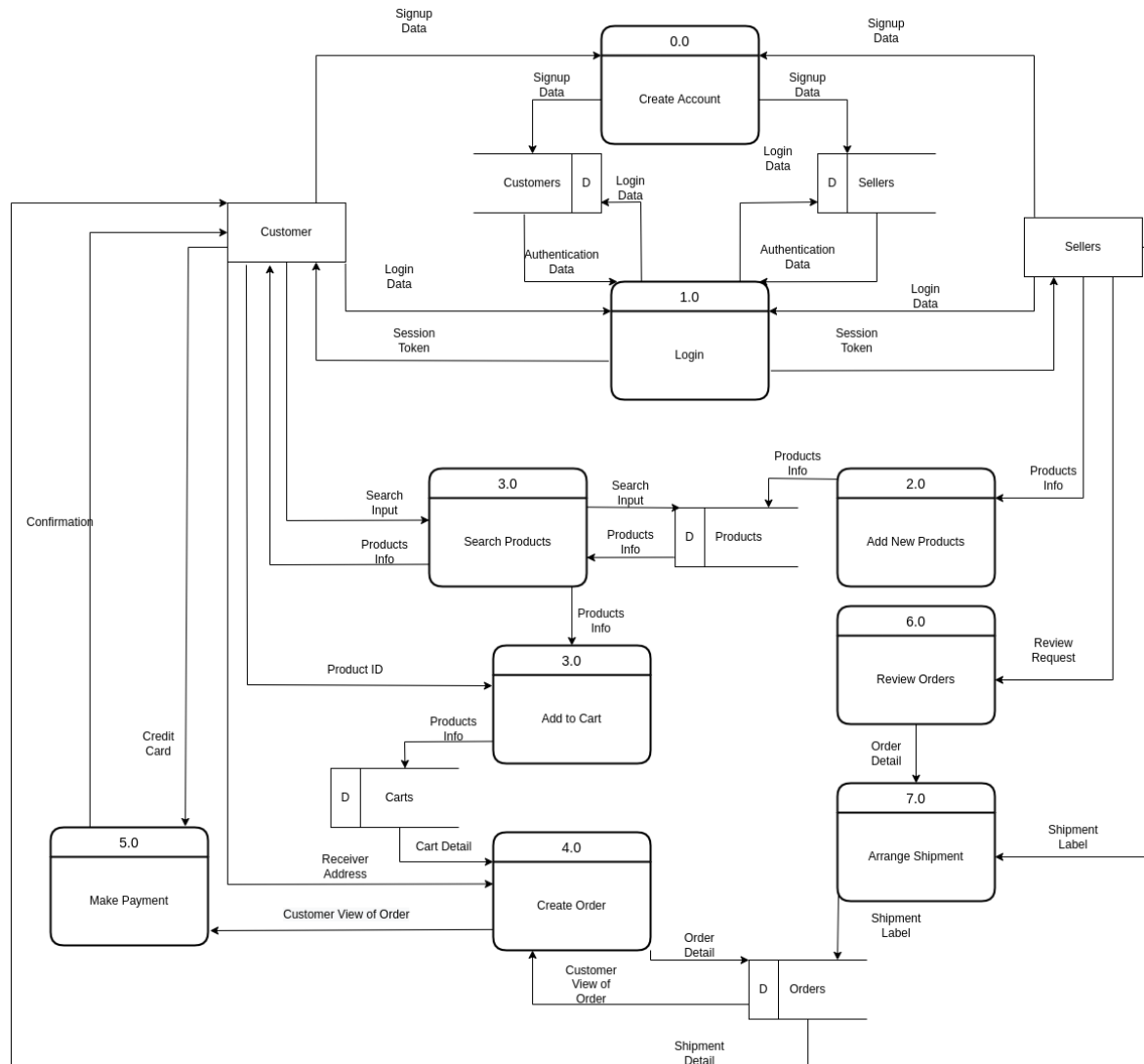
Our project **YourStore** will be built with **HTML,CSS,Javascript, Bootstrap,ReactJS, ExpressJS, MongoDB**.

At this stage, we have initial design on the API, what data would be stored, and the static UI. However, software engineering is an iterative refining process, we would expect some changes down the way, and some initial design might be modified in the Final Submission.

HIPO & DFD
Contributors

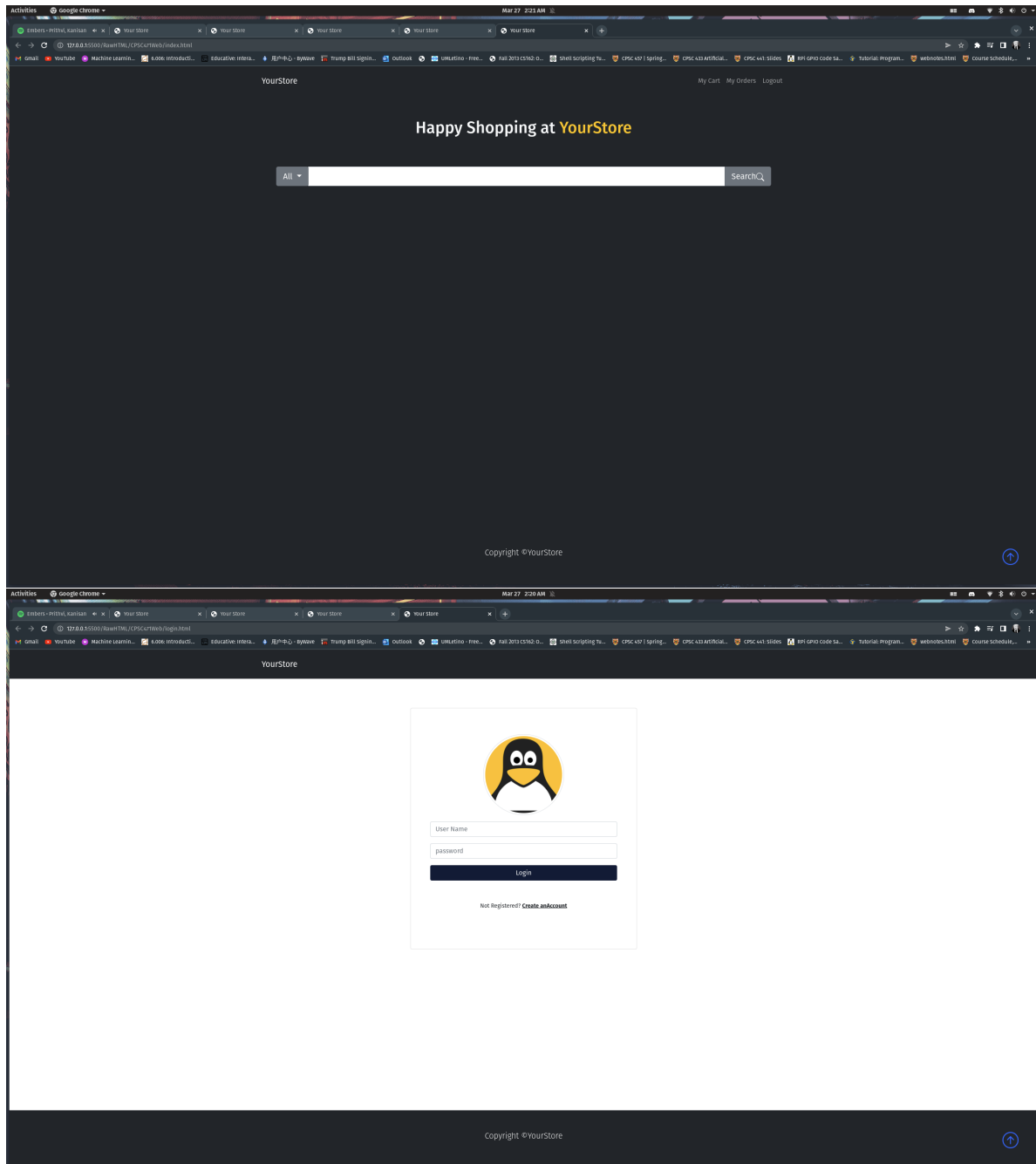
1. Zhijie Xia
2. Zixing Wei

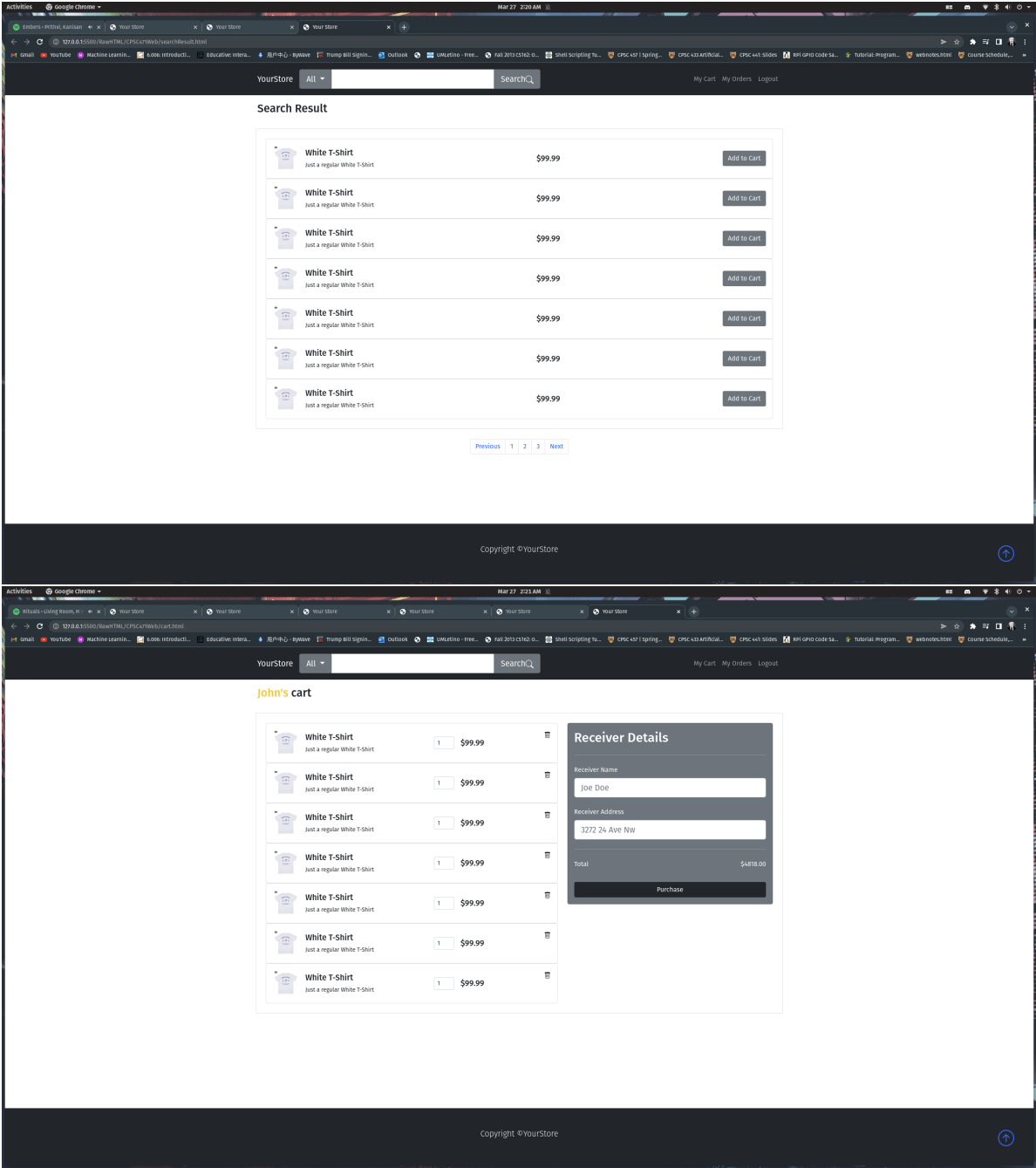




Website Contributors

1. Zhijie Xia
2. Yuan Liu





Activities

Google Chrome

Mar 27 2:38 AM

Enders - Pritish, Kaitian

your store

127.0.0.1:5555/yourStore/CPSCodeWeb/index.html

Google

YouTube

Machine Learning

6.006 Introduction to Computer Science in Python

Educative: Intro...

知乎 - 中国

Bye Bye

Trump Bill Sign...

Outlook

Unlimited - Free...

Real 2013 CS162 C...

Shell Scripting T...

CPS 401 | Spring...

CPS 433 Artificial...

CPS 443 Intro...

MIT Open Code Sa...

Tutorial: Program...

yourStore

Course Schedule...

My Products

My Orders










Logout

john, search within your products

All

Search

Add New Product

Image	Name	Description	Price	Inventory	Actions
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete
	White T-Shirt	Just a regular White T-Shirt	\$99.99	5	Update Delete

[Previous](#) [1](#) [2](#) [3](#) [Next](#)

Mar 27 2:38 AM

yourStore

My Products

My Orders

Logout

john, search within your orders

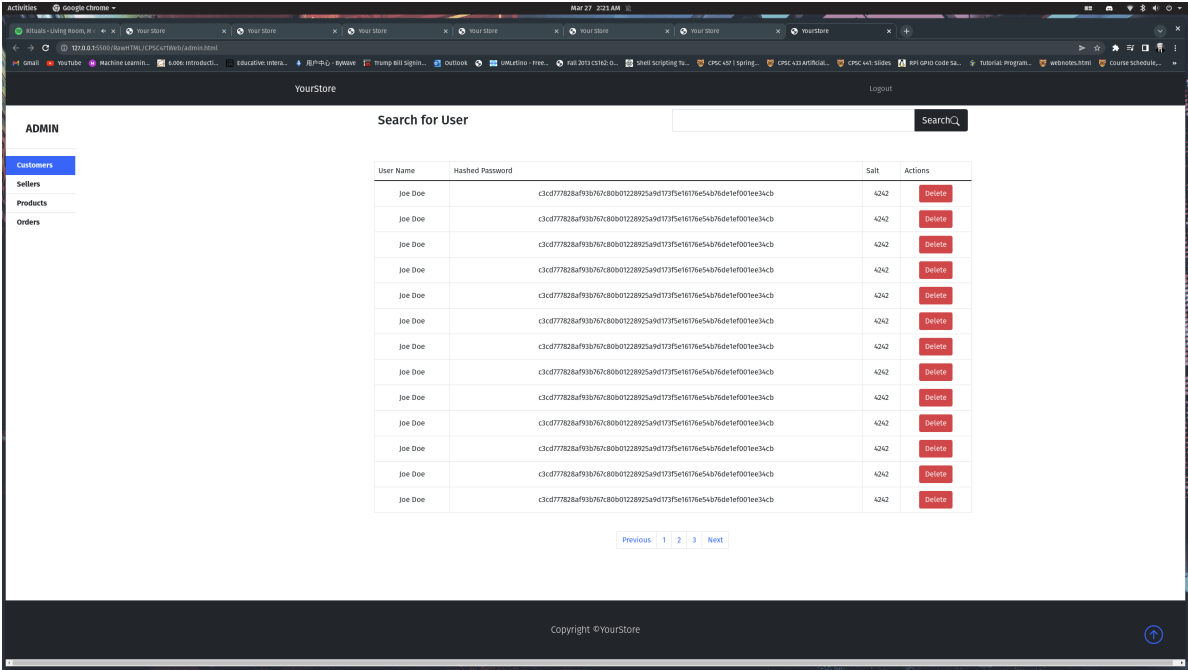
All

Search

Order ID	Customer Name	Total	Receiver Name	Receiver Address	Status	Actions
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipment Required	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Shipped	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Waiting for Payment	Detail Ship Cancel
80808080	Random Customer	\$99.99	Dummy Receiver	3272 24 Ave NW	Cancelled	Detail Ship Cancel

[Previous](#) [1](#) [2](#) [3](#) [Next](#)

Copyright ©YourStore



Contributors

1. Zhijie Xia

Description: Users(customer,seller, and admin) login

URL: /login

METHOD: PUT

INPUT:

```
[{
    "user_name":string,
    "password":string
}]
```

OUTPUT:

```
[{
    "session_token":string
}]
```

/**

* APIs about SELLER

*/

Description: Get a list of sellers with page size and page number

URL: /sellers?pageSize=m&pageNumber=n

METHOD: GET

INPUT: N/A

OUTPUT:

```
[ {
    "user_name":string,
    "password":string,
    "card_number":string
},
...
]
```

Description: Get a particular seller

URL: /sellers/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[
    {
        "user_name":string,
        "password":string,
```

```

        "card_number":string
    }
]

```

Description: Create new seller

URL: /sellers

METHOD: POST

INPUT:

```

    [{
        "user_name":string,
        "password":string,
        "card_number":string
    }]

```

OUTPUT: N/A

Description: Delete a particular seller

URL: /sellers/:id

METHOD: DELETE

OUTPUT: N/A

Description: Get a list orders of a particular seller with page size and page number

URL: /sellers/:id/orders?pageSize=m&pageNumber=n

METHOD: GET

INPUT: N/A

OUTPUT:

```

    [
        {
            "order_id":number,
            "customer_id":string,
            "seller_id":string,
            "total":float,
            "receiver_name":string,
            "receiver_address":string,
            "payment":boolean,
            "shipped":boolean,
            "cancelled":boolean,
            "shipment_label":string,
            "products":[{"product_id":number, "quantity":number},...]
        },
        ...
    ]

```

Description: Get a particular order of a particular seller

URL: /sellers/:id/orders/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[ {
    "order_id":number,
    "customer_id":string,
    "seller_id":string,
    "total":float,
    "receiver_name":string,
    "receiver_address":string,
    "payment":boolean,
    "shipped":boolean,
    "cancelled":boolean,
    "shipment_label":string,
    "products":[{"product_id":number, "quantity "=number},...]
  }
]
```

```
/**
 * APIs about CUSTOMER
 */
```

Description: Get a list of customers with page size and page number

URL: /customers?pageSize=m&pageNumber=n

METHOD: GET

INPUT: N/A

OUTPUT:

```
[ {
    "user_name":string,
    "password":string,
    "cart_id":string
  },
  ...
]
```

Description: Get a particular customer

URL: /customers/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[
  {
    "user_name":string,
    "password":string
    "cart_id":string
  }
]
```

```
    }
  ]

```

Description: Create new customer

URL: /customers

METHOD: POST

INPUT:

```
[{
    "user_name":string,
    "password":string
}]

```

OUTPUT: N/A

Description: Delete a particular customer

URL: /customer/:id

METHOD: DELETE

OUTPUT: N/A

Description: Get a list orders of a particular customer with page size and page

URL: /customers/:id/orders?pageSize=m&pageNumber=n

METHOD: GET

INPUT: N/A

OUTPUT:

```
[  {
    "order_id":number,
    "customer_id":string,
    "seller_id":string,
    "total":float,
    "receiver_name":string,
    "receiver_address":string,
    "payment":boolean,
    "shipped":boolean,
    "cancelled":boolean,
    "shipment_label":string,
    "products":[{"product_id":number, "quantity "=number},...]
  },
  ...
]
```

Description: Get a particular order of a particular customer

URL: /customers/:id/orders/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[  {

```

```

        "order_id":number,
        "customer_id":string,
        "seller_id":string,
        "total":float,
        "receiver_name":string,
        "receiver_address":string,
        "payment":boolean,
        "shipped":boolean,
        "cancelled":boolean,
        "shipment_label":string,
        "products":[{"product_id":number, "quantity "=number},...]
    }
]

```

Description: Get a customer's cart

URL: /customers/:id/cart

METHOD: GET

INPUT: N/A

OUTPUT:

```

    [  {
        "cart_id":string,
        "products":[{"product_id":number,"quantity "=number}...],
        "total":number
    }
]

```

Description: Add a product to the customer's cart

URL: /customers/:id/cart/product:id

METHOD: PUT

INPUT: N/A

OUTPUT: N/A

Description: Delete a product from the customer's cart

URL: /customers/:id/cart/product:id

METHOD: DELETE

INPUT: N/A

OUTPUT: N/A

```

/**
 * APIs for ORDER
 */

```

Description: Get a list of all orders with page size and page number

URL: /orders?pageSize=m&pageNumber=n

METHOD: GET

INPUT: N/A

OUTPUT:

```
[ {
    "order_id":number,
    "customer_id":string,
    "seller_id":string,
    "total":float,
    "receiver_name":string,
    "receiver_address":string,
    "payment":boolean,
    "shipped":boolean,
    "cancelled":boolean,
    "shipment_label":string,
    "products":[{"product_id":number, "quantity "=number},...]
  },
  ...]
```

Description: Get a particular order

URL: /orders/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[{
    "order_id":number,
    "customer_id":string,
    "seller_id":string,
    "total":float,
    "receiver_name":string,
    "receiver_address":string,
    "payment":boolean,
    "shipped":boolean,
    "cancelled":boolean,
    "shipment_label":string,
    "products":[{"product_id":number, "quantity "=number},...]
}]
```

Description: Update a order

URL: /orders/:id

METHOD: PUT

INPUT: // the updated method does not necessary include every element of the fol

```
[{
```

```

        "customer_id":string,
        "seller_id":string,
        "total":float,
        "receiver_name":string,
        "receiver_address":string,
        "payment":boolean,
        "shipped":boolean,
        "cancelled":boolean,
        "shipment_label":string,
        "products":[{"product_id":number, "quantity "=number},...]
    ]}

```

OUTPUT: N/A

Description: Create a order

URL: /orders/:id

METHOD: POST

INPUT:

```

    [{
        "customer_id":string,
        "seller_id":string,
        "total":float,
        "receiver_name":string,
        "receiver_address":string,
        "products":[{"product_id":number, "quantity "=number},...]
    }]

```

OUTPUT:

```

    [{
        "order_id":string
    }]

```

Description: Delete a particular order

URL: /orders/:id

METHOD: Delete

INPUT: N/A

OUTPUT: N/A

```

/**
 * APIs for PRODUCT
 */

```

Description: Get a list of products with search keyword, category, page size and

URL: /products?pageSize=m&pageNumber=n&key=k&category=c

METHOD: GET

INPUT: N/A

OUTPUT:

```
[ {
    "product_id":string,
    "seller_id":string,
    "price":float,
    "search_keys":[string,...],
    "description":string
  },
  ...
]
```

Description: Get a particular product

URL: /products/:id

METHOD: GET

INPUT: N/A

OUTPUT:

```
[{
    "product_id":string,
    "seller_id":string,
    "price":float,
    "search_keys":[string...],
    "description":string
}]
```

Description: Create a product

URL: /products

METHOD: POST

INPUT:

```
[{
    "seller_id":string,
    "price":float,
    "search_keys":[string...],
    "description":string
}]
```

OUTPUT:

```
[{
    "product_id":string
}]
```

Description: Update a particular product

URL: /products/:id

METHOD: PUT

INPUT:

```
[{
```

```

        "seller_id":string,
        "price":float,
        "search_keys":[string...],
        "description":string
    }]

```

OUTPUT: N/A

Description: DELETE a particular product

URL: /products/:id

METHOD: DELETE

INPUT: N/A

OUTPUT: N/A

```

/**
 * APIs for Category
 */

```

Description: Get the complete list of all category

URL: /categories

METHOD: GET

INPUT: N/A

OUTPUT:

```

    [
        {
            "category_id":string,
            "title":string
        }
        ...
    ]

```

Description: Create a new category

URL: /categories

METHOD: POST

INPUT:

```

    [{
        "title":string
    }]

```

OUTPUT:

```

    [{
        "category_id":string
    }]

```

Description: Update a category

URL: /categories/:id

METHOD: PUT

INPUT:
 [{
 "title":string
 }]

OUTPUT: N/A

Description: Delete a category

URL: /categories/:id

METHOD: DELETE

INPUT: N/A

OUTPUT: N/A

Description: Add a product into a category

URL: /categories/:id/products/:id

METHOD: PUT

INPUT: N/A

OUTPUT: N/A

Description: Delete a product from a category

URL: /categories/:id/products/:id

METHOD: DELETE

INPUT: N/A

OUTPUT: N/A