Retail Supply Chain System

Handbook

2024-05

**Table of Content**

[1. Install the Blockchain Environment 3](#_Toc166702756)

[2. Install the Back-end Environment 4](#_Toc166702757)

[3. Install the Front-end Environment 8](#_Toc166702758)

[4. System Operation Instruction 11](#_Toc166702759)

[4.1 Administration Management Module 11](#_Toc166702760)

[4.2 Category Management Module 12](#_Toc166702761)

[4.2 Product Management Module 13](#_Toc166702762)

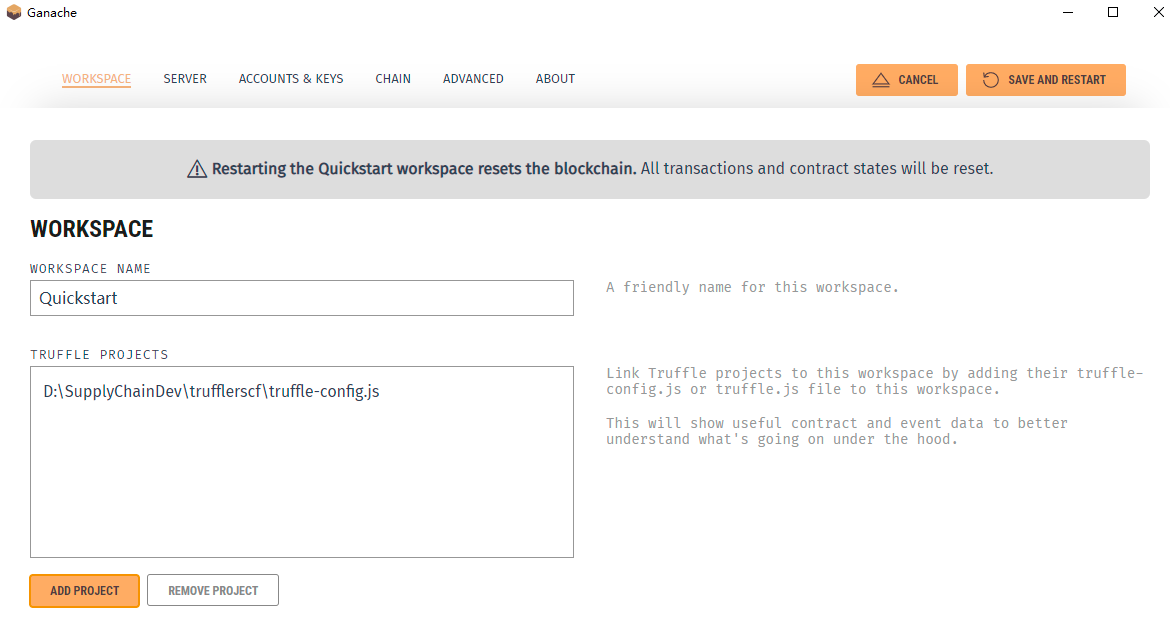
[4.3 Order Management Module 14](#_Toc166702763)

[4.4 Document Management Module 15](#_Toc166702764)

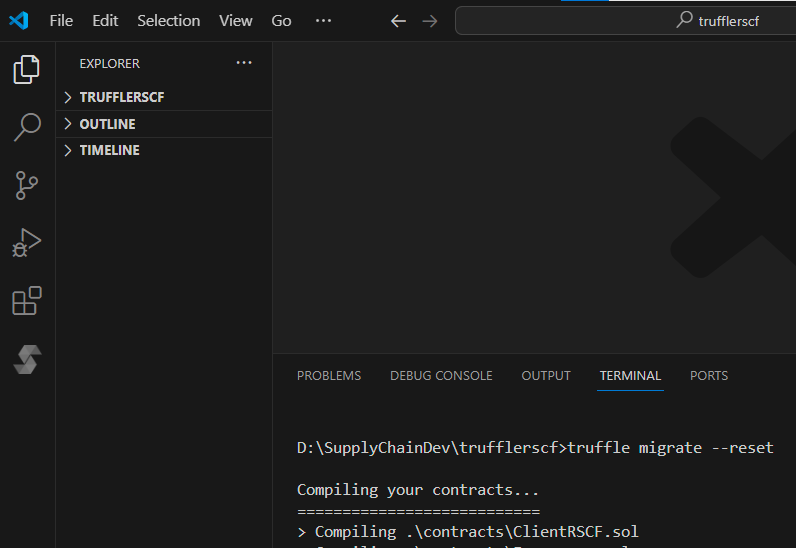
[4.6 Trace Management Module 16](#_Toc166702765)

# 1. Install the Blockchain Environment

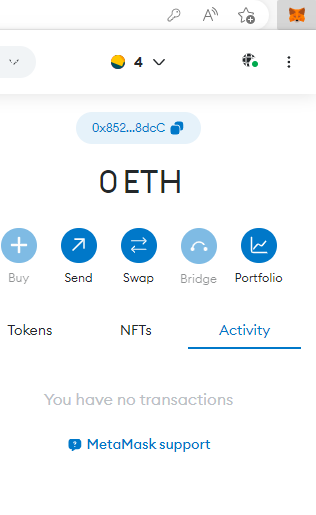
Install the private blockchain Ganache and start it, then import the truffle configuration file truffle-config.js into the Ganache and restart Ganache:



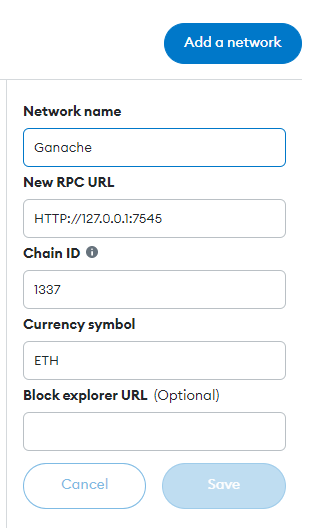
In Visual Studio Code, open the Truffle project “TRUFFLRSCF”. To deploy all the smart contracts to Ganache, in the terminal, run the command: truffle migrate --reset:



Install MetaMask, create a new account to log in, and open it in the browser:

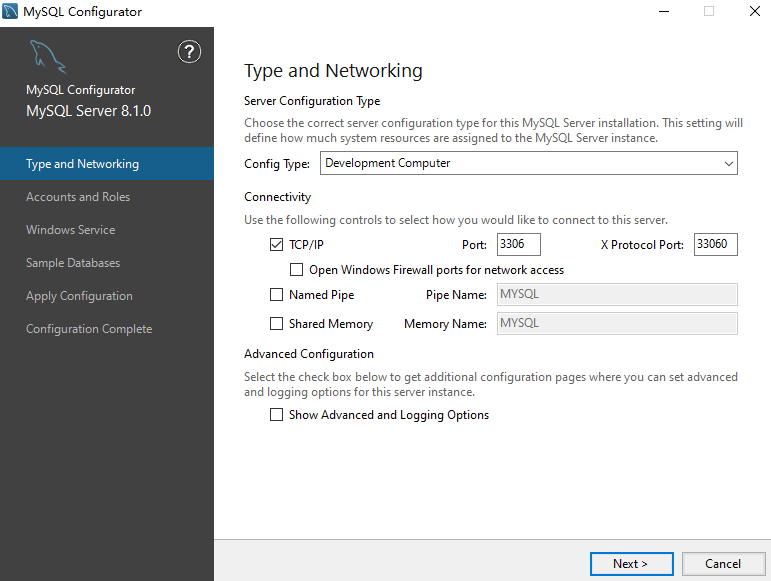


Add the Ganache network to MetaMask:



# 2. Install the Back-end Environment

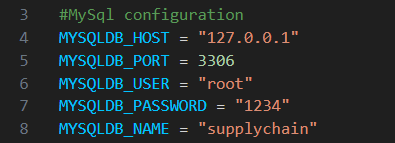
Install MySQL Server 8.1.0:



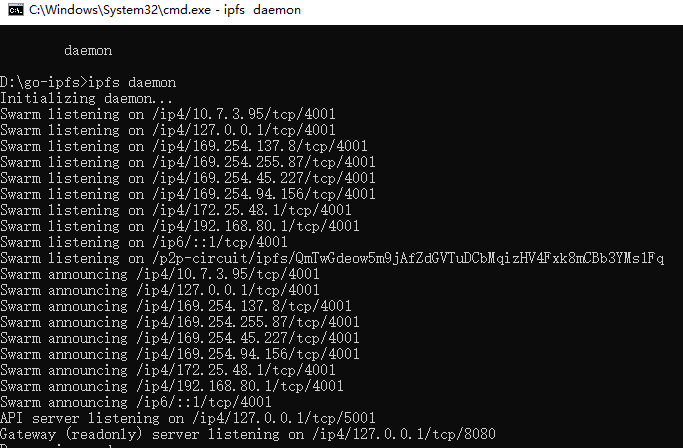
In another Visual Studio Code, open the Flask project “FlaskEndRSCF”.

In the MySQL database, execute all the scripts in the folder: FlaskEndRSCF\db.

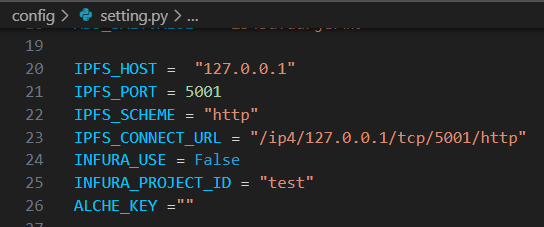
In the Flask project “FlaskEndRSCF”, configure the MySQL database in setting.py:



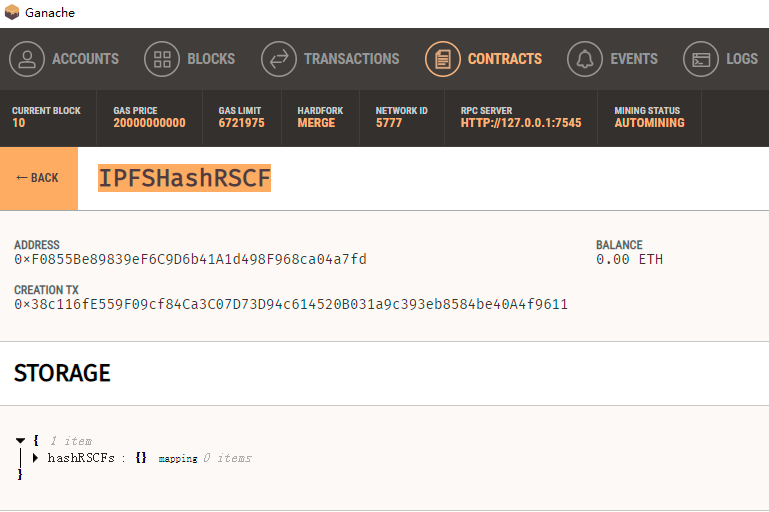
Install IPFS, open the folder of IPFS, start IPFS by executing the command: ipfs daemon. Then the IPFS daemon is up and running on the machine, and the default server site is at [http://localhost:5001/.](http://localhost:5001/)



Configure the IPFS information in setting.py of the Flask project “FlaskEndRSCF”:



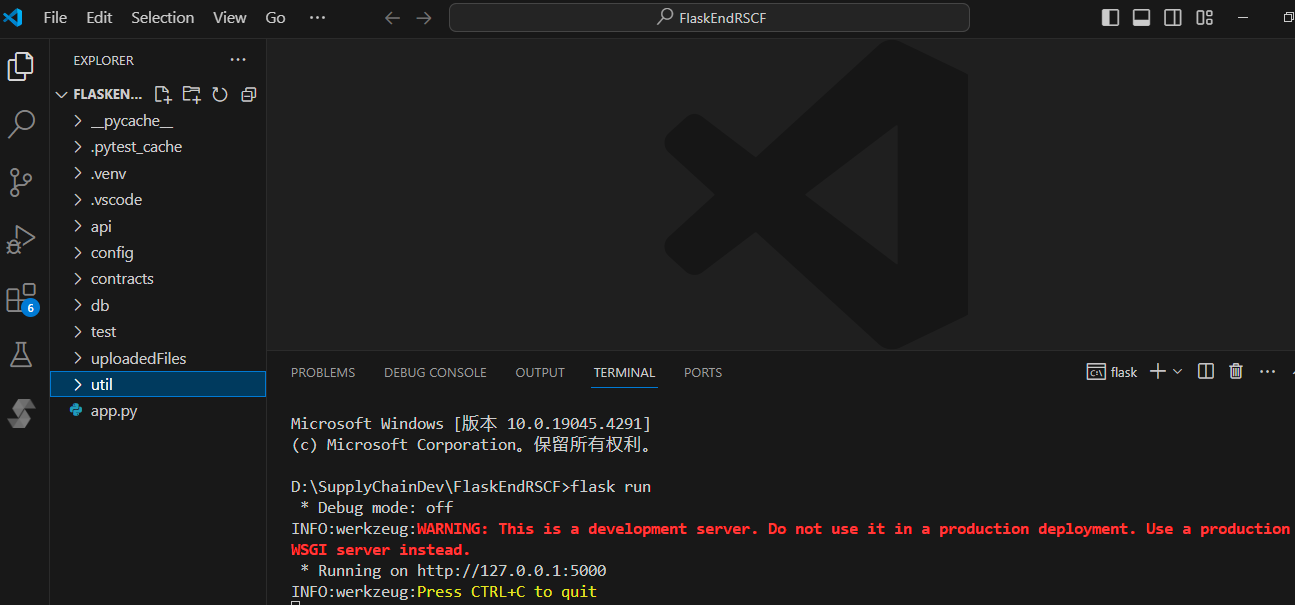
Get the address of the smart contract IPFSHashRSCF in Ganache:



Configure the address of the smart contract IPFSHashRSCF in setting.py of the Flask project “FlaskEndRSCF”:

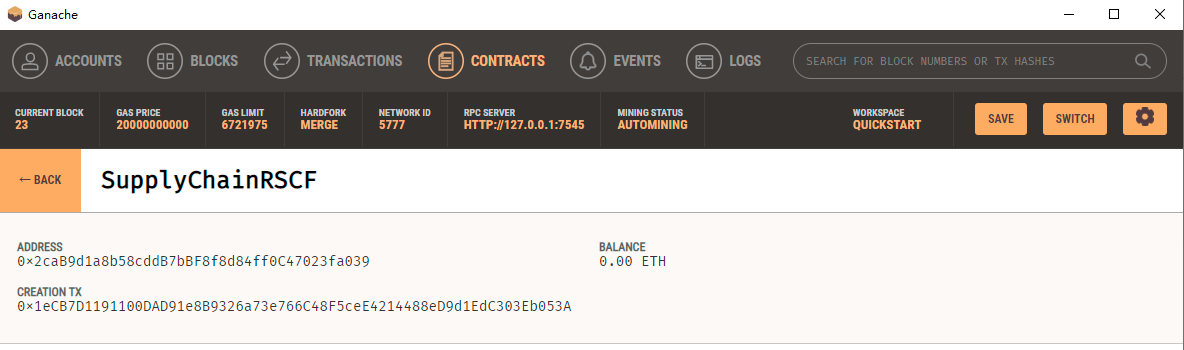


To start the back-end server, in the terminal, run the command: flask run:

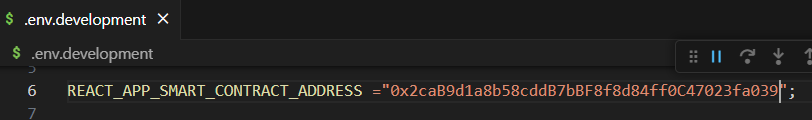


# 3. Install the Front-end Environment

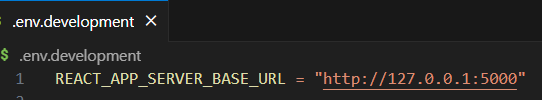
Get the address of the smart contract “SupplyChainRSCF” in Ganache:



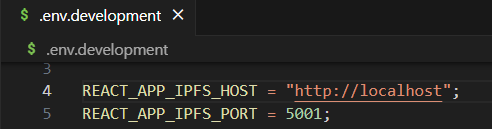
In another Visual Studio Code, open the front-end project “frontendrscf”. In the file “.env.development”, set the address of the smart contract “SupplyChainRSCF” to REACT\_APP\_SMART\_CONTRACT\_ADDRESS:



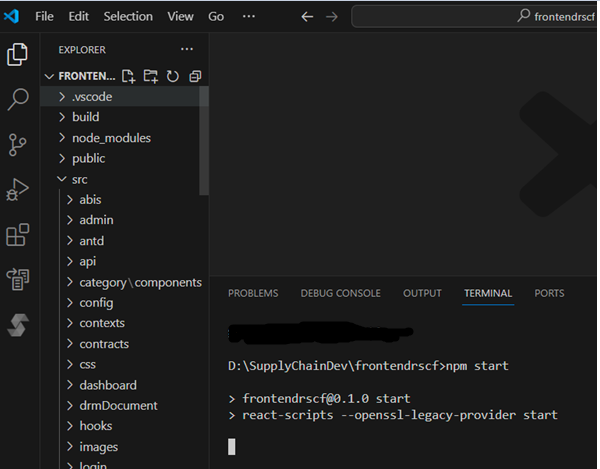
Set the back-end server URL to REACT\_APP\_SERVER\_BASE\_URL:



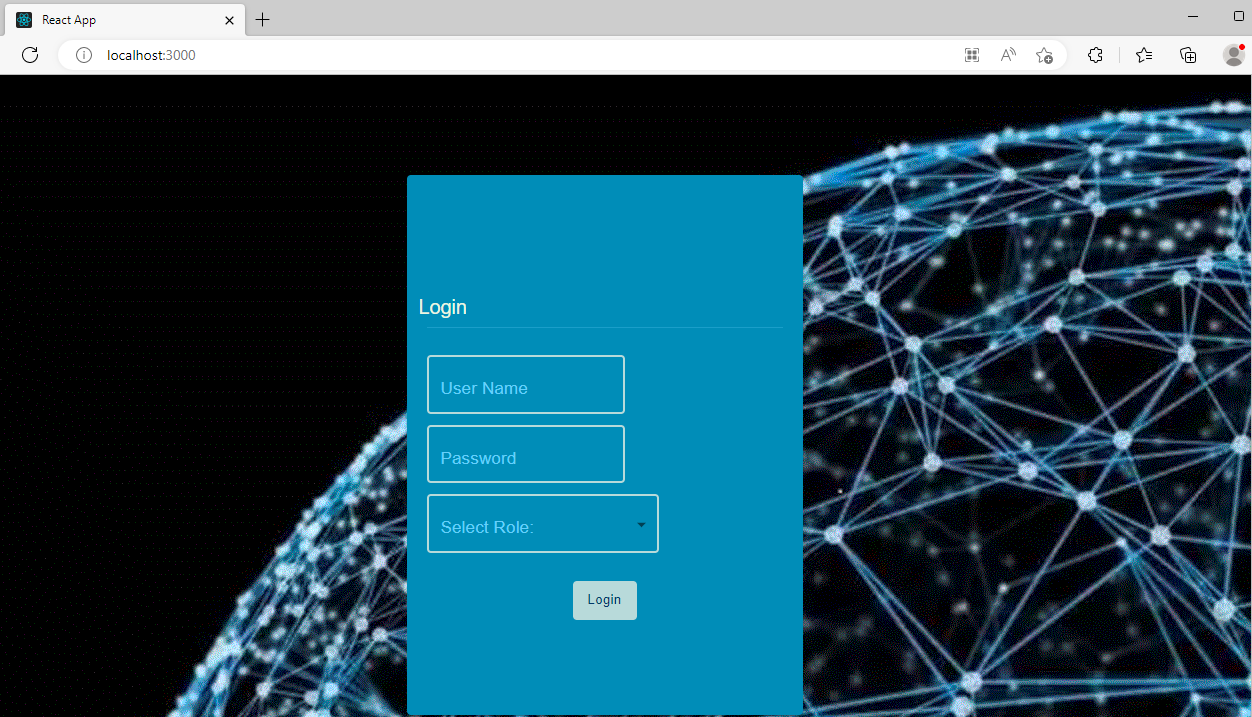
Set IPFS server details to REACT\_APP\_IPFS\_HOST and REACT\_APP\_IPFS\_PORT:



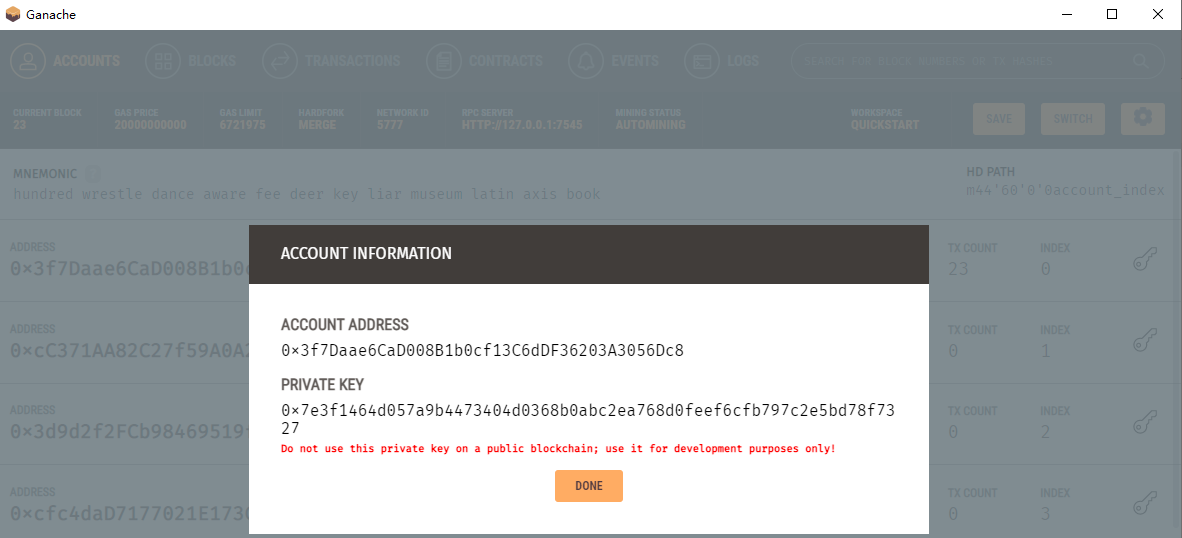
To start the front-end project, in the terminal, run the command: npm start:



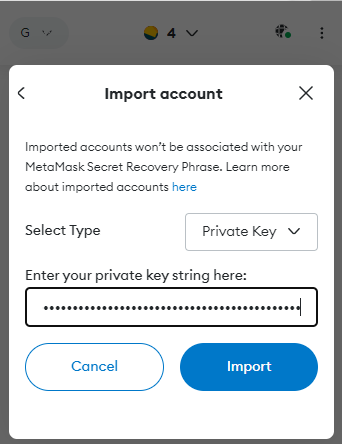
In the browser, the web application is opened automatically:



Get the private key to the first account of Ganache:

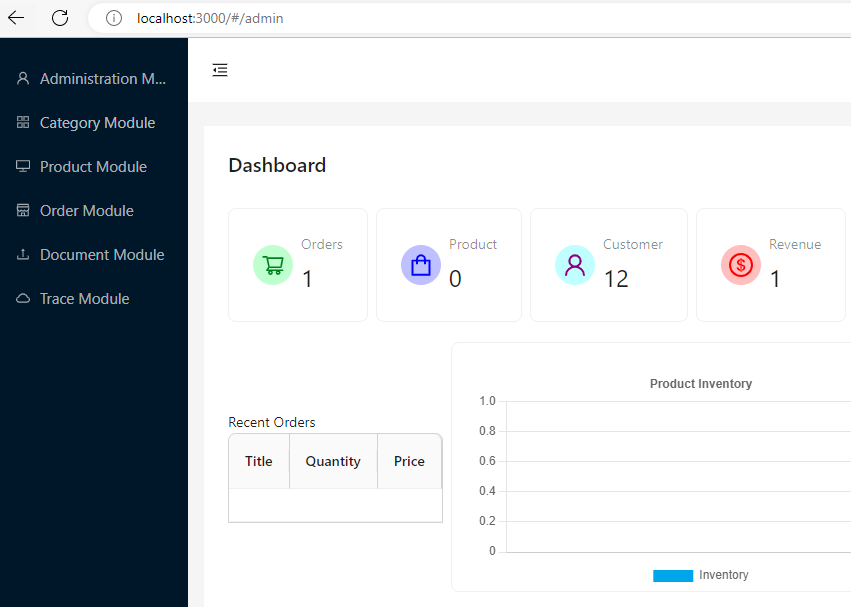


Import this private key to MetaMask to create a new account in MetaMask and connect to this new account:



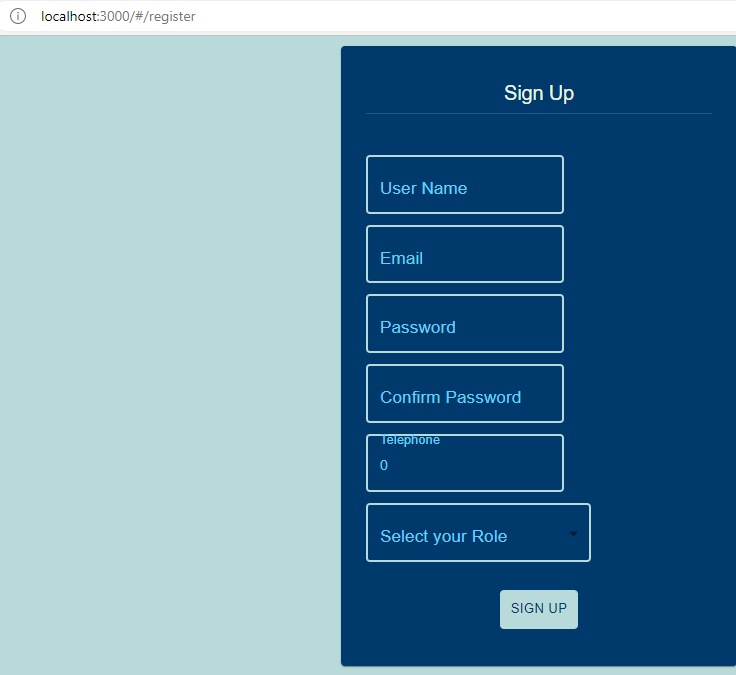
# 4. System Operation Instruction

Log in to the system by default with the user name “sa” and password “12345678”, select the role “Administrator”. After entering the system, users could operate the relative modules according to their roles. The administrator can operate all the functions:

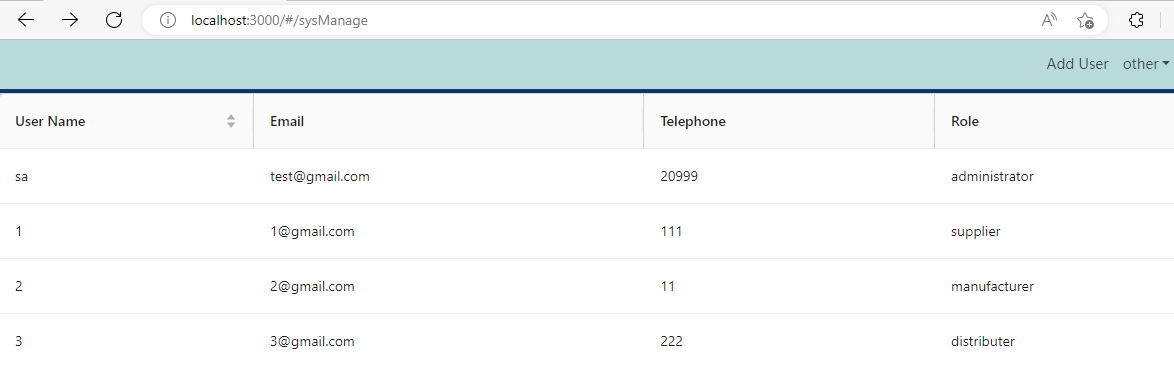


## 4.1 Administration Management Module

The category management module can add users:

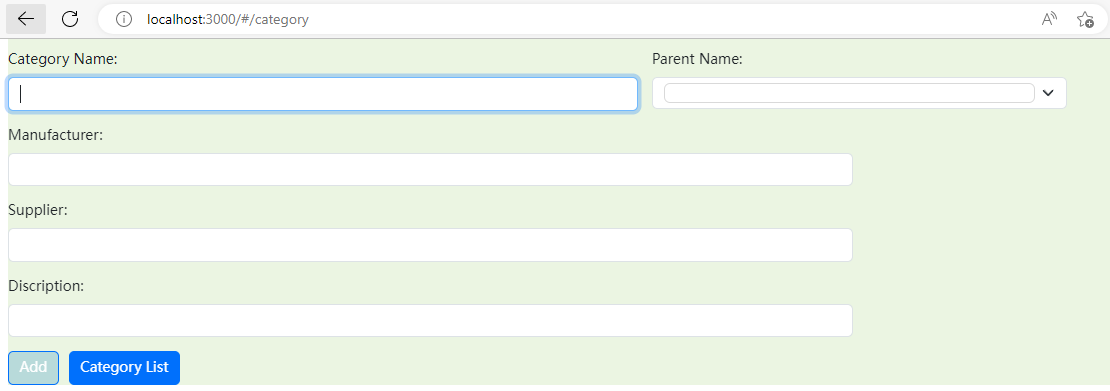


Customers can search for user information by the Web page:

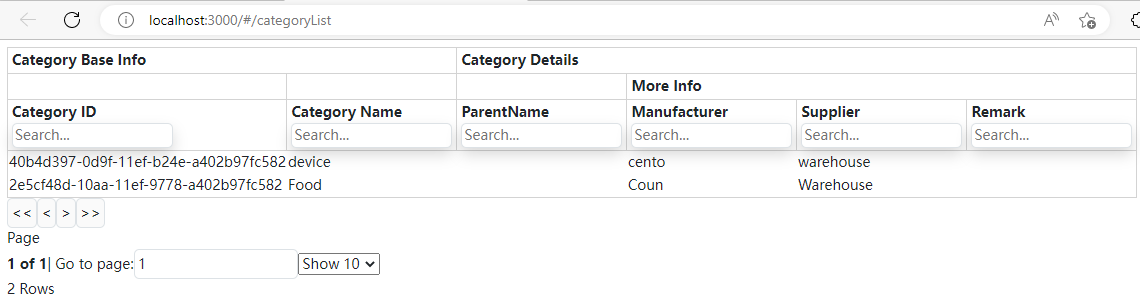


## 4.2 Category Management Module

The category management module can add categories according to similar features of products:

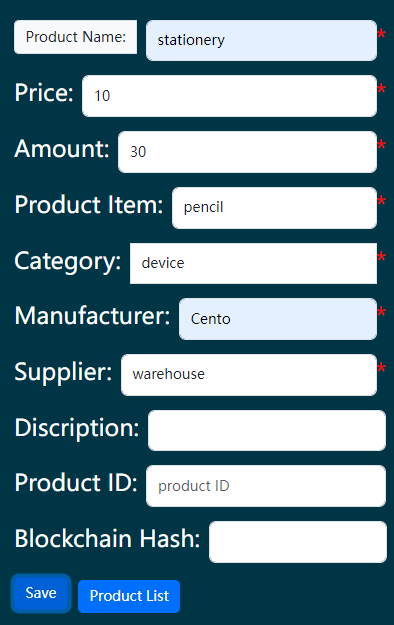


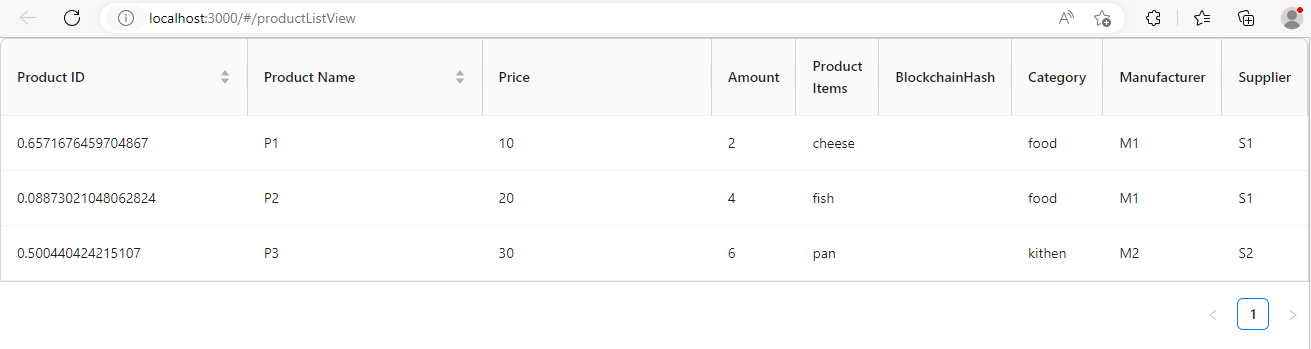
Users can search for category information by clicking the button “Category List” :



## 4.3 Product Management Module

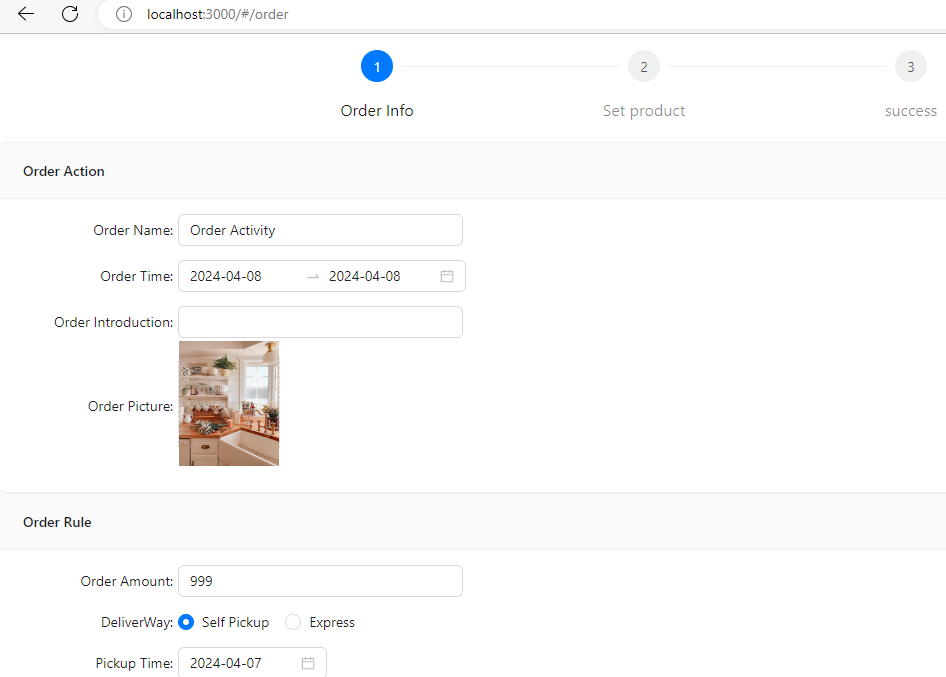
System users can have permission to add product details according to their roles. The product base information is stored in the blockchain network through the smart contract to track the product information in the system.



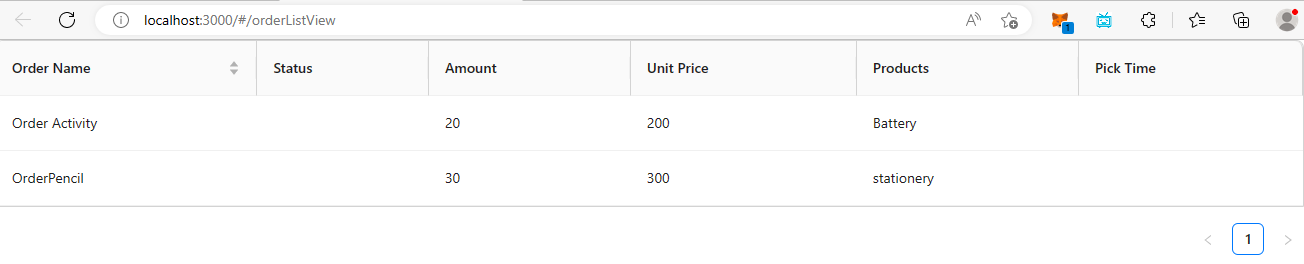
Users can search for product information by clicking the button “Product List” :**

## 4.4 Order Management Module

All the users could create an order in this module:

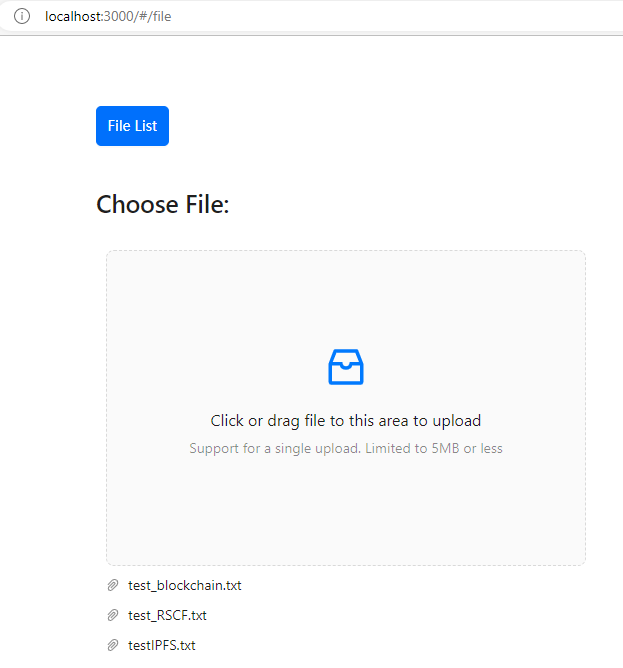


Users can search for the order information after adding it:

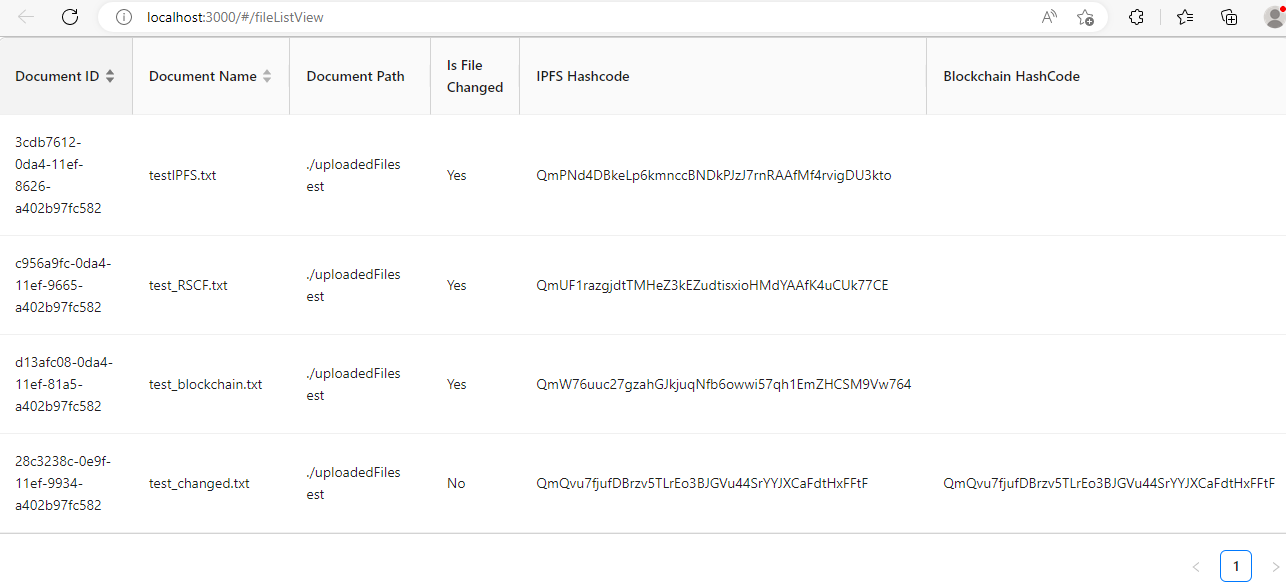


## 4.5 Document Management Module

In the document management module, financial files could be uploaded to the system:

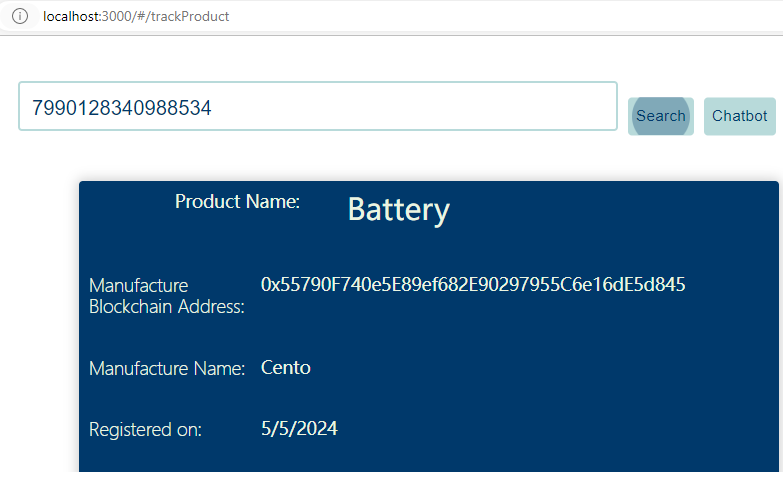


To determine whether the file has been altered, customers can compare the hash codes stored in the MySQL database and on the blockchain on the following page:

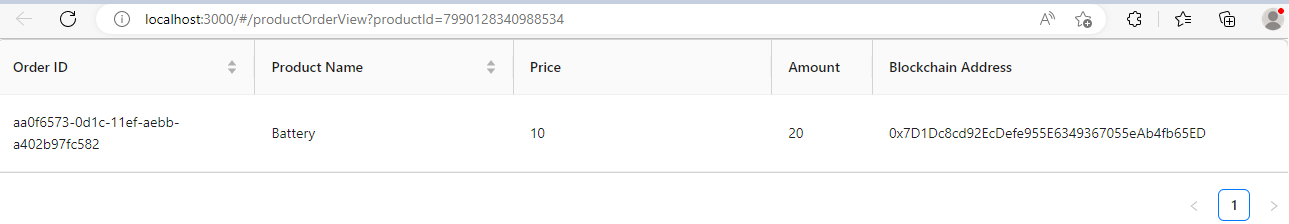


## 4.6 Trace Management Module

Once the consumer has entered the product's unique serial number, they can obtain detailed information about it on the blockchain:



Customers can access relevant information about their product orders on the blockchain by clicking the button “related orders”:



The Trace Management Module also includes the Chatbot feature, which allows customers to track products and report any issues about the product:

