Resi Buyer
Data Management Proof of Concept

Team members

Abbey McCarthy - a1744594

Brandon Smart - a1743623

Janhvi Sirohi - a1745611

Jihua Shi (Chris) - a1739502

Ragav Sachdeva - a1720657

Rijul Ramkumar - a1725709

Rory Martin - a1740203

William Godfrey - a1743033

Xiaofan Lu (Cade) - a1780911

Overview

This introduction provides an overview of the Web App Resi buyer. It includes the purpose, scope, overview of the architecture of the application,main component design, and detailed user guide.

Document Scope and Purpose

This document provides a description of the technical design of an online food ordering web app demo that incorporates a blockchain system to track and update the states of the order delivery. This document’s primary purpose is to describe the functionality of this web app to any new users, and a developer tooltips for any future works.

Introduction

This is a web app project showcasing an online shopping website for three sample products (Milk, Beef and Yoghurt). It is capable of tracking the order information disseminated to all customers throughout the ordering and delivering journey, all of the data are attached and implemented as blocks of blockchain, and could be referenced with blockchain hash search.

System Enviroment

Development: Python+Bootstrap+Django

Database: Django database sqlite3

Revison control: https://github.cs.adelaide.edu.au/a1790336/TPRB2

Installation & Usage

This project is built in and requires a Python Environment to operate. Anaconda has a great guide on obtaining Python,In case of any installation helps, https://docs.anaconda.com/anaconda/install/ will tell you how to install it.

Install pip on your Python environment

```
apt install python3-pip
```

Install Django:

```
python -m pip install Django
```

Install django-crispy-forms:

```
pip3 install --user django-crispy-forms
```

Install Faker (for tracking info generation):

```
 pip3 install Faker
```

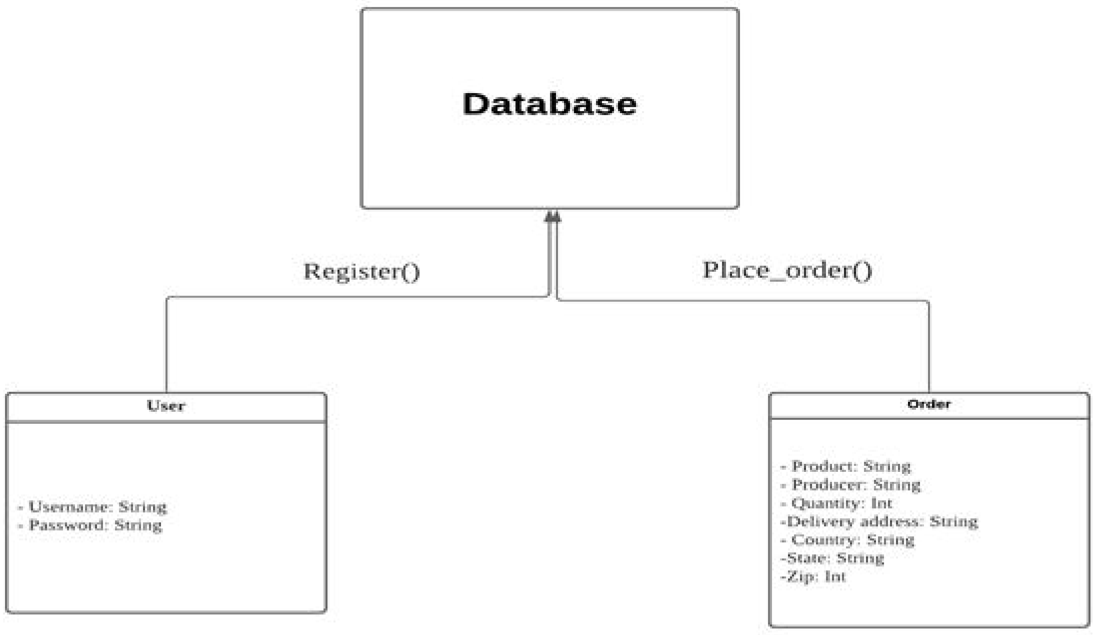
Run server using:

```
python manage.py runserver
```

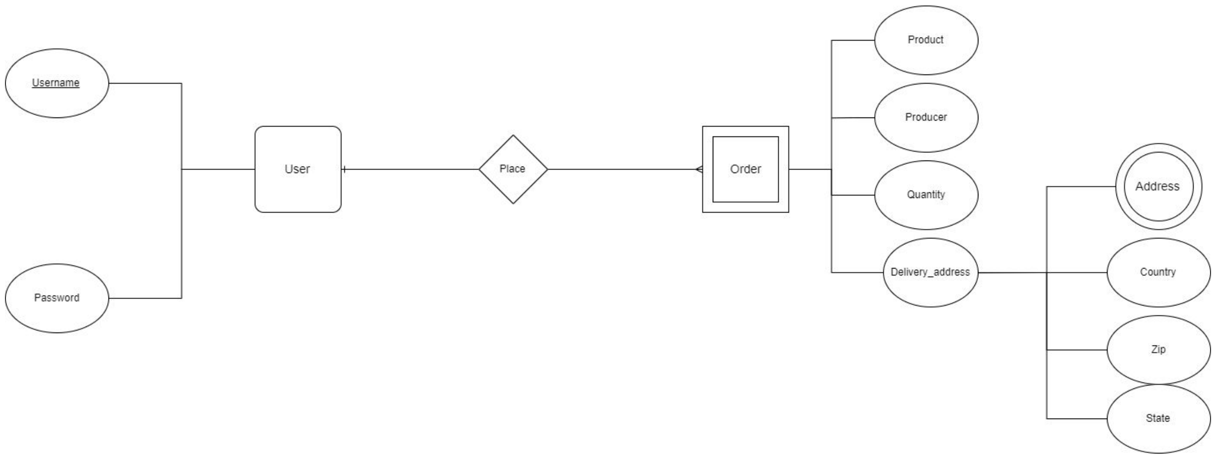
Install Selenium:

```
pip install selenium
```

Architecture Design



Database Design



Module

Sign-up

The sign-up page will ask the user to input a username and a password. the username must be unique, and was not registered and used by other customers.The password needs to be strong and complicated that meets the requirements displayed above the input section. The user will be notified if the username is unique or if the password is strong enough by submitting the form.If all requirements are met, the profile will be successfully created and saved to the database, and then the page will be redirected to the login page where all blanks are prefilled with this newly registered account.

Login

To login, the user is required to input their own username and the matching password, after it gets verified with the local database, the user will be redirected to the index page.

If the wrong combination has been inputted, an alert message will be displayed and the user is asked to re-enter their login info.

Order placing:

Any logged in user is able to place an order on the order placing page,the user is required to pick a product,choose a corresponding producer among all three of them, and specify the amount they would like to order, as well as a detailed delivery address. When an order has been placed on the order placing page, an ‘Order’ object will be created， and will then be saved to the local datgabase.

View_order

If the user has passed the authentication and logged in to the website, he/she will be able to view all the past orders collectively on the page, the information is retrieved from the local database. For each order, the displayed information includes the product catalog,the producer, the amount and the delivery address.

View_Blockchain

The user can view the tracking information of orders placed under the current account on the broadcasted blockchain. It also allows the user to search for certain transactions by its hash value.

Join Blockchain network

The user is allowed to register and publicate his local transaction to all peers in the blockchain system with a custom address. All registered peers would be displayed on the page as well.

View All Fulfilled Order

The user is allowed to view all the delivered order that was registered and saved on the blockchain by clicking on the “

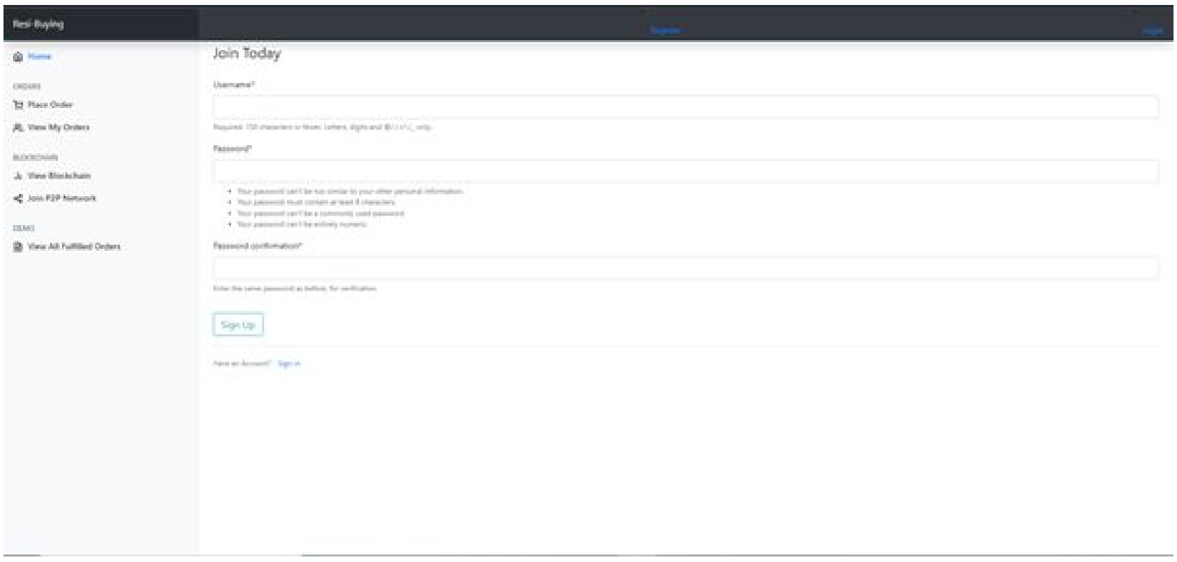
View All Fulfilled Order” button on the side panel, and all previous orders placed and fulfilled under current user will be displayed.

User Guide

Registration

This is the **registration page** where users could set up an account with a unique ID and a corresponding password, the password will need to be typed in twice to verify its correctness.

(could add a functionality to make it visible to avoid mistyping)


Resi-Buying

Register Login

Home

ORDERS

Place Order

View My Orders

BLOCKCHAIN

View Blockchain

Join P2P Network

DEMO

View All Fulfilled Orders

Join Today

Username*

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password*

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

Password confirmation*

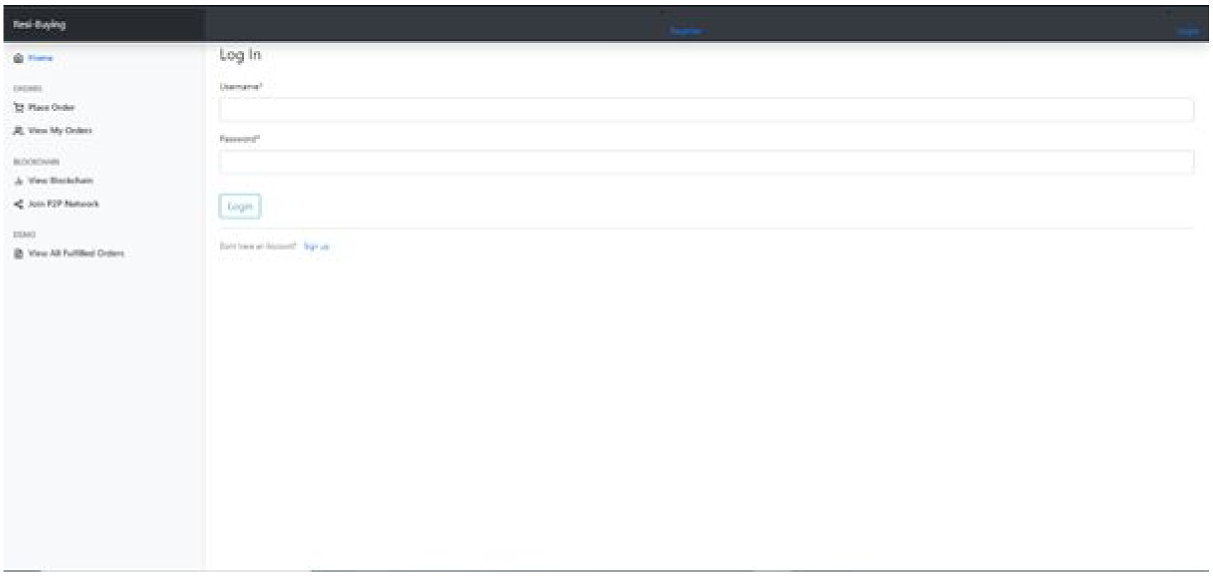
Enter the same password as before, for verification.

Sign Up

Have an Account? Sign in

Log in

A user needs to type his/her registered username and the matching password to access the website for order-placing or any order history.


Resi-Buying

Register Login

Home

ORDERS

Place Order

View My Orders

BLOCKCHAIN

View Blockchain

Join P2P Network

DEMO

View All Fulfilled Orders

Log In

Username*

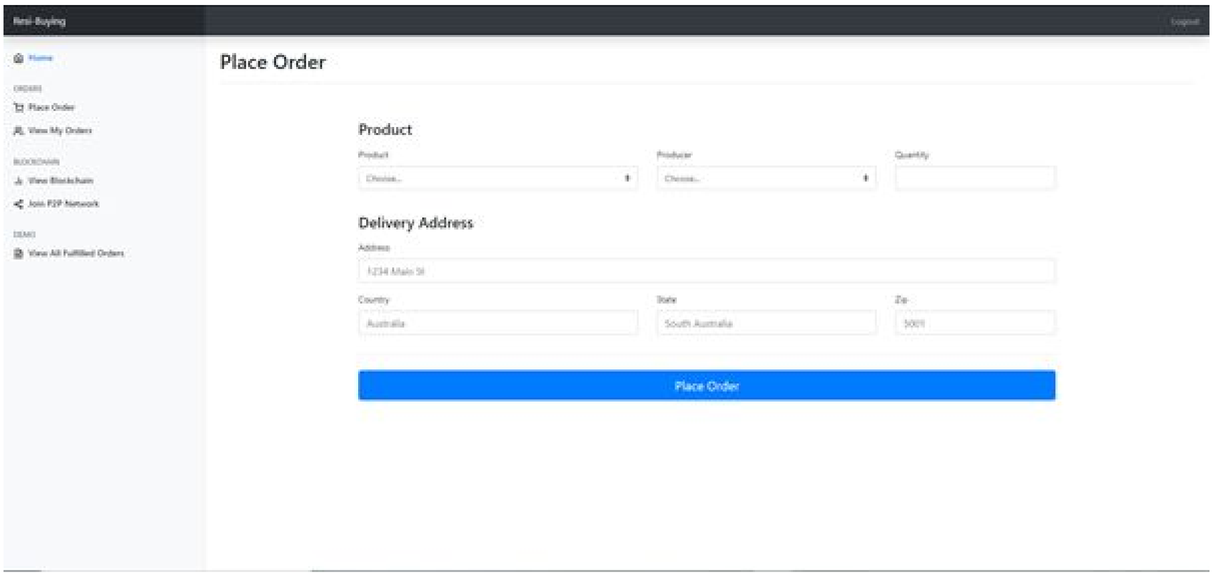
Password*

Login

Don't have an Account? Sign up

Order placing

The order could be placed via **Place Order Page**,our web-app currently provides choices for three product types, Milk, Yogurt and Beef, and three major manufacturers to order from.The users are also required to specify the amount they would like to order through a scrollable list as well as a collection of standard address information.



Product

Product	Producer	Quantity
Choose...	Choose...	

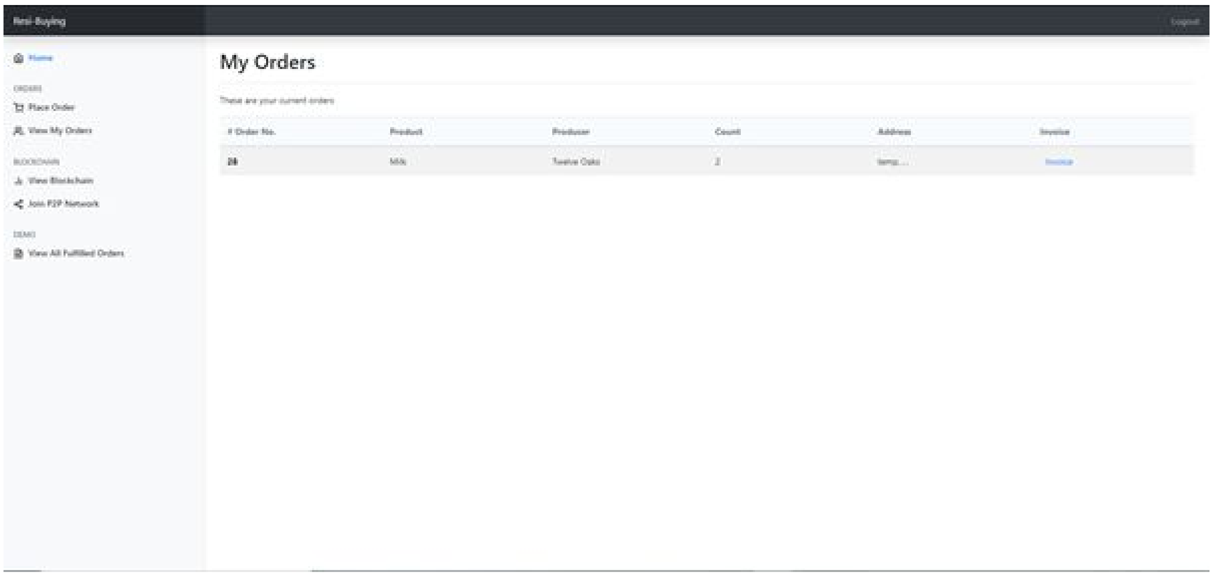
Delivery Address

Address: 1234 Main St

Country	State	Zip
Australia	South Australia	5001

View My Orders

By clicking the **View My Orders** button on the side panel, user could view all the previous orders in the list



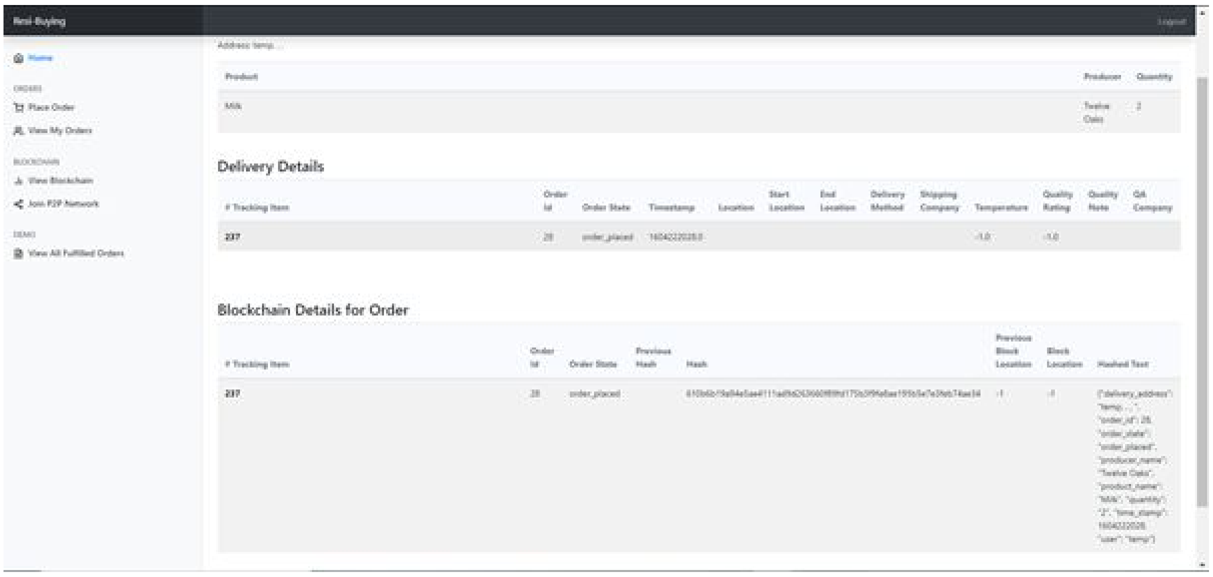
My Orders

These are your current orders

# Order No.	Product	Producer	Count	Address	Invoice
28	Milk	Twelve Oaks	2	temp....	Invoice

View invoice

The invoice of each order could be accessed from the **View My Orders** page,it locates at the end of each item.It provides a detailed information display of the order along with the has value of the order information



Address: temp, ...

Product	Producer	Quantity
Milk	Twelve Oaks	2

Delivery Details

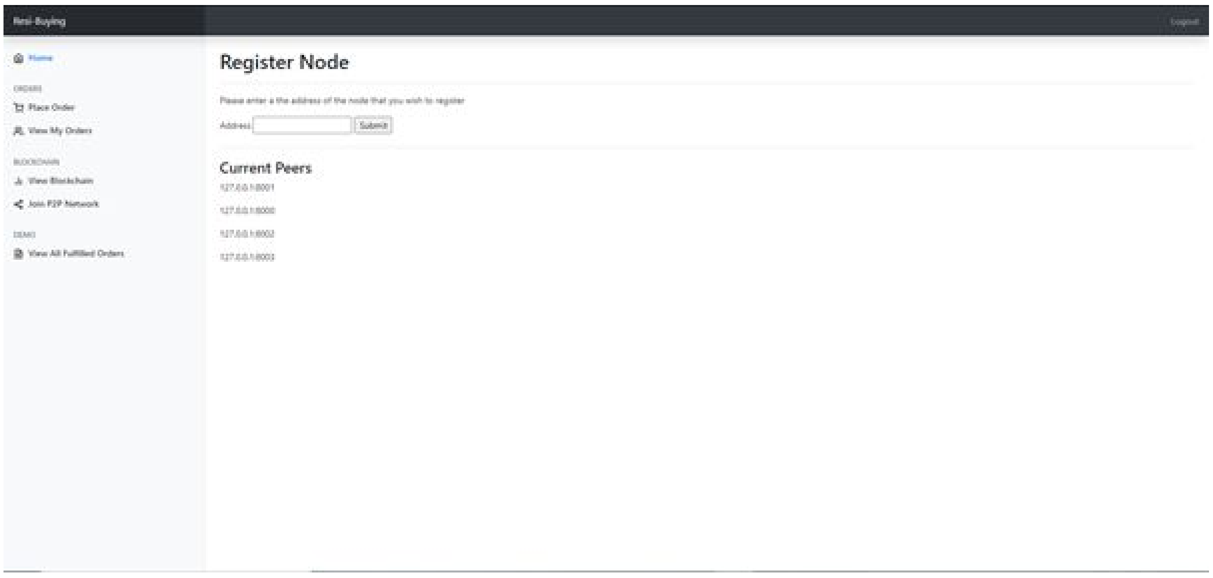
# Tracking Item	Order Id	Order State	Timestamp	Location	Start Location	End Location	Delivery Method	Shipping Company	Temperature	Quality Rating	Quality Note	QA Company
237	28	order_placed	1604222028.0						-1.0	-1.0		

Blockchain Details for Order

# Tracking Item	Order Id	Order State	Previous Hash	Hash	Previous Block Location	Block Location	Hashed Text
237	28	order_placed		[illegible]	-1	-1	{"delivery_address": "temp, ... ", "order_id": 28, "order_state": "order_placed", "producer_name": "Twelve Oaks", "product_name": "Milk", "quantity": "2", "time_stamp": 1604222028, "user": "temp"}

Join BlockChain

The user can register his order by manually input an address for a node, the web-app will automatically hash the order and append it onto the blockchain.



Register Node

Please enter a the address of the node that you wish to register

Address:

Current Peers

127.0.0.1:8001

127.0.0.1:8000

127.0.0.1:8002

127.0.0.1:8003

For Developers

Frontend

The front-end pages of this web-app are designed using the Bootstrap library,and are being served up with Django. You should not have to download anything related to Bootstrap in order to correct view these pages

The basic template for each page is in the basic_template folder, and it primarily consists of the navigation bar at the top and the sidebar. People can feel free to edit either of these elements, but editing should be done in the basic_template.html page rather than in the subpages in order to preserve consistency across the pages.

If you want to create a new page, you should take advantage of Django templates to use the existing navigation bar and sidebar. The order_place, order_view and index pages show examples on how to do that. The HTML you put in the 'content' block will appear in the main section of the page. You will likely also need to add your page to the views.py and urls.py files.

Database

Any product, order and tracking information of an order are saved as an independent object. They could be viewed and edited under ResiBuyer/Models.py

Product	
Data Type	Data Name
String	Name
String	Producer

Order	
Data Type	Data Name
String	User
String	Product_name
String	Producer_name
Int	Quantity
String	Delivery_address

Information			
Data Type	Data Name	Data Type	Data Name
Int	Order_id	String	Qa_company
String	Order_state	String	Previous_hash
Float	Time_stamp	String	Hash
String	Location	Int	Previous_blcok_loc

String	Start_location	Int	Block_inc
String	End_location	String	Hashed_text
String	Delivery_method	String	User
String	Shipping_company	String	Product_name
Float	Temperature	String	Producer_name
Float	Quality_rating	Int	Quantity
String	Quality_note	String	Delivery_address