

Starbucks Project

Supply Chain & Solidity



# Team Members

- - - - X

1. Hala Sedki (212200229)
2. Tony Begemy (212200274)
3. Ali Mohamed (212200217)
4. Ahmed Galal (212200215)
5. Hatem ElZomor (212200234)

Project: Supply Chain Blockchain Management using Solidity

# Step 1: Defining the Supply Chain Process

- - - - X

The supply chain process is the series of steps involved in getting a product from the supplier to the end-user. It can be divided into the following stages:

1. Procurement: This stage involves sourcing raw materials and components from suppliers.
2. Manufacturing: This stage involves transforming raw materials and components into finished products.
3. Warehousing: This stage involves storing finished products until they are ready to be shipped.
4. Transportation: This stage involves moving finished products from warehouses to distribution centers or retailers.
5. Retail: This stage involves selling finished products to end-users.

**Procurement**

The procurement stage involves sourcing raw materials and components from suppliers. This can be a complex process, as it involves finding suppliers who can provide the necessary materials at a competitive price. It is also important to ensure that the suppliers are reliable and can deliver the materials on time.

**Manufacturing**

The manufacturing stage involves transforming raw materials and components into finished products. This can be a highly automated process, or it can be done more manually. The manufacturing process will vary depending on the type of product being produced.

**Warehousing**

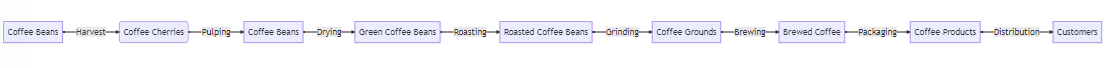
The warehousing stage involves storing finished products until they are ready to be shipped. This is an important stage, as it ensures that the products are stored in a safe and secure environment. It is also important to track the inventory levels, so that the company knows how many products are in stock.

**Transportation**

The transportation stage involves moving finished products from warehouses to distribution centers or retailers. This can be done by truck, rail, or air. The mode of transportation will vary depending on the distance the products need to be shipped and the cost of transportation.

**Retail**

The retail stage involves selling finished products to end-users. This can be done through brick-and-mortar stores, online stores, or a combination of both. The retail stage is important, as it is where the company generates revenue.



## Challenges and Inefficiencies

The current supply chain process is facing a number of challenges, including:

* Inefficiency: The current supply chain process is often inefficient, with products sitting in warehouses for long periods of time before they are sold. This can lead to increased costs and decreased profits.
* Complexity: The current supply chain process is complex, with many different stakeholders involved. This can make it difficult to track products and ensure that they are delivered on time and in good condition.
* Vulnerability: The current supply chain process is vulnerable to disruptions, such as natural disasters or cyberattacks. This can lead to shortages and price increases.

## What is a Smart contract-based supply chain management system?

A smart contract-based supply chain management system can help to address some of the challenges and inefficiencies of the current supply chain process. Smart contracts are self-executing contracts that are stored on a blockchain. This means that they are transparent and immutable, which can help to improve trust and efficiency. A smart contract-based supply chain management system can track products from the supplier to the end-user. This can help to ensure that products are delivered on time and in good condition. The system can also help to reduce costs by eliminating the need for intermediaries.

# 

# Step 2: Identify the Stakeholders

- - - - X

The stakeholders in a supply chain are the individuals or organizations that have a vested interest in the success of the supply chain. They can include suppliers, manufacturers, distributors, retailers, and end-users.

**Suppliers**

Suppliers are the individuals or organizations that provide the raw materials and components that are used to make the products in the supply chain. They are an important part of the supply chain, as they provide the essential materials that are needed to produce the products.

**Manufacturers**

Manufacturers are the individuals or organizations that transform the raw materials and components into finished products. They are an important part of the supply chain, as they are responsible for creating the products that are sold to end-users.

**Distributors**

Distributors are the individuals or organizations that store and transport the finished products from the manufacturers to the retailers. They are an important part of the supply chain, as they help to ensure that the products are available to the retailers when they need them.

**Retailers**

Retailers are the individuals or organizations that sell the finished products to end-users. They are an important part of the supply chain, as they are the ones who ultimately sell the products to the people who will use them.

**End-users**

End-users are the individuals or organizations that purchase the finished products from the retailers. They are the ultimate consumers of the products in the supply chain, and their satisfaction is essential to the success of the supply chain.

### 

## Stakeholders Roles and Responsibilities

Each stakeholder in a supply chain has a different role and set of responsibilities. These roles and responsibilities are important to the smooth operation of the supply chain.

**Suppliers**

The roles and responsibilities of suppliers include:

* Providing high-quality raw materials and components on time
* Meeting customer expectations
* Working with manufacturers to develop new products
* Keeping costs low

**Manufacturers**

The roles and responsibilities of manufacturers include:

* Transforming raw materials and components into finished products
* Meeting customer expectations
* Keeping costs low
* Working with suppliers to develop new products

**Distributors**

The roles and responsibilities of distributors include:

* Storing finished products in a safe and secure environment
* Transporting finished products from manufacturers to retailers
* Meeting customer expectations
* Keeping costs low

**Retailers**

The roles and responsibilities of retailers include:

* Selling finished products to end-users
* Meeting customer expectations
* Keeping costs low
* Providing excellent customer service

**End-users**

The roles and responsibilities of end-users include:

* Purchasing finished products from retailers
* Using finished products in a safe and responsible manner
* Providing feedback to retailers and manufacturers

## Conflicts and Collaboration

There are a number of potential conflicts or areas of collaboration among stakeholders in a supply chain. For example, suppliers and manufacturers may have conflicting goals. Suppliers may want to sell raw materials and components at a high price, while manufacturers may want to buy them at a low price. This can lead to conflict between the two parties. However, there are also areas where suppliers and manufacturers can collaborate. For example, they can work together to develop new products or to improve the efficiency of the supply chain.

## Step 3: Design the Smart Contract

- - - - X

## Determine the Requirements for the Smart Contract

The first step in designing the smart contract is to determine the requirements. This includes identifying the functionality and features that our smart contract needed to have. For example, we wanted the smart contract to be able to track the movement of products, store data about the products, and execute transactions.

The following are the requirements for the smart contract:

* The smart contract should be able to track the supply chain of a coffee product from procurement to distribution.
* There should be four stages in the supply chain: procurement, manufacturing, packaging, and distribution.
* The smart contract should be able to identify suppliers who are authorized to participate in the supply chain.
* The smart contract should be able to identify the current stage of a product in the supply chain.
* The smart contract should be able to distribute a product to a distributor if there is enough inventory space.
* The smart contract should be able to prevent distribution if there is not enough inventory space.
* The smart contract should be able to store the details of each product, including its ID and current stage in the supply chain.
* The smart contract should have an owner who is authorized to add users and assign roles.

Starbucks Specific Requirements:

1. The system should be able to manage the inventory of coffee beans, milk, syrups, and other ingredients used in making coffee and other beverages.
2. The system should be able to track the sales of different types of beverages, including hot and cold coffee, tea, and other drinks.
3. The system should be able to calculate the cost of each beverage based on the quantity and cost of the ingredients used.
4. The system should be able to generate reports on sales, inventory levels, and other metrics to help managers make informed decisions.

## Define the Data Structures and Variables that Will be Used in the Smart Contract

Since we have determined the above requirements, the next step is to define the data structures and variables that will be used in the smart contract. This includes identifying the types of data that will be stored in the smart contract and the relationships between the data.

The following are the data structures and variables that will be used in the smart contract:

* Product: A struct that represents a coffee product. It should have two fields: id, which is a unique identifier for the product, and stage, which represents the current stage of the product in the supply chain.
* Supplier: A struct that represents a supplier. It should have one field: supplierAddress, which is the Ethereum address of the supplier.
* products: A mapping that maps a product ID to a Product object.
* suppliers: A mapping that maps a supplier address to a Supplier object.
* inventorySpace: An integer that represents the available inventory space.
* roles: A mapping that maps a role name (as a string) to a mapping of addresses to booleans indicating whether the address has that role.
* identifySupplier(): A function that identifies a new supplier and adds them to the suppliers mapping.
* inventoryHasSpace(): A private function that checks whether there is enough inventory space to distribute a product.
* distributeProduct(): A private function that performs the distribution logic and updates the inventory and product stage.
* identifySuppliers(): A function that identifies a new product and updates its stage in the supply chain based on the current stage and the role of the caller.

CODE:

// SPDX-License-Identifier: GPL-3.0

pragma solidity >=0.8.20;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";

contract CoffeeSupplyChain is ERC20 {

constructor(uint256 initialSupply) ERC20("Coffee", "CHOC") {

\_setRole("owner", msg.sender);

\_mint(msg.sender, initialSupply);

}

enum Stage {

Procurement,

Manufacturing,

Packaging,

Distribution

}

struct Supplier {

address supplierAddress;

bool exists;

}

struct Product {

uint256 id;

Stage stage;

}

mapping(uint256 => Product) public products;

mapping(address => Supplier) private suppliers;

uint256 private inventorySpace = 10; // Example inventory space, you can change it

// Roles

mapping(string => mapping(address => bool)) private \_roles;

modifier onlyRole(string memory role) {

require(

hasRole(role, msg.sender),

"CoffeeSupplyChain: Only authorized role can access this function"

);

\_;

}

function hasRole(string memory role, address account) public view returns (bool) {

return \_roles[role][account];

}

function AddUser(string memory role, address account) public onlyRole("owner") {

\_setRole(role, account);

}

function \_setRole(string memory role, address account) internal {

\_roles[role][account] = true;

}

function identifySupplier(address supplierAddress) public onlyRole("owner") {

require(!suppliers[supplierAddress].exists, "CoffeeSupplyChain: Supplier already exists");

suppliers[supplierAddress] = Supplier(supplierAddress, true);

}

function inventoryHasSpace() private view returns (bool) {

return inventorySpace > 0;

}

function distributeProduct(uint256 productId) private {

inventorySpace--; // Reduce inventory space by 1

products[productId].stage = Stage.Distribution;

// Perform distribution logic here

// Update inventory and track distribution details

}

function identifySuppliers() public onlyRole("owner") {

uint256 productId = uint256(keccak256(abi.encodePacked(block.timestamp, msg.sender)));

Product storage product = products[productId];

require(product.id == 0, "CoffeeSupplyChain: Product with this ID already exists");

product.id = productId;

product.stage = Stage.Procurement;

if (suppliers[msg.sender].exists) {

product.stage = Stage.Manufacturing;

// Packaging

product.stage = Stage.Packaging;

// Distribution

if (inventoryHasSpace()) {

distributeProduct(productId);

} else {

revert("CoffeeSupplyChain: Inventory is full, distribution stopped");

}

}

}

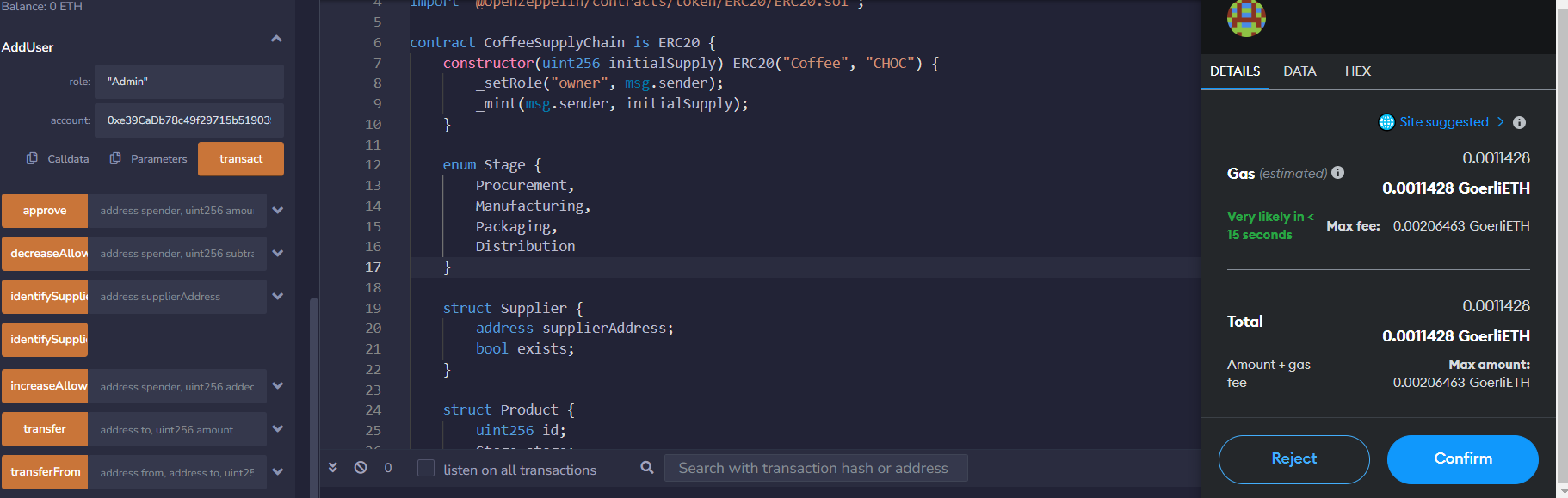
}

Testing:

Smart contract Address:

0x102501670FdDfe87e43ADe695BfEEeD61B836dfb

AddUser: “Admin”



# 

# 

# Step 4: DeFi Apps

- - - - X

Alchemy

API KEY:

L6eIu8hbA8lHu5MngiPVZR723UF7hbW8

HTTPS:

<https://eth-mainnet.g.alchemy.com/v2/L6eIu8hbA8lHu5MngiPVZR723UF7hbW8>

WEBSOCKETS:

wss://eth-mainnet.g.alchemy.com/v2/L6eIu8hbA8lHu5MngiPVZR723UF7hbW8

Metamask

Hardhat Project

Install Env