SMARTER
EXPERIENTIAL LEARNING TOOLS

SupplyChainSpace - Quick use and configuration guide

2022-1-FI01-KA220-HED-000086152

Co-funded by the
Erasmus+ Programme
of the European Union

Index

Introduction

SupplyChainSpace is an educational tool, specifically a decentralized application (dApp), designed to enhance experiential learning in smart supply chain management. By leveraging blockchain technology and smart contracts, this tool enables students to construct, modify, and analyze various supply chain processes—from the initial production stage to final consumer delivery. Additionally, it offers a hands-on experience with emerging technologies such as blockchain and smart contracts, fostering familiarity with their concepts and applications.

Understanding the Difference Between Traditional Web Applications & dApps

To better grasp how a dApp functions, it is useful to compare it with a traditional web application.

In a **traditional web application**—such as e-commerce platforms like Amazon—users interact with a privately managed web server, which coordinates with a centralized database. This database, controlled by a company, stores all user-related data, including personal information (e.g., name, address, credit card details). Authentication requires users to create an account and submit their personal details to the private server, where all transactions are processed.

In contrast, a **decentralized application (dApp)** operates differently. While the web server may be either private or public, its primary function is to facilitate interaction with the blockchain—a distributed, peer-to-peer network that securely stores both user data and smart contracts governing their management.

The **Ethereum network**, which underpins many dApps, employs public-key cryptographic techniques to protect user privacy. To engage with a dApp, users must have a cryptocurrency wallet that manages their account. This wallet may optionally store personal details but must contain the user's **private key pair**, which is required for signing transactions.

Within this framework, the blockchain transparently records transaction data, which can then be accessed via the web interface. When a transaction is initiated, the cryptocurrency wallet signs it using the user's private key before broadcasting it to the blockchain. Consequently, all transactions occur in a secure, decentralized manner, eliminating the need for a centralized entity to manage user data and financial transactions.

By using **SupplyChainSpace**, students not only gain insight into supply chain dynamics but also develop practical experience with blockchain-based applications, learning how digital transactions are secured, validated, and executed in a decentralized environment.

Metamask: Configuration and Usage Guide

Introduction to Metamask

Metamask is a widely used cryptocurrency wallet and serves as a gateway to decentralized applications (dApps) operating on the Ethereum blockchain. It facilitates secure management of digital assets and interactions with smart contracts. In Metamask, digital identity is established through the generation and management of cryptographic keys, ensuring a high level of security and privacy.

Core Functions of Metamask

1. **Key Generation** – When a user creates a new Metamask wallet, the software automatically generates a **pair of cryptographic keys**:
    - The **private key**, which serves as the user's digital identity and grants access to funds and transactions.
    - The **public key**, which derives a unique Ethereum address associated with the wallet.
2. **Ethereum Addresses** – The public key generates an **Ethereum address**, which acts as a user's unique identifier on the blockchain. This address is used to receive funds and interact with smart contracts.
3. **Transaction Signing** – When a user initiates a transaction, such as transferring cryptocurrency or executing a smart contract, Metamask uses the **private key** to

digitally sign the transaction. This process verifies authenticity and ensures that the transaction originates from the rightful owner of the wallet.

4. **Key Management and Security** – Metamask offers backup and recovery options, including:
    - A **recovery phrase** for wallet restoration.
    - Additional security measures such as two-factor authentication (2FA).

Privacy and Anonymity

Metamask does not store any **personally identifiable information** beyond cryptographic keys and Ethereum addresses. This design enhances user privacy and anonymity, ensuring that blockchain transactions remain pseudonymous.

Student Accounts and Network Balance

Each student will be provided with a **preconfigured Metamask account** with a balance allocated for transactions within the tool. However, it is important to note that each Ethereum account holds a balance on a specific network. This means that the displayed balance can only be used for transactions on the network to which the wallet is connected.

In this educational setup, the tool operates on a **private Ethereum network**, where the network administrator assigns funds to user accounts. While real blockchain technology is being used, the financial transactions occur with simulated tokens—similar to using "Monopoly money" in a controlled learning environment.

Metamask Configuration Guide

Metamask is available as a **browser extension** and can be installed from the extension stores of Firefox or Chrome-based browsers (Brave is recommended for an optimized experience).

Step 1: Installing Metamask

1. Download Metamask from the **Firefox** or **Chrome** extension store.
2. Install the extension and launch it for the first time.
3. Follow the on-screen instructions to create a new wallet and store the **recovery phrase** securely.

Step 2: Exploring the Metamask Interface

Once Metamask is set up, users may familiarize themselves with its interface through the built-in guidance provided within the extension.

Step 3: Adding a Custom Ethereum Network

Since the tool operates on a **private Ethereum network**, users must manually configure the network within Metamask:

1. Click on the **network selector** (top left corner).
2. Select **“Add network” → “Add a network manually”**.
3. Enter the following network details:

- **Network Name:** SMARTER
- **RPC URL:** `http://157.88.123.28:8406`
- **Chain Identifier:** 12345
- **Currency Symbol:** ETH
- **(Leave optional fields blank)**

4. Click **"Save"** and switch to the newly added SMARTER network.

Step 4: Importing the Provided Student Account

Students must import the account assigned to them by the instructor:

1. Click on the **account name** at the top center.
2. Select **"Add an account to the wallet" → "Import account"**.
3. Copy and paste the **private key** provided by the instructor into the designated field.
4. Click **"Import"** to finalize the process.

Once successfully imported, students can now perform transactions within the **SupplyChainSpace dApp** using Metamask.

A brief guide to the dApp SupplyChainSpace

Overview

The **SupplyChainSpace** dApp has been developed as a **sandbox environment**, allowing users to model various supply chain processes and, if necessary, interconnect them. Additionally, it facilitates **financial transactions** between different agents involved in the supply chain using **Ether (ETH), the cryptocurrency of the Ethereum network**.

At its core, **a supply chain consists of sequential steps that track the transformation and distribution of a product (batch or item) or service from its origin to the final consumer**. The application enables students to **simulate real-world supply chain events dynamically**, adding steps as necessary to reflect the processes involved in the distribution of a product.

Each step represents a **significant event** in the supply chain, **storing relevant information on the blockchain**. These steps may involve:

- **A single agent** (e.g., the current owner of the batch at a specific stage), or
- **Two agents** (e.g., a seller and a buyer, or a producer and a manufacturer), including financial transactions between them.

This **sandbox approach** empowers students to **experiment with different supply chain models**, combining multiple chains when necessary, and encouraging **creativity, critical thinking, and problem-solving**. Through this process, users gain insights into the **complexities, interdependencies, and potential efficiencies** that blockchain technology can introduce into supply chain management.

Blockchain Integration and Transactions

Each recorded step is stored on the **Ethereum blockchain**, ensuring transparency, immutability, and security. If an economic transaction is required, a **smart contract is automatically created** to facilitate and regulate the transaction.

All financial transactions—including payments between agents and associated fees—are conducted in **Ether (ETH)**. This mechanism introduces students to the practical applications of **digital assets, tokenized transactions, and blockchain-based payments**.

Types of Steps in the Supply Chain

The tool provides **five primary step types**, each serving a distinct function within the supply chain:

Step	Description
Produce	• First step in the supply chain, representing the production of a good. • Requires a unique reference for batch/item identification and a description. • The **owner of the batch/item is the creator** of this step.
Manufacture	• Represents the creation of a new product using existing batches/items already stored in the system. • Requires a unique reference for the new batch, a description, and identification of contributing batches/items. • The **owner of the new product is the creator** of this step.
Process	• Represents **a transformation or modification of a product**. • Can be performed by the owner or a third party (in which case an **economic transaction** is required). • Requires a description, recipient account (if applicable), and the transaction amount in Ether.
Transport	• Represents the **logistical movement of a product**. • **Always performed by a third party** who receives payment for the service. • Requires a description, recipient account, and the transaction amount in Ether.
Sell	• Represents **the transfer of ownership of a product**. • Upon receipt of payment, ownership is transferred to the buyer. • Requires a description, sender's account, and the transaction amount in Ether.

Educational Constraints and Features

Since the software is an **educational tool**, certain **rules and restrictions** have been implemented to **simplify the learning process** and illustrate key blockchain functionalities:

- **Step Ownership:** Only the **current owner of a batch/item** can add new steps to the supply chain.

- **Mandatory Third-Party Transport:** The **transportation of goods must always be performed by an external agent**, reflecting real-world logistics.
- **Blocked Transactions:** When a financial transaction is required, the corresponding step is temporarily **"blocked"** until payment is completed. This prevents further steps from being added until all pending transactions are resolved. *(Note: This constraint simplifies the educational experience; real-world applications are typically more complex.)*
- **Step Status Updates:** Additional status updates can be added to a step, **even if the step is blocked**. These updates can be made by the owner or relevant agents involved in the economic transaction.
- **Step Deletion:** The **last step of a supply chain can be removed**, provided at least one other step remains. However, **any fees or payments associated with the removed step remain recorded on the blockchain** and are not reversed.

Illustrative Example: A Coffee Supply Chain

To demonstrate how the tool operates, consider a **coffee supply chain**, which could be represented as follows:

1. **Produce:** A farmer cultivates and harvests coffee beans.
2. **Sell:** The farmer sells the beans to a coffee processing company, receiving payment.
3. **Transport:** The company arranges for the coffee beans to be transported via a logistics provider, paying for the service.
4. **Process:** The factory **roasts and grinds** the coffee beans.
5. **Process:** The factory **packages the coffee** into consumer-ready units.
6. **Sell:** The company sells the packaged coffee to a retailer, receiving payment.
7. **Transport:** The retailer arranges transportation of the packaged coffee to stores, paying a logistics provider.

Each step in this supply chain is **securely recorded on the blockchain**, along with its associated data. Additionally, each financial transaction is governed by **smart contracts**, ensuring that details such as **agents involved, transaction amounts, and timestamps** are securely stored. Transactions are executed using **Ether**, with funds automatically transferred between accounts.

Navigating the SupplyChainSpace dApp. User Interface Overview

Upon accessing the tool (currently in its development phase), users will encounter several key interface elements:

1. **Help Section:**
   - Located at the top of the screen and available on all pages.
   - Can be toggled on or off for guidance.
2. **Creating a New Supply Chain:**
   - A button at the **top-right corner** allows users to create a new supply chain, either by **producing a new product** or **manufacturing an item from existing products**.
3. **Viewing Existing Supply Chains:**

- Users can **select one or more supply chains** from those stored on the blockchain.
- Pressing the **refresh button** updates the displayed chains.

4. **Graphical Representation of Supply Chains:**
   - Displays the **name, number of steps, and chronological ordering** of steps within a selected supply chain.
   - Provides a **management button** for adding new steps or making pending payments.
5. **Step Information Display:**
   - Hovering over a step **reveals detailed information**, including:
     - **Creation date**
     - **Current owner of the product**
     - **Additional recorded information**
   - If a step includes a financial transaction, the associated payment details are also displayed.

By utilizing these features, students can effectively **model, analyze, and interact with supply chains** in a **realistic yet controlled blockchain environment**. The **SupplyChainSpace dApp** provides a **hands-on learning experience**, enhancing understanding of **decentralized networks, smart contracts, and supply chain management** in a practical, engaging manner.