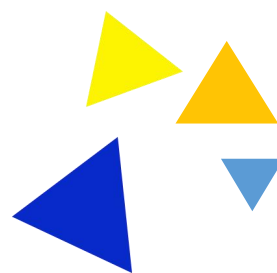# Test Project (Offline)

BRICS-FS-07

Blockchain-based Solutions

2022 BRICS Skills Competition

CONTENTS

Module 1: Engineering documentation (10 points)

Question 1: Improve the requirements document

[Mission Requirements]

Based on the provided product requirement document, improve the functions of "Food Traceability Information Storage" in the requirement document. Submit a screenshot of the revised document to the project document.

[Project Code]

Get the "01_Requirements_Document.docx" file in the "Competition /01_Engineering_Document_Writing/" folder on the desktop, edit it, and save the result to the "Submit_Document/01/" folder on the desktop.

Question 2: Improve design documentation

[Mission Requirements]

Refine the functions of the "Farm" role in the design document based on the product design document Submit a screenshot of the revised document to the project file.

[Project Code]

Get the "02_design_document.docx" file in the folder of "Competition/01_Engineering_document_writing/" on your desktop, edit it, and save the result to the folder of "Submit_document/01/" on your desktop.

Module 2: Blockchain System Operation and Maintenance (20 points)

Question 3: Installation of FISCO BCOS Blockchain System

[Mission Requirements]

Install blockchain related dependencies in Linux operating system, download FISCO BCOS blockchain system installation package and related tools, and install FISCO BCOS blockchain system. Submit the installation process and running results to the project documentation.

[Project Code]

Get the code in the folder "03_code" in the folder "Question/02_Blockchain System Operation and Maintenance/" on the desktop, and open the file "03_Installation of FISCO BCOS Blockchain System.docx" in the folder "Question/02_Blockchain System Operation and Maintenance/". Run and save the screenshot of the file, and then save the file to the desktop "Submission Documents/02/" folder after completing the question.

Question 4: Building the FISCO BCOS blockchain system

[Mission Requirements]

Configure blockchain build configuration file in Linux OS, use blockchain builder tool to generate single 4-node blockchain system, set starting ports of p2p, channel, jsonrpc to 30300, 20200, 8545, start blockchain system, check blockchain process, listen to ports, log. Submit the installation process and running results to the project documentation.

[Project Code]

Get the file "04_Build FISCO BCOS Blockchain System.docx" in the folder "Competition Questions/02_Blockchain System Operation and Maintenance/" on your desktop, save the screenshot of the file, and then save the file to your desktop after you finish the question. Save the file to the "Submit Documents/02/" folder.

Question5: Permission assignment

[Mission Requirements]

Install and configure the console for the built FISCO BCOS blockchain system, use console commands to set 2 committee accounts (A and B), 1 operation and maintenance account, and set the weight of committee A to 2, and finally pass the operation and maintenance account. Deploy the HelloWorld contract. Submit the rights assignment process and the screenshot of the operation result to the project document.

[Project Code]

Take the "05_Permission Assignment" file in the "Question/02_Blockchain System Operation and Maintenance/" folder on the desktop, save the screenshot of the result in this file, and save the file to the desktop "Submit File/" after completing this question 02" folder.

Question6: Building WeBASE-Front node front services

[Mission Requirements]

Download the WeBASE-Front installation package in the Linux operating system, configure it so that it connects to the local FISCO BCOS node, start the service after configuration and use a browser to access the web interface. Submit the installation process and running results to the project documentation.

[Project Code]

Get the code in the folder "05_code" in the folder "Race /02_Blockchain System Operation and Maintenance/" on the desktop, and open the folder "06_code" in the folder "Race /02_Blockchain System Operation and Maintenance/" on the desktop. "06_Build WeBASE-Front node front service.docx" file in the folder, save the screenshot of the file, and save the file to the desktop "Submit Documents/02/" folder after finishing the problem. Save the file to your desktop in the "Submit Documents/02/" folder.

Module 3: Traceability Depository Smart Contract Development (40 points)

Question 7: New farm role

[Mission Requirements]

Visit the fisco-bcos:5002/WeBASE-Front address in the browser to enter the smart contract IDE. Modify the Food.sol file, add the newFood function, add a new farm role, complete the basic vegetable information creation, farm role creation and verification, and complete the farm upload operation. Submit a

screenshot of the completed code to the project file.

[Project Code]

Get the file "07_New Farm Role.docx" in the folder "03_Traceability Smart Contract Development/" on your desktop, save the screenshot of the file, and then save the file to your desktop after you finish the question. "Save the file to your desktop after you finish the question.

Question 8: New middlemen role

[Mission Requirements]

Visit the fisco-bcos:5002/WeBASE-Front address in the browser and enter the smart contract IDE. Modify the food.sol file, add the addTraceInfoByMiddlemen function, add the middlemen role, middlemen role creation and verification, and complete the middlemen uplink operation. Submit the screenshot of the completed code to the project document.

[Project Code]

Get the file "08_Add Intermediary Role.docx" in the folder of "Question/03_Traceability Smart Contract Development/" on your desktop, save the screenshot of the file, and then save the file to your desktop after you finish the question. Save the file to the "Submit Documents/03/" folder.

Question 9: New supermarket role

[Mission Requirements]

Visit the fisco-bcos:5002/WeBASE-Front address in the browser and enter the smart contract IDE. Modify the Food.sol file, add the addTraceInfoBySupermarket function, add the supermarket role, supermarket role creation and verification, and complete the supermarket uplink operation. Submit the screenshot of the completed code to the project document.

[Project Code]

Get the file "09_Add supermarket role.docx" in the folder of

"Question/03_Trace and Store Smart Contract Development/" on your desktop, save the screenshot of the file, and save the file to "Submit Documents/03/" folder   on your desktop after you finish the question.

Question 10: New traceability function

[Mission Requirements]

Visit the fisco-bcos:5002/WeBASE-Front address in your browser to access the smart contract IDE. M/odify the Trace.sol file to supplement the getTraceInfo function to return traceability information. Submit a screenshot of the completed code to the project documentation.

[Project Code]

Get the file "10_Add Traceability Function.docx" in the folder of "Question/03_Traceability Smart Contract Development/" on your desktop, save the screenshot of the file, and save the file to your desktop after you finish the question. "Save the file to your desktop after you finish the question.

Question 11: Smart Contract Invocation

[Mission Requirements]

Visit the fisco-bcos:5002/WeBASE-Front address in your browser and enter the smart contract IDE. Compile and deploy the completed Food.sol file and Trace.sol file, call the newFood, addTraceInfoByMiddlemen, addTraceInfoBySuppermarket, and getTraceInfo functions.   Submit screenshots of the call parameters and running results to the project document.

[Project Code]

Get the file "11_Smart Contract Call.docx" in the folder "Question/03_Traceability Smart Contract Development/" on your desktop, save the screenshot of the file, and then save the file to "Submit Documents/03/" folder on your desktop after you finish the question. "

Module 4: Traceability Depository Application Development (25 points)

Question 12: Call smart contract using Go SDK

[Mission Requirements]

Visit fisco-bcos:5002/WeBASE-Front address in the browser, enter the smart contract IDE, Export the food-trace folder, compile the abigen program in the FISCO BCOS Go SDK, use the abigen program to generate the go file from the abi file and the bin file, introduce the generated go file into the web backend program, use the Go SDK to call the newFood function and getTraceInfo function in the smart contract. The generated go file is introduced into the web backend application, and the Go SDK is used to call the newFood function and getTraceInfo function in the smart contract in the go application. Submit a screenshot of the runtime and results to the project documentation.

[Project Code]

Get the code in the "12_code" folder in the "Question/04_Trace and Traceability Application Development/" folder on the desktop, and open the "12_Application_Development_using_the_Go_SDK_invoke.docx" file in the "Question/04_Trace and Traceability Application Development/" folder, save the file as a screenshot of the operation, and save the file to the "Submit Documents/04/" folder on the desktop after completing this question.

Question 13: Interacting with Smart Contracts in Web

Pages

[Mission Requirements]

The code realizes the functions of each page and role in the food traceability application system：

1、 Farm: Support adding basic information of vegetable, and after successful addition, require the intermediary page to also display corresponding food information and be able to query traceability records through traceability codes. In WeBASE-Front node console-contract management-contract list-contract call position, call the contract to query the traceability records by traceability code, and submit the result screenshot to the project document.

2. Intermediaries: Support adding the quality information of vegetable. After successful addition, the supermarket page is required to display the corresponding food information and query the traceability records through the traceability code. In WeBASE-Front node console-Contract Management-Contract List-Contract Call position, call the contract to query the traceability records by traceability code and submit the result screenshot to the project document.

3、Supermarket: Support adding vegetable and food quality information. After successful addition, you can query the traceability records through the traceability code and submit the result screenshot to the project document.

[Project Code]

Obtain the code in the "13_code" folder in the "Question/04_Trace and Traceability Application Development/" folder on the desktop, and open the "13_Interacting with Smart Contracts in Web Pages.docx" file in the "Question/04_Trace and Traceability Application Development/" folder. Save the screenshot of the file, and save the file to the "Submit Documents/04/" folder on your desktop after completing the question.

Module 5: Professionalism (5 points)
Question 14: Professionalism

[Mission Requirements]

The contestants should operate in a standardized manner, comply with the discipline of the examination room, organized clean and tidy storage, good safety awareness and civilized competition.