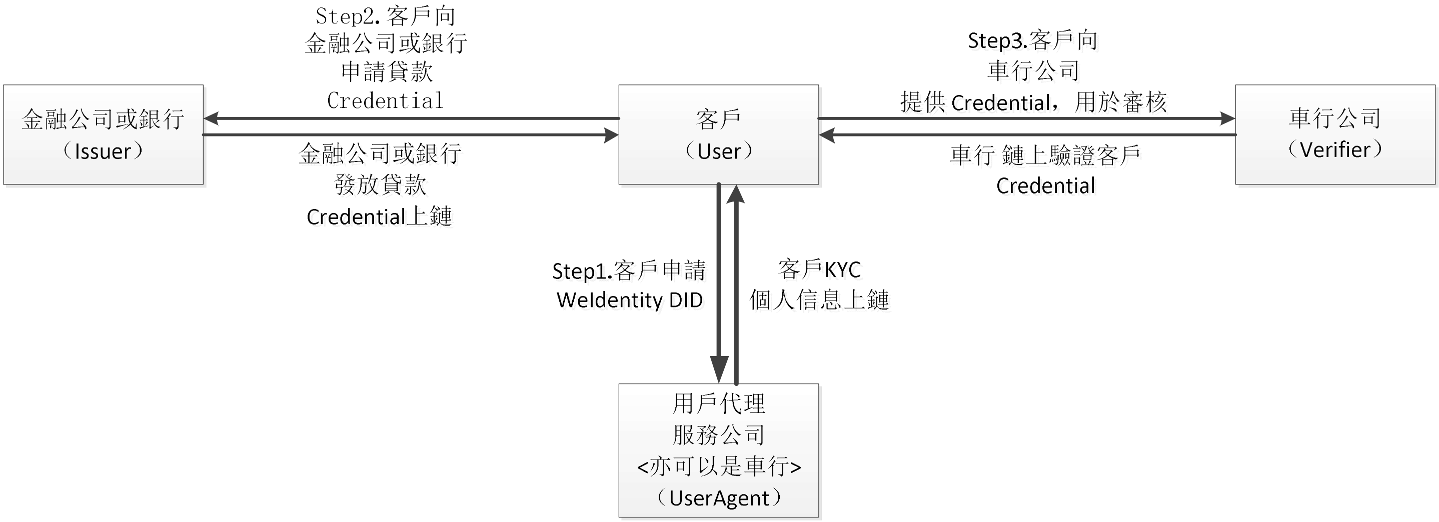
課堂大作業：應用 **WeIdentity** 實現一個‘金融貸款購車案例’

■ **WeIdentity 大作業**（金融貸款憑證----------實現一個‘金融貸款購車案例’）：

作業場景描述：一家車行，在爲客戶辦理購車手續時需要驗證客戶的貸款是否已發放，確定客戶貸款信息的真實性，當前存在的問題是，對于車行而言，缺乏高效即時驗證貸款及客戶信息真實性的手段，對于客戶而言，可以避免在貸款發放過程中二頭跑。基于 WeID，用戶身份同電子身份 ID 一一對應，真實金融貸款信息電子化，幷一同保存在區塊鏈上，保證身份信息真實不可篡改，驗證身份信息快速高效。



-------客戶進行 WeIdentity DID 注冊

1.客戶進入用戶代理公司的客戶信息（KYC）注册頁面，提交個人信息

2.用戶代理公司<亦可以是車行>審核申請，成功後授予該客戶 Weidentity DID

------客戶向金融公司或銀行申請購車貸款憑證（Credential）

1.客戶向金融公司或銀行提交購車貸款憑證（Credential）申請

2. 金融公司或銀行審核申請，成功後授予該客戶購車貸款憑證 Credential 上鏈

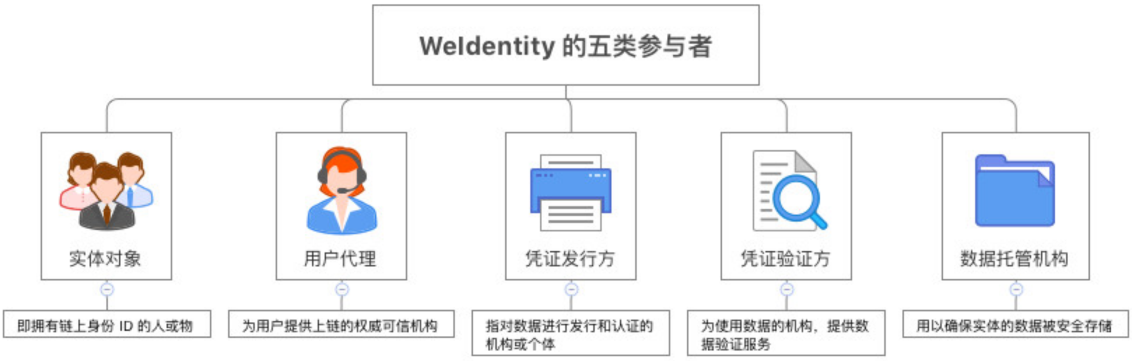
------客戶提交貸款憑證（Credential），用于驗證

1.客戶向車行提交自己的貸款憑證 Credential，車行鏈上驗證真實性後交付汽車

Final Project：實現一個‘金融貸款購車案例’				
No.編號	**Name 名稱**	**Role 角色**	**Resp 說明**	現場講解
1	Davis	小組長	案例描述及場景說明；角色分配；基本流程測試與設計；	一至四（概述）
2	Matthew	副小組長/金融公司或銀行(Issuer)	基本流程測試及跑通；金融公司或銀行(Issuer)方編碼；	五（概述、金融公司或銀行(Issuer)方）
3	Nelson	組員/金融公司或銀行(Issuer)	金融公司或銀行(Issuer)方編碼；配合基本流程測試及跑通；	五（概述、金融公司或銀行(Issuer)方）
4	Tao	組員/客戶(User)	組員/客戶(User) 方編碼；配合基本流程測試及跑通；	五（概述、客戶(User)）
5	Henry	組員/車行公司(Verifier)	車行公司(Verifier)方編碼；配合基本流程測試及跑通；	五（概述、車行公司(Verifier)）
備注：請大家按如下實驗作业練習過程步驟（一至四步驟，參考“3、WeIdentity 课堂练习.pptx”及 weid-sample 及 HTTP-Service 構建配合完成大作業！**最後我組決定選用 HTTP-Service+PostMan 方式實現案例。**）				

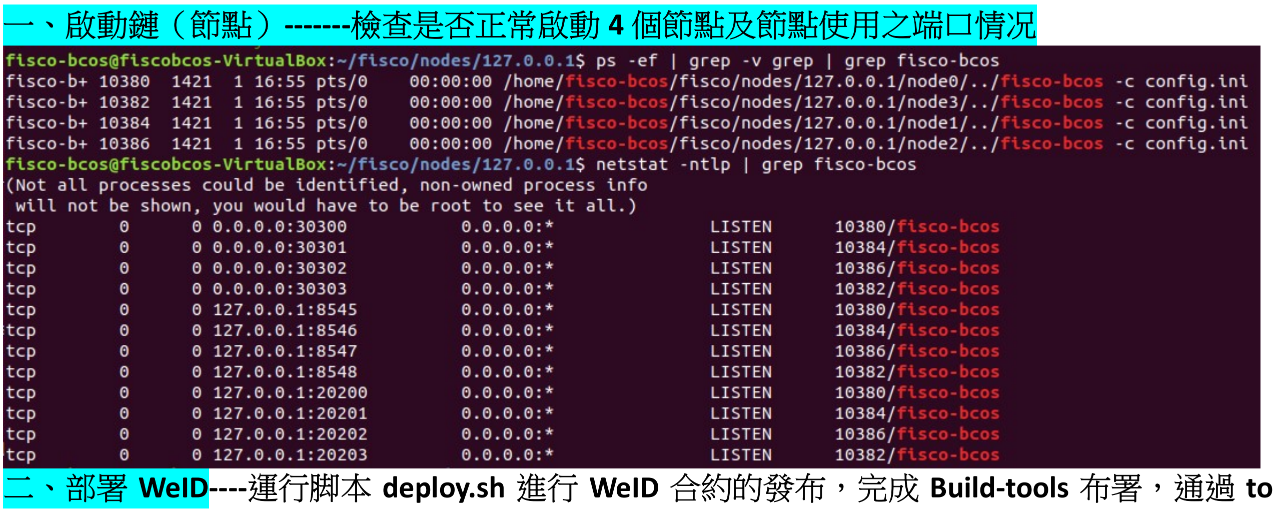
WeIdentity 方案	VS	傳統方案
電子化貸款憑證成本低，貸款證明文件轉為 Credential，可信存儲，易於管理，可以即時查詢及驗證，客戶不需要二頭跑；	**成本與便利性**	紙質貸款憑證文件管理成本高，易丟失，不易更新，客戶需要二頭跑；
鏈上客戶身份與現實身份一一對應；密碼學確保真實性和有效性，區塊鏈記錄確保不可篡改性；	**真實性**	紙質貸款文件有造假可能性，需銀行-車行互相驗真周期長、效率低；
全流程無第三方機構參與，金融公司或銀行可以選擇性向車行公司披露客戶信息；用戶授權後，車行公司方能獲取貸款批核信息，隱私保護性強；	**隱私保護**	金融公司或銀行需要給車行披露客戶資料，資料傳遞的中間過程可能導致用戶隱私洩漏問題；
方案可擴展性強，增加金融公司或銀行、車行數量，工作量不會相應增加，甚至可以聯合貸款	**可擴展性**	方案可擴展性差，每增加一個金融公司或銀行、車行公司，工作量會線性增加

參考：以下 WeIdentity 的五類參與者角色分工如下------------------
實體對象：客戶；用戶代理：用戶代理服務公司<亦可以是車行>；憑證發行方：金融公司或銀行；憑證驗證方：車行公司；數據托管機構（VM 模擬器）；


WeIdentity 的五类参与者
实体对象
用户代理
凭证发行方
凭证验证方
数据托管机构
即拥有链上身份 ID 的人或物
为用户提供上链的权威可信机构
指对数据进行发行和认证的机构或个体
为使用数据的机构，提供数据验证服务
用以确保实体的数据被安全存储

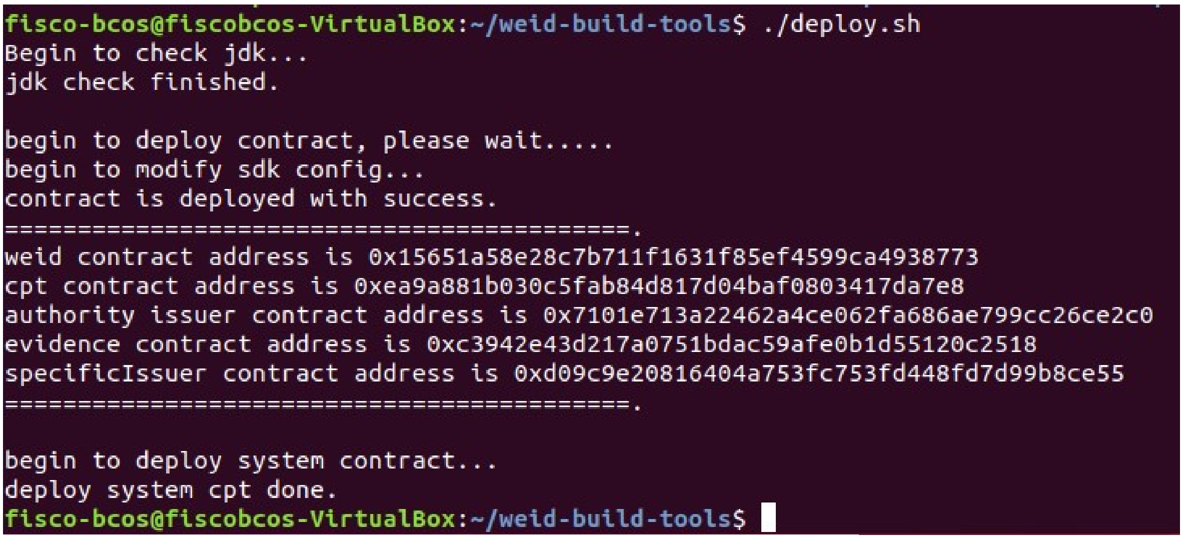
其它：實驗大作业過程說明如下：

一、啟動鏈（節點）-------檢查是否正常啟動 4 個節點及節點使用之端口情況

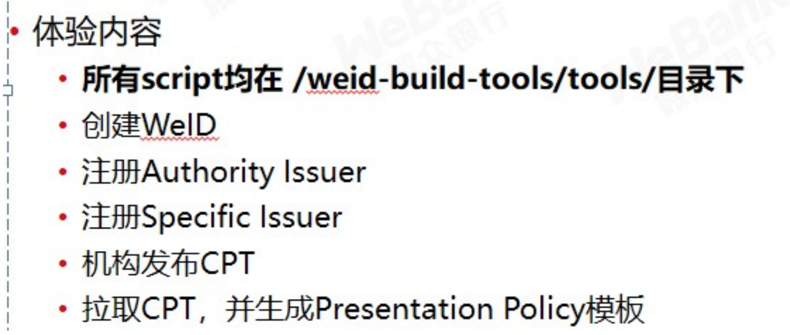


```
fisco-bcos@fiscobcos-VirtualBox:~/fisco/nodes/127.0.0.1$ ps -ef | grep -v grep | grep fisco-bcos
fisco-b+ 10380  1421  1 16:55 pts/0    00:00:00 /home/fisco-bcos/fisco/nodes/127.0.0.1/node0/../fisco-bcos -c config.ini
fisco-b+ 10382  1421  1 16:55 pts/0    00:00:00 /home/fisco-bcos/fisco/nodes/127.0.0.1/node3/../fisco-bcos -c config.ini
fisco-b+ 10384  1421  1 16:55 pts/0    00:00:00 /home/fisco-bcos/fisco/nodes/127.0.0.1/node1/../fisco-bcos -c config.ini
fisco-b+ 10386  1421  1 16:55 pts/0    00:00:00 /home/fisco-bcos/fisco/nodes/127.0.0.1/node2/../fisco-bcos -c config.ini
fisco-bcos@fiscobcos-VirtualBox:~/fisco/nodes/127.0.0.1$ netstat -ntlp | grep fisco-bcos
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 0.0.0.0:30300           0.0.0.0:*               LISTEN      10380/fisco-bcos
tcp        0      0 0.0.0.0:30301           0.0.0.0:*               LISTEN      10384/fisco-bcos
tcp        0      0 0.0.0.0:30302           0.0.0.0:*               LISTEN      10386/fisco-bcos
tcp        0      0 0.0.0.0:30303           0.0.0.0:*               LISTEN      10382/fisco-bcos
tcp        0      0 127.0.0.1:8545          0.0.0.0:*               LISTEN      10380/fisco-bcos
tcp        0      0 127.0.0.1:8546          0.0.0.0:*               LISTEN      10384/fisco-bcos
tcp        0      0 127.0.0.1:8547          0.0.0.0:*               LISTEN      10386/fisco-bcos
tcp        0      0 127.0.0.1:8548          0.0.0.0:*               LISTEN      10382/fisco-bcos
tcp        0      0 127.0.0.1:20200         0.0.0.0:*               LISTEN      10380/fisco-bcos
tcp        0      0 127.0.0.1:20201         0.0.0.0:*               LISTEN      10384/fisco-bcos
tcp        0      0 127.0.0.1:20202         0.0.0.0:*               LISTEN      10386/fisco-bcos
tcp        0      0 127.0.0.1:20203         0.0.0.0:*               LISTEN      10382/fisco-bcos
```

二、部署 **WeID**----運行脚本 **deploy.sh** 進行 **WeID** 合約的發布，完成 **Build-tools** 布署，通過 **tools** 快速體驗 **WeID** 的核心功能，如下：



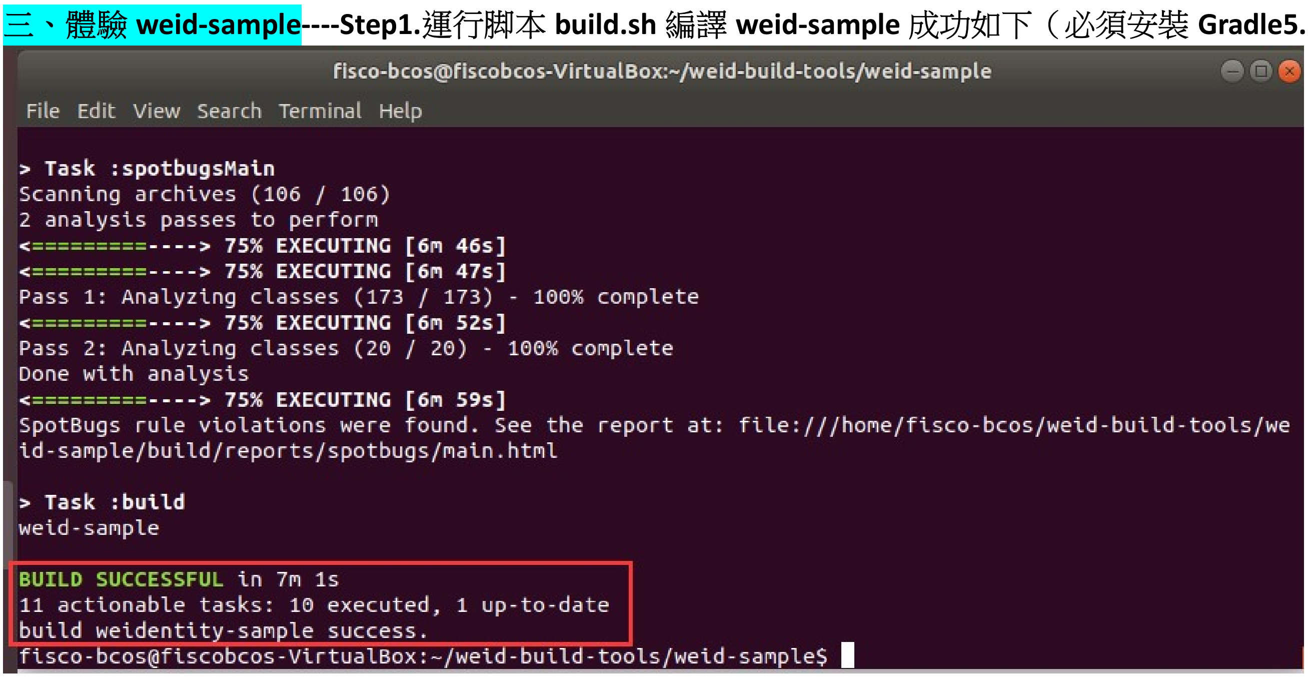
```
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools$ ./deploy.sh
Begin to check jdk...
jdk check finished.

begin to deploy contract, please wait.....
begin to modify sdk config...
contract is deployed with success.
===========================================.
weid contract address is 0x15651a58e28c7b711f1631f85ef4599ca4938773
cpt contract address is 0xea9a881b030c5fab84d817d04baf0803417da7e8
authority issuer contract address is 0x7101e713a22462a4ce062fa686ae799cc26ce2c0
evidence contract address is 0xc3942e43d217a0751bdac59afe0b1d55120c2518
specificIssuer contract address is 0xd09c9e20816404a753fc753fd448fd7d99b8ce55
===========================================.

begin to deploy system contract...
deploy system cpt done.
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools$
```



- 体验内容
  - **所有script均在 /weid-build-tools/tools/目录下**
  - 创建WeID
  - 注册Authority Issuer
  - 注册Specific Issuer
  - 机构发布CPT
  - 拉取CPT，并生成Presentation Policy模板

https://weidentity.readthedocs.io/zh_CN/latest/docs/weidentity-quick-tools.html

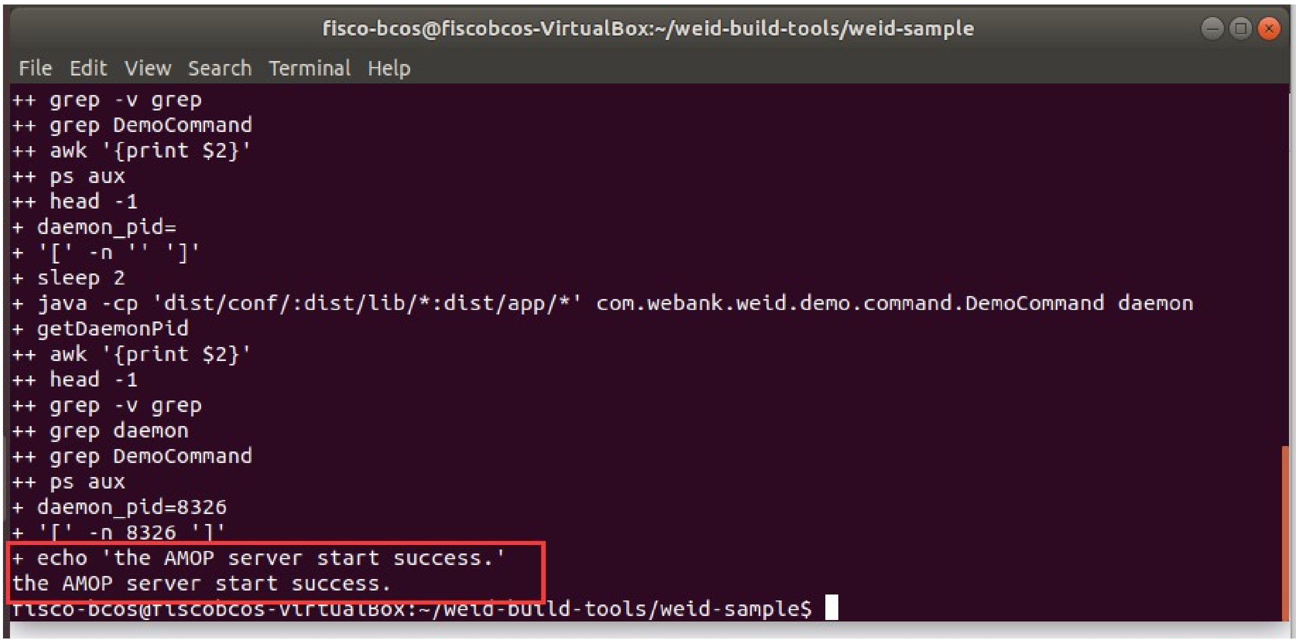
三、體驗 **weid-sample----Step1.**運行脚本 **build.sh** 編譯 **weid-sample** 成功如下（必須安裝 **Gradle5.2.1**）



fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample
File Edit View Search Terminal Help

```
> Task :spotbugsMain
Scanning archives (106 / 106)
2 analysis passes to perform
<=========----> 75% EXECUTING [6m 46s]
<=========----> 75% EXECUTING [6m 47s]
Pass 1: Analyzing classes (173 / 173) - 100% complete
<=========----> 75% EXECUTING [6m 52s]
Pass 2: Analyzing classes (20 / 20) - 100% complete
Done with analysis
<=========----> 75% EXECUTING [6m 59s]
SpotBugs rule violations were found. See the report at: file:///home/fisco-bcos/weid-build-tools/we
id-sample/build/reports/spotbugs/main.html

> Task :build
weid-sample

BUILD SUCCESSFUL in 7m 1s
11 actionable tasks: 10 executed, 1 up-to-date
build weidentity-sample success.
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample$
```

Step2.啓動 **AMOP** 服務：**#./command.sh daemon**


fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample
File Edit View Search Terminal Help

```
++ grep -v grep
++ grep DemoCommand
++ awk '{print $2}'
++ ps aux
++ head -1
+ daemon_pid=
+ '[' -n '' ']'
+ sleep 2
+ java -cp 'dist/conf/:dist/lib/*:dist/app/*' com.webank.weid.demo.command.DemoCommand daemon
+ getDaemonPid
++ awk '{print $2}'
++ head -1
++ grep -v grep
++ grep daemon
++ grep DemoCommand
++ ps aux
+ daemon_pid=8326
+ '[' -n 8326 ']'
+ echo 'the AMOP server start success.'
the AMOP server start success.
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample$
```

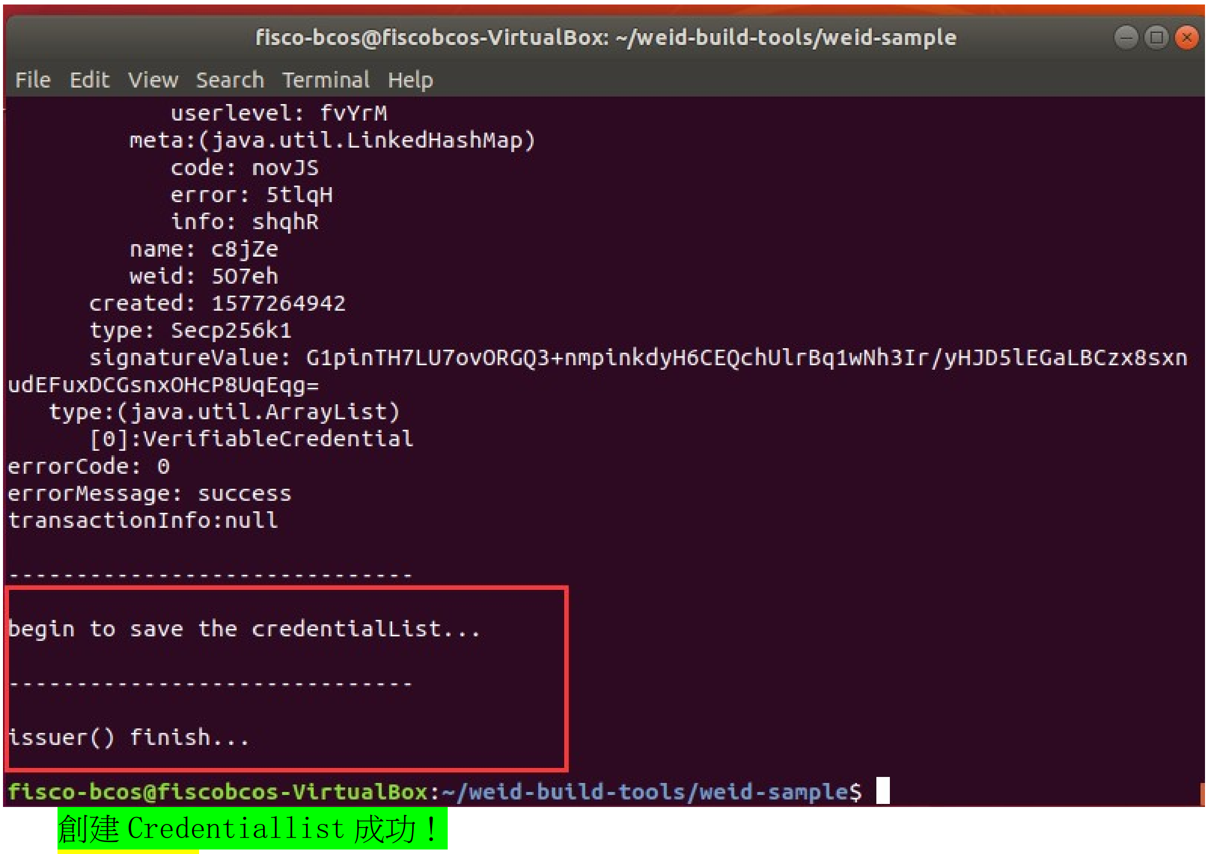
Step3. 修改 **User Agent** 配置：

#vi 下述三個文件的節點機構名爲 **organizationA**，節點 **IP** 用 **127.0.0.1:20200**
/weid-sample/src/main/resources/weidentity.properties
/weid-sample/build/resources/main/weidentity.properties
/weid-sample/dist/conf/weidentity.properties

#/weid-build-tools/dist/conf/weidentity.properties 之下的節點機構名爲 **organizationB**，節點 **IP** 用 **127.0.0.1:20201**

Step4.執行情況如下：

1、**Issuer** 視角如下：

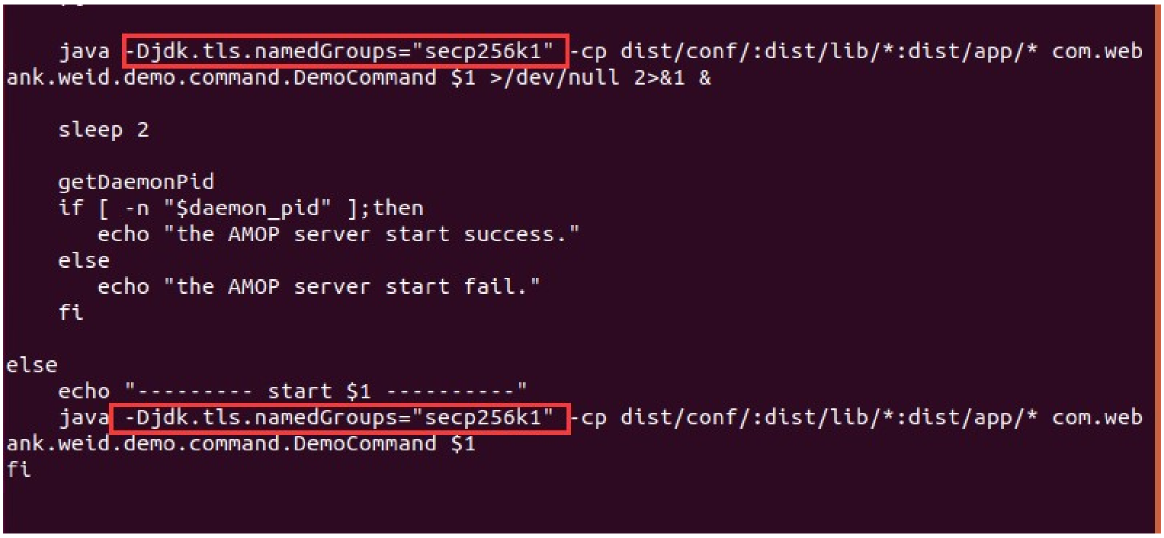

fisco-bcos@fiscobcos-VirtualBox: ~/weid-build-tools/weid-sample
File Edit View Search Terminal Help

```
            userlevel: fvYrM
         meta:(java.util.LinkedHashMap)
            code: novJS
            error: 5tlqH
            info: shqhR
         name: c8jZe
         weid: 507eh
      created: 1577264942
      type: Secp256k1
      signatureValue: G1pinTH7LU7ovORGQ3+nmpinkdyH6CEQchUlrBq1wNh3Ir/yHJD5lEGaLBCzx8sxn
udEFuxDCGsnxOHcP8UqEqg=
   type:(java.util.ArrayList)
      [0]:VerifiableCredential
errorCode: 0
errorMessage: success
transactionInfo:null

------------------------------

begin to save the credentialList...

------------------------------

issuer() finish...
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample$
```

創建 Credentiallist 成功！

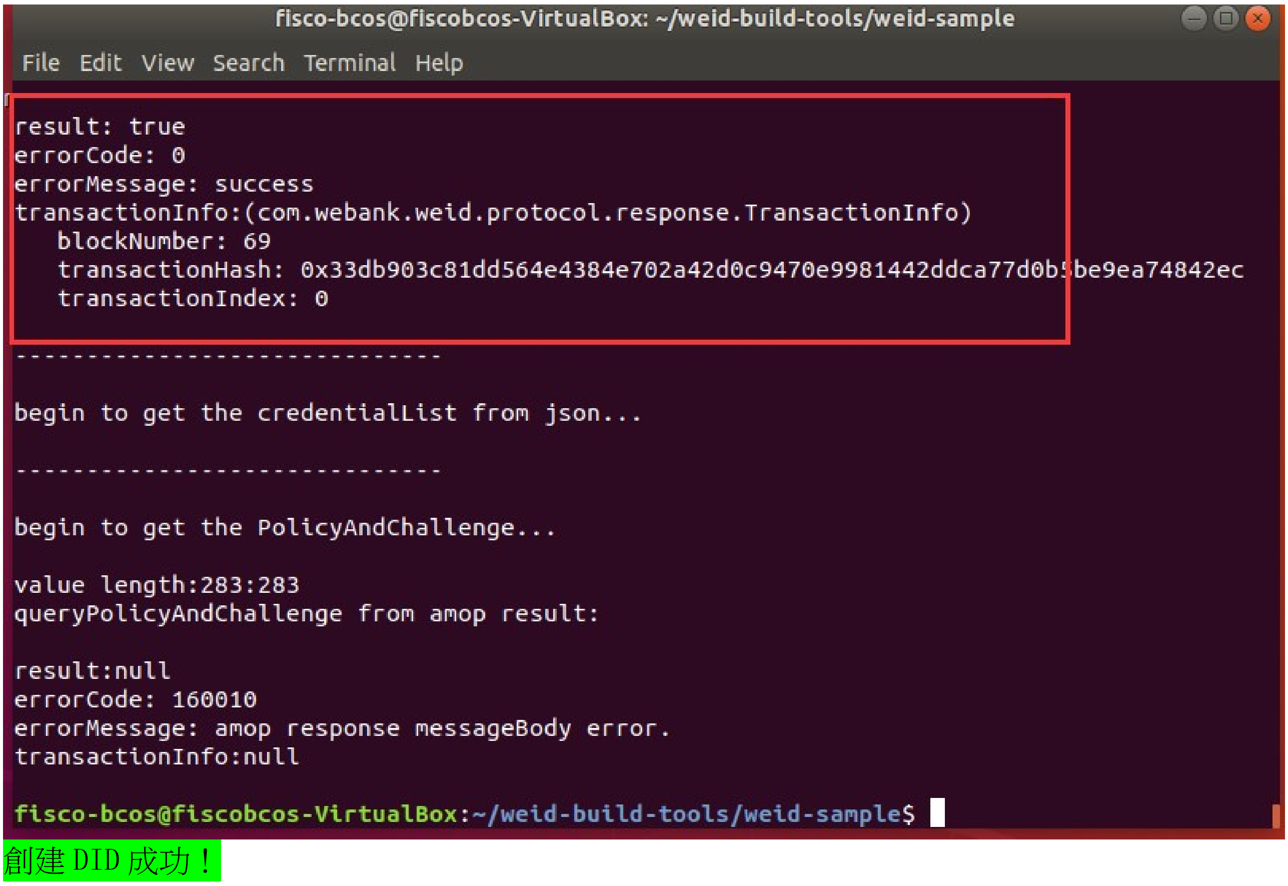
另外注意：需要修改/weid-sample/.commond.sh 找到如下內容，並添加-Djdk.tls.namedGroups="secp256k1"，

這是因為 JDK 的版本太高引起的兼容性問題。



```
    java -Djdk.tls.namedGroups="secp256k1" -cp dist/conf/:dist/lib/*:dist/app/* com.web
ank.weid.demo.command.DemoCommand $1 >/dev/null 2>&1 &

    sleep 2

    getDaemonPid
    if [ -n "$daemon_pid" ];then
       echo "the AMOP server start success."
    else
       echo "the AMOP server start fail."
    fi

else
    echo "--------- start $1 ----------"
    java -Djdk.tls.namedGroups="secp256k1" -cp dist/conf/:dist/lib/*:dist/app/* com.web
ank.weid.demo.command.DemoCommand $1
fi
```

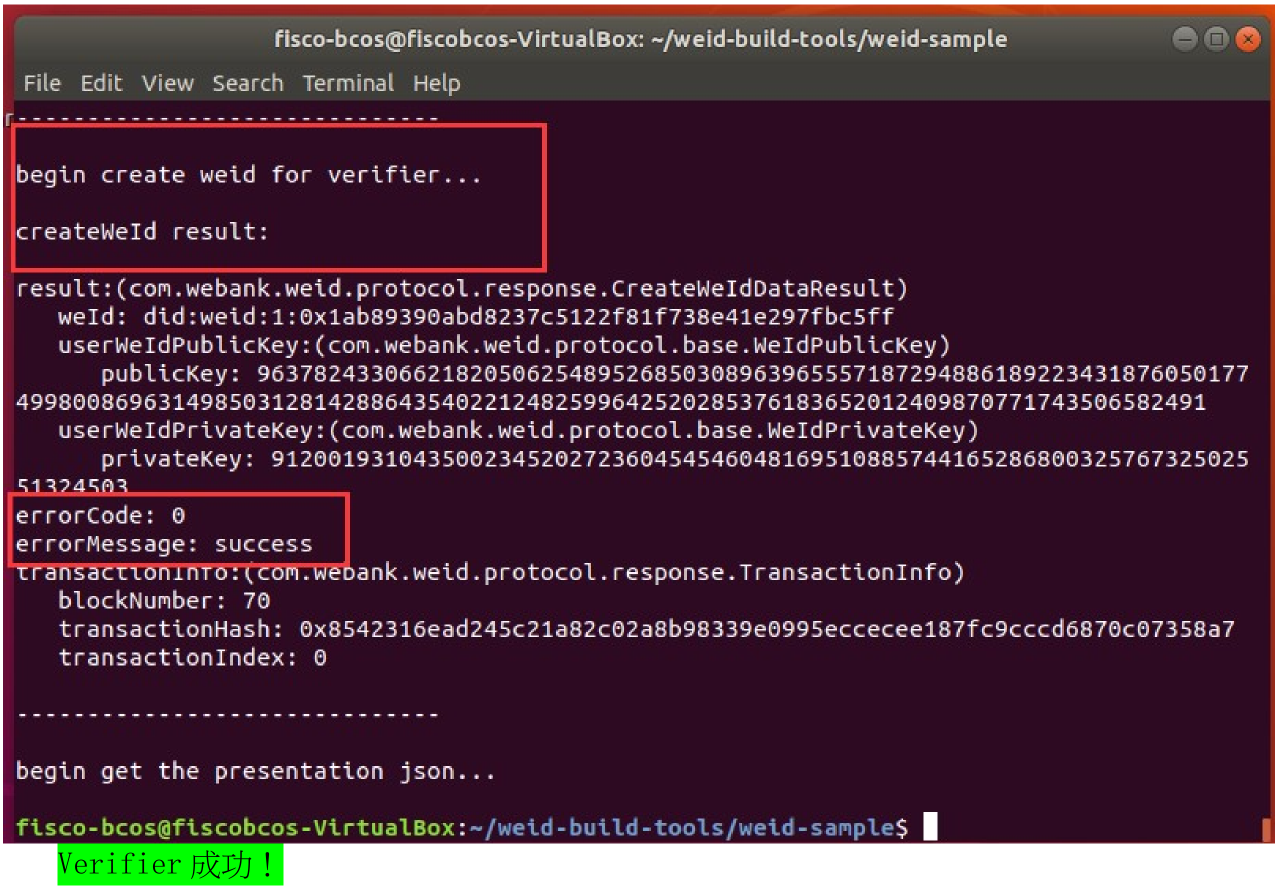
2、User Agent 視角如下：


fisco-bcos@fiscobcos-VirtualBox: ~/weid-build-tools/weid-sample
File Edit View Search Terminal Help

```
result: true
errorCode: 0
errorMessage: success
transactionInfo:(com.webank.weid.protocol.response.TransactionInfo)
   blockNumber: 69
   transactionHash: 0x33db903c81dd564e4384e702a42d0c9470e9981442ddca77d0b5be9ea74842ec
   transactionIndex: 0

------------------------------

begin to get the credentialList from json...

------------------------------

begin to get the PolicyAndChallenge...

value length:283:283
queryPolicyAndChallenge from amop result:

result:null
errorCode: 160010
errorMessage: amop response messageBody error.
transactionInfo:null

fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample$ 
```

創建 DID 成功！

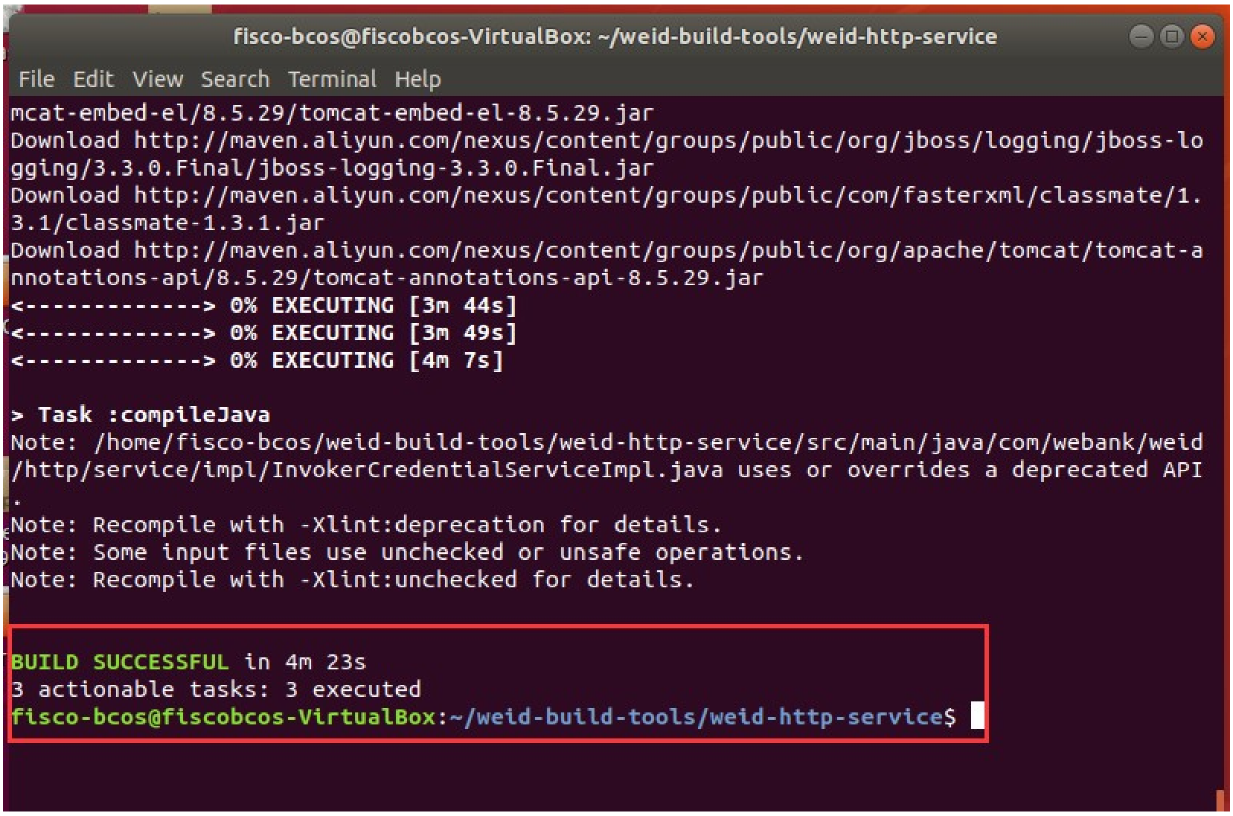
3、Verifier 視角如下：


fisco-bcos@fiscobcos-VirtualBox: ~/weid-build-tools/weid-sample
File Edit View Search Terminal Help

```
------------------------------

begin create weid for verifier...

createWeId result:

result:(com.webank.weid.protocol.response.CreateWeIdDataResult)
   weId: did:weid:1:0x1ab89390abd8237c5122f81f738e41e297fbc5ff
   userWeIdPublicKey:(com.webank.weid.protocol.base.WeIdPublicKey)
      publicKey: 9637824330662182050625489526850308963965557187294886189223431876050177499800869631498503128142886435402212482599642520285376183652012409870771743506582491
   userWeIdPrivateKey:(com.webank.weid.protocol.base.WeIdPrivateKey)
      privateKey: 91200193104350023452027236045454604816951088574416528680032576732502551324503
errorCode: 0
errorMessage: success
transactionInfo:(com.webank.weid.protocol.response.TransactionInfo)
   blockNumber: 70
   transactionHash: 0x8542316ead245c21a82c02a8b98339e0995eccecee187fc9cccd6870c07358a7
   transactionIndex: 0

------------------------------

begin get the presentation json...

fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-sample$
```

Verifier 成功！

四、體驗 HTTP-Service-----Step1.運行#gradle build -x test 編譯 weid-http-service 成功如下（必須安裝 Gradle5.2.1）


fisco-bcos@fiscobcos-VirtualBox: ~/weid-build-tools/weid-http-service
File Edit View Search Terminal Help

```
mcat-embed-el/8.5.29/tomcat-embed-el-8.5.29.jar
Download http://maven.aliyun.com/nexus/content/groups/public/org/jboss/logging/jboss-logging/3.3.0.Final/jboss-logging-3.3.0.Final.jar
Download http://maven.aliyun.com/nexus/content/groups/public/com/fasterxml/classmate/1.3.1/classmate-1.3.1.jar
Download http://maven.aliyun.com/nexus/content/groups/public/org/apache/tomcat/tomcat-annotations-api/8.5.29/tomcat-annotations-api-8.5.29.jar
<-------------> 0% EXECUTING [3m 44s]
<-------------> 0% EXECUTING [3m 49s]
<-------------> 0% EXECUTING [4m 7s]

> Task :compileJava
Note: /home/fisco-bcos/weid-build-tools/weid-http-service/src/main/java/com/webank/weid/http/service/impl/InvokerCredentialServiceImpl.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

BUILD SUCCESSFUL in 4m 23s
3 actionable tasks: 3 executed
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service$
```

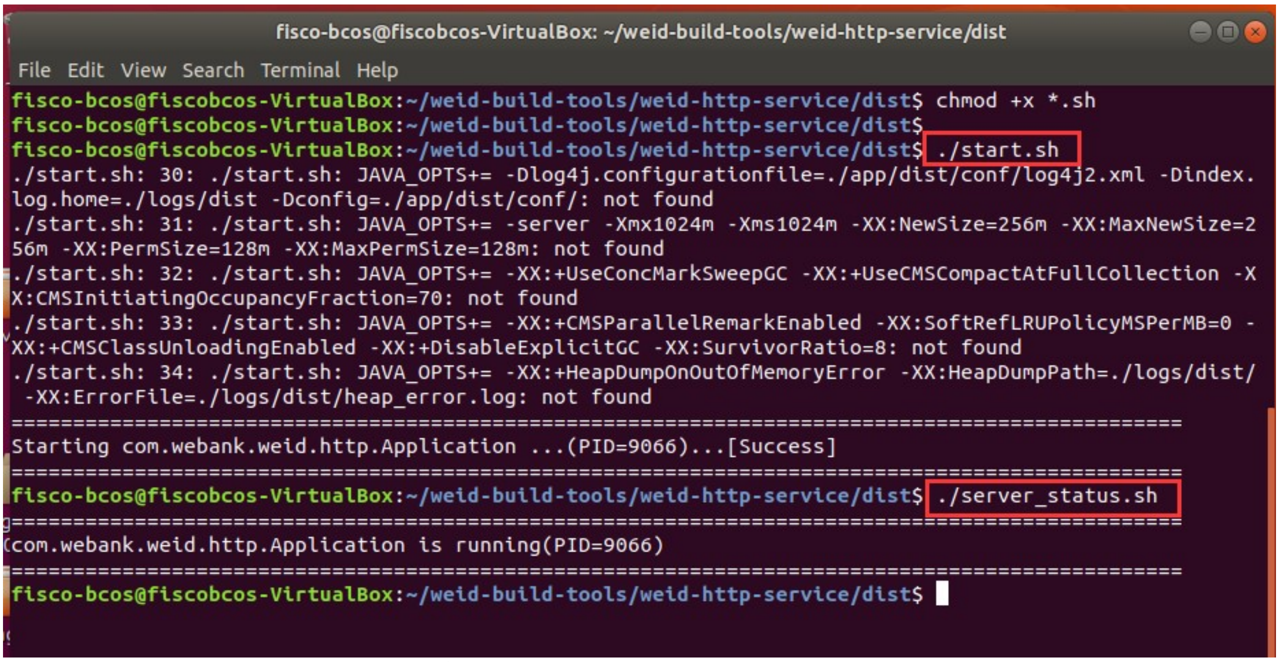
Step2. 修改配置文件：

1.确认 WeIdentity 合约已部署;

2.修改合约地址（部署工具或者源码部署）及区块链节点信息修改；

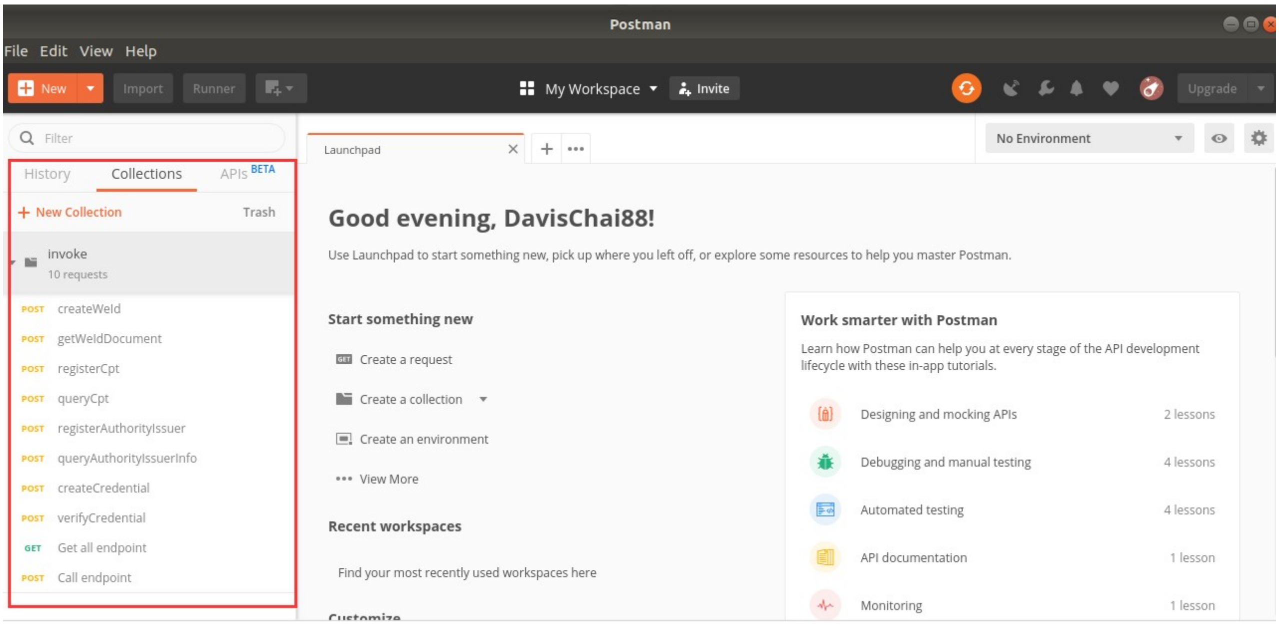
參考：https://weidentity.readthedocs.io/zh_CN/latest/docs/weidentity-rest-deploy.html

Step3.執行情況如下：啟動 **HTTP-Service** 及觀察應用狀態


fisco-bcos@fiscobcos-VirtualBox: ~/weid-build-tools/weid-http-service/dist
File Edit View Search Terminal Help

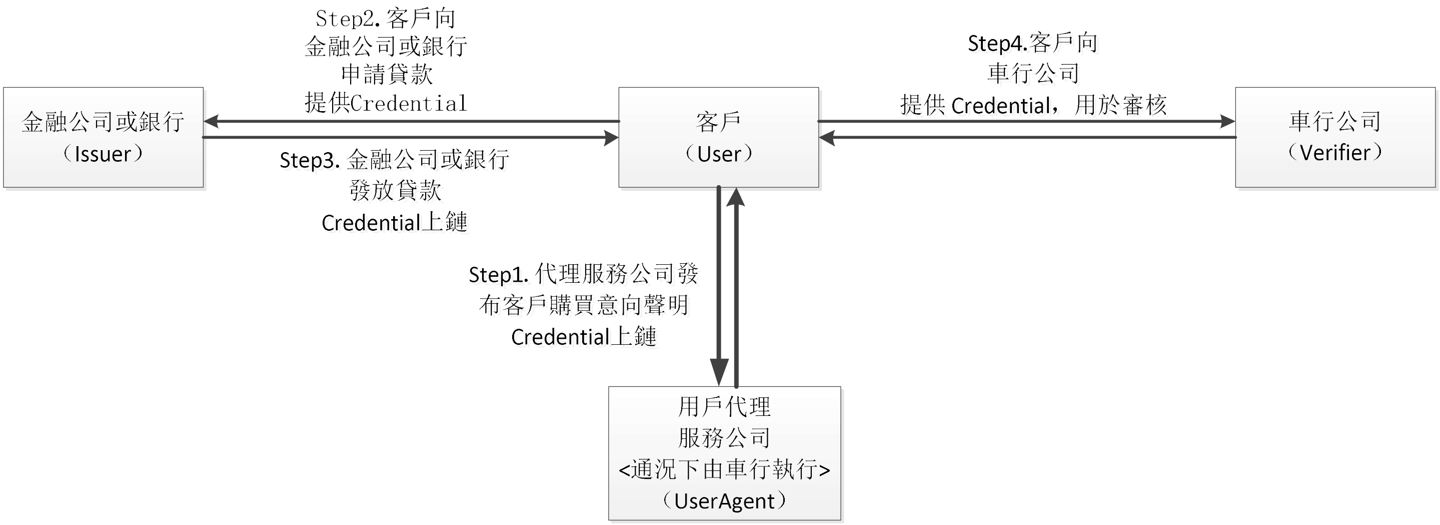
```
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service/dist$ chmod +x *.sh
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service/dist$
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service/dist$ ./start.sh
./start.sh: 30: ./start.sh: JAVA_OPTS+= -Dlog4j.configurationfile=./app/dist/conf/log4j2.xml -Dindex.
log.home=./logs/dist -Dconfig=./app/dist/conf/: not found
./start.sh: 31: ./start.sh: JAVA_OPTS+= -server -Xmx1024m -Xms1024m -XX:NewSize=256m -XX:MaxNewSize=2
56m -XX:PermSize=128m -XX:MaxPermSize=128m: not found
./start.sh: 32: ./start.sh: JAVA_OPTS+= -XX:+UseConcMarkSweepGC -XX:+UseCMSCompactAtFullCollection -X
X:CMSInitiatingOccupancyFraction=70: not found
./start.sh: 33: ./start.sh: JAVA_OPTS+= -XX:+CMSParallelRemarkEnabled -XX:SoftRefLRUPolicyMSPerMB=0 -
XX:+CMSClassUnloadingEnabled -XX:+DisableExplicitGC -XX:SurvivorRatio=8: not found
./start.sh: 34: ./start.sh: JAVA_OPTS+= -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=./logs/dist/
 -XX:ErrorFile=./logs/dist/heap_error.log: not found
===============================================================================================
Starting com.webank.weid.http.Application ...(PID=9066)...[Success]
===============================================================================================
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service/dist$ ./server_status.sh
===============================================================================================
com.webank.weid.http.Application is running(PID=9066)
===============================================================================================
fisco-bcos@fiscobcos-VirtualBox:~/weid-build-tools/weid-http-service/dist$ 
```

Step4. 安裝 **Postman** 及 **Invoke** 測試


Postman
File Edit View Help
New
Import
Runner
My Workspace
Invite
Upgrade
Filter
History
Collections
APIs BETA
New Collection
Trash
invoke
10 requests
POST createWeId
POST getWeIdDocument
POST registerCpt
POST queryCpt
POST registerAuthorityIssuer
POST queryAuthorityIssuerInfo
POST createCredential
POST verifyCredential
GET Get all endpoint
POST Call endpoint
Launchpad
No Environment
Good evening, DavisChai88!
Use Launchpad to start something new, pick up where you left off, or explore some resources to help you master Postman.
Start something new
Create a request
Create a collection
Create an environment
View More
Recent workspaces
Find your most recently used workspaces here
Customize
Work smarter with Postman
Learn how Postman can help you at every stage of the API development lifecycle with these in-app tutorials.
Designing and mocking APIs 2 lessons
Debugging and manual testing 4 lessons
Automated testing 4 lessons
API documentation 1 lesson
Monitoring 1 lesson

然後按 Invoke 裏面不同的方法做測試即可。

五、（第 1 組）選用 **HTTP-Service+Postman** 方式構建實現大作業------一個‘金融貸款購車案例’，具體過程如下：



-------客戶進行 WeIdentity DID 注册

~~1.客戶進入用戶代理公司的客戶信息（KYC）注册頁面，提交個人信息~~

客戶進入代理公司無需提供個人信息，保證客戶個人私隱（代理公司只需認錢不需認人）

~~2.用戶代理公司<亦可以是車行>審核申請，成功後授予該客戶 Weidentity DID~~

~~——同上~~

------- STEP1. 用戶代理為客戶註冊購車意向合同聲明

1. 為註冊代理公司申請 DID 並註冊成為 Issuer
2. 由代理公司往鏈上發布購買意向聲明（客戶無需向代理服務公司提供任何個人資料/或僅提供基本資料，以保障個人隱私）

------STEP2. 客戶向金融公司或銀行申請購車貸款憑證（Credential）

1. 客戶向金融公司或銀行提交購車貸款憑證（Credential）申請
2. 金融公司或銀行審核申請，成功後授予該客戶購車貸款憑證 Credential 上鏈（**STEP3**）

------STEP4 客戶提交貸款憑證（Credential），用于驗證

1.客戶向車行提交自己的貸款憑證 Credential，車行驗證真實性後交付汽車

Prepare01．註冊業務實體

- **技術實現**

- 通过 WeIdService 的 createWeId()生成一个 WeIdentity DID 并注册到链上；
- 通过 WeIdService 的 getWeIdDocument()獲取對應 DID 的相關 public key；
- 通過 WeIdService 的 registerAuthorityIssuer()註冊有關機構成為 AuthorityIssuer

- **為代理服務公司註冊 DID：**

代理公司	functionName : CreateWeId, getWeIdDocument (public key)
車行 A	did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362 11155932033748773519505914719313752212213348045073475523740549015135636699414065490680829863281888604325934923295416875762229721153897766840531447814654 94
車行 B	did:weid:1:0x8a76b3c83dc508d704b77d2b3da81174e92a0823

	9404446263392011469416841937203143821909499703691587888030022312834603529982710728388654397923988744478691721007011776686676154642118015491133058571184495

- **為貸款機構註冊 DID：**

貸款機構	functionName : CreateWeId, getWeIdDocument (public key)
銀行 A	did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd 6801105129696240739616695658401489095517671600782623003065594152758122299844447215935468351306337427417793616085979604586715724896530005609815370730374034
借貸公司 B	did:weid:1:0xde318bf343e3a07dd250adec6699519e31f1c60c 5507875379932321264906146818377171990368867956603657290445514810082965511112080100889435176561146746437114519602104925651990900526273229345733915546730068

- **為代理服務公司註冊 及 貸款機構 註冊成為授权机构(AuthorityIssuer)**

  - 通過 WeIdService 的 registerAuthorityIssuer()實現，並使用 queryAuthorityIssuer 進行驗證

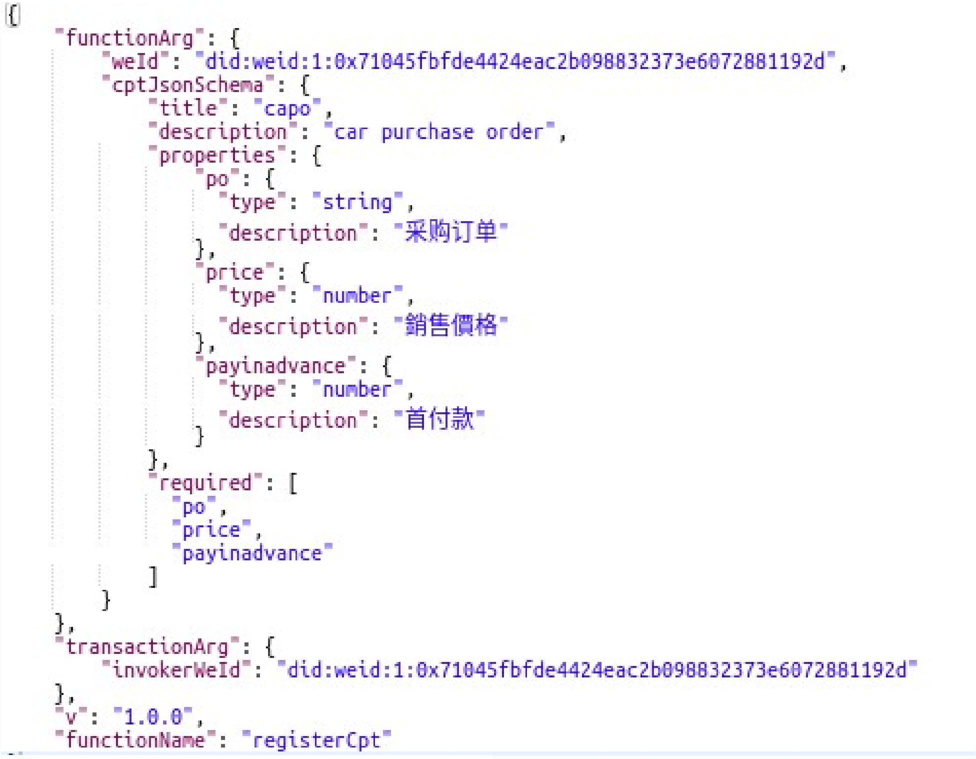
- **其他測試用途 DID**

  did:weid:1:0x71045fbfde4424eac2b098832373e6072881192d

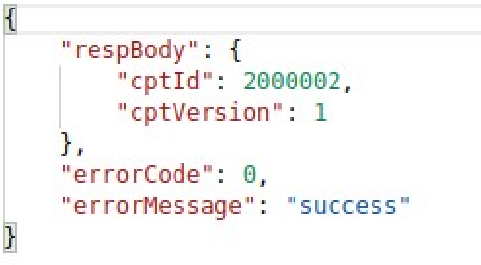
Prepare02. 註冊 CPT （CAPO）

- **技術實現**

- 通过 WeIdService 的 registercpt () 註冊 cpt ；



```
{
    "functionArg": {
        "weId": "did:weid:1:0x71045fbfde4424eac2b098832373e6072881192d",
        "cptJsonSchema": {
            "title": "capo",
            "description": "car purchase order",
            "properties": {
                "po": {
                    "type": "string",
                    "description": "采购订单"
                },
                "price": {
                    "type": "number",
                    "description": "銷售價格"
                },
                "payinadvance": {
                    "type": "number",
                    "description": "首付款"
                }
            },
            "required": [
                "po",
                "price",
                "payinadvance"
            ]
        }
    },
    "transactionArg": {
        "invokerWeId": "did:weid:1:0x71045fbfde4424eac2b098832373e6072881192d"
    },
    "v": "1.0.0",
    "functionName": "registerCpt"
```

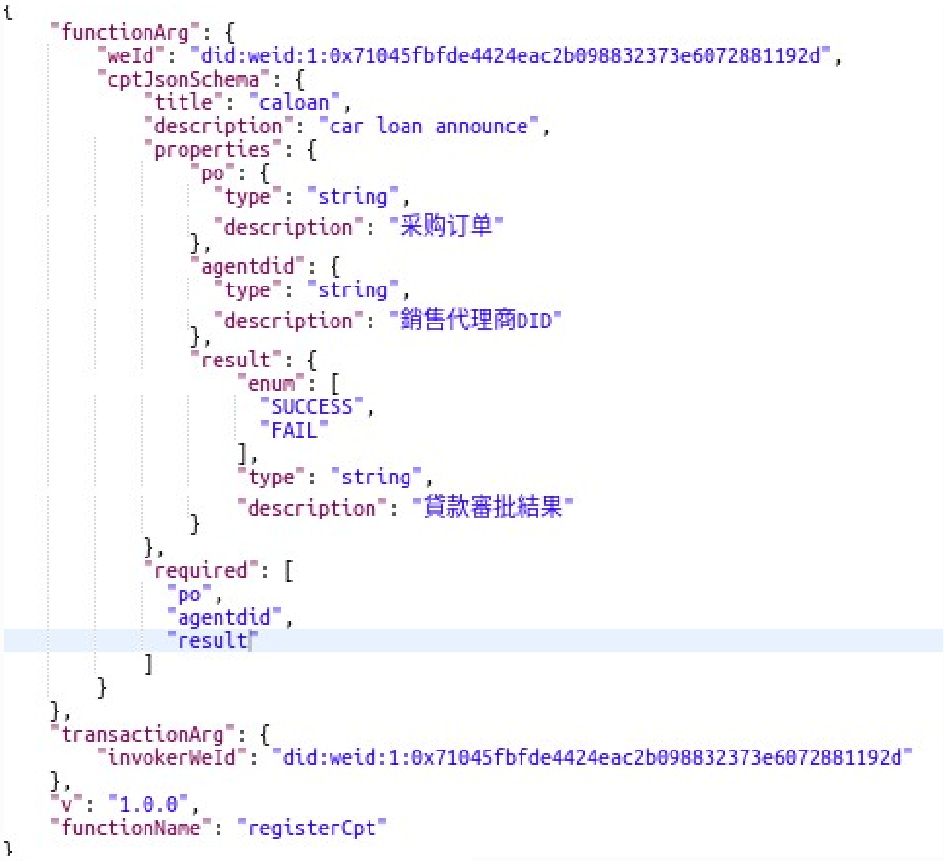


```
{
    "respBody": {
        "cptId": 2000002,
        "cptVersion": 1
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

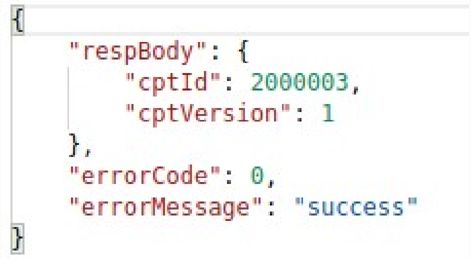
Prepare03. 註冊 CPT （CALOAN）

- 技術實現

- 通过 WeIdService 的 registercpt () 註冊 cpt；



```
{
    "functionArg": {
        "weId": "did:weid:1:0x71045fbfde4424eac2b098832373e6072881192d",
        "cptJsonSchema": {
            "title": "caloan",
            "description": "car loan announce",
            "properties": {
                "po": {
                  "type": "string",
                  "description": "采购订单"
                },
                "agentdid": {
                  "type": "string",
                  "description": "銷售代理商DID"
                },
                "result": {
                    "enum": [
                      "SUCCESS",
                      "FAIL"
                    ],
                    "type": "string",
                    "description": "貸款審批結果"
                }
            },
            "required": [
              "po",
              "agentdid",
              "result"
            ]
        }
    },
    "transactionArg": {
        "invokerWeId": "did:weid:1:0x71045fbfde4424eac2b098832373e6072881192d"
    },
    "v": "1.0.0",
    "functionName": "registerCpt"
}
```

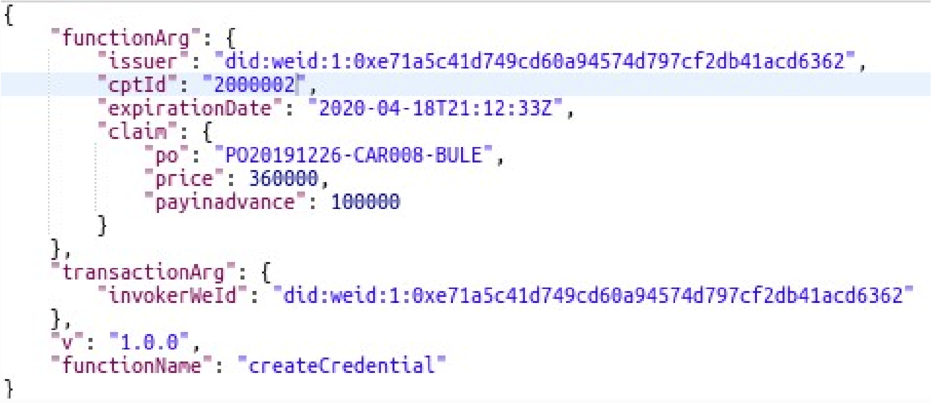


```
{
    "respBody": {
        "cptId": 2000003,
        "cptVersion": 1
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

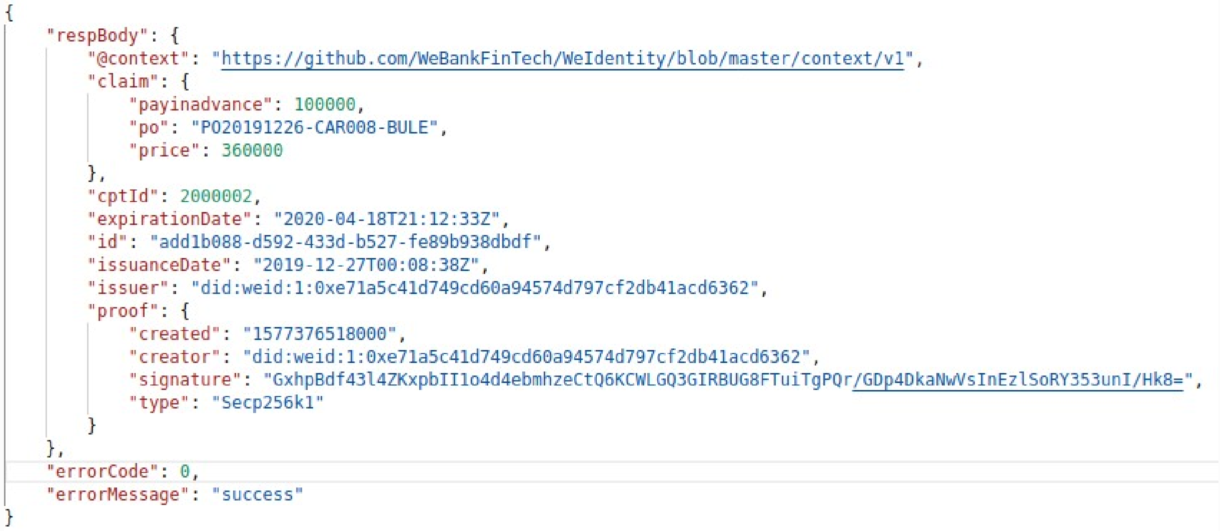
Car Purchasing（Step1）．購車貸款申請（車行 A 代理發佈邏輯）

- 技術實現

- 客戶與汽車銷售公司洽商銷售價格，達成購買相關產品意向；
- 針對需要進行貸款購買的客戶，將其採購信息上鏈，以便有關貸款機構進行審批核實 (通過 WeIdService 的 createCredential ()實現)



```
{
    "functionArg": {
        "issuer": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
        "cptId": "2000002",
        "expirationDate": "2020-04-18T21:12:33Z",
        "claim": {
            "po": "PO20191226-CAR008-BULE",
            "price": 360000,
            "payinadvance": 100000
        }
    },
    "transactionArg": {
        "invokerWeId": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362"
    },
    "v": "1.0.0",
    "functionName": "createCredential"
}
```

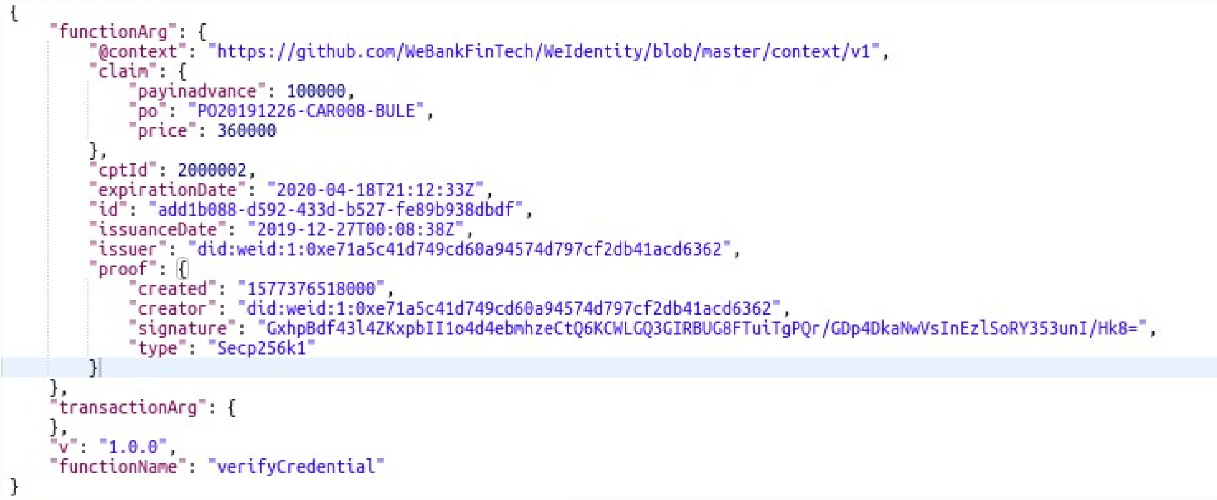


```
{
    "respBody": {
        "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
        "claim": {
            "payinadvance": 100000,
            "po": "PO20191226-CAR008-BULE",
            "price": 360000
        },
        "cptId": 2000002,
        "expirationDate": "2020-04-18T21:12:33Z",
        "id": "add1b088-d592-433d-b527-fe89b938dbdf",
        "issuanceDate": "2019-12-27T00:08:38Z",
        "issuer": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
        "proof": {
            "created": "1577376518000",
            "creator": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
            "signature": "GxhpBdf43l4ZKxpbII1o4d4ebmhzeCtQ6KCWLGQ3GIRBUG8FTuiTgPQr/GDp4DkaNwVsInEzlSoRY353unI/Hk8=",
            "type": "Secp256k1"
        }
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

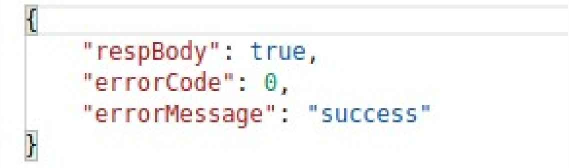
Loan Request（Step2）. 貸款申請

- 技術實現

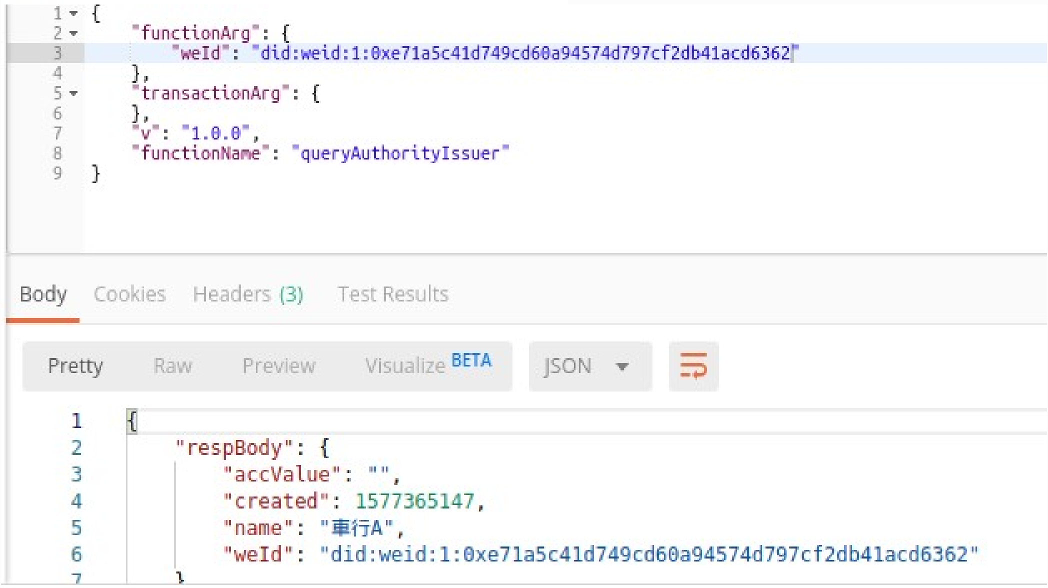
- 客戶向指定銀行或貸款機構發出申請貸款請求，並出示有關 CAPO credential;
- 銀行或貸款機構驗證 CAPO credential 的真實性及有效性
  (通過 WeIdService 的 verifyCredential ()實現)
- 銀行或貸款機構驗證 issuer 信息，確保其銷售業務符合該公司的貸款政策
  (通過 WeIdService 的 queryAuthorityIssuer ()實現)



```
{
    "functionArg": {
        "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
        "claim": {
            "payinadvance": 100000,
            "po": "PO20191226-CAR008-BULE",
            "price": 360000
        },
        "cptId": 2000002,
        "expirationDate": "2020-04-18T21:12:33Z",
        "id": "add1b088-d592-433d-b527-fe89b938dbdf",
        "issuanceDate": "2019-12-27T00:08:38Z",
        "issuer": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
        "proof": {
            "created": "1577376518000",
            "creator": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
            "signature": "GxhpBdf43l4ZKxpbII1o4d4ebmhzeCtQ6KCWLGQ3GIRBUG8FTuiTgPQr/GDp4DkaNwVsInEzlSoRY353unI/Hk8=",
            "type": "Secp256k1"
        }
    },
    "transactionArg": {
    },
    "v": "1.0.0",
    "functionName": "verifyCredential"
}
```



```
{
    "respBody": true,
    "errorCode": 0,
    "errorMessage": "success"
}
```



```
{
    "functionArg": {
        "weId": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362"
    },
    "transactionArg": {
    },
    "v": "1.0.0",
    "functionName": "queryAuthorityIssuer"
}
```

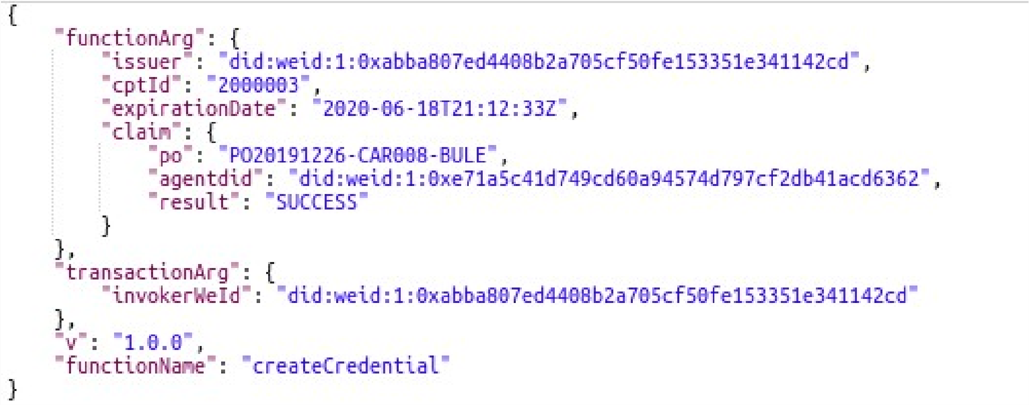
```
{
    "respBody": {
        "accValue": "",
        "created": 1577365147,
        "name": "車行A",
        "weId": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362"
    }
```

Loan Request（Step3）. 發放貸款 （銀行 A）

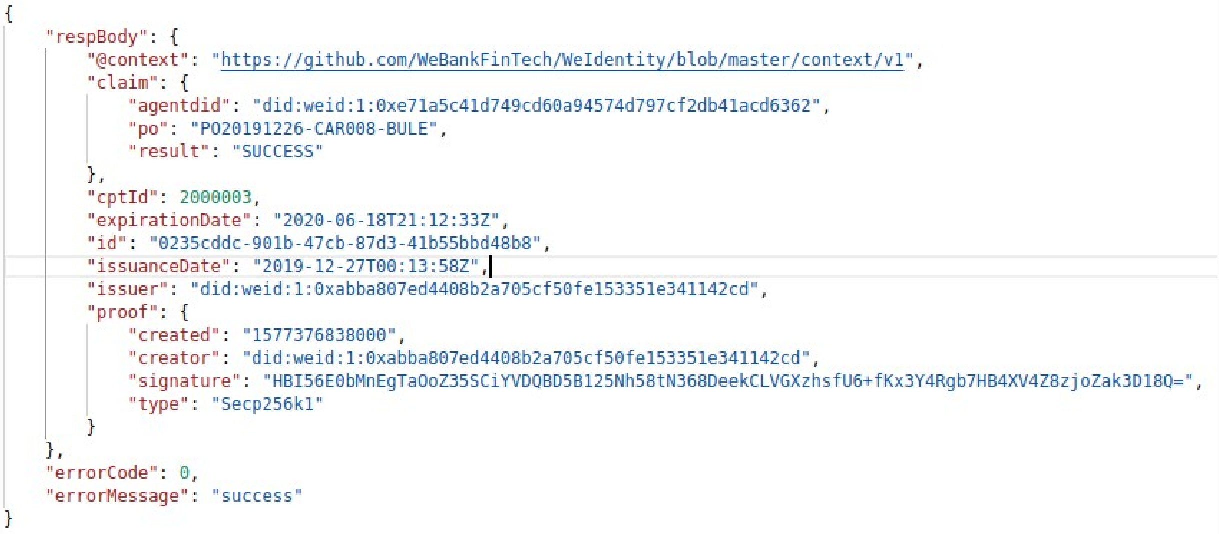
- 技術實現

- 銀行或貸款機構審批客戶貸款請求，並及時將其審批結果上鏈;

(通過 WeIdService 的 createCredential ()實現)



```
{
    "functionArg": {
        "issuer": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd",
        "cptId": "2000003",
        "expirationDate": "2020-06-18T21:12:33Z",
        "claim": {
            "po": "PO20191226-CAR008-BULE",
            "agentdid": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
            "result": "SUCCESS"
        }
    },
    "transactionArg": {
        "invokerWeId": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd"
    },
    "v": "1.0.0",
    "functionName": "createCredential"
}
```



```
{
    "respBody": {
        "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
        "claim": {
            "agentdid": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
            "po": "PO20191226-CAR008-BULE",
            "result": "SUCCESS"
        },
        "cptId": 2000003,
        "expirationDate": "2020-06-18T21:12:33Z",
        "id": "0235cddc-901b-47cb-87d3-41b55bbd48b8",
        "issuanceDate": "2019-12-27T00:13:58Z",
        "issuer": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd",
        "proof": {
            "created": "1577376838000",
            "creator": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd",
            "signature": "HBI56E0bMnEgTaOoZ35SCiYVDQBD5B125Nh58tN368DeekCLVGXzhsfU6+fKx3Y4Rgb7HB4XV4Z8zjoZak3D18Q=",
            "type": "Secp256k1"
        }
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

GotCar（Step4）. 提車 （客戶）

- 技術實現

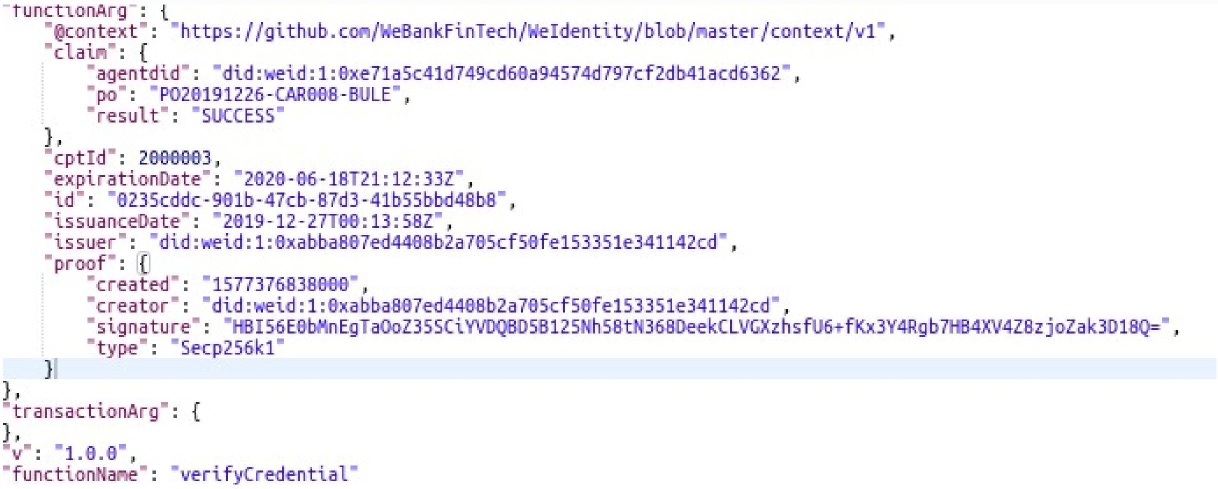
- 客戶向車行展示貸款結果

- 車行驗證 CALOAN credential 的真實性及有效性

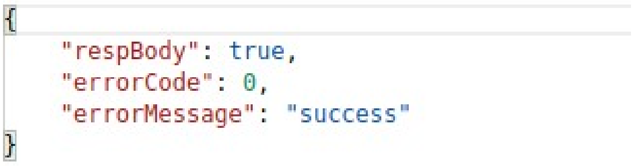
(通過 WeIdService 的 verifyCredential ()實現)

- 車行驗證 CALOAN credential 發自於銀行機構

- 車行根據 CALOAN 所描述的 PO 及 AgentDID，確保有關的貸款發放對象及購買合約正確後，給予客戶提車權利，同時保留憑證，及時向有關銀行收取有關款項

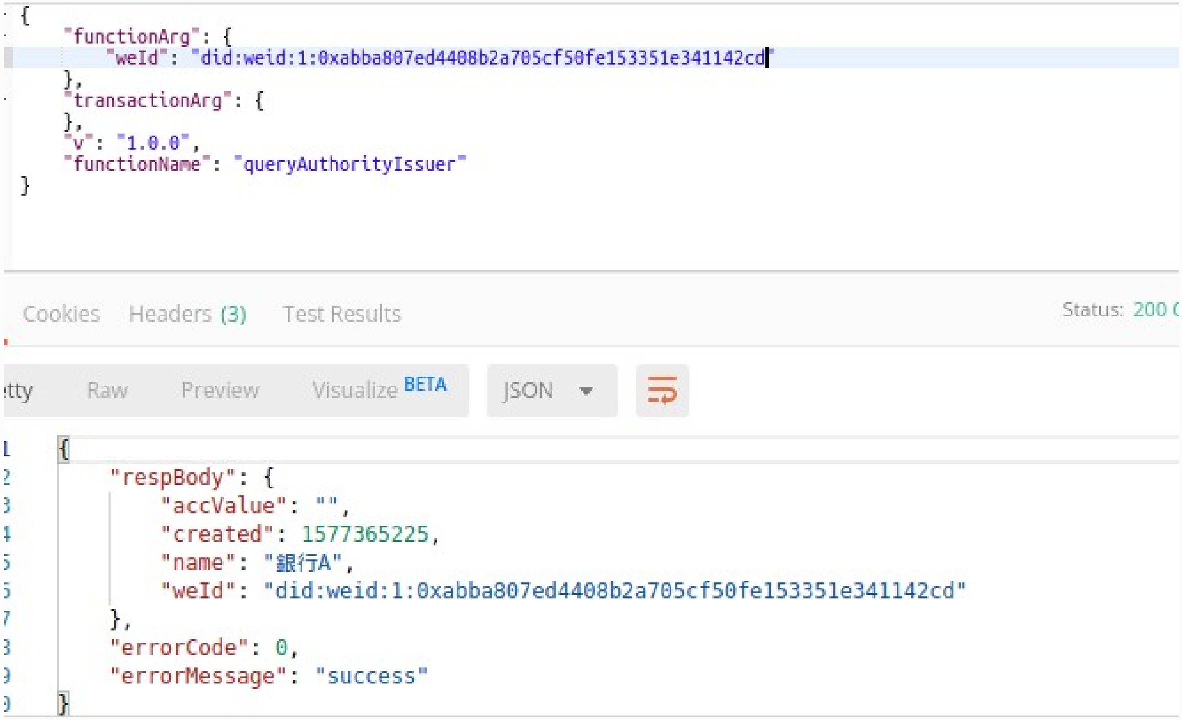


```
"functionArg": {
    "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
    "claim": {
        "agentdid": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362",
        "po": "PO20191226-CAR008-BULE",
        "result": "SUCCESS"
    },
    "cptId": 2000003,
    "expirationDate": "2020-06-18T21:12:33Z",
    "id": "0235cddc-901b-47cb-87d3-41b55bbd48b8",
    "issuanceDate": "2019-12-27T00:13:58Z",
    "issuer": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd",
    "proof": {
        "created": "1577376838000",
        "creator": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd",
        "signature": "HBI56E0bMnEgTaOoZ35SCiYVDQBD5B125Nh58tN368DeekCLVGXzhsfU6+fKx3Y4Rgb7HB4XV4Z8zjoZak3D18Q=",
        "type": "Secp256k1"
    }
},
"transactionArg": {
},
"v": "1.0.0",
"functionName": "verifyCredential"
```



```
{
    "respBody": true,
    "errorCode": 0,
    "errorMessage": "success"
}
```

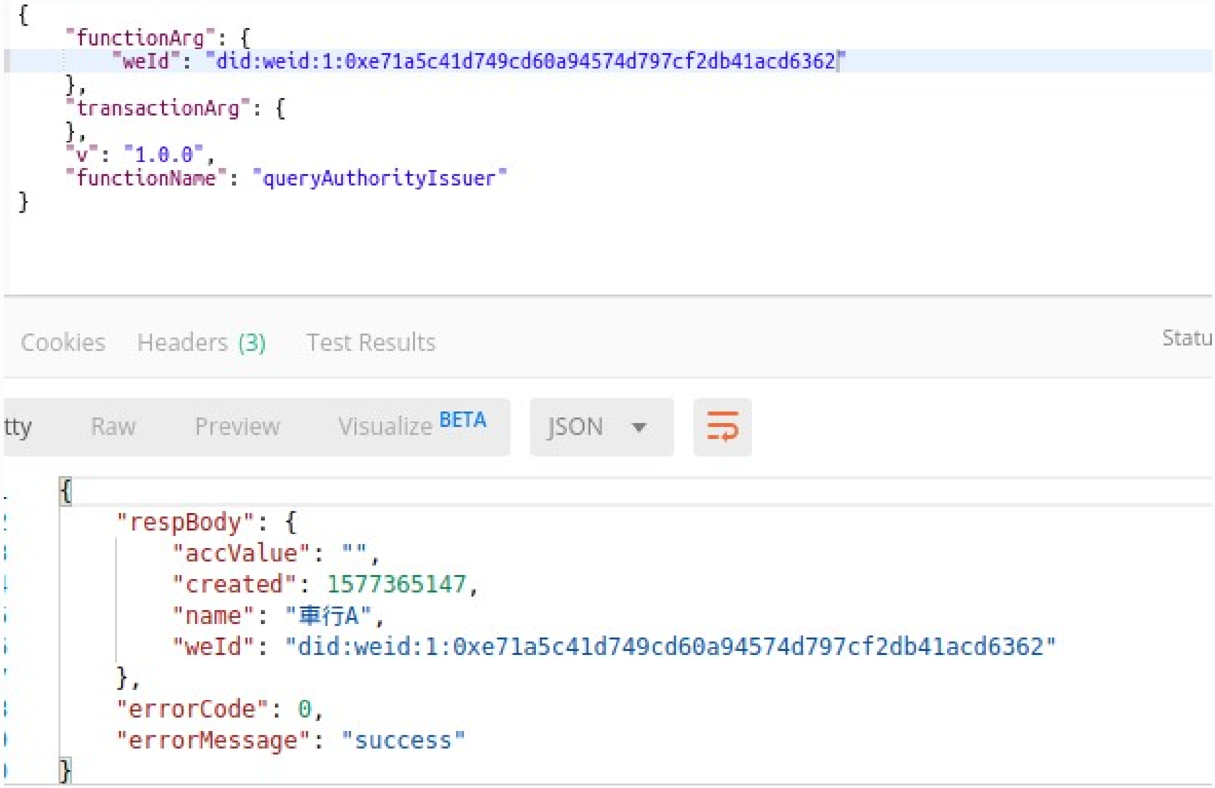
檢查 Issuer 是否銀行



```
{
    "functionArg": {
        "weId": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd"
    },
    "transactionArg": {
    },
    "v": "1.0.0",
    "functionName": "queryAuthorityIssuer"
}
```

```
{
    "respBody": {
        "accValue": "",
        "created": 1577365225,
        "name": "銀行A",
        "weId": "did:weid:1:0xabba807ed4408b2a705cf50fe153351e341142cd"
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

檢查 agentdid 是否自己



```
{
    "functionArg": {
        "weId": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362"
    },
    "transactionArg": {
    },
    "v": "1.0.0",
    "functionName": "queryAuthorityIssuer"
}
```

```
{
    "respBody": {
        "accValue": "",
        "created": 1577365147,
        "name": "車行A",
        "weId": "did:weid:1:0xe71a5c41d749cd60a94574d797cf2db41acd6362"
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```

核對 PO 及提車人身份後發放車輛