

中山大学数据科学与计算机学院本科生实验报告

(2019年秋季学期)

【大作业（三）】最终成品

课程名称：区块链原理与技术

任课教师：郑子彬

年级	17级	专业（方向）	软件工程
学号	17343125	姓名	伍祥坤
电话	13411610063	Email	1285149329@qq.com
开始日期	2019-10-18	完成日期	2019-12-23

一、项目背景

https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/introduction.html

基于区块链的供应链金融平台

必要功能

基于已有的开源区块链系统 FISCO-BCOS

（<https://github.com/FISCO-BCOS/FISCO-BCOS>），以联盟链为主，开发基于区块链或区块链智能合约的供应链金融平台，实现供应链应收账款资产的溯源、流转。



场景介绍：

传统供应链金融：

某车企（宝马）因为其造车技术特别牛，消费者口碑好，所以其在同行业中占据绝对优势地位。因此，在金融机构（银行）对该车企的信用评级将很高，认为他有很大的风险承担的能力。在某次交易中，该车企从轮胎公司购买了一批轮胎，但由于资金暂时短缺向轮胎公司

签订了 1000 万的应收账款单据，承诺 1 年后归还轮胎公司 1000 万。这个过程可以拉上金融机构例如银行来对这笔交易作见证，确认这笔交易的真实性。在接下里的几个月里，轮胎公司因为资金短缺需要融资，这个时候它可以凭借跟某车企签订的应收账款单据向金融结构借款，金融机构认可该车企（核心企业）的还款能力，因此愿意借款给轮胎公司。但是，这样的信任关系并不会往下游传递。在某个交易中，轮胎公司从轮毂公司购买了一批轮毂，但由于租金暂时短缺向轮胎公司签订了 500 万的应收账款单据，承诺 1 年后归还轮胎公司 500 万。当轮毂公司想利用这个应收账款单据向金融机构借款融资的时候，金融机构因为不认可轮胎公司的还款能力，需要对轮胎公司进行详细的信用分析以评估其还款能力同时验证应收账款单据的真实性，才能决定是否借款给轮毂公司。这个过程将增加很多经济成本，而这个问题主要是由于该车企的信用无法在整个供应链中传递以及交易信息不透明化所导致的。

区块链+供应链金融：

将供应链上的每一笔交易和应收账款单据上链，同时引入第三方可信机构来确认这些信息的交易，例如银行，物流公司等，确保交易和单据的真实性。同时，支持应收账款的转让，融资，清算等，让核心企业的信用可以传递到供应链的下游企业，减小中小企业的融资难度。

实现功能：

功能一：实现采购商品—签发应收账款 交易上链。例如车企从轮胎公司购买一批轮胎并签订应收账款单据。

功能二：实现应收账款的转让上链，轮胎公司从轮毂公司购买一笔轮毂，便将于车企的应收账款单据部分转让给轮毂公司。轮毂公司可以利用这个新的单据去融资或者要求车企到期时归还钱款。

功能三：利用应收账款向银行融资上链，供应链上所有可以利用应收账款单据向银行申请融资。

功能四：应收账款支付结算上链，应收账款单据到期时核心企业向下游企业支付相应的欠款。

二、 方案设计

前端：用java实现一个UI界面

后端：https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/sdk/java_sdk.html

链段：合约等的部署都是照着下面的网址进行的https://fisco-bcos-documentation.readthedocs.io/zh_CN/latest/docs/tutorial/sdk_application.html

部署合约

file : Web3Contoller

```
public void deployAsset() {  
    try {  
        Test_v1 asset = Test_v1.deploy(web3j, credentials, new  
StaticGasProvider(gasPrice, gasLimit)).send();  
        System.out.println(" deploy success, contract address is " +  
asset.getContractAddress());  
        recordAddress(asset.getContractAddress());  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

调用部署合约的函数

file : UIViewContoller.

```
void addDeploy() {  
    JPanel jpanel = new JPanel();  
    jpanel.setLayout(new FlowLayout());  
    jpanel.setOpaque(false);  
  
    JButton jbutton = new JButton("部署合约");  
    jbutton.setOpaque(false);  
  
    jbutton.addActionListener(new ActionListener() {  
        public void actionPerformed(ActionEvent e) {  
            System.out.println("Deploying.....");  
            client.deployAsset();  
        }  
    })  
}
```

```
});
```

```
jpanel.add(jbutton);
```

```
this.add(jpanel);
```

```
}
```

资金转移

file : Web3Contoller

```
public void tranBillsFromToNumber(String from,String to,BigInteger number) {  
  
    try {  
  
        String contractAddress = loadAddress();  
  
        Test_v1 asset = Test_v1.load(contractAddress, web3j, credentials, new  
StaticGasProvider(gasPrice, gasLimit));  
  
        asset.tran_bills(from, to, number).send();  
  
    }  
  
    catch(Exception e) {  
  
        e.printStackTrace();    }  
  
}
```

查询资金

file : Web3Contoller

```
public String getUserBillMoney(String user) {  
  
    try {  
  
        String string = loadAddress();  
  
        Test_v1 asset = Test_v1.load(string, web3j, credentials, new  
StaticGasProvider(gasPrice, gasLimit));  
  
        Tuple2<BigInteger,BigInteger> result = asset.get_bills_money(user).send();
```

```

        System.out.println("money: ");

        if(result.getValue1().compareTo(new BigInteger("0"))==0){

            return new String("-"+result.getValue2());

        }

        System.out.println("outmoney: "+result.getValue2());

        return result.getValue2().toString();

    } catch (Exception e) {

        e.printStackTrace();    }

    return "0";

}

```

File:Asset.sol

功能一：function creat_bill(string lender, string from, int amount) public
returns (int)

功能二：function trans_bill(string lender, string from, int amount) public returns
(int)

功能三：function askwith_bill(string lender, string from, int amount) public
returns (int)

功能四：function remove_bill(string lender, string borrower) public returns (int)

三、 功能测试






```
UiViewController [Java Application] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (Dec 13, 2019, 8:19:17 PM)
Deploying.....
  deploy Asset success, contract address is 0x3b5fe9039b08e23d475f551ecbd1ac5719317c15
Transfer 100 from Alice to Bob
result: 0x0000000000000000000000000000000000000000000000000000000000000001
get the money result:
user: Alice account: -100
get the money result:
outmoney: 100
user: Bob account: 100
Pay back successfully!
get the money result: |
user: Alice account: -0
get the money result:
user: Bob account: -0
```

四、 界面展示



五、 心得体会

这次实验让我学到了很多区块链应用实现的知识，大大地提高了我的动手能力。实验过程中，不但学习到了fisco-bcos链的配置，还学到了基于fisco-bcos的应用开发的技巧。