

ZJU-blockchain-course-2023

如何运行

补充如何完整运行你的应用。

1. 在本地启动ganache应用。
2. 在 `./contracts` 中安装需要的依赖，运行如下的命令：

```
npm install
```

3. 在 `./contracts/contracts/BorrowYourCar.sol` 中的mint函数中设置每个人初始拥有的汽车id。
4. 在 `./contracts` 中编译合约，运行如下的命令：

```
npx hardhat compile
```

5. 在 `./hardhat.config.ts` 中修改account为自己部署的链中的地址
6. 在 `./frontend/src/utis/address.json` 中修改地址为自己合约中的地址
7. 在 `./frontend` 中安装需要的依赖，运行如下的命令：

```
npm install
```

8. 在 `./frontend/src/App.tsx` 中的数组中设置每个人初始拥有的汽车id和拥有者，以便正确展示
9. 在 `./frontend` 中启动前端程序，运行如下的命令：

```
npm run start
```

功能实现分析

1. 显示用户拥有的车辆和当前可租用的车辆

[连接钱包](#)

车友车行 当前用户: 0x16eddb83b0b733bae796e04992ed0671cd977d83
当前用户拥有的车辆编号: 2
可被借用的汽车编号: :1,2,4

连接钱包

当前用户和租有的车辆是常量，所以在前端中用常量数组直接显示，可被借用的汽车列表是通过后端遍历获得

```
function getAvailableCars() public view returns(uint256[] memory){
    uint256 cnt=0;
    for(uint256 i=0;i<5;i++){
        {
            if(userOf(i)==address (0))
            {
                cnt++;
            }
        }
    }
    uint256[] memory indices = new uint256[](cnt);
    uint256 index = 0;
    for(uint256 i=0;i<5;i++){
        {
            if(userOf(i)==address (0))
            {
                indices[index] = i;
                index ++;
            }
        }
    }
    return indices;
}
```

调用了userof函数，如果有租用者且尚未过期，则添加到可用汽车列表。

2. 查询一辆汽车的主人及其借用者

主人通过前端静态数组查询，借用者则通过后端userof函数调用加前端useeffect渲染实时更新。



车辆编号: 0

拥有者: 0xDE21c4f130ab2036D552Bc63C22a8012fc2C55C1

借用者: 0x16eddB83B0B733BaE796E04992ED0671cd977d83

借用期限: 1710013770234

3. 借车

通过输入车的编号和借用天数，可以借用汽车一段时间。通过模仿ERC4907实现

```
function borrowCar (uint256 tokenId, address user,uint64 expires) public{
    Car storage carInfo = cars[tokenId];
    carInfo.borrower=user;
    carInfo.borrowUntil=block.timestamp+expires;
    emit CarBorrowed(tokenId,user,block.timestamp,expires);
}
```

项目运行截图

合约成功部署

```
PS C:\Users\86152\Desktop\qukuailian\ZJU-blockchain-course-2023-main> cd contracts
PS C:\Users\86152\Desktop\qukuailian\ZJU-blockchain-course-2023-main\contracts> npx hardhat run ./scripts/deploy.ts --network ganache
Compiled 1 Solidity file successfully (evm target: paris).
BorrowYourCar deployed to 0x4624C96aC37699C80a3b4CF3Efd35Ae21bE51B31
```

初始默认五辆车，均可用

当前用户: 0x16eddb83b0b733bae796e04992ed0671cd977d83
当前用户拥有的车辆编号: 2
可被借用的汽车编号: :0,1,2,3,4

输入车的编号和借车事件后，即可借车

当前用户: 0x16eddb83b0b733bae796e04992ed0671cd977d83
当前用户拥有的车辆编号: 2
可被借用的汽车编号: :0,1,2,3,4



车辆编号: 0
拥有者: 0xDE21c4f130ab2036D552Bc63C22a8012fc2C55C1
借用者: 0x00
借用期限: 无借用者

随后观察到汽车借用成功，借用者和可用车辆均发生改变

当前用户: 0x16eddb83b0b733bae796e04992ed0671cd977d83

当前用户拥有的车辆编号: 2

可被借用的汽车编号: :1,2,3,4

请输入车的编号

请输入借车时间 (天)

借用



车辆编号: 0

拥有者: 0xDE21c4f130ab2036D552Bc63C22a8012fc2C55C1

借用者: 0x16eddB83B0B733BaE796E04992ED0671cd977d83

借用期限: 1701505741720

参考内容

- 课程的参考Demo见: [DEMOS](#)。
- ERC-4907 [参考实现](#)