|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编辑时间 | 2019-12-19 11:35:03 |  |  |  |
| 版本 | v0.01 |  |  |  |
| 编辑者 | Joker |  |  |  |

## 运行环境

Go 1.10 以上版本、ActiveMQ 5.0.0以上版本（注意是ActiveMQ不是RabbitMQ）；

机器最低配置： 2核CPU，4G内存，32/64位系统均可；

mysql 5.0 以上版本；

## 项目说明

管理员密码 10086 admin2020

数据库配置位于 service/conf/config.go

MQ配置位于 service/common/mq\_config.go

jhagent项目：

* 为代理商户管理系统，包含前后端逻辑
* 账户管理、商户管理、订单管理、财务管理
* conf：app.conf：runmode=dev表示开发模式，=pro表示运营模式，二者不能同时存在；httpport是端口号，任意设置，不冲突即可
* controllers：控制器
* logs：存放日志文件
* routers：路由文件
* static：css/js/图片文件
* sys：系统常量配置、session配置
* utils：日志配置、短信对接云片
* views：html文件

jhboss项目

* 包含管理后台的前后端所有逻辑
* 通道管理、商户管理、代理管理、支付订单管理、代付管理、自助服务、数据管理等
* 项目结构同上

jhgateway项目

* 为支付网关，主要是为下游的商户提供API，包括以下功能
* 1：支付请求逻辑处理，返回支付链接或者重定向到支付收银台。
* 2：支付订单状态查询接口，代付状态结果的查询接口。
* 3：上游支付订单的回调处理，上游代付回调的处理。
* 项目结构同上

jhmerchant项目

* 商户管理系统，包含前后端逻辑
* 账户管理、订单管理、财务管理
* 项目结构同上

jhmicro项目

* 包含三个微服务：
* 1：订单查询（向上游查询），其包括2个小应用，支付订单查询和代付订单查询。
* 2：订单回调（向下游商户），其包括2个小应用，支付订单回调和代付订单回调。
* 3：结算处理，每个一段时间将待结算的金额结算到可用金额里面，时间间隔由运营人员或者开发人员制定。

service项目

* 包含一些公共的工具文件，以及对接上游通道
* 1：model是操作数据库的文件
* 2：utils是一些基本的工具函数
* 3：common包含公用的数据和方法
* 4：controller处理用户加减款的主要逻辑
* 5：message\_queue消息队列，订单查询，订单回调，代付查询，代付回调
* 6：service\_init包含该系统的所有需要初始化的工作
* 7：supplier处理上游的通道接口

shop项目

* 测试网关和通道的可用性

## 安装使用

* 1. 创建数据库：juhe\_pay，导入sql文件：mysql.sql
* 2. 以jhagent项目为例，在win环境下
* 3. 进入项目的jhagent目录，进入此目录下的命令行模式
* 4. 输入go build，会在jhagent目录下生成一个jhagent.exe文件，点击即可运行，如下图：



* 5. 在地址栏输入<http://localhost:12308/>即可运行，若出现404，重新进行第四步

## 在windows下编译成Linux下可执行的二进制文件并且执行

* 1. 以jhagent项目为例，在win环境下
* 2. 进入项目的jhagent目录，进入此目录下的命令行模式
* 3. 执行下面的命令：set GOARCH=amd64；set GOOS=linux
* 4. 输入go build，会在jhagent目录下生成一个没有后缀的jhagent二进制文件
* 5. 将该文件放入linux系统某个文件夹下
* 6. 赋予文件权限：chmod 777 jhagent；执行：./jhagent
* 7. 二进制文件不需要go的任何依赖，可以直接运行
* 8. 在地址栏输入<http://localhost:12308/>即可运行，若出现404，重新进行第四步

