

美妆叠叠乐游戏软件

使用手册

目录

| | |
|-------------|----|
| 目录 | 1 |
| 文件修订记录 | 2 |
| 1. 总体功能描述 | 1 |
| 2. 运行环境 | 2 |
| 3. 软件编译环境 | 3 |
| 4. 软件安装说明 | 5 |
| 5. 游戏启动 | 6 |
| 6. 游戏玩法 | 7 |
| 7. 游戏结束 | 9 |
| 8. 排行榜功能 | 10 |
| 9. 数据库软件 | 11 |
| 10. 数据库脚本文件 | 12 |
| 11. 规范文件 | 13 |

文件修订记录

| 版本号 | 生成日期 | 作者 | 修订内容 |
|------|------------|-----|------|
| V1.0 | 2026-01-23 | 王永成 | 初始版本 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

1. 总体功能描述

系统采用三层结构，采用主流的 C#语言进行开发。

玩家需要将杂乱堆叠的各种化妆品，

通过点击，归类放入上方指定的“化妆包”中。

每个化妆包都有特定的槽位需求，装满后即可消除。

2. 运行环境

硬件要求

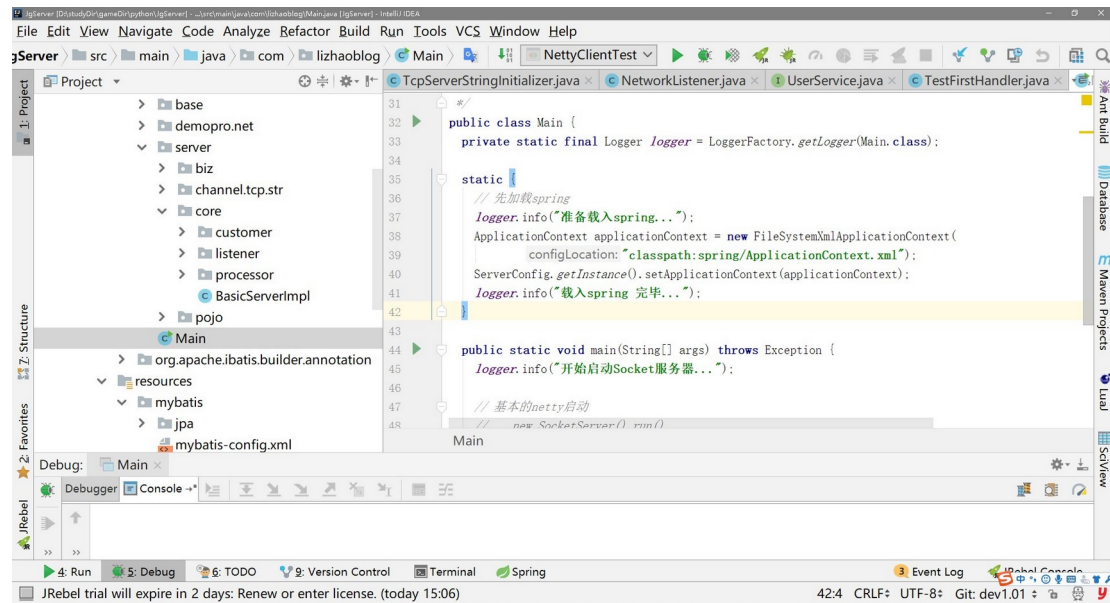
| 类 别 | 基 本 要 求 |
|------|--------------------------------|
| 服务器端 | CPU 2G 内存 2G 以上；硬盘剩余空间不低于 50G； |
| 客户端 | 手机 1G 内存及以上；硬盘空间 40M 及以上 |

软件要求：

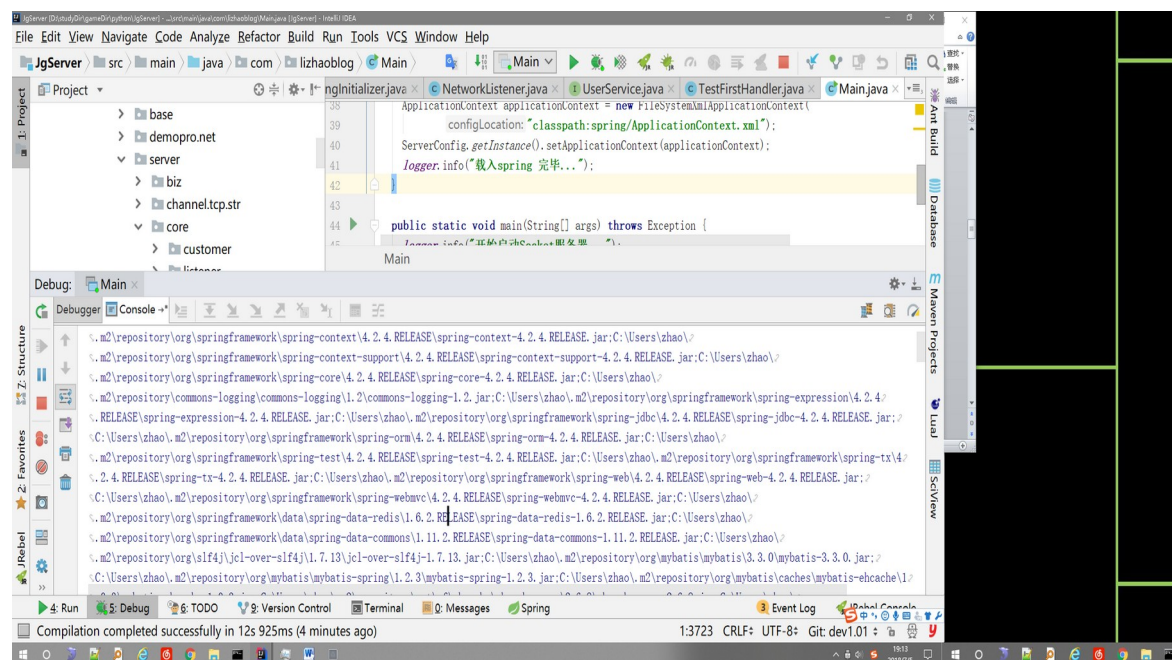
| 类别 | 名 称 | 基本环境 |
|------|-------|--------------------------------|
| 服务器端 | 操作系统 | 支持 Windows2008 32 位/64 位中文企业版； |
| | 数据库软件 | 支持 MySQL5.6 |
| 客户端 | 操作系统 | Android |
| | 其它软件 | 微信 |

3. 软件编译环境

本软件使用 Idea 进行开发，需要使用相同软件进行开发编译。



编译成功画面



4. 软件安装说明

本软件将发布至微信小游戏，可以在微信中搜索相关名称，然后点击使用。

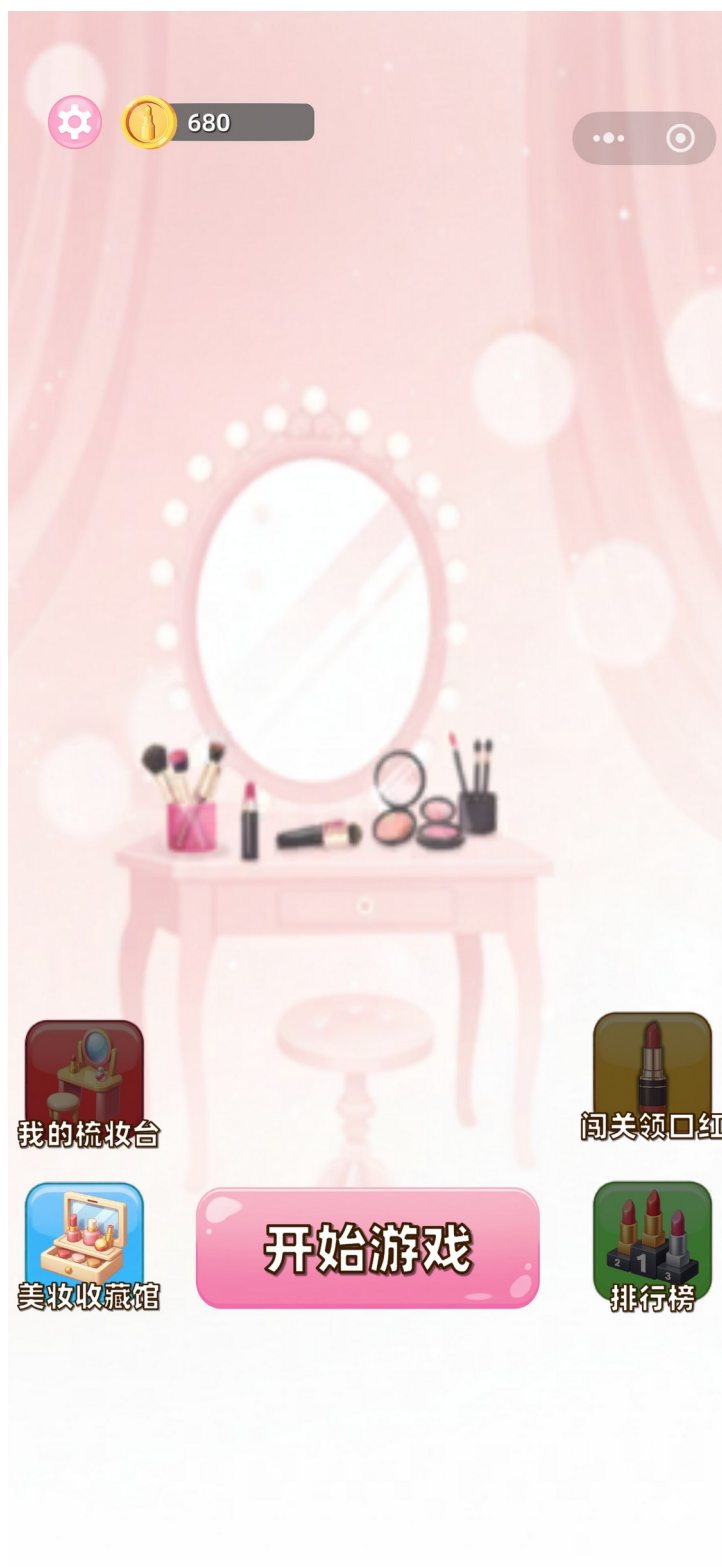
第一次打开软件将会进入加载界面，效果如图。



5. 游戏启动

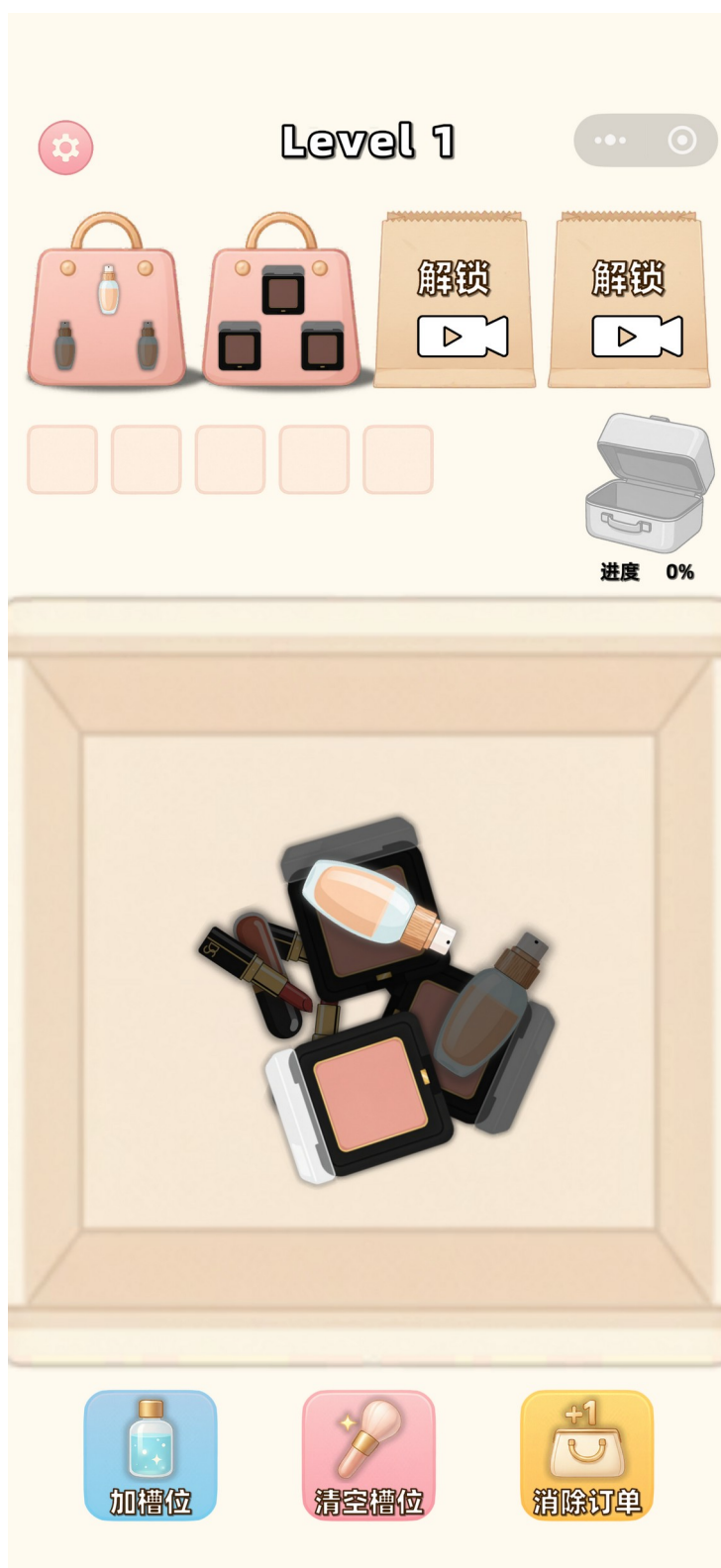
待游戏安装加载完成，可以在主页选择进入游戏，进行游戏。

效果如图：



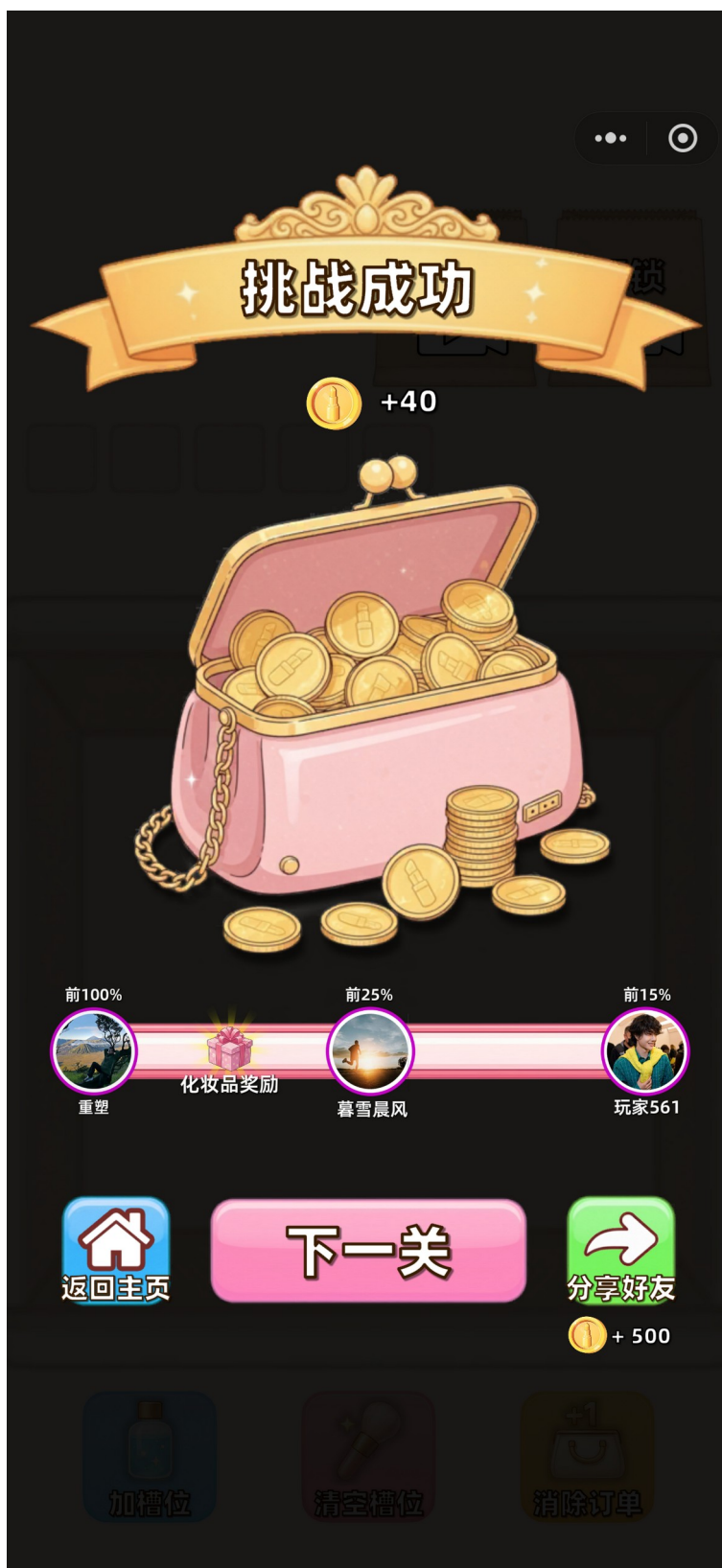
6. 游戏玩法

玩家需要将杂乱堆叠的各种化妆品 ,通过点击 ,归类放入上方指定的“化妆包”中。每个化妆包都有特定的槽位需求 ,装满后即可消除。



7. 游戏结束

游戏结束，玩家可以选择回到主界面，或者分享游戏



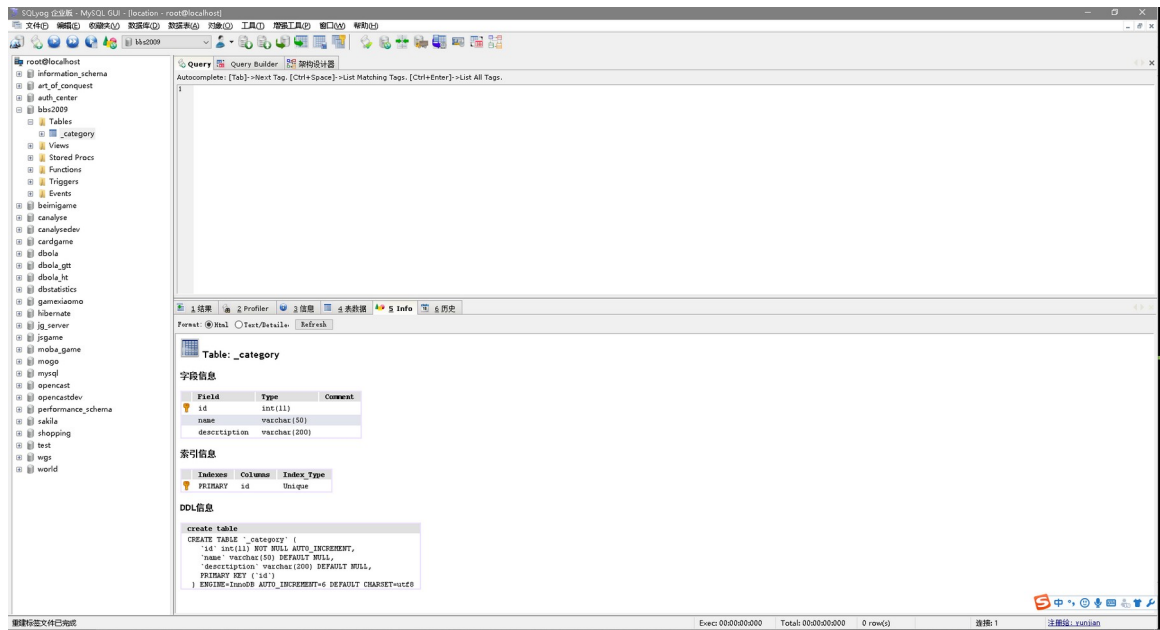
8. 收藏馆功能

在主页中选择收藏馆，可以看到自己通关获取的化妆品奖励，效果如图所示：

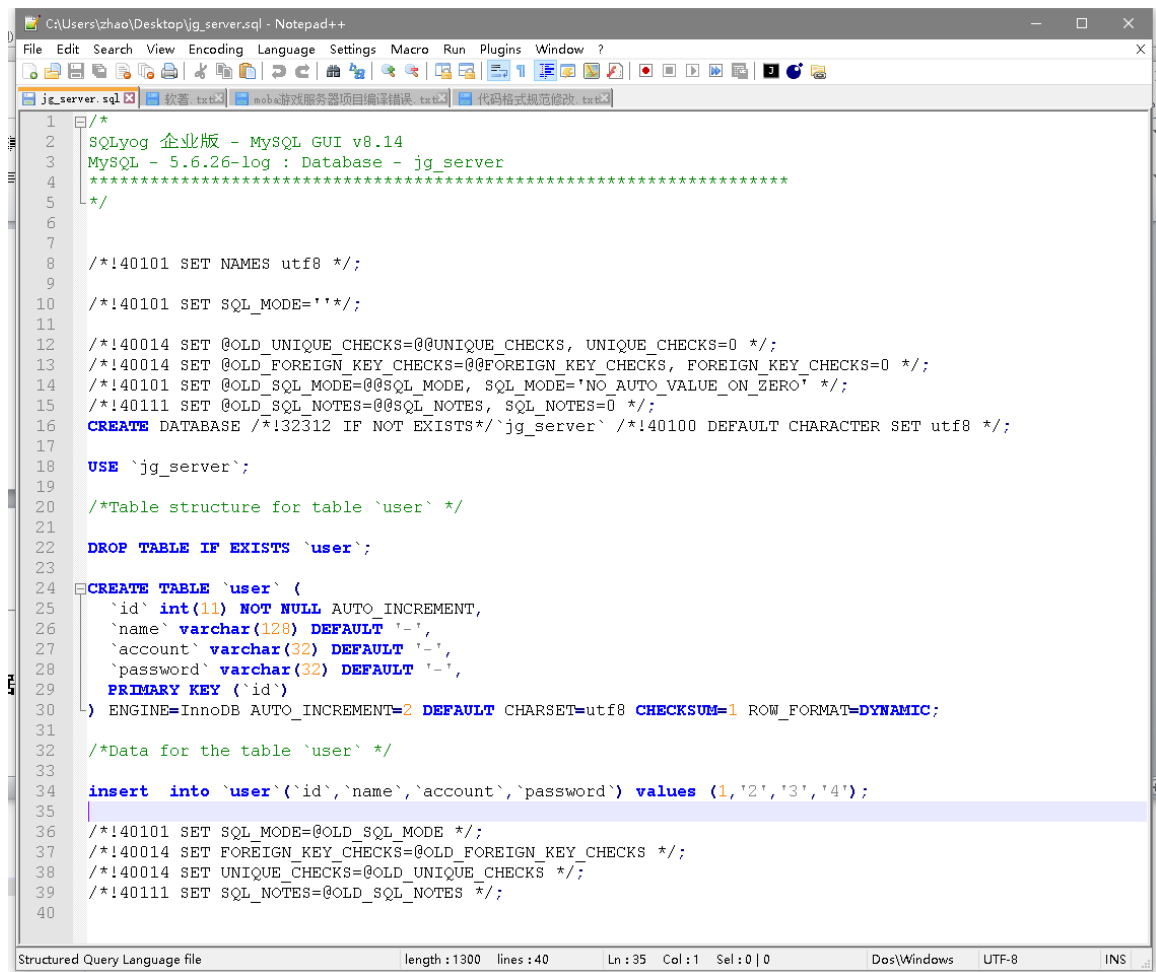


9. 数据库软件

数据库使用的是 mysql5.6，表如下



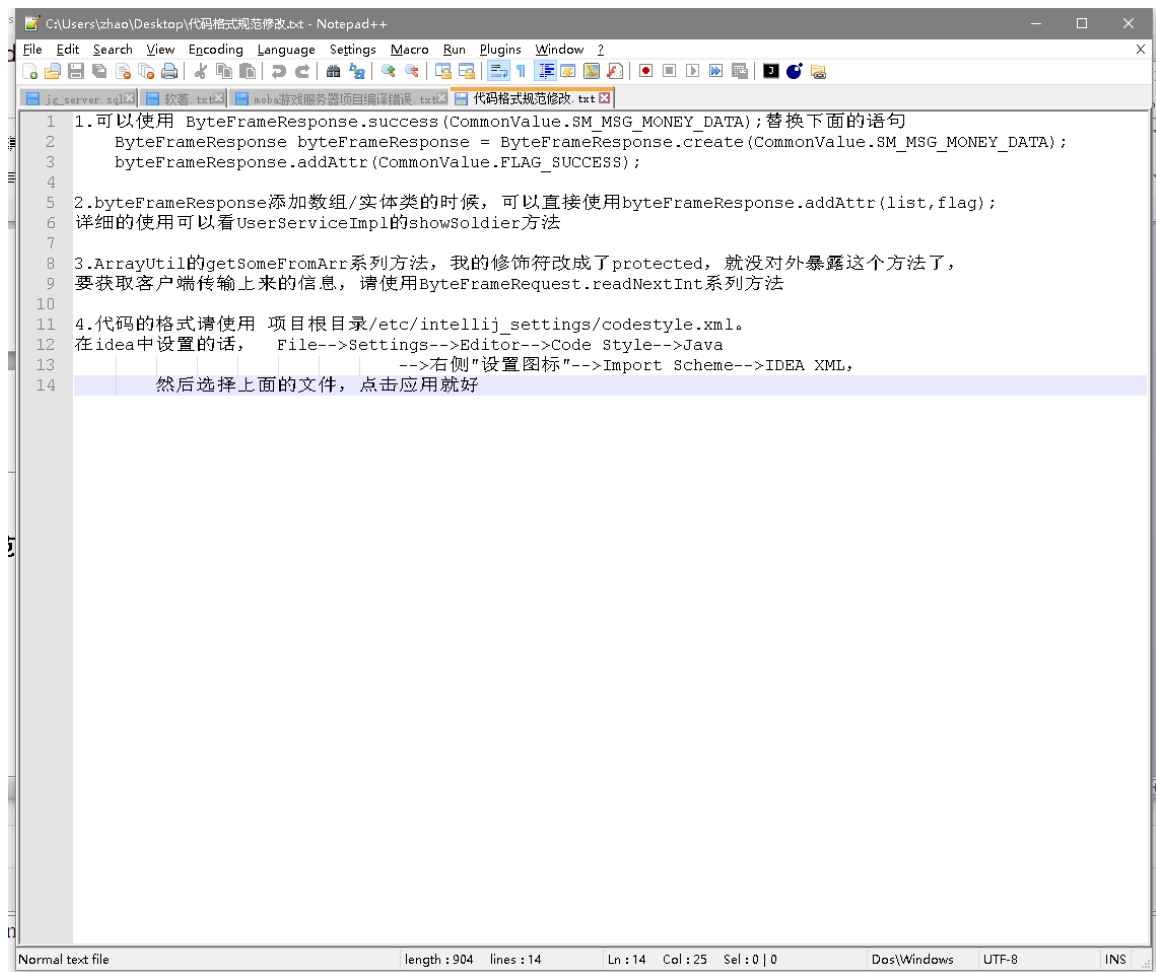
10. 数据库脚本文件



```
1  /*
2  SQLyog 企业版 - MySQL GUI v8.14
3  MySQL - 5.6.26-log : Database - jg_server
4  ****
5  */
6
7
8  /*!40101 SET NAMES utf8 */;
9
10 /*!40101 SET SQL_MODE='*/;
11
12 /*!40014 SET @OLD_UNIQUE_CHECKS=@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
13 /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
14 /*!40101 SET @OLD_SQL_MODE=@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
15 /*!40111 SET @OLD_SQL_NOTES=@SQL_NOTES, SQL_NOTES=0 */;
16 CREATE DATABASE /*!32312 IF NOT EXISTS*/ `jg_server` /*!40100 DEFAULT CHARACTER SET utf8 */;
17
18 USE `jg_server`;
19
20 /*Table structure for table `user` */
21
22 DROP TABLE IF EXISTS `user`;
23
24 CREATE TABLE `user` (
25   `id` int(11) NOT NULL AUTO_INCREMENT,
26   `name` varchar(128) DEFAULT '-',
27   `account` varchar(32) DEFAULT '-',
28   `password` varchar(32) DEFAULT '-',
29   PRIMARY KEY (`id`)
30 ) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8 CHECKSUM=1 ROW_FORMAT=DYNAMIC;
31
32 /*Data for the table `user` */
33
34 insert into `user` (`id`,`name`,`account`,`password`) values (1,'2','3','4');
35
36 /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
37 /*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
38 /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
39 /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
40
```

Structured Query Language file length: 1300 lines: 40 Ln: 35 Col: 1 Sel: 0 | 0 Dos\Windows UTF-8 INS

11. 规范文件



```

1 1. 可以使用 ByteFrameResponse.success(CommonValue.SM_MSG_MONEY_DATA); 替换下面的语句
2   ByteFrameResponse byteFrameResponse = ByteFrameResponse.create(CommonValue.SM_MSG_MONEY_DATA);
3   byteFrameResponse.addAttr(CommonValue.FLAG_SUCCESS);
4
5 2. byteFrameResponse添加数组/实体类的时候, 可以直接使用byteFrameResponse.addAttr(list,flag);
6   详细的使用可以看UserServiceImpl的showSoldier方法
7
8 3. ArrayUtil的getSomeFromArr系列方法, 我的修饰符改成了protected, 就没对外暴露这个方法了,
9   要获取客户端传输上来的信息, 请使用ByteFrameRequest.readNextInt系列方法
10
11 4. 代码的格式请使用 项目根目录/etc/intellij_settings/codestyle.xml。
12   在idea中设置的话,   File-->Settings-->Editor-->Code Style-->Java
13                       -->右侧"设置图标"-->Import Scheme-->IDEA XML,
14   然后选择上面的文件, 点击应用就好

```

Normal text file length: 904 lines: 14 Ln: 14 Col: 25 Sel: 0 | 0 Dos\Windows UTF-8 INS