**概要设计规约（说明书）**

修订历史:

修改或初始编写日期、 SEPG、 版本、 说明、 作者、 评审时间、 评审参与人员、 评审后修改批准日期、确认签字人员

目　录

1. 引言

1.1. 概要设计依据

1.2. 参考资料

1.3. 假定和约束

2. 概要设计

2.1. 系统总体架构设计

2.2. 系统软件结构设计

2.3. 接口设计

2.4. 界面设计

2.5 数据库设计

2.6 系统出错处理设计

2.6.1 出错信息

2.6.2 补救措施

2.6.3 系统维护设计

附录

**软件需求规格(说明书)模板解释及说明：**

修订历史：描述修订人、时间及在哪个版本上修订。如果是迭代开发每个版本对应一次迭代。非迭代开发每次正式的修改标记为一个版本。可以用以下的表来体现：

修订历史记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **编写日期** | **SEPG** | **版本** | **说明** | **作者** | **。。。。** |
| 2019.11.16 | All members | 1.0.0 | 基于Netty的mvc | 崔垚、蒋伟博、吴国栋、杨雨奇 |  |
| 2019.12.8 | All members | 1.0.1 | 基于websocket的client-server直连 | 崔垚、蒋伟博、吴国栋、杨雨奇 |  |
| 2020.01.04 | All members | 1.0.2 | 基于websocket的client-server直连  基于spring的mvc  基于flask的ai服务器 | 崔垚、蒋伟博、吴国栋、杨雨奇 |  |

1. 引言

1.1. 概要设计依据

需求分析规约

1.2. 参考资料

软件工程：实践者的研究方法（第八版）Roger S. Pressman著

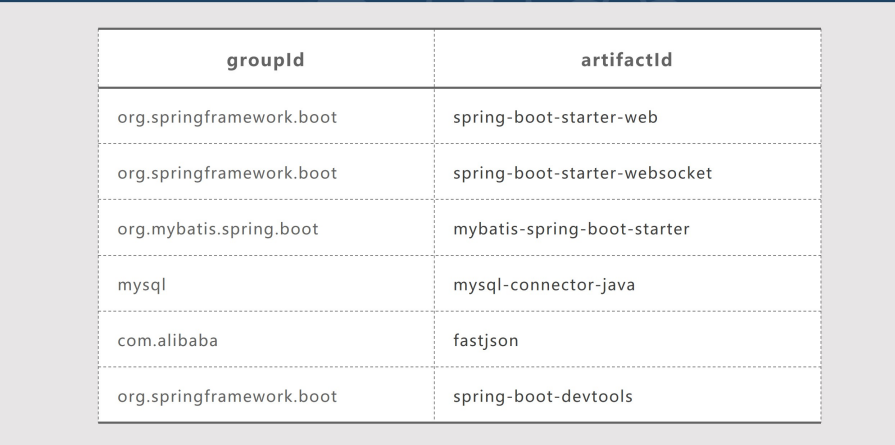
1.3. 假定和约束

开发期限：至2020.1.8

版本控制：git

2. 概要设计

2.1. 系统总体架构设计



至于架构，要实现的架构是用源/架构编写的。让我们看看项目中包含的依赖项。1. Spring Boot Starter网站。 它允许我们创建一个基于Spring Boot的MVC Web程序。

1. 网络套接字启动器。 它允许会话进行全双工连接，这意味着浏览器和服务器并发送消息种子主动消息，只要保持会话连接活。

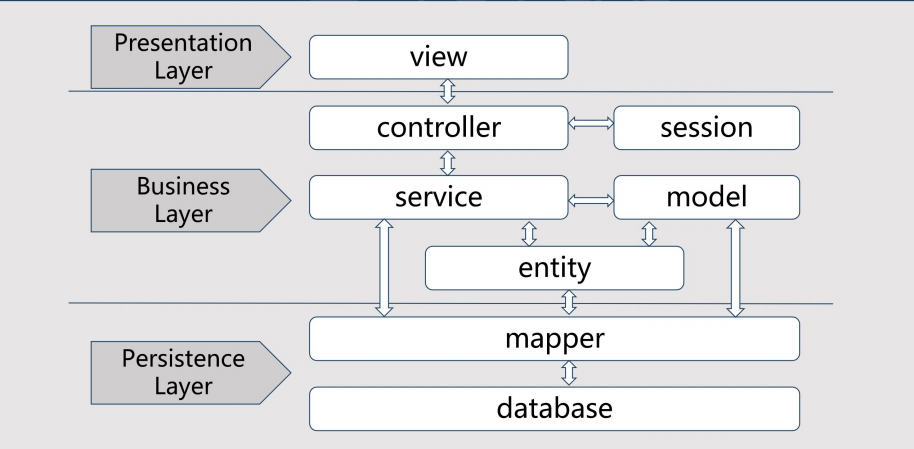
3. mybatis。 这是一个广泛使用的持久层框架，用户包括阿里巴巴等主流公司。

4.数据库我们使用MySql作为存储库，使用jdbc进行Java连接。

5.我们以json格式来回发送消息。 因此，我们还导入了json依赖项。

6. Dev-tools用于热部署，编译器可以在其中重新构建并重新运行项目时间很短。

2.2. 系统软件结构设计



上图是我们的主要架构。我们采用了经典的3层Web框架。

当消息从表示层到达业务层时，第一个模块是控制者。首先负责取消请求并处理消息开始。它还用于控制网络连接和响应发送。例如，在Web套接字，控制器需要记住会话，以便以后使用消息交换。然后，请求进入服务层。它用作操作逻辑处理。最简单的在不涉及特定数据的情况下，它可以计算然后返回结果而无需持久层。但是，这种情况越来越少，因为我们可以考虑到BS，几乎可以使用JavaScript在表示层完成所有这些操作框架项目。当涉及数据库连接时，使用映射器进行数据库访问管理，有时不同线程中的服务可能需要数据库同时。关于发送和接收对象，除了普通对象（如int或字符串，一个实体对象适用于此功能。对于一些不需要的临时对象存储在数据库中，将它们存储到模型中，它们就可以进行数据自我管理。另外，模型可以与持久层交互。

2.3. 接口设计

对2.2节中的类之间的接口，模块之间的接口，对接口作详细描述。

本系统和其他系统的接口，对接口作详细描述。

2.3.1 登陆接口



2.3.2 注册接口



2.3.3 历史战绩接口



2.3.4 好友列表接口



2.3.5 任务列表接口

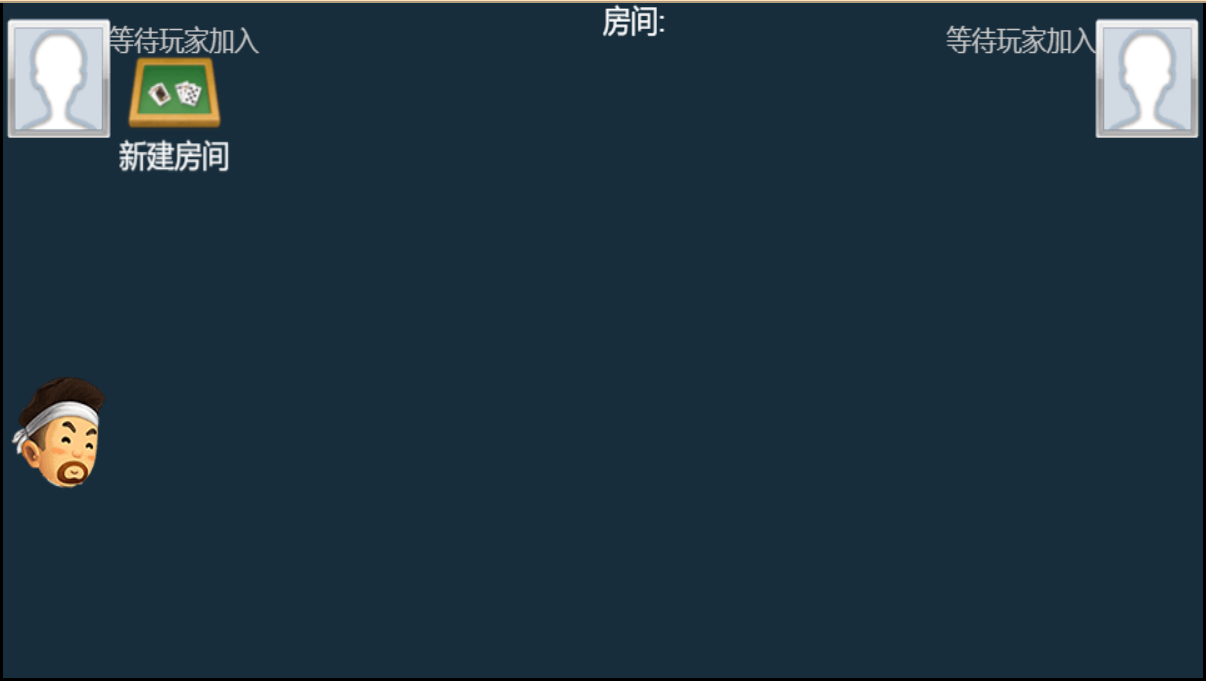


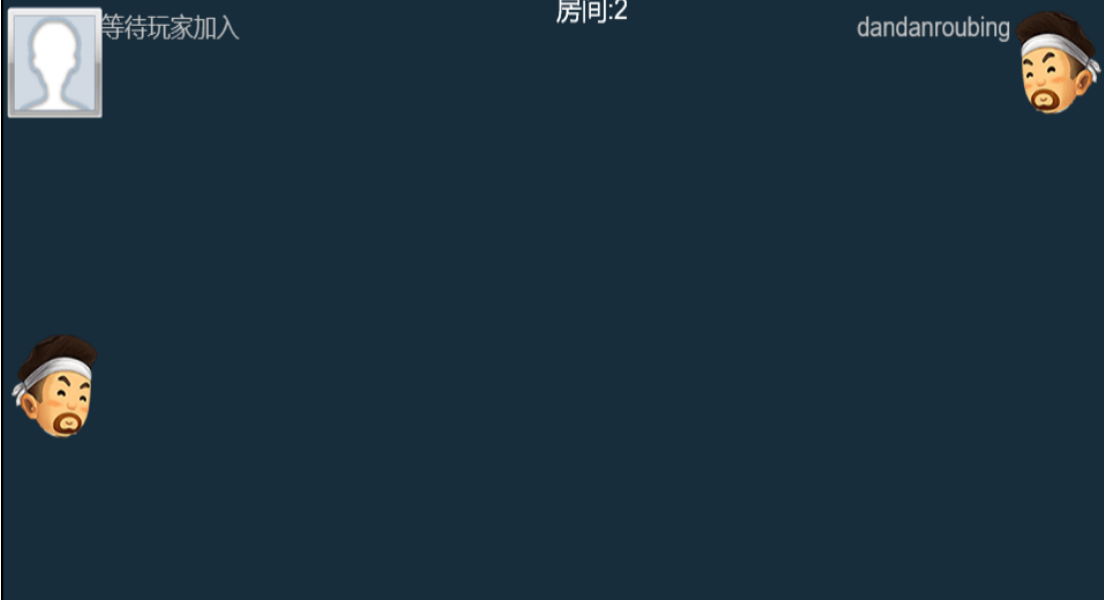
* + 1. 简单AI出牌接口



2.4. 界面设计

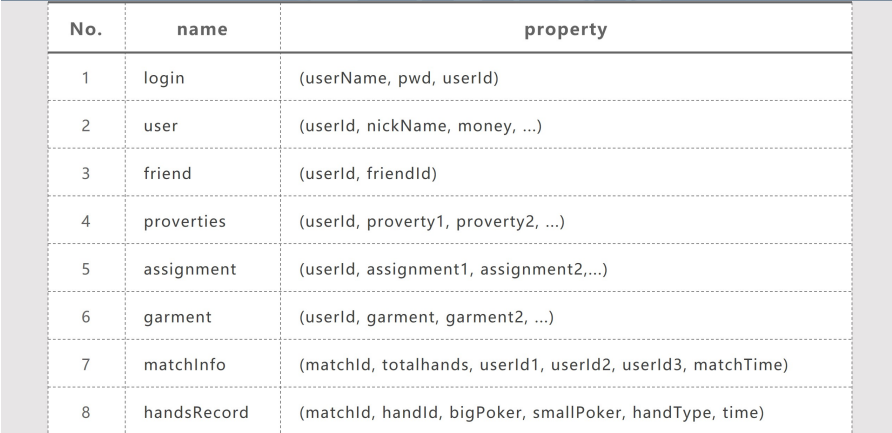








2.5 数据库设计



Login：用于登录和注册功能。

User：有关用户的基本信息。 大多数都可以在之后查看和修改用户进入游戏。

Friend：这是一个连接表，用于将彼此的朋友关系与用户ID的格式。

Task: 这是一个实时记录玩家任务完成情况的表。

Proverties：对于这些奖励和未使用的衣服，它们将被存储在这里。

Assgnment：它记录分配的共谋性。

Garment：对于仍在使用但尚未淘汰的持久服装，时间开始使用服装记录在这里。

MatchInfo：对于记录的特定比赛，一些基本信息存储在此表。

HandsRecord：在一场比赛中，每位球员进行很多手牌。 这里我们会记录时间，扑克信息，以重新显示比赛。

2.6 系统出错处理设计

2.6.1 出错信息

可以用一览表的形式说明每种可能出现错误或故障情况发生时，系统输出信息的形式、含义及处理的方法。

2.6.2 补救措施

说明故障出现后可能采取的变通措施。

2.6.3 系统维护设计

说明为了系统维护的方便而在程序内部设计中作出的安排，包括在程序中专门安排用于系统的检查与维护的检测点和专用模块（或）类。