文件编号：MAC-SWE-TMP-07 密级：■ 保密 □ 通用

**详细设计说明书**

# Detail Design Description

本程序属MAC公司所有，未经书面许可，

不得以任何形式复印或传播。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | | | | |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
|  |  |  |  | |  |
| V1.0 | 楼康 | 孙潇桐 | 孙潇桐 | |  |
| 版 本 | 编写/修改 | 审 核 | 批 准 | | 修改原因 |
| 受控编号： | | 正文页数： | | 附件页数： | |
| 存储编号： | | 管理部门： | | 审查期： | |

**修 改 记 录**

|  |  |  |  |
| --- | --- | --- | --- |
| 日 期 | 版 本 | 修 改 内 容 | 修 改 人 |
| 2023/05/12 | <1.0.0> | 初步完成文档 | 楼康 |
| 2023/06/20 | <1.0.1> | 检查并完善文档内容 | 孙潇桐 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

文件编号： 密级：■ 保密 □ 通用

**详细设计说明书**

|  |  |  |
| --- | --- | --- |
| 项目名称：OBattle源神斗 | 拟制：楼康 | 审核：孙潇桐 |
| SQA：测试小组 | 批准：孙潇桐 |
| 项目代号：OBattle | 收文： | |
| 产品版本：1.0 | 抄送： | |

**1．0 引言**

**1．1 编写目的**

目的:

1. 提供技术指导和参考：详细设计说明书为项目团队成员提供了技术指导和参考，以便他们在开发过程中正确地实施设计。它描述了系统的详细设计方案、架构、模块和组件的设计原理、算法和数据结构等技术细节，帮助开发人员理解和实现系统的功能和特性。
2. 促进团队合作和沟通：详细设计说明书作为一个共享文档，可以促进团队成员之间的合作和沟通。它提供了一个统一的设计规范和标准，让团队成员可以共同理解和讨论系统的设计方案，并在开发过程中协同工作。通过详细设计说明书，团队成员可以更好地协调工作，减少误解和冲突。
3. 支持代码开发和测试：详细设计说明书为代码开发和测试提供了基础。它将系统的设计细节转化为实际的代码实现指南，包括数据结构、接口定义、模块划分等。开发人员可以根据详细设计说明书编写代码，测试人员可以根据其中的设计规范进行测试。通过详细设计说明书，可以提高代码的质量和可维护性。
4. 为系统维护和升级提供支持：详细设计说明书为系统的后续维护和升级提供了支持。它记录了系统的设计决策和实现细节，包括系统结构、模块功能、接口规范等。在后续的维护和升级过程中，开发人员可以参考详细设计说明书，了解系统的设计意图和实现细节，更好地理解和修改系统。
5. 提供审查和评估依据：详细设计说明书作为一个正式的文档，可以用于进行审查和评估。项目管理人员、技术专家和利益相关者可以根据详细设计说明书对系统的设计方案进行审查和评估，以确保设计的合理性、可行性和符合需求。通过审查和评估，可以发现潜在的问题和风险，并提出改进建议。

总之，详细设计说明书的编写目的是为了提供技术指导和参考，促进团队合作和沟通，支持代码开发和测试，为系统维护和升级提供支持，以及提供审查和评估依据，用于记录系统的详细设计方案和实现细节，并在整个项目生命周期中发挥重要作用。另外，在软件开发过程中，详细设计说明书也可以用作沟通的工具，以促进团队成员之间的有效协作。

预期的读者包括以下人员：

1. 软件开发人员：他们需要了解联机斗地主软件的整体结构和各个子系统之间的关系与交互，以便开发出高质量、符合要求的代码。
2. 审计人员和监管机构：在某些情况下，体系结构文档可能需要被审计人员和监管机构审查和评估，以确保系统的合规性、安全性和稳定性。
3. 软件测试人员：在测试阶段，他们需要了解联机斗地主软件的整体结构和各个子系统之间的关系与交互，以便更有效地设计和执行测试用例。

**1．2 背景**

说明：

a)

名称：Origin War of Spirit 源神斗

b)

任务提出者:

孙潇桐，俞家宝

开发者:

孙潇桐，俞家宝，任庆华

测试小组：

孙潇桐，俞家宝，任庆华，纪国梁，楼康以及部分组外成员

文档小组：

楼康，孙潇桐

用户:

1.个人玩家：对游戏感兴趣的普通玩家.

2.团队玩家：例如朋友、家人、同事等，他们可以组队进行游戏。

3.竞技玩家：喜欢参加各种比赛和竞赛，挑战自我，展示自己的棋牌技能。

运行软件的计算站:

数据库服务器:负责存储和管理斗地主游戏数据和玩家信息

Web服务器：同步用户的应用内信息，控制用户行为。支持玩家的登录与注册。

**1．3 定义**

①软件体系结构：指软件系统的组织结构、组件和连接方式等方面的描述，用于指导软件的设计和开发。

②模块化：指将软件系统划分为独立的模块或组件，以便于管理和维护。

③分层结构：指将软件系统按照功能分为不同层次，每层之间通过明确定义的接口进行交互，以实现代码的复用和模块化。

④客户端/服务器结构：指软件系统将功能划分为客户端和服务器两部分，客户端负责显示界面和用户交互，服务器负责后台逻辑和数据管理。

⑤MVC架构：指软件系统采用Model-View-Controller的架构，将业务逻辑、用户界面和逻辑控制分别独立分层，以实现模块化和代码的复用。

⑥API：指软件系统提供的应用程序接口，用于与其他系统进行交互和数据传输。

⑦SDK：指软件系统提供的软件开发工具包，用于帮助开发者快速地开发和集成软件系统。

⑧GUI：指软件系统的图形用户界面，用于与用户进行交互和数据显示。

**1．4 参考资料**

①斗地主的基本规则

②原神人物建模

③JavaFX官方文档 <https://openjfx.io/javadoc/20/>

④Java8官方文档 <https://docs.oracle.com/javase/8/docs/api/overview-summary.html>

⑤MySQL官方文档<https://docs.oracle.com/javase/8/docs/api/overview-summary.html>

**2．0 程序系统的结构**

这三个模块之间的层次结构如下：

游戏客户端模块 → 游戏服务器模块 → 数据库模块

其中，游戏客户端模块是位于最外层的用户界面模块，游戏服务器模块是位于中间层的业务逻辑模块，数据库模块是位于底层的数据存储模块。游戏客户端模块通过与游戏服务器模块的通信，与游戏服务器模块交互；游戏服务器模块通过与数据库模块的通信，与数据库模块进行数据读写操作。

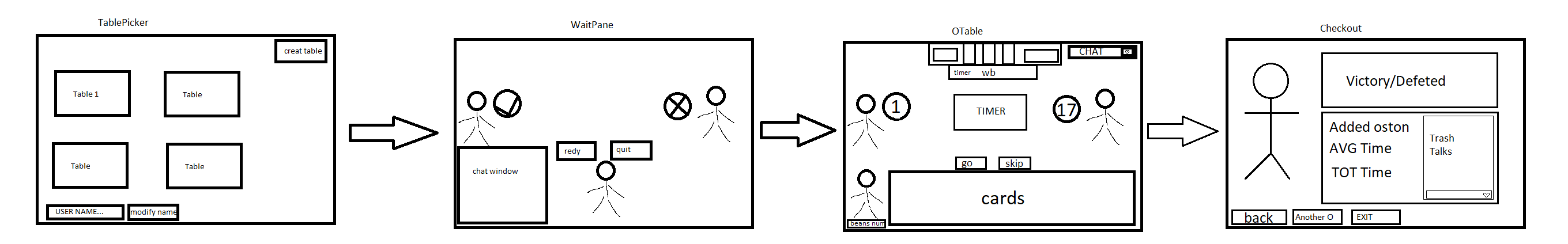
|  |  |  |
| --- | --- | --- |
| 客户端模块 | 服务器模块 | 数据库模块 |
| 玩家聊天 | 玩家注册 | 用户信息存取 |
| 登录界面 | 玩家登录 | 数据管理 |
| 等待界面 | 牌局生成 |  |
| 游戏界面 | 玩家状态 |  |
| 游戏逻辑 |  |  |
| 结算界面 |  |  |

**3．0 模块综述**

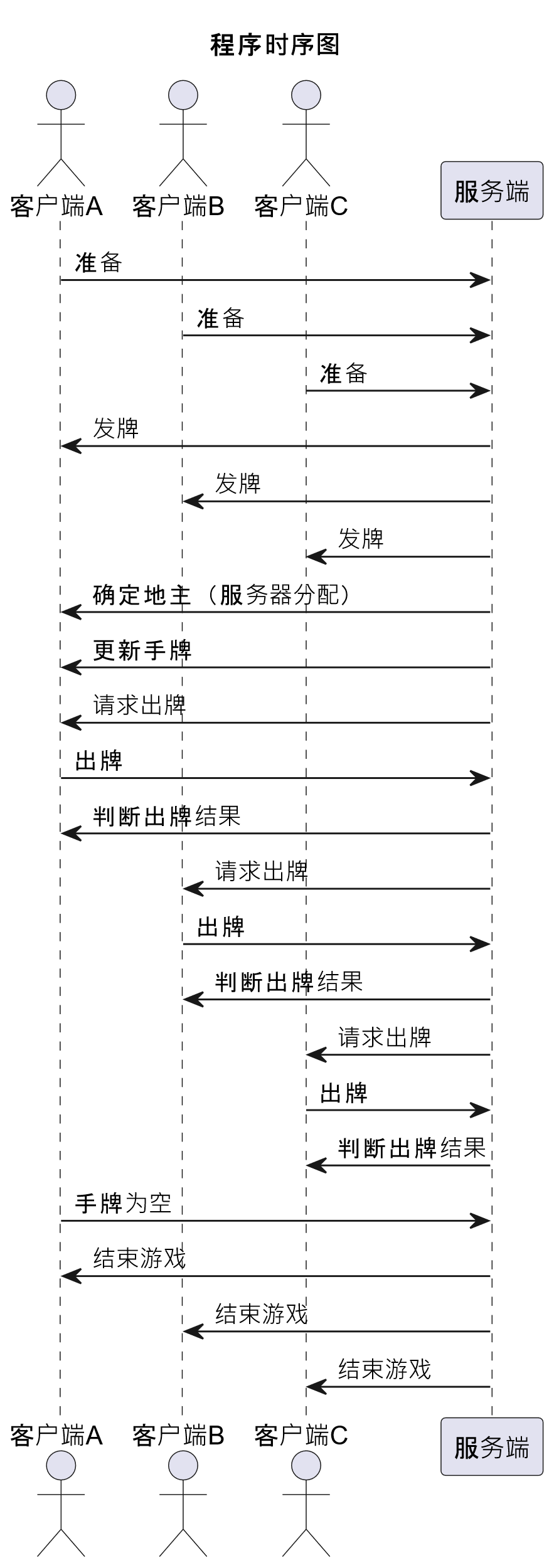
**各模块功能描述,输入项,输出项,接口，注释设计**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 模块名称 | 所属端号 | 功能描述 | 输入项 | 输出项 | 接口 | 注释设计 |
| 玩家聊天 | 客户端 | 实现玩家之间的聊天交互，增加互动性。 | 玩家输入的字符串 | 将玩家输入字符串，玩家UID，时间等输出到后端 | 客户端更新客户信息申请，到服务器和数据库进行更改 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 登陆界面 | 客户端 | 每个玩家都有一个自己的账号，增加个人归属感 | 无 | 登陆界面展示 | 客户端更新客户信息申请，到服务器和数据库进行更改 | 顶层注释:模块信息, |
| 等待界面 | 客户端 | 玩家登陆账户后，可以等待，自己决定什么时候开始玩 | 准备就绪和决定退出两个按键 | 按准备就输出开始，进入下一个游戏界面，退出则退出GUI | 客户端更新客户信息申请，到服务器和数据库进行更改 | 顶层注释:模块信息, |
| 游戏界面 | 客户端 | 开始游戏后的GUI，包括自己的牌，地主牌等 | 起始:叫或不叫地主  中途:出牌或者跳过 | 起始:叫地主则成为地主，输出地主信息到服务器  中途:出牌则更新牌库，跳过则将出牌权限给顺位 | 客户端更新客户信息申请，到服务器和数据库进行更改 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 结算界面 | 客户端 | 决出胜负后的等待结算界面 | 返回，下一局，退出三个按键 | 按返回:则查看上局情况  按下一局则进入准备阶段  按退出，则退出GUI | 客户端更新客户信息申请，到服务器和数据库进行更改 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 玩家注册 | 服务器 | 玩家注册账号 | 玩家输入账号，密码 | 若账号不存在于数据库，则将此账号注册给该玩家 | 服务器获得客户端数据信息 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 玩家登录 | 服务器 | 玩家登录自己的账号 | 玩家输入账号，密码。 | 若账号密码正确，则进入GUI | 服务器获得客户端数据信息 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 牌局生成 | 服务器 | 每一把牌都随机生成 | 无 | 随机生成牌局(地主牌和玩家手牌) | 服务器获得客户端服务器获得客户端数据信息数据信息 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 玩家状态 | 服务器 | 有时玩家可能中途退出，转为机器人智能出牌 | 玩家可能处于的状态(离线,在线) | 若离线，则转为机器人智能出牌 | 服务器获得客户端数据信息 | 顶层注释:模块信息,  分支注释:各数据项的功能,取值范围的确定 |
| 游戏逻辑 | 服务器 | 关于斗地主的游戏规则书写 | 用户出牌是否符合规则 | 不符合则拒绝出牌 | 服务器获得客户端数据信息 | 顶层注释:模块信息, |
| 用户信息存取 | 数据库 | 用户的账号密码存取 | 无 | 无 | JDBC，获取数据库连接,交互服务器 | 顶层注释:模块信息, |
| 牌局信息 | 数据库 | 当前牌局的信息记录 | 无 | 无 | JDBC，获取数据库连接,交互服务器 | 顶层注释:模块信息, |
| 战绩信息 | 数据库 | 用户输赢战绩保存 | 用户账号，密码 | 可查对应用户战绩 | JDBC，获取数据库连接,交互服务器 | 顶层注释:模块信息, |

GUI初步设计图图:



时序图：



流程图：

