<酒吧管理系统>

软件需求规约

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <19/06/2023> | <1.0> | 基本完成易用性、可靠性、性能、可支持性、设计约束的定义 | 陆逸凡 陈馨桐 王琪祥 VAHAGN GHAZARYAN |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

2. 整体说明 4

2.1 背景 4

2.2 总体效果及功能 4

2.3 约束 4

3. 具体需求 4

3.1 功能 4

3.1.1 <Use case 图> 4

3.1.2 <Use case1 规约> 4

3.1.3 <Use case2 规约> 4

3.2 易用性 5

3.2.1 <易用性需求一> 5

3.2.2 <易用性需求二> 5

3.2.3 <易用性需求三> 5

3.3 可靠性 5

3.3.1 <可靠性需求一> 5

3.3.2 <可靠性需求二> 5

3.3.3 <可靠性需求三> 5

3.3.4 <可靠性需求四> 5

3.3.5 <可靠性需求五> 5

3.3.6 <可靠性需求六> 5

3.4 性能 5

3.4.1 <性能需求一> 5

3.4.2 <性能需求二> 5

3.4.3 <性能需求三> 6

3.4.4 <性能需求四> 6

3.4.5 <性能需求五> 6

3.5 可支持性 6

3.5.1 <可支持性需求一> 6

3.5.2 <可支持性需求二> 6

3.6 设计约束 6

3.6.1 <设计约束一> 6

3.6.2 <设计约束二> 6

3.7 联机用户文档和帮助系统需求 6

3.8 接口 6

3.8.1 用户界面 6

3.8.2 硬件接口 6

3.8.3 软件接口 6

3.8.4 通信接口 7

3.9 适用的标准 7

软件需求规约 (简化版)

# 简介

## 目的

本文档首先给出了整个酒吧管理系统的整体效果与分功能效果的概貌，从总体架构上给出了整个系统的轮廓，然后对功能需求、性能需求及非功能需求进行了详细描述。其中功能需求描述采用了UML的用例模型方式，给出了基本事件流、备选流以及直观的用例图。

本文档还详尽说明了这一款产品的接口要求及适用的标准，这些规格说明是进行设计的基础，也是编写测试用例和进行系统测试的主要依据。同时，本文档也是用户确定软件功能需求的主要依据。

# 整体说明

## 背景

对于当代年轻人而言，日常休闲多多少少会有饮酒需求。但是通常小酒吧老板身兼多职，采购、调酒、服务，工作忙的不可开交。而目前国内市场缺少一种面向小型酒吧的酒吧管理系统（国内的少有的酒吧管理系统只有面向大型高档酒吧）。

## 总体效果及功能

该产品是一款面向酒吧管理经营人员的集点单收银、菜单管理、仓库管理、财务管理、活动发布于一体的酒吧管理系统。

## 约束

该产品受功能、性能及非功能需求约束。

# 具体需求

## 功能

[此节为以UseCase模型和自然语言风格表达的需求说明为此设计的系统功能性需求。对于许多应用程序，此节会成为 **SRS** 包的主体部分，所以应仔细考虑此节的组织方式。此节通常按ＵseCase或特性来组织，但也可能会有其他适用的组织方式，例如按用户或子系统组织的方式。功能性需求可能包括特性集和安全性。]

### <Use case 图>

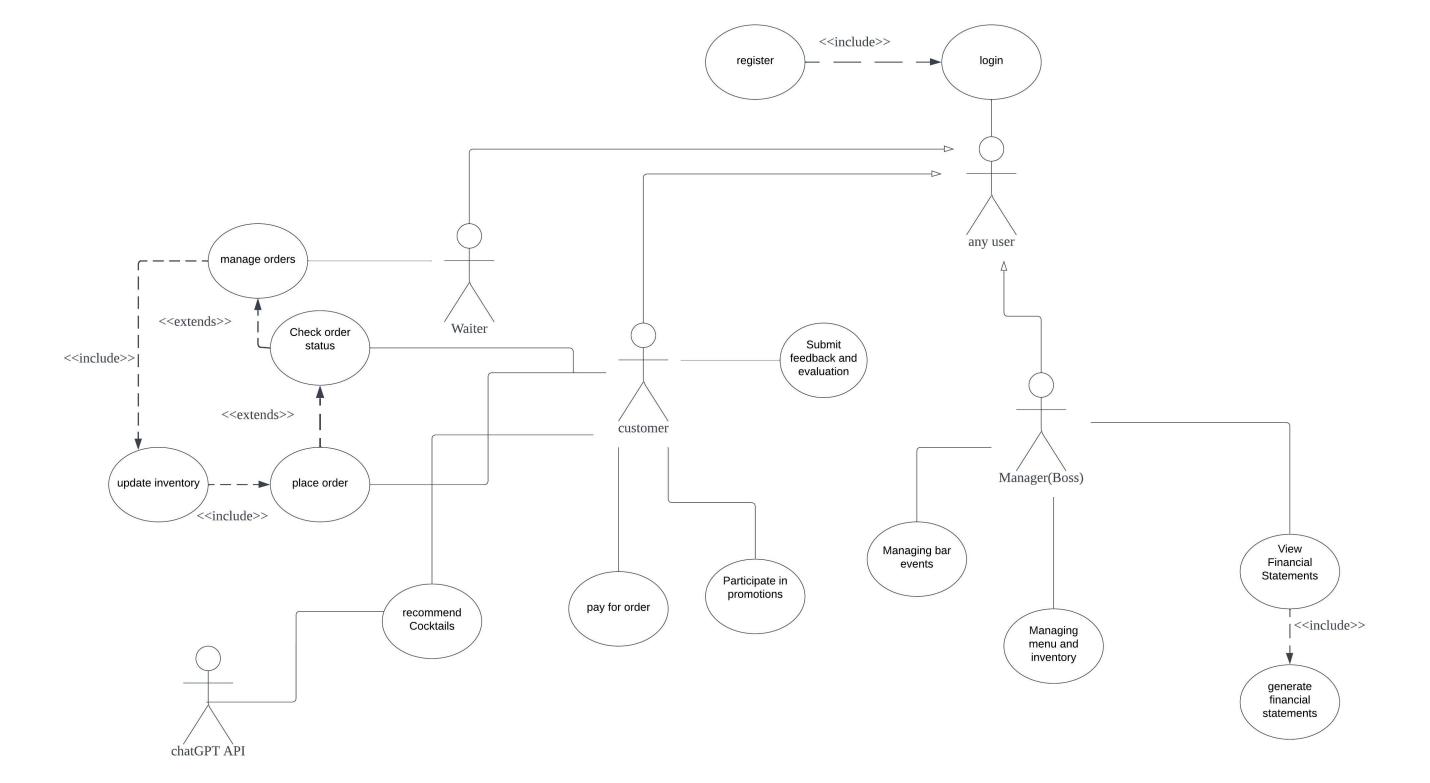


图1-use case模型

Actor：

1. Customer（顾客）即消费者
2. Manager（Boss）即酒吧老板兼管理员
3. Cashier（收银员）主要处理酒吧的财务
4. Waiter（服务生）主要处理顾客下的订单

Usecase：

1. Register：提供用户注册功能
2. Login：用户登录
3. Place order：用户下订单
4. Check order status：查看订单状态
5. Manage orders：管理用户订单
6. Recommend cocktails：调用chatGPT API 为用户提供鸡尾酒推荐
7. Pay for order：用户支付订单
8. Participate in promotions：用户参与酒吧促销活动
9. Submit feedback and evaluation：用户为酒吧服务、餐饮等提供反馈和评价
10. Managing bar events：管理酒吧活动（如节日促销活动等）
11. Managing menu and inventory：管理酒吧菜单和商品（酒水、食物）库存
12. Generate financial statements：生成财务报表
13. Managing financial statements：管理财务报表
14. Update inventory：菜品、酒水库存自动更新

[Use case 图，并对每个actor或usecase有简要说明。如果系统比较大，则可按用户或子系统进行组织]

### <Use case1 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：register 用户注册**

**前置条件**：用户打开注册界面

**事件流步骤**：

1. 用户打开注册页面。
2. 系统显示注册表单。
3. 用户输入用户名、电子邮件和密码。
4. 用户点击"提交"按钮。
5. 系统验证输入数据是否有效。
6. 如果输入数据有效，系统创建新用户账户并保存信息。
7. 系统显示注册成功消息并跳转到登录页面。
8. 用户完成注册流程

**替代流**：

· 如果输入数据无效（如无效的电子邮件格式），系统显示错误消息，并要求用户重新输入。

· 如果用户名已经被占用，系统显示用户名已存在的错误消息，并要求用户选择不同的用户名。

**后置条件：**

用户成功注册并可以进入登陆界面

### <Use case2 规约>

Usecase： Login 用户登录

**前置条件**：用户打开注册界面

**事件流步骤**：

1. 用户打开登录页面。
2. 系统显示登录表单。
3. 用户输入用户名、密码。
4. 用户点击"提交"按钮。
5. 系统验证输入数据是否有效。
6. 如果输入数据有效，系统显示登录成功消息并跳转到主页面。
7. 用户完成登录流程

**替代流**：

· 如果输入数据无效（如无效的用户名或错误密码），系统显示错误消息，并要求用户重新输入。

**后置条件：**

用户成功登录并进入主界面

### <Use case3 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：place order 用户下订单**

**前置条件**：顾客登录成功，已经浏览菜单

**事件流步骤**：

1. 顾客登陆后进入主界面。
2. 顾客选择用餐人数。
3. 系统显示酒吧菜单。
4. 顾客浏览菜单。
5. 顾客点击"加入购物车"按钮，将选好的餐食放入购物车中。
6. 顾客进入下订单页面之后点击“提交订单”按钮。
7. 界面提示顾客是否要下单，或是修改订单。
8. 顾客点击“确认下单”，将订单发送至系统
9. 发送订单成功后界面显示“您已下单”消息。

**替代流**：

· 顾客在购物车为空时就点击“提交订单”按钮，界面会提示“您的订单为空”

**后置条件：**

用户成功下订单，并可以查看自己的订单状态。

### <Use case4 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：check order 查看订单**

**前置条件**：顾客已经下订单；服务生已经登录主界面

**事件流步骤**：

1. 顾客下订单成功/服务生已经成功登录，进入主界面的订单页面
2. 订单界面自动生成顾客的订单号，能显示订单状态；如：已下单、店家已接单、正在制作、已完成。/服务生在接收到用户发送的订单后，选择接受订单。
3. 用户/服务生可查看订单状态

**替代流**：

·

**后置条件：**

顾客/服务生查看订单状态成功

### <Use case5 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：manage orders 管理用户订单**

**前置条件**：顾客已经下订单；服务生已经登录主界面

**事件流步骤**：

1. 顾客下订单成功/服务生已经成功登录，进入主界面的订单页面
2. 若顾客寻求服务生帮助要求进行订单的修改，服务生可以在自己的系统界面上进行顾客订单的修改。
3. 服务生可查看修改后订单状态

**替代流**：

·

**后置条件：**

服务生修改订单状态成功

### <Use case6 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：recommend cocktails 鸡尾酒推荐**

**前置条件：顾客已经登录**

**事件流步骤**：

1. 顾客浏览菜单，进入菜单的酒水推荐页面
2. 系统调用chatGPT Api为用户随机推荐鸡尾酒，如果顾客不满可以点击“重新推荐”按钮，让系统进行新的推荐。
3. 顾客获得推荐的酒水

**替代流**：

·

**后置条件：**

顾客获取推荐成功

### <Use case7 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：pay for order**

**前置条件**：顾客已经下订单

**事件流步骤**：

1. 顾客下订单成功，完成进食
2. 顾客在订单页面选择支付方式
3. 用户选择支付方式后付款

**替代流**：

·若用户超过十五分钟没有付款，？

**后置条件：**

顾客付款成功

### <Use case7 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：submit feedback and evaluation**

**前置条件：顾客已经支付订单成功**

**事件流步骤**：

1. 支付成功后，系统自动跳转到评价表单页
2. 顾客填写反馈和评价，点击“确认提交按钮”，信息被发送至系统

**替代流**：

·顾客若不愿评价，可点击评价页面表单中的“跳过”按钮，不进行评价。

**后置条件：**

顾客评价成功

### <Use case8 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：Managing bar events**

**前置条件：顾客已经登陆账号**

**事件流步骤**：

1. 顾客进入主界面中的活动页面
2. 页面显示当前时期酒吧的促销活动等
3. 顾客点击“加入活动”，如果顾客的自身条件符合活动要求，则页面显示“加入活动成功”

**替代流**：

·若顾客条件不符合活动条件，则页面显示“您无法参加该活动”

**后置条件：**

顾客加入活动成功，可获得优惠

### <Use case9 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：manage bar events**

**前置条件：酒吧老板已经登录**

**事件流步骤**：

1. 管理员（老板）进入管理员界面的发布活动页面
2. 管理员（老板）编辑活动内容并发布

**替代流**：

·

**后置条件：**

管理员（老板）发布活动成功

### <Use case10 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：manage menu and inventory**

**前置条件：**管理员（老板）已经登录成功

**事件流步骤**：

1. 管理员（老板)进入主界面的管理菜单与库存页面
2. 管理员（老板）进行库存、菜单的修改

**替代流**：

·

**后置条件：**

管理员（老板）修改菜单、库存成功

### <Use case11 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：Generate financial statements**

**前置条件：**管理员（老板）已经登录成功

**事件流步骤**：

1. 管理员（老板）进入主界面的财务页面
2. 老板点击“生成财务报表”菜单，页面自动显示财务报表（通过选择菜单中的周、月度、季度、年度选项可生成响应财务报表）

**替代流**：

·若营业时间不足（如不足一年），点击年度选项，页面会显示“数据不足”

**后置条件：**

老板（管理员）获取财务报表成功

### <Use case12 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：Managing financial statements**

**前置条件：**管理员（老板）已经登录成功

**事件流步骤**：

1. 顾客浏览菜单，进入菜单的酒水推荐页面
2. 系统调用chatGPT Api为用户随机推荐鸡尾酒，如果顾客不满可以点击“重新推荐”按钮，让系统进行新的推荐。
3. 顾客获得推荐的酒水

**替代流**：

·

**后置条件：**

顾客获取推荐成功

### <Use case13 规约>

[对每个usecase 要有详细规约，说明其事件流等信息。]

**Use case：update inventory**

**前置条件：用户成功下订单或服务生修改用户订单成功**

**事件流步骤**：

1. 订单成功下好或修改好之后，系统自动从当前库存中减少订单中所需要的库存。

**替代流**：

·

**后置条件：**

库存更新成功

## 易用性（待定）

### <易用性需求一>

酒吧老板可在一小时培训后即可上手即可熟练使用该系统；可在三小时内创建自己的账号并成功上传有关自己的酒吧的全部数据，上传完毕后即可开始正常营业。

### <易用性需求二>

酒该系统廉价简约，适合工作人员为 人的小酒吧使用。

### <易用性需求三>

该系统符合CUA标准方面在一致的用户界面、用户友好的操作方式、易于理解的文本和标签及上下文相关的帮助和提示四个方面的需求。

## 可靠性 （待定）

### <可靠性需求一>

该系统可用时间百分比为 ，可使用 小时，仅允许酒吧老板访问，仅允许开发团队维护，当软件发生故障时， 功能将暂停使用。

### <可靠性需求二>

该系统平均故障时间为 。

### <可靠性需求三>

该系统平均修复时间为 。

### <可靠性需求四>

该系统支持的画面分辨率为 ，精确度为 。

### <可靠性需求五>

该系统每千行代码错误数目不大于 ，每个功能点错误数目不大于 。

### <可靠性需求六>

小错误（点单失败、编辑菜单失败、物品出库入库失败、物品仓储不足未提醒、活动发布失败）出现概率为 ；

大错误（收银失败、菜单仓库数据清空、活动数据清空）出现概率为 ；

重大错误（财务状况清空或泄露、个人数据全部清空、支付信息泄露）出现概率为 。

## 性能 （待定）

### <性能需求一>

在1000并发的场景下，系统平均响应时间0.5s，最长响应时间3s。

### <性能需求二>

系统每秒可处理的事务数是 1000+ 。

### <性能需求三>

系统菜单可容纳的最大项目数是 300 ，仓库可容纳的最大项目数是 10000 ，财务管理可储存最大订单数是 100k 。

### <性能需求四>

当系统发生故障时，系统平均响应时间 30ms ，最长响应时间 300ms ，系统每秒可处理的事务数是 1000+ 。

### <性能需求五>

系统占用磁盘存储空间最大是 500Mb ，占用内存最大是 100Mb 。

## 可支持性（待定）

### <可支持性需求一>

系统遵循 的编码标准，遵循 的命名规定。

### <可支持性需求二>

系统遵循 的维护访问权和维护实用程序。

## 设计约束(待定)

### <设计约束一>

系统的编程语言是 ，设计流程遵守 ，开发工具仅用于设计开发并维护该系统，系统前段采用react native，后端采用 ，要求在手机上使用。

### <设计约束二>

为开发系统，购买的构件为 ，购买的类库为 。

## 联机用户文档和帮助系统需求 （待定）

[如果存在对联机用户文档、帮助系统、关于声明的帮助等的需求，请在此说明。]

## 接口 （待定）

[此节规定应用程序必须支持的接口/界面。它应非常具体，包含协议、端口和逻辑地址等，以便于按照接口/界面需求开发并检验软件。]

### 用户界面（待定）

网站及客户端app主页

管理员

库存管理界面

订单管理界面

菜单管理界面

员工管理界面

用户

菜单界面

订单界面

个人信息界面

[说明软件将实现的用户界面。]

### 硬件接口

无

[此节指出软件所支持的所有硬件接口，其中包括逻辑结构、物理地址、预期行为等。]

### 软件接口

待定

[此节说明软件系统中与其他构件之间的软件接口。这些构件可以是购入的构件、取自其他应用程序重新利用的构件，也可以是为此 **SRS** 范围之外的子系统开发，但该软件应用程序必须与之交互的构件。]

### 通信接口 （待定）

传输层TCP

网络层协议IPv4

[说明与其他系统或设备（如局域网、远程串行设备等）的所有通信接口。]

## 适用的标准

[通过引用，此节说明了所有适用的标准以及适用于所述系统的相应标准的具体部分。例如，其中可以包括法律、质量及法规标准；业界在易用性、互操作性、国际化、操作系统相容性等方面的标准。]