# 需求规格说明书

目录

[需求规格说明书 1](#_Toc6939557)

[一、引言 3](#_Toc6939558)

[1.1 项目的目的与目标 3](#_Toc6939559)

[1.1.1 目的 3](#_Toc6939560)

[1.1.2 目标 3](#_Toc6939561)

[1.2 术语定义 3](#_Toc6939562)

[1.2.1 虚拟账户 3](#_Toc6939563)

[1.2.2 支付 3](#_Toc6939564)

[1.2.3 支付工具 3](#_Toc6939565)

[1.2.4 电子支付 3](#_Toc6939566)

[1.3 参考资料 4](#_Toc6939567)

[1.4 相关文档 4](#_Toc6939568)

[二、现有系统描述 4](#_Toc6939569)

[2.1 作业流程 4](#_Toc6939570)

[三、非技术要求 5](#_Toc6939571)

[四、系统环境 5](#_Toc6939572)

[4.1 软件运行环境 5](#_Toc6939573)

[4.2 开发环境 5](#_Toc6939574)

[五、目标系统功能需求 5](#_Toc6939575)

[5.1 用户登录 6](#_Toc6939576)

[5.2 用户注册 6](#_Toc6939577)

[5.3 用户信息查询 7](#_Toc6939578)

[5.4 用户交易查询 7](#_Toc6939579)

[5.5 转账 7](#_Toc6939580)

[5.6 支付 8](#_Toc6939581)

[六、目标系统的其他需求 8](#_Toc6939582)

[6.1 安全性 8](#_Toc6939583)

[6.2 可靠性 8](#_Toc6939584)

[6.3 稳定性 8](#_Toc6939585)

[6.4 故障处理能力 8](#_Toc6939586)

[6.5 特殊需求 8](#_Toc6939587)

## 一、引言

## 1.1 项目的目的与目标

## 1.1.1 目的

“支付系统”要求后端人员设计与编写全套接口，供前端开发人员调用。前端人员设计界面，并调用后端人员提供的接口实现整个支付系统供用户使用。

## 1.1.2 目标

（1）后端人员为前端开发人员提供安全稳定的接口；

（2）前端人员提供美观的可视化界面；

（3）确保“支付系统”内功能完备。

## 1.2 术语定义

## 1.2.1 虚拟账户

虚拟账户是有用户在网站注册时自行设置，进行交易活动时的一个中介，是根据会计科目设置，具有一定结构，对各种经济业务进行分类和系统、连续的记录，反映资产、负债和所有者权益增减变动的记账实体。支付服务商为客户提供的实现交易资金的收付和暂存管理的支付账户，一般可以通过银行账户进行资金的充值或提取。

## 1.2.2 支付

指货币债券从付款人向收付人的转移，是为清偿商品交换和劳务活动所引起的债权债务关系。

## 1.2.3 支付工具

使货币债券发生转移的发起工具与支付流程。

支付工具的演变过程：实物货币（黄金）—信用货币（纸币）—电子货币。

非现金支付工具包括：支票、转账支付、银行卡、网上银行等。

## 1.2.4 电子支付

电子支付是指消费者、商家和金融机构之间使用安全电子手段把支付信息通过信息网络安全地传送到银行或相应的处理机构，用来实现货币支付或资金流转的行为。电子支付过程中，货币债券以数字信息的方式被持有、处理、接收，由电子支付工具发起实现货币债券的转移。

## 1.3 参考资料

[1]吕云翔，软件工程实用教程，清华大学出版社，2015

[2]吕云翔，刘瀚诚，刘天毅，软件工程项目实训教程，清华大学出版社，2016

[3]Roger S. Pressman，Bruce R. Maxim 著，软件工程：实践者的研究方法（原书第8版），郑人杰等译，机械工业出版社，2016

[4]Ian Sommerville著，软件工程（原书第10版），彭鑫等译，机械工业出版社，2018

[5]张海藩，吕云翔，软件工程（第4版），人民邮电出版社，2013

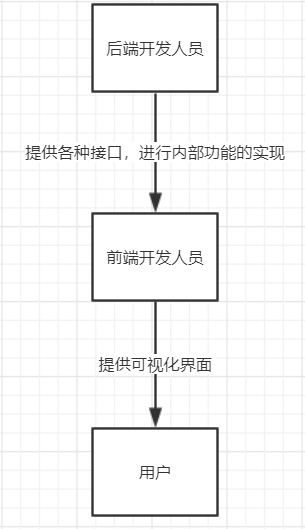
## 1.4 相关文档

“支付系统”的《软件开发计划书》；需求规格说明书编写指南。

## 二、现有系统描述

## 2.1 作业流程

“支付系统”的作业流程如图2.1所示：



**图2.1 “支付系统”的作业流程**

## 三、非技术要求

（1）系统开发周期为两个月左右。开发流程为：需求分析、设计、编码实现、单元测试、集成和系统测试、交付。

（2）要交付的工作产品有：软件开发计划书、需求规格说明书、软件设计说明书、测试报告、用户使用说明书、部署文档、源代码、可执行程序（拥有可视化界面）。

## 四、系统环境

## 4.1 软件运行环境

Ubuntu 16.04 TLS 64位

Microsoft SQL Server 2017

apache-tomcat 9.0.19

JRE 1.8.0\_201

## 4.2 开发环境

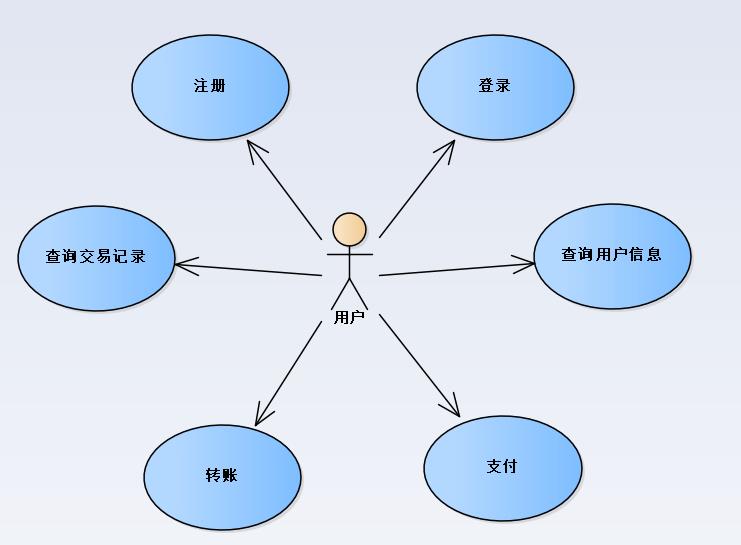
IntelliJ IDEA

Microsoft SQL Server 2017

JDK 1.8.0及以上

## 五、目标系统功能需求

支付系统的用例图如图5.1所示，对用例的说明如表5.1-表5.14所示。



**图5.1 支付系统的用例图**

## 5.1 用户登录

**表5.1 用户登录**

|  |  |
| --- | --- |
| **功能名称** | 用户登录 |
| **功能描述** | 用户在平台上通过用户名与密码进行登录操作。 |
| **输入项** | 用户名（string）、密码（string） |
| **处理描述** | 输入用户名，密码及验证码后点击登录按钮进行登录，前端首先进行验证码验证，如果验证失败则直接提示登录失败并清空验证码框。前端将用户名与密码传递给后端，后端调用数据库进行验证，并返回相应值，根据登录情况进行页面跳转或刷新。 |
| **输出项** | 用户ID（int,-1表示用户登录失败） |

## 5.2 用户注册

**表5.2 用户注册**

|  |  |
| --- | --- |
| **功能名称** | 用户注册 |
| **功能描述** | 用户根据提示输入姓名、手机号、邮箱等信息进行注册开户，验证信息正确后传回后端，进行账户创建，注册成功后自动跳转到用户信息页面。 |
| **输入项** | 用户名（string）、密码（string）、用户姓名（string）、用户手机号（string）、用户邮箱（string）、用户身份凭证（string） |
| **处理描述** | 用户点击注册后将注册信息传回后端，由后端添加用户信息，同时为该用户创建并添加账户，如果用户名被占用则注册失败，返回结果。 |
| **输出项** | 用户ID（int,-1表示用户登录失败） |

## 5.3 用户信息查询

**表5.3 用户信息查询**

|  |  |
| --- | --- |
| **功能名称** | 用户信息查询 |
| **功能描述** | 用户进行基本信息和账户信息查询。 |
| **输入项** | 用户ID（int） |
| **处理描述** | 进入页面后自动向后端数据库发送当前用户ID及信息查询请求，将返回的数据进行解析后进行可视化输出。 |
| **输出项** | 用户信息 |

## 5.4 用户交易查询

**表5.4 用户交易查询**

|  |  |
| --- | --- |
| **功能名称** | 用户交易查询 |
| **功能描述** | 对用户交易信息进行约束查询，约束涉及交易时间和交易类型。 |
| **输入项** | 用户ID（int）、查询起始日期（string符合Date格式）、查询终止日期（string符合Date格式）、交易类型（int, 0充值、1提现、2转账与消费） |
| **处理描述** | 系统通过日期进行约束范围内的账户流水查询，将流水查询信息处理后返回。 |
| **输出项** | 交易流水信息 |

## 5.5 转账

**表5.5 转账**

|  |  |
| --- | --- |
| **功能名称** | 转账 |
| **功能描述** | 用户进行转账操作。 |
| **输入项** | 付款方ID（int）、收款方ID（int）、转账金额（double）、 |
| **处理描述** | 如果转账金额大于付款方账户可用余额，则返回false。否则生成交易ID，将付款方ID、收款方ID、交易时间、交易ID、交易类型、交易金额等信息存入交易信息表和区块链，最后返回交易状态。 |
| **输出项** | 交易状态（boolean） |

## 5.6 支付

**表5.6 支付**

|  |  |
| --- | --- |
| **功能名称** | 支付 |
| **功能描述** | 用户进行支付操作。 |
| **输入项** | 付款方ID（int）、收款方ID（int）支付金额（double）、订单编号（int） |
| **处理描述** | 如果付款金额大于账户可用余额，则返回false，否则生成交易ID，减少账户余额，并将付款方ID、收款方ID、交易时间、交易ID、交易类型、交易金额等信息存入交易信息表和区块链，最后返回交易状态。 |
| **输出项** | 交易状态（boolean） |

## 六、目标系统的其他需求

## 6.1 安全性

只向前端提供接口，内部逻辑在后端完成。

## 6.2 可靠性

程序正常运行时不会出现不可控的未知错误。

## 6.3 稳定性

该系统部署后，在硬件设备和软件条件没有发生变化的情况下，能够一直保持运行状态，保证各接口功能的正常运行。

## 6.4 故障处理能力

可能发生的故障为系统意外崩溃或网络波动，在这种情况下应能将正在操作的信息进行保存，防止数据丢失。

## 6.5 特殊需求

（1）进度需求：系统的阶段进度要求。

（2）资金需求：服务器搭建。

（3）运行环境需求：参见第四章。