**需求规约文档**

修订版历史

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <05日/10月/2022年> | <1.0> | < T18组ATM机模拟项目需求规约文档> | <宣傲> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 1.简介

## 1.1    目的和范围

该文档处于该软件的问题提出阶段，该阶段的关键任务是确切地定义用户要求解决的问题，也就是确定问题的性质，工程的目标和规模。该系统规约说明书的目的是：让分析员把自己对问题的明确认识写成书面报告，提交用户和使用部门的负责人审查，以检验分析员对问题的理解是否正确。为以后的工作确定方向，所以这份报告是非常重要的，这对确保今后开发工作的成功是至关重要的。

## 1.2    定义、首字母缩写词和缩略语

账号：在银行中，事务应用的单个账号。账号可以为多种类型，但是至少包括支票和存款。每个顾客可以拥有多个账号。

ATM:一个工作站终端，使得顾客能够使用现金卡在ATM上进行自己的事务处理。ATM同顾客进行交互，收集事务信息，并发送事务信息到中心计算机，由中心计算机确认和处理信息之后，将现金通过ATM提供给顾客。

银行:一个金融机构，负责保存顾客的账号信息，执行授权卡通过ATM网访问账号。

银行计算机:银行拥有的计算机，同ATM网和银行自己的现金工作站进行交互。银行可以拥有自己的内部计算机网处理账号，但是我们只关心同网络进行交互的计算机。

现金卡:每张卡提供给一个银行客户，授权客户可以使用ATM机访问自己的账号。每张卡包含一个银行代码和一个卡号，银行代码根据信用卡的国际标准进行编码，卡号确定卡能够访问的账户。一张卡不能访问客户的所有账户。每张卡只能有一个持卡人，但是多个复本可能存在，所以必须考虑从不同的ATM机同时使用相同卡的行为。

客户:拥有银行的一个或者多个账号的人。客户可以包括一个人或者多个人，或者是公司。相同的人，拥有不同银行的账号被认为是不同的客户。

事务:对单个客户账号的单个完整的操作请求。

## 1.3     概述

ATM应用系统为更进一步的提供银行交易服务，实现24小时不间断自助服务，对提高银行间的业务竞争能力有很大的帮助。项目目标是实现一个银行业务自动取款系统，客户通过ATM系统可以实现24小时的随时银行业务，其中包括客户自动取款存款功能，账户信息查询功能，信用卡密码更改功能。

# 2.整体说明

当用户用现金兑换卡插入ATM机之后，ATM就与用户交互，以获取有关这次事务的信息，并与中央计算机交换关于事务的信息，首先，ATM要求用户输入密码，接下来ATM把从这张卡读到的信息以及用户输入的密码传给中央计算机，请求中央计算机核对这些信息并处理这次事务，中央计算机根据卡上的分行代码确定这次事务与分行的对应关系，并且委托相应的分行计算机验证用户密码，如果用户输入的密码正确，ATM就要求用户选择事务类型(取款和查询等)。当用户选择取款时，ATM请求用户输入取款额，最后，ATM从现金出口吐出现金，并且打印出帐单交给用户。

# 3.具体需求

## 3.1功能性

## 系统需要有确认用户(系统检验客户个人账户是否与客户的账户上记载的账户一致)功能，账户要有密码保护功能，只有输入相应的密码才可以进入该账户，登陆界面中密码框须使用‘\*’保护。

## 用户登录后，进入服务选择界面，浏览各种服务，选择所需要的服务。服务包括(取款，存款，查询余额，修改密码).

**一、取款：**

1进入取款后选择账户类型（活期账户，定期账户，信用卡帐户），选好账户类型后，提示该ATM机一次能取款的最大金额－2000元

2该用户的一个帐户（用户可能有多个帐户）一天取款累计的最大金额为50000。

3用户选择“取款”后，选取系统界面上的固定金额（500，1000，1500，2000），或是输入要取的金额，进行取款

4若是取款金额大于余额，提示余额不足

5若是所取金额大于ATM机一次能取款的最大金额2000，给出提示，重新输入取款金额或是选择退卡

6若是一天所取累计金额大于该帐户一天取款累计金额（50000）的限制，给出提示信息，重新输入取款金额或是选择退卡

7信用卡帐户可透支取款，透支额为2000元，其他帐户不允许透支取款

8取款后，给出提示选择退卡或是继续服务

9信用卡帐户取款，业务规则无（不用实现）

10完成后用户可选择退卡或是继续选择服务

二、**存款：用户选择“存款”后，输入要存的金额，进行存款。存的钱必须是面值100一张的。存款后，ATM给出提示选择**退卡或是继续服务。

**三、查询余额：**

1选择帐户后，进入查询余额后，显示界面信息包括如下：

2完成后用户可选择退卡或是继续选择服务

**四、转账：**

1进入转账后，（用下拉框，给出该客户所有帐户）用户选择转帐帐户源（该客户的一个帐户），及转帐目标账户（该客户的另一个帐户），以及转帐数额

2如转帐数额大于该帐户余额，给出提示信息，要求用户重新输入转帐数额

3转账服务按所转账的1%收取手续费，从余额中扣除，（若是余额不足，转账服务取消），转账一次的金额不能超过1万元

4完成后用户可选择退卡或是继续选择服务

5转帐交易包括两个交易事务，即从一帐户中取款和另一帐户中取款，这两个交易事务（Transaction）信息须记录到数据库。

**五、修改密码：**

1进入修改密码选项，提示输入新密码，再次输入新密码进行确认修改

2修改密码完成后选择退卡或是继续服务

**六、查询历史事务记录：**

1进入查询历史交易选项，交易指取款、转帐、汇入帐户款等，应提供不同查询条件（例如最近的交易、近一个月交易，过去一年交易等），查询结果应包括交易类型、发生时间、数额）

## 3.2可用性

普通用户和高级用户要高效地执行特定操作不需要培训

符合公认的可用性标准： **CIF，ISO 13407，ISO 9241**

## 3.3可靠性

规定的时间:24小时运行。

平均故障间隔时间： 50天。

平均修复时间：1天。

精确度：100%

## 3.4性能

对事务的响应时间（平均：2秒、最长：5秒）

## 3.5可支持性

多操作系统：WINDOWS XP ，WINDOWS CE .NET

采用x86或非x86芯片组和配备不少于 64 MB 的RAM内存的单功能设备

**广泛的CPU选择：**ARM，MIPS，SHx，X86

**小内存体积**

## 3.6设计约束

开发环境：  
    OS: Windows  
    IDE: eclipse   
架构设计：  
    主要是C/S三层结构：1表示层2业务逻辑层3数据存储层。1-2之间采用Java RMI，实现分布式服务；2－3直接采用JDBC衔接。  
安全设计：  
    在本次开发后期可以加入MD5算法对用户密码进行加密传输  
    在企业级应用中，登录验证用户时应该考虑是用JAAS  
      
数据库设计：  
    查询用户可用帐户的时候连接数据库，返回可用帐户信息  
    这里采用分次连接数据库的方式主要是考虑到不要一次返回较多的数据量，减少用户的等待。但是分次连接的方式将加大应用服务器和数据库服务器的负载：  
    (1) 应用服务器要负责数据库连接对象的管理，分步的方式早成了数据库连接对象过多。  
    (2) 数据库服务器要连接多次，负载加大  
    如果本系统在实际使用中出现数据库连接负载过大，可以考虑用数据库连接池加以改进。  
      
客户端界面：  
    1. 采用Java SWT开发，实现GUI的本地化。

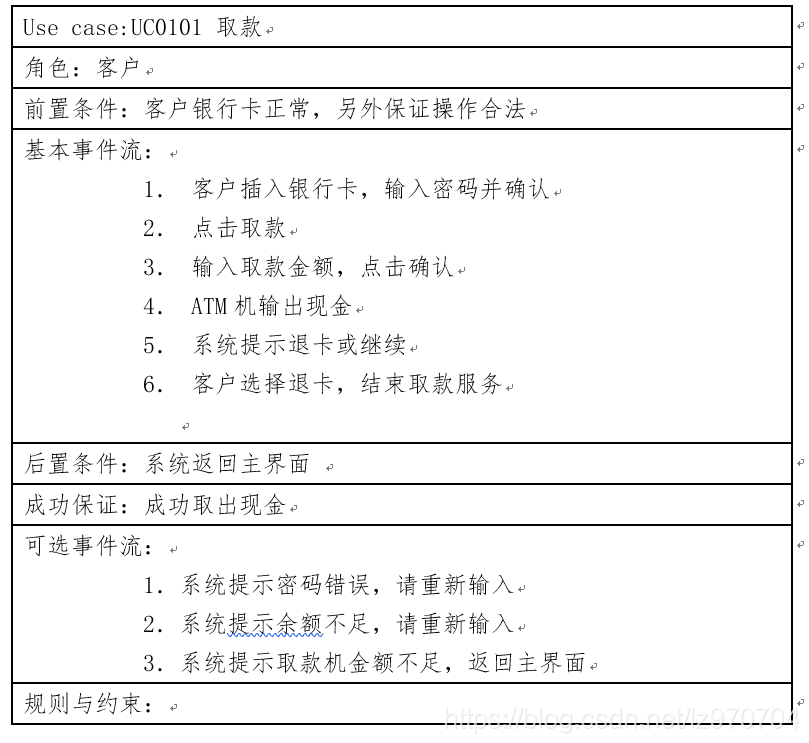
# 4整体流程

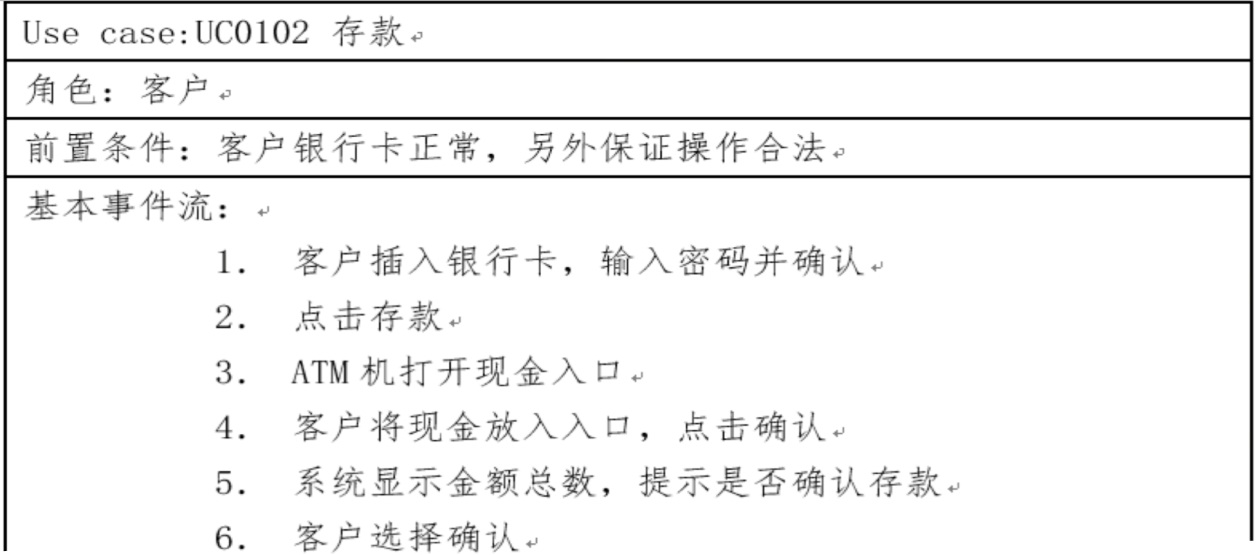
**5用例概述**

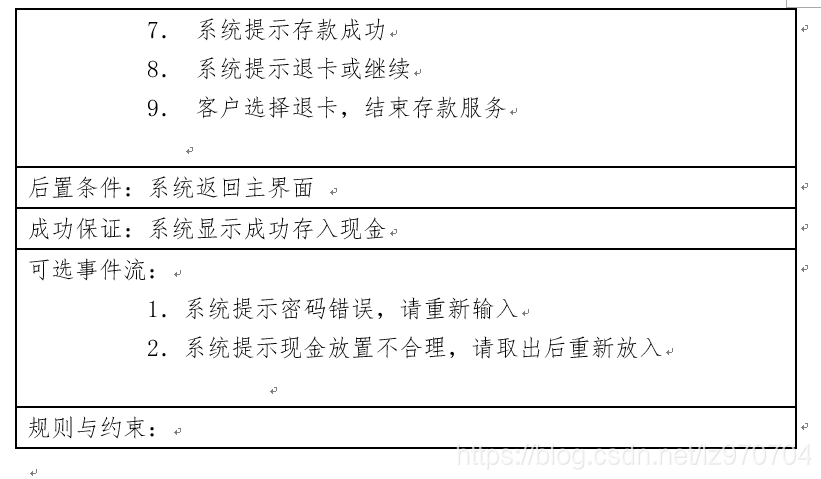


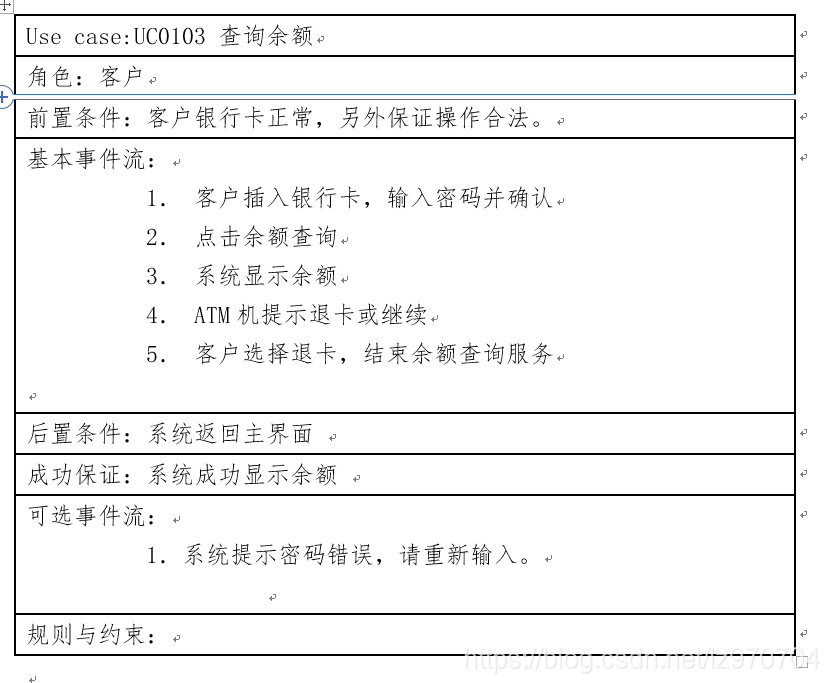


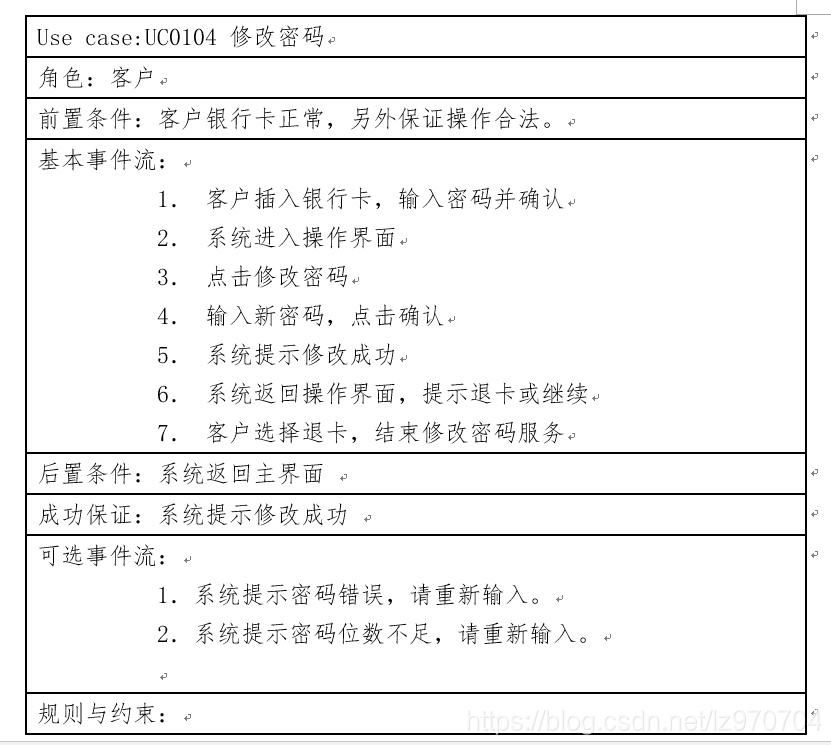
**6用例详述**

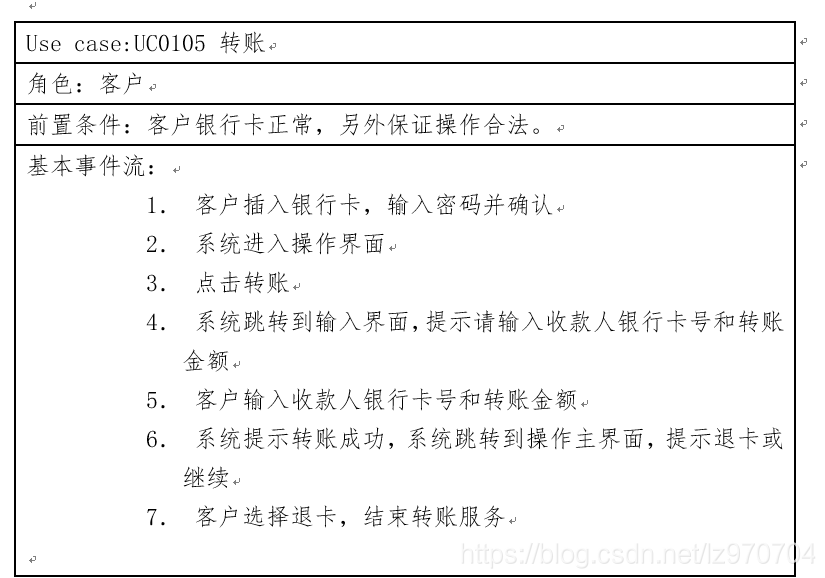
****

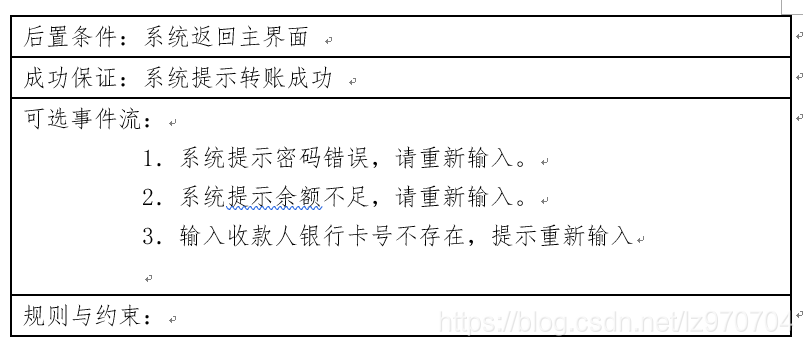


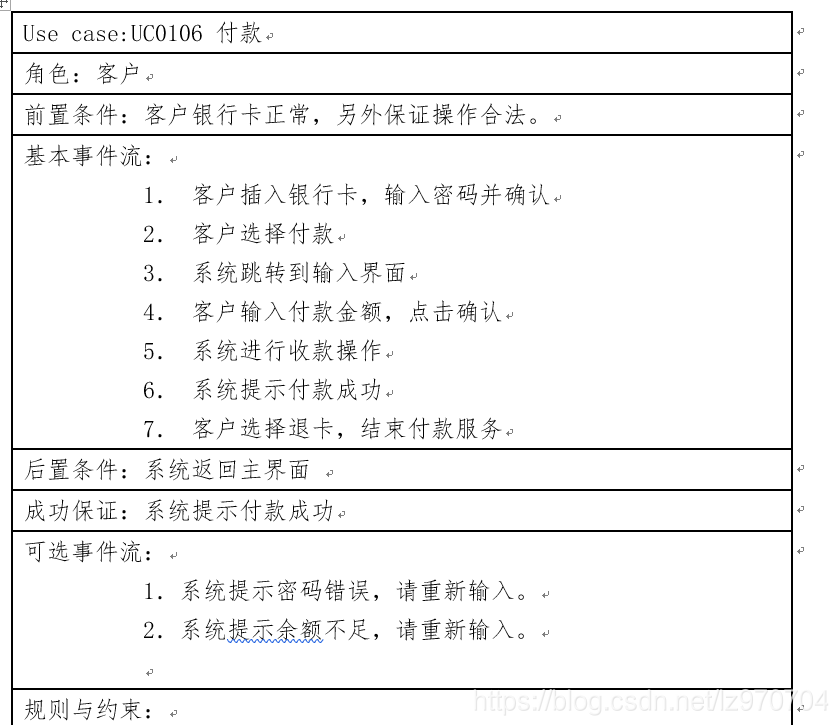
****

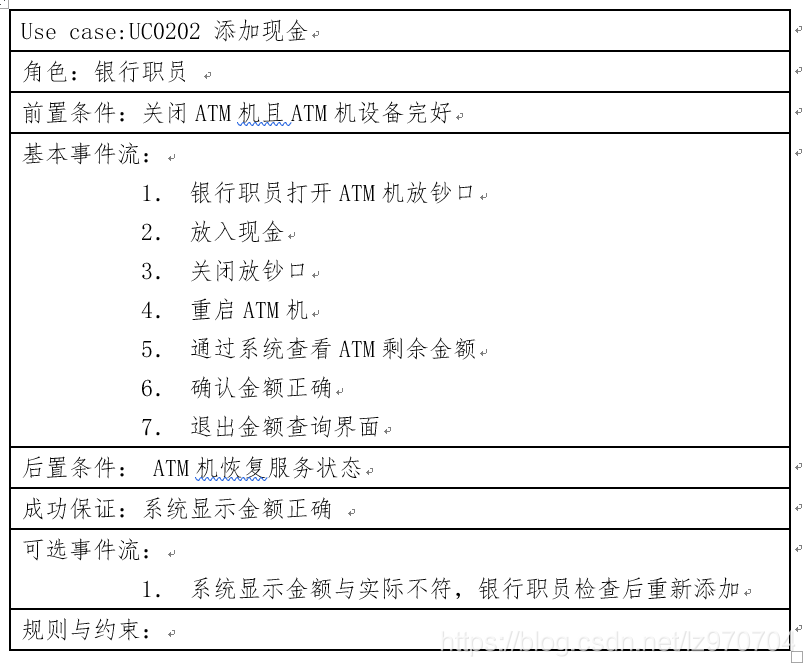
****

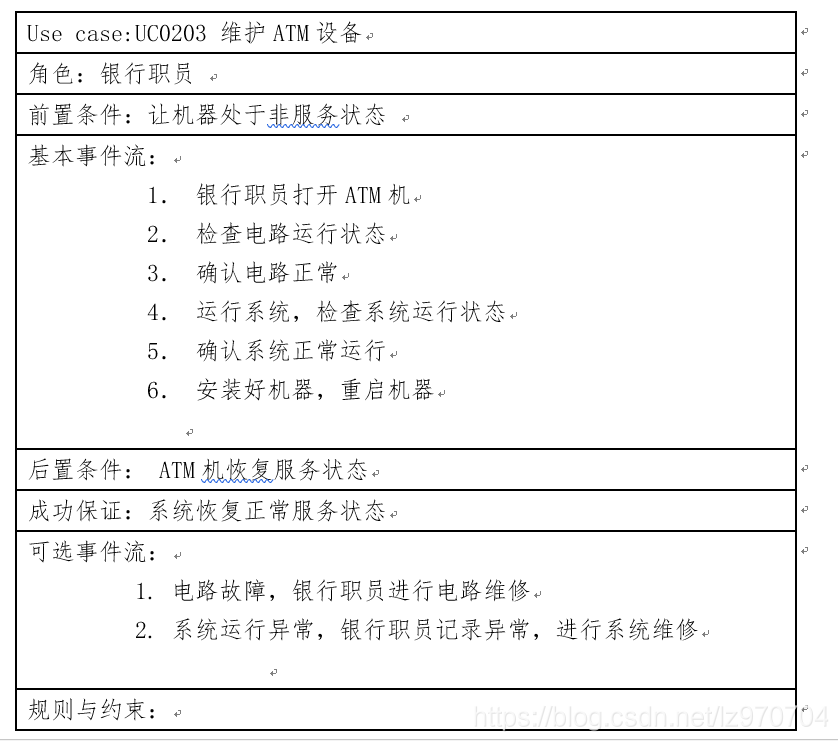
****

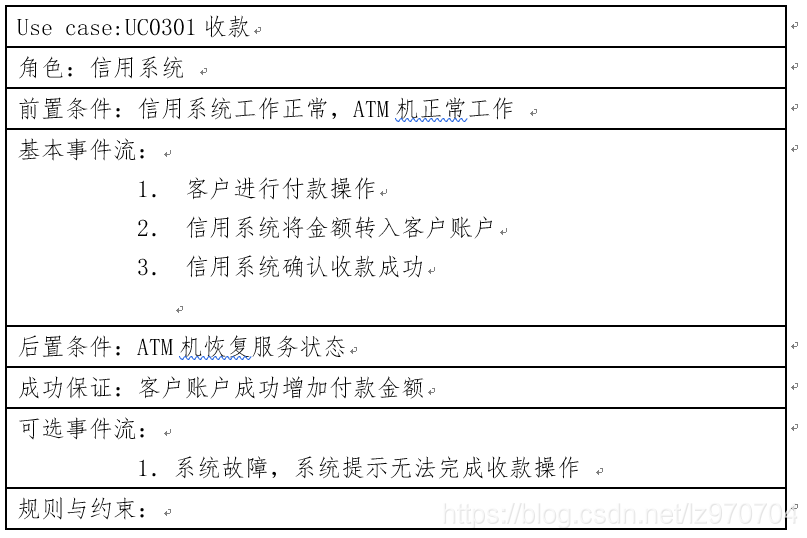
****

****

****

****

****

****

**7、用例图**

