题目：

用java编程实现银行ATM自动存取款柜员机程序。

具体要求：

1. 用户登录自己的银行账户（此处用手动输入账户来代替插卡动作）并输入密码来到达业务操作页面。当输入密码错误次数超过三次，该用户即被冻结
2. 实现基本的存取款功能，此处具体规范可取100,300,500，1000数额，同时可以自己手动输入取钱数额，但是不可以超过2000元每次，不可透支（即不能够取钱总数大于自己的余额），不能有负数或零等。存款时只接受100的整数倍的钱。
3. 可以进行余额查询并有利息更新，并能够查询交易明细清单。

需求规格说明书

[1．引言 1](#_Toc506358922)

[1.1编写目的 1](#_Toc506358923)

[1.2项目背景 2](#_Toc506358924)

[1.3定义 2](#_Toc506358925)

[1.4参考资料 2](#_Toc506358926)

[2．任务概述 2](#_Toc506358927)

[2.1目标 2](#_Toc506358928)

[2.2运行环境 2](#_Toc506358929)

[2.3条件与限制 2](#_Toc506358930)

[3．数据描述 3](#_Toc506358931)

[3.1静态数据 3](#_Toc506358932)

[3.2动态数据 3](#_Toc506358933)

[3.3数据库介绍 3](#_Toc506358934)

[3.4数据词典 3](#_Toc506358935)

[3.5数据采集 3](#_Toc506358936)

[4．功能需求 3](#_Toc506358937)

[4.1功能划分 3](#_Toc506358938)

[4.2功能描述 3](#_Toc506358939)

[5．性能需求 3](#_Toc506358940)

[5.1数据精确度 3](#_Toc506358941)

[5.2时间特性 3](#_Toc506358942)

[5.3适应性 3](#_Toc506358943)

[6．运行需求 3](#_Toc506358944)

[6.1用户界面 3](#_Toc506358945)

[6.2硬件接口 3](#_Toc506358946)

[6.3软件接口 3](#_Toc506358947)

[6.4故障处理 3](#_Toc506358948)

[7．其它需求 3](#_Toc506358949)

# 1．引言

## 1.1编写目的

本次软件开发是一个模拟银行ATM自动柜员机软件，撰写需求说明书是为了后期开发的需要。需求说明明确了整个工程的目标，从整个工程的角度对一个ATM机所需求的功能进行了分析。本需求说明书主要是方便组内成员对整个工程的掌握和了解，促进开发的效率，避免盲目性。

## 1.2项目背景

本项目受委托，由两人组成的研发团队进行开发。

## 1.3定义

【列出文当中所用到的专门术语的定义和缩写词的原文。】

## 1.4参考资料

1项目开发计划：

需求分析说明书

概要设计说明书

详细计设说明书

1. 资料

《软件工程--原理、方法、应用》第二版 史济民著

# 2．任务概述

## 2.1目标

用java编程实现银行ATM自动存取款柜员机程序。

具体要求：

1．用户登录自己的银行账户（此处用手动输入账户来代替插卡动作）并输入密码来到达业务操作页面。当输入密码错误次数超过三次，该用户即被冻结

2．实现基本的存取款功能，此处具体规范可取100,300,500，1000数额，同时可以自己手动输入取钱数额，但是不可以超过2000元每次，不可透支（即不能够取钱总数大于自己的余额），不能有负数或零等。存款时只接受100的整数倍的钱。

1. 可以进行余额查询并有利息更新，并能够查询交易明细清单。

## 2.2运行环境

操作系统：Microsoft Windows7

支持环境：JDK虚拟机

数 据 库： MySQL

## 2.3条件与限制

|  |  |  |
| --- | --- | --- |
| 约束源 | 约束 | 理由 |
| 开发人员人数 | 2人团队 | 人数限制了程序不能设计的很完善 |
| 开发语言 | Java语言 | Java语言是一种高级开发语言，便于代码编写 |
| 开发工具 | Myeclipse | Myeclipse支持Java界面开发， |
| 软件架构 | Hibernate架构 | 比那与对数据库的链接和处理 |
| 界面 | 美观，有好的用户体验 | 漂亮的界面和用户体验决定着软件能否推广 |
| 可靠性 | 程序运行稳定，数据可靠，能够容错处理 | 程序的稳定性决定着软件的质量，可靠数据保证数据的准确性。容错处理保证用户的错误操作不会引起程序的出错。 |
| 安全型 | 数据库的备份和保密 | 错误的操作可能引起数据的丢失，对银行造成损失。 |
| 高效性 | 操作的简练和程序运行流畅 | 节省用户的时间 |

# 3．数据描述

## 3.1静态数据

|  |  |
| --- | --- |
| 条目 | 说明 |
| 顾客账户数据 | 顾客在银行开好账户，设置好密码 |
| 定额取款金额数据 | 设置好定额取款金额节省自定义的时间 |

## 3.2动态数据

|  |  |
| --- | --- |
| 条目 | 说明 |
| 顾客账户余额数据 | 根据顾客存取钱而动态的变化 |
| 用户自定义存取款金额数据 | 用户存取金额不定 |

## 3.3数据库介绍

【给出使用数据库的名称和类型。】

|  |  |
| --- | --- |
| 数据库名称 | MySql |
| 类型 | Mysql+hibernate+myEclipse |

## 3.4数据词典

## 3.5数据采集

数据的采集大概有以下几种：

1. 系统的初始化的一部分
2. 根据时间的动态添加
3. 顾客的数据输入

# 4．功能需求

## 4.1功能划分

|  |  |
| --- | --- |
| 功能 | 描述 |
| 用户存款 | 金额可自定 |
| 用户取款 | 100的整数倍，且小于总钱数 |
| 查询余额 | 账户余额根据存取款动作动态改变 |
| 查询交易明细 | 交易记录 |

## 4.2功能描述

**用例图：**

**4.3.1 用户总用例图：**



**4.3.2 取款用例图：**

****

**4.3.3查询账户用例图：**



**4.3.4 ATM机功能用例图：**



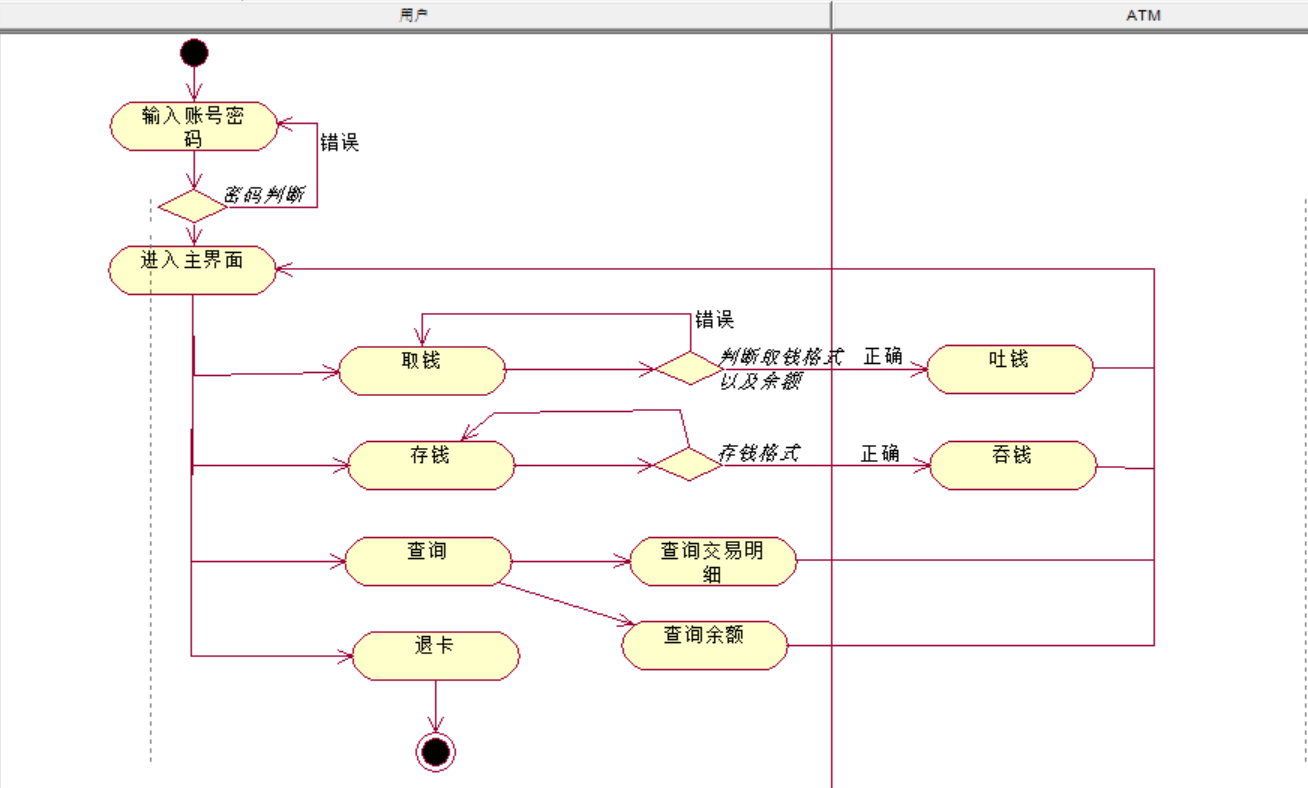
新增用例描述：

|  |
| --- |
| 用例：用户取钱 |
| 用例ID：1 |
| 简要描述：  用户取款。 |
| 第一参与者：  用户 |
| 第二参与者：  ATM |
| 前置条件：  ATM必须有钱，账户必须有余额 |
| 主流：   1. 用户进入界面登陆之后，选择取钱功能。 2. 输入取钱金额，可自定义，可选择。 3. 如果ATM有钱，则吐钱。 |
| 后置条件：  账户和ATM余额改变 |
| 附流：  无 |

|  |
| --- |
| 用例：用户存钱 |
| 用例ID：2 |
| 简要描述：  用户输入存钱的数目，并存入 |
| 第一参与者：  用户 |
| 第二参与者：  ATM |
| 前置条件：  无 |
| 主流：   1. 用户输入选择存钱功能 2. 输入存钱金额 |
| 后置条件：  账户和ATM余额改变 |
| 附流：  无 |

|  |
| --- |
| 用例：查询交易明细余额 |
| 用例ID：3 |
| 简要描述：  用户查询交易明细 |
| 第一参与者：  用户 |
| 第二参与者：  ATM |
| 前置条件：  用户已经正确登陆。 |
| 主流：   1. 用户点击查询功能 2. 出现查询余额和交易明细两种功能 3. 点击查询 |
| 后置条件：无 |
| 附流：  无 |

**活动图（业务流程图）：**



**领域模型（业务概念图）：**



# 5．性能需求

## 5.1数据精确度

**数据的精确度为： 100%**

## 5.2时间特性

对所有数据的增删改查都是运用sql语句，运用封装好的方法，响应时间非常短，数据转换与传输时间也很短。

【如响应时间、更新处理时间、数据转换与传输时间、运行时间等。】

## 5.3适应性

代码的适应性比较强，按照统一的封装模式，DAO+Control+Factory，在数据操作上统一，团队整合时也比较方便。

【在操作方式、运行环境、与其它软件的接口以及开发计划等发生变化时，应具有的适应能力。】

# 6．运行需求

## 6.1用户界面

|  |  |
| --- | --- |
| 屏幕格式 | 借鉴简单ATM机系统 |
| 报表格式 | Table格式 |
| 菜单格式 | 图形化的按键式的 |

【如屏幕格式、报表格式、菜单格式、输入输出时间等。】

## 6.2硬件接口

## 6.3软件接口

## 6.4故障处理

1. 输入信息的处理
2. 输出信息的处理
3. 读数据库异常的处理
4. 软件异常的恢复

等等

# 7．其它需求

【如可使用性、安全保密、可维护性、可移植性等。】