

**软件工程概要设计报告**

课程名称： 软件工程与实践

所在系别： 信息与软件工程学院

学 期： 2022-2

课题名称： “智帐”记账软件

指导教师： 邱元杰

学生信息：

|  |  |  |
| --- | --- | --- |
| 序号 | 学号 | 姓名 |
| 1（组长） | 2021090920004 | 詹锐艺 |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |

目录

[1. 引言 3](#_Toc128664354)

[1.1 编写目的 3](#_Toc128664355)

[1.2 项目风险 3](#_Toc128664356)

[1.3 预期读者和阅读建议 3](#_Toc128664357)

[1.4 参考资料 4](#_Toc128664358)

[2. 设计概述 4](#_Toc128664359)

[2.1 限制和约束 5](#_Toc128664360)

[2.2 设计原则和设计要求 5](#_Toc128664361)

[3. 系统逻辑设计 6](#_Toc128664362)

[3.1 系统组织设计 6](#_Toc128664363)

[3.2 系统结构设计 6](#_Toc128664364)

[3.3 设计类结构 8](#_Toc128664365)

[3.4 关键处理过程或者算法的设计 8](#_Toc128664366)

[3.5 系统接口设计 9](#_Toc128664367)

[3.5.1 系统接口表 9](#_Toc128664368)

[4. 系统出错处理设计 11](#_Toc128664369)

[4.1 系统出错处理表 11](#_Toc128664370)

[4.2 维护处理过程表 12](#_Toc128664371)

[5. 数据库设计 14](#_Toc128664372)

[6. 词汇表 15](#_Toc128664373)

# 引言

本份概要设计报告旨在说明记账软件系统的整体设计，包括系统的功能模块划分、系统架构设计等方面。该报告主要面向开发团队和管理人员，以便更好地理解系统的功能和架构，并为后续的详细设计和开发提供参考。

本文档的主要内容有：编写目的、项目风险、、阅读建议、设计概述、系统逻辑设计、错误处理及维护、数据库设计以及词汇表。

需要说明的是，本文档并不是最终的详细设计文档，其中的一些细节和具体实现方式还需要进一步讨论和细化。但本文档中的设计思路和框架，对于后续的详细设计和开发具有指导和参考意义。

## 编写目的

说明这份软件系统概要设计报告是基于哪份软件产品需求规格说明书编写的，开发这个软件产品意义、作用、以及最终要达到的意图。通过这份软件系统概要设计报告详尽说明了该软件产品的软件结构，包括数据库结构和出错处理，从而对该软件产品的结构的描述。

如果这份软件系统概要设计报告只与整个系统的某一部分有关系，那么只定义软件系统概要设计报告中说明的那个部分或子系统。

该概要设计基于智账软件的需求规格说明书编写。

意义和作用：记账软件是一种帮助用户记录和管理个人或企业财务状况的工具。在日常生活中，人们需要对自己的收支情况进行记录和管理，以便更好地掌握自己的财务状况，制定更好的消费计划和理财策略。记账软件能够让用户更加方便快捷地记录自己的消费情况，并对数据进行分类、统计和分析，从而更好地了解自己的财务状况，帮助用户更好地管理自己的财务。

最终要达到的意图：

1. 提供方便快捷的消费记录功能，让用户可以随时随地记录自己的消费情况，包括消费时间、地点、用途等信息。
2. 提供数据分类、统计和分析功能，让用户可以更好地了解自己的财务状况，例如支出占比、消费趋势等等。
3. 提供良好的用户体验，包括简洁明了的界面设计、操作便捷性等。

## 项目风险

任务提出者：需求变更、需求不明确、需求矛盾等风险。主要风险是未能清晰地表达需求，导致软件开发过程中产生不必要的返工和重复劳动。

软件开发者：技术风险、进度风险和质量风险等。主要风险是技术实现难度较大，导致无法按照需求规定的时间节点完成开发任务。

产品使用者：数据安全性、用户体验和功能缺陷等。主要风险是软件使用过程中出现的未知问题或错误，导致数据损失或其他问题。此外，如果软件的用户体验不佳，也会影响到产品使用者的体验和满意度。

## 预期读者和阅读建议

列举本软件系统概要设计报告所针对的各种不同的预期读者，例如，可能的读者包括：

预期读者：用户、开发人员、文档编写人员、测试人员

* 用户；阅读14.4顺序图、15.1和15.2错误处理及维护、17词汇表
* 开发人员；13设计概述、14系统设计、16数据库设计
* 测试人员；15出错处理及维护
* 文档编写人员；12.1编写目的、13设计概述

## 参考资料

列举编写软件产品概要设计报告时所用到的参考文献及资料，可能包括：

* 本项目的合同书；
* 上级机关有关本项目的批文；
* 本项目已经批准的计划任务书；
* 用户界面风格指导；
* 开发本项目时所要用到的标准；
* 系统规格需求说明；
* 使用实例文档；
* 属于本项目的其它已发表文件；
* 本软件系统概要设计报告中所引用的文件、资料：
* 相关软件系统概要设计报告：
* 等等。

为了方便读者查阅，所有参考资料应该按一定顺排列。如果可能，每份资料都应该给出：

* 标题名称；
* 作者或者合同签约者；
* 文件编号或者版本号；
* 发表日期或者签约日期；
* 出版单位或者资料来源。

# 设计概述

现有开发条件：本项目需要开发一个安卓记账软件，用户可以登录后记录自己的消费记录，并保存到软件中。软件需要记录消费的时间和地点，并允许用户添加备注，生成用户的总支出，并生成饼状图说明各类消费占比。

需要实现的目标：为用户提供一个方便快捷的记账工具，用户可以通过软件轻松地记录自己的消费记录和支出情况，并通过软件生成的报表快速了解自己的消费情况，从而更好地控制自己的消费。

设计原则：进行概要设计时应该遵循以下设计原则：

1. 可靠性：软件应该能够准确地记录用户的消费记录，保证数据的安全性和稳定性。
2. 可维护性：软件应该易于维护和升级，方便后续的功能扩展和改进。
3. 可扩展性：软件应该具有较强的扩展性，以便在用户需求变化时快速适应。
4. 用户友好性：软件应该具有良好的用户体验，易于操作，符合用户习惯，提高用户使用体验。

设计方法：进行概要设计时应该采用以下设计方法：

1. 模块化设计：将软件系统划分为若干个独立的模块，每个模块实现独立的功能，便于后续维护和升级。
2. 面向对象设计：采用面向对象的设计思想，将软件系统的组成部分抽象成对象，以便于实现复杂的功能和保证软件的可维护性和可扩展性。
3. 数据库设计：采用合理的数据库设计，保证数据的安全性和数据的存储和检索的效率。同时，还需要保证数据的一致性和完整性。
4. UI设计：采用人性化的UI设计，使得软件具有良好的用户体验和易于操作的特点。

## 限制和约束

1. 技术条件：本软件需要运行在安卓平台上，因此需要使用Java开发语言进行开发，并需要了解安卓平台的开发框架和技术。
2. 资金状况：本项目的资金来源是自身资金，因此需要在资金有限的情况下进行开发，避免不必要的浪费。
3. 开发环境：本项目需要使用开发工具Android Studio，以及一些第三方库来实现某些功能，例如用于生成饼状图的MPAndroidChart库。
4. 时间限制：本项目需要在一个学期内完成，因此需要合理规划开发进度和时间，并且避免不必要的延误。

在以上条件的基础上，本软件的目标是开发一款方便用户进行记账的安卓应用程序，能够记录用户的消费记录、时间地点和备注信息，并能够生成用户的总支出和各类消费占比的饼状图，从而帮助用户更好地了解自己的消费情况和掌握自己的财务状况。同时，本软件需要具备简洁、易用、稳定、安全等特点，以提高用户的使用体验和保证数据的安全。

## 设计原则和设计要求

1. 命名规则：采用规范的命名规则，例如采用驼峰命名法来命名变量、函数、类等，并尽量见名知意，能够方便地理解代码。
2. 模块独立性原则：应该遵循模块独立性原则，即每个模块应该尽可能独立，模块之间的耦合度应该尽可能低，以方便模块的复用和维护。
3. 边界设计原则：在设计系统的边界时，应该考虑清楚边界内外的数据流和控制流，确保边界内的数据和操作都是安全和正确的。
4. 数据库设计规则：在进行数据库设计时，应该遵循一些基本的设计规则，例如数据表之间的关联应该合理、数据表的字段应该具备唯一性等。
5. 必须的安全措施：为了确保系统的安全性，应该采取必要的安全措施，例如密码加密、用户身份验证等。
6. 安全性和保密原则：应该遵循安全性和保密原则，确保用户的数据和隐私不会被泄露。
7. 系统灵活性要求：在系统设计中，应该考虑到系统的灵活性，能够适应不同的用户需求，并且能够方便地进行功能扩展和升级。
8. 系统易操作性要求：为了提高用户的使用体验，系统应该具备易操作性，能够方便地进行操作和交互。
9. 系统可维护性要求：为了方便系统的维护和升级，系统应该具备良好的可维护性，能够方便地进行代码修改和功能扩展。

# 系统逻辑设计

本节内容主要根据软件产品需求规格说明书和软件产品数据字典建立系统的逻辑模型。此种模型暂时与系统的物理因素(例如：计算机、数据库管理系统)无关。它是系统需求与物理实现的中间结构，它的主要结果是建立：类图、系统结构图、系统界面结构图、系统出错处理、以及系统开发技术说明。

## 系统组织设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 子系统编号 | 英文名称 | 中文名称 | 业务职能 | 备注 |
| 1 | user\_manage | 用户管理 | 注册、登录 | 无 |
| 2 | expan\_recore | 消费记录 | 添加、修改 | 无 |
| 3 | expan\_analyse | 消费分析 | 生成饼状图 | 无 |
| 4 | data\_manage | 数据管理 | 保证数据安全 | 无 |
| 5 | sys\_manage | 系统管理 | 运营人员维护 | 无 |

## 系统结构设计

本节将对系统特性作较为详细的描述，并给出系统特性结构图。

系统用例是系统中完成某项具体操作的基本单元，它由入口参数，出口参数以及处理过程等组成，是系统要实现的对象。

系统用例可以具有操作界面，也可以没有操作界面；可以被其它操作界面、或者系统用例调用，也可以调用其它操作界面、非操作界面、或者系统用例；但是不允许递归调用(调用自己)，包括间接递归调用。

系统用例模块映射表说明各个系统用例和各个子模块的关系，即每个子模块实现那些用例或者用例的一部分。

当系统由多个子系统(模块)组成时，每个子系统分别使用一张系统用例模块映射表进行描述。系统用例模块映射表的格式如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号： 1  子系统英文名称：user\_manage  子系统中文名称：用户管理 | | | | | | |
| 用例编号 | 系统用例  英文名称 | 系统用例  中文名称 | 操作功能 | 调用对象 | 被调用  对象 | 备注 |
| 1 | sign\_in | 注册 | 注册新用户 | 操作界面 | 无 | 无 |
| 2 | login\_in | 登录 | 用户登录 | 操作界面 | 无 | 无 |
| 3 | alter\_info | 更改信息 | 更改个人信息 | 操作界面 | 无 | 无 |
| 说明：用户管理子系统：用于管理用户信息，包括用户的注册、登录、个人信息修改等。 | | | | | | |

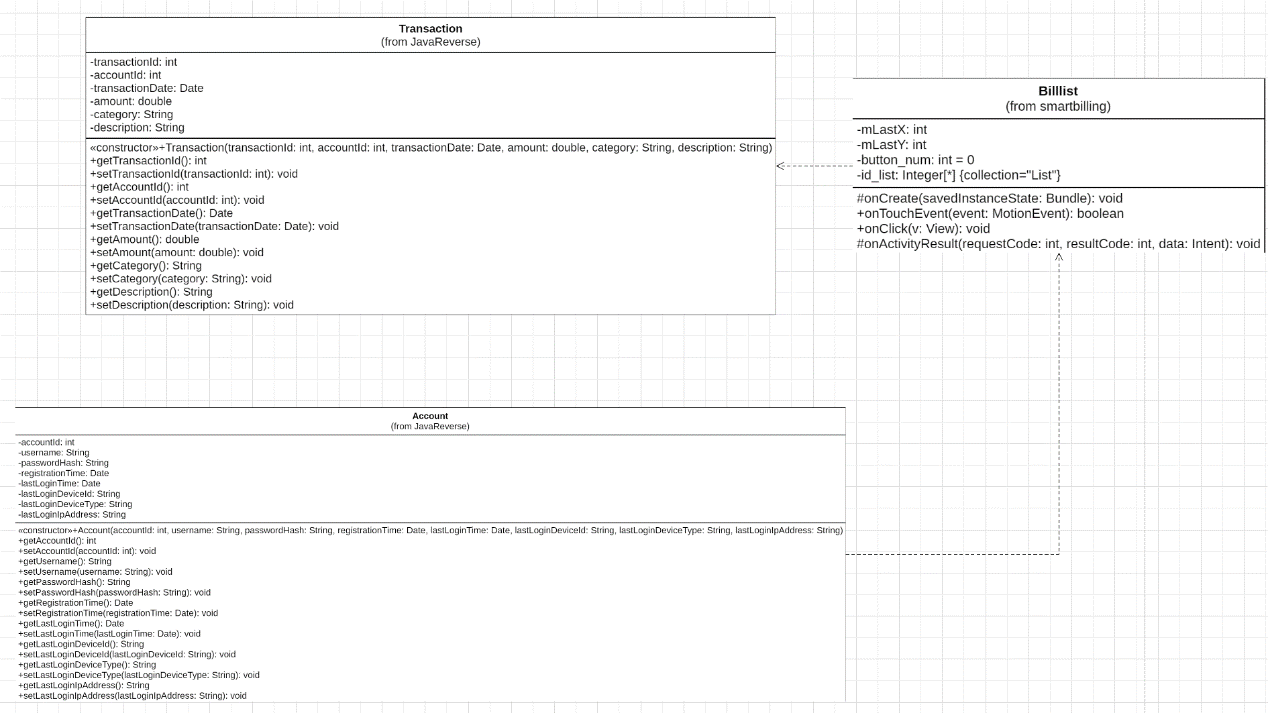
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号： 2  子系统英文名称：enpan\_recore  子系统中文名称：消费记录 | | | | | | |
| 用例编号 | 系统用例  英文名称 | 系统用例  中文名称 | 操作功能 | 调用对象 | 被调用  对象 | 备注 |
| 1 | add\_expan | 添加记录 | 增加消费记录 | 操作界面 | 无 | 无 |
| 2 | dele\_expan | 删除记录 | 删除消费记录 | 操作界面 | 无 | 无 |
| 3 | alter\_expan | 更改记录 | 更改消费记录 | 操作界面 | 无 | 无 |
| 4 | inquir\_expan | 查询记录 | 查找消费记录 | 操作界面 | 无 | 无 |
| 说明：消费记录管理子系统：用于管理用户的消费记录，包括添加、修改、删除、查询等功能。 | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号： 3  子系统英文名称：expan\_analyse  子系统中文名称：消费分析 | | | | | | |
| 用例编号 | 系统用例  英文名称 | 系统用例  中文名称 | 操作功能 | 调用对象 | 被调用  对象 | 备注 |
| 1 | gen\_chart | 生成图表 | 生成饼状图 | 系统 | add\_expan | 无 |
| 2 | classify | 分类 | 对消费分类 | 系统 | add\_expan | 无 |
| 说明：消费分析子系统：用于对用户的消费数据进行分析，生成支出统计报表、消费类别饼状图等功能 | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号： 4  子系统英文名称：data\_manage  子系统中文名称：数据管理 | | | | | | |
| 用例编号 | 系统用例  英文名称 | 系统用例  中文名称 | 操作功能 | 调用对象 | 被调用  对象 | 备注 |
| 1 | backup\_data | 备份数据 | 备份用户数据 | 系统 | 无 | 无 |
| 2 | restore | 恢复数据 | 恢复用户数据 | 系统 | 无 | 无 |
| 说明：数据管理子系统：用于对用户数据进行备份、恢复操作，保证数据的安全性。 | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号： 5  子系统英文名称：sys\_manage  子系统中文名称：系统管理 | | | | | | |
| 用例编号 | 系统用例  英文名称 | 系统用例  中文名称 | 操作功能 | 调用对象 | 被调用  对象 | 备注 |
| 1 | user\_manage | 用户管理 | 管理所有用户 | 系统 | sign\_in | 无 |
| 2 | log\_manage | 日志管理 | 检测日志 | 系统 | 无 | 无 |
| 说明：系统管理子系统：用于软件运营人员对软件进行管理和维护，包括用户管理、日志管理等功能。 | | | | | | |

## 设计类结构



## 关键处理过程或者算法的设计

JWT鉴权的设计

  项目使用maven对第三方库进行管理，在maven的配置文件pom.xml中填写如下依赖，安装 jjwt 库：

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt</artifactId>

<version>0.9.1</version>

</dependency>

接口实际实现如下：

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

import java.util.Date;

import javax.ws.rs.POST;

import javax.ws.rs.Path;

import javax.ws.rs.core.Response;

@Path("/login")

public class LoginResource {

@POST

public Response login(String credentials) {

// 从请求体中获取账号密码

JsonObject jsonCredentials = Json.createReader(new StringReader(credentials)).readObject();

String username = jsonCredentials.getString("username");

String password = jsonCredentials.getString("password");

// 根据账号密码查询数据库

String passwordHash = getPasswordHashFromDatabase(username, password);

if (passwordHash == null) {

return Response.status(Response.Status.UNAUTHORIZED).build();

}

// 生成 JWT Token

String token = Jwts.builder()

.setSubject(username)

.setIssuedAt(new Date())

.setExpiration(new Date(System.currentTimeMillis() + EXPIRATION\_TIME))

.signWith(SignatureAlgorithm.HS512, SECRET\_KEY)

.compact();

// 返回 Token

return Response.ok()

.header("Authorization", "Bearer " + token)

.build();

}

private String getPasswordHashFromDatabase(String username, String password) {

// 从数据库中查询账号密码哈希值

// 此处具体实现见下

}

private static final String SECRET\_KEY = "secret-key"; // JWT 密钥

private static final long EXPIRATION\_TIME = 86400000; // Token 过期时间（1 天）

}

**从Account表取账号密码详细实现**

  先用 maven添加jBCrypt依赖从而实现哈希值的生成

<!-- https://mvnrepository.com/artifact/org.mindrot/jbcrypt -->

<dependency>

<groupId>org.mindrot</groupId>

<artifactId>jbcrypt</artifactId>

<version>0.4</version>

</dependency>

然后具体实现如下：

private String getPasswordHashFromDatabase(String username, String password) {

String passwordHash = null;

try (Connection conn = DriverManager.getConnection(DATABASE\_URL, DATABASE\_USERNAME, DATABASE\_PASSWORD);

PreparedStatement stmt = conn.prepareStatement("SELECT password\_hash FROM Account WHERE username = ?")) {

stmt.setString(1, username);

ResultSet rs = stmt.executeQuery();

if (rs.next()) {

String storedHash = rs.getString("password\_hash");

// 检查密码是否匹配

if (BCrypt.verifyer().verify(password.toCharArray(), storedHash).verified) {

passwordHash = storedHash;

}

}

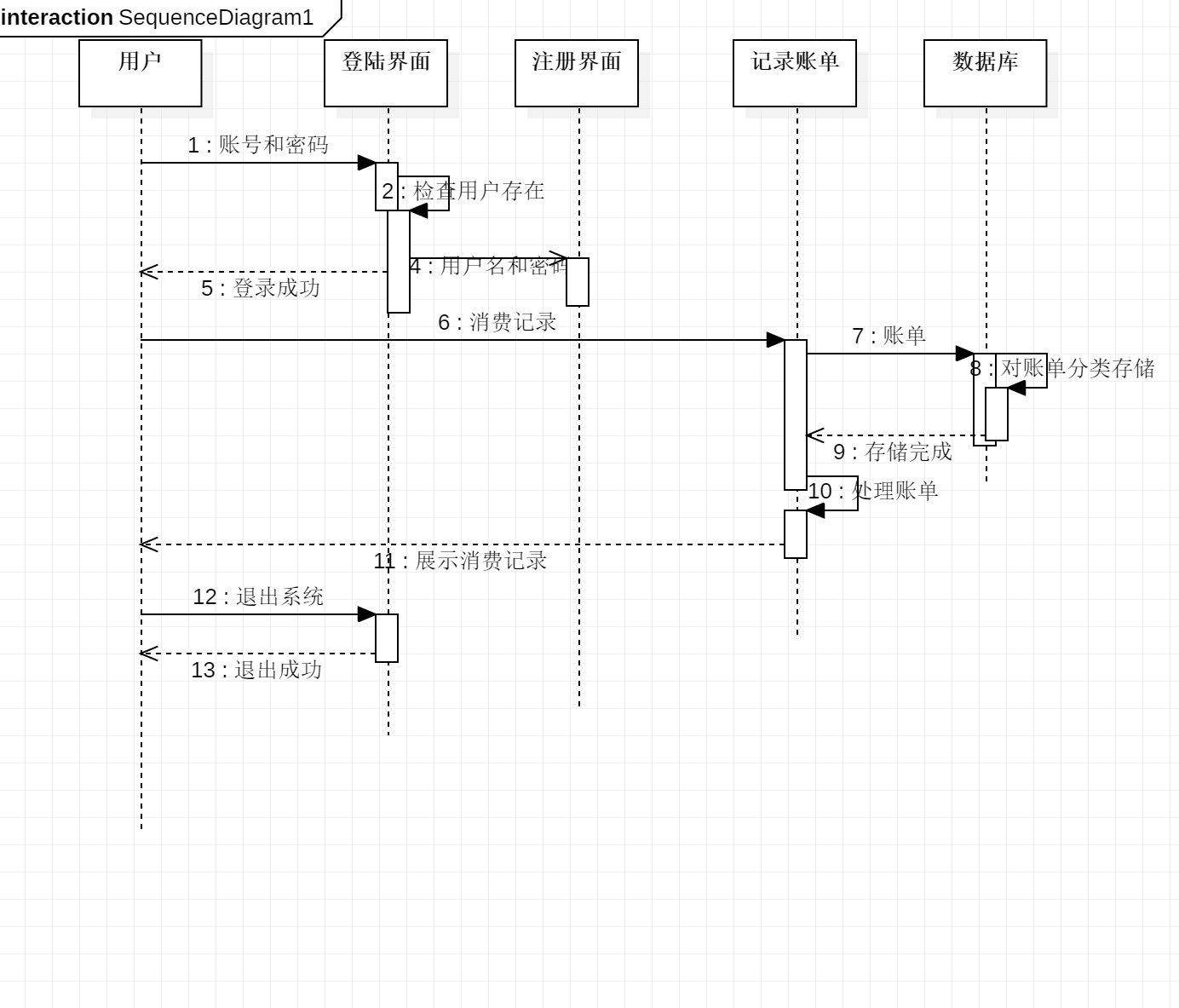
} catch (SQLException e) {

e.printStackTrace();

}

return passwordHash;

}



## 系统接口设计

系统接口是一种非可视的系统界面，在多数情况下，它对用户是透明的。

本节将对系统接口作较为详细的描述，并给出接口说明清单。

### 系统接口表

接口作为系统的一种输入／输出形式，分为网络接口、数据库接口、RS-232串行通讯接口、IEEE—485串行总线接口、并行I/O接口等等多种类型。

对于一些为可视界面服务的接口，例如：打印机接口、显示器接口等，因为这类接口对应用软件是透明的，所以不在本节描述范围内。

当系统由多个子系统(模块)组成时，每个子系统分别使用一张系统接口表进行描述。系统接口表的格式如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：1  子系统英文名称：user\_manage  子系统中文名称：用户管理 | | | | | | |
| 接口  编号 | 接口  名称 | 接口  类型 | 接口  性质 | 接口  速率 | 接口  协议 | 备注 |
| 101 | signup | 注册 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 102 | signin | 登录 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 103 | resert\_password | 重置密码 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：2  子系统英文名称：expan\_recore  子系统中文名称：消费记录 | | | | | | |
| 接口  编号 | 接口  名称 | 接口  类型 | 接口  性质 | 接口  速率 | 接口  协议 | 备注 |
| 201 | add\_bill | 增加记录 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 202 | modify\_bill | 修改记录 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 203 | delete\_bill | 删除记录 | 输入 | 根据网络 | HTTP/HTTPS |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：3  子系统英文名称：expan\_analyse  子系统中文名称：消费分析 | | | | | | |
| 接口  编号 | 接口  名称 | 接口  类型 | 接口  性质 | 接口  速率 | 接口  协议 | 备注 |
| 301 | gen\_chart | 显示图表 | 输出 | 根据网络 | HTTP/HTTPS |  |
| 302 | classify | 分类数据 | 输出 | 根据网络 | HTTP/HTTPS |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：4  子系统英文名称：data\_manage  子系统中文名称：数据管理 | | | | | | |
| 接口  编号 | 接口  名称 | 接口  类型 | 接口  性质 | 接口  速率 | 接口  协议 | 备注 |
| 401 | backup\_data | 备份数据 | 双向 | 根据网络 | HTTP/HTTPS |  |
| 402 | restore | 恢复数据 | 输出 | 根据网络 | HTTP/HTTPS |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：5  子系统英文名称：sys\_manage  子系统中文名称：系统管理 | | | | | | |
| 接口  编号 | 接口  名称 | 接口  类型 | 接口  性质 | 接口  速率 | 接口  协议 | 备注 |
| 501 | user\_manage | 管理用户 | 双向 | 根据网络 | HTTP/HTTPS |  |
| 502 | log\_manage | 管理日志 | 双向 | 根据网络 | HTTP/HTTPS |  |
| 说明： | | | | | | |

# 系统出错处理设计

本节描述系统发生外界及内在错误时，所提供的错误信息及处理方法，它包括系统出错处理表及维护处理过程表。

## 系统出错处理表

本表给出有关出错处理的产生原因、提示信息、以及建议处理方法。

当系统由多个子系统(模块)组成时，每个子系统分别使用一张系统出错处理表进行描述。系统出错处理表的格式如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 子系统编号：1  子系统英文名称：user\_manage  子系统中文名称：用户管理 | | | | | |
| 错误编号 | 错误名称 | 错误原因 | 错误信息 | 处理方式 | 备注 |
| 101 | NetworkError | 网络中断 | 无法连接到网络，请检查网络连接并重试 | 系统记录错误信息并提供重试机制 |  |
| 102 | inputerror | 输入错误 | 输入数据不合法，请重新输入 | 给出明确的错误提示信息，指导用户如何正确输入数据 |  |
| 说明： | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 子系统编号：2  子系统英文名称：expan\_recore  子系统中文名称：消费记录 | | | | | |
| 错误编号 | 错误名称 | 错误原因 | 错误信息 | 处理方式 | 备注 |
| 201 | NetworkError | 网络中断 | 无法连接到网络，请检查网络连接并重试 | 系统记录错误信息并提供重试机制 |  |
| 202 | inputerror | 输入错误 | 输入数据不合法，请重新输入 | 给出明确的错误提示信息，指导用户如何正确输入数据 |  |
| 说明： | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 子系统编号：3  子系统英文名称：expan\_analyse  子系统中文名称：消费分析 | | | | | |
| 错误编号 | 错误名称 | 错误原因 | 错误信息 | 处理方式 | 备注 |
| 301 | NetworkError | 网络中断 | 无法连接到网络，请检查网络连接并重试 | 系统记录错误信息并提供重试机制 |  |
| 302 | inputerror | 输入错误 | 输入数据不合法，请重新输入 | 给出明确的错误提示信息，指导用户如何正确输入数据 |  |
| 说明： | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 子系统编号：4  子系统英文名称：data\_manage  子系统中文名称：数据管理 | | | | | |
| 错误编号 | 错误名称 | 错误原因 | 错误信息 | 处理方式 | 备注 |
| 401 | NetworkError | 网络中断 | 无法连接到网络，请检查网络连接并重试 | 系统记录错误信息并提供重试机制 |  |
| 402 | OutOfMemory | 资源耗尽 | 无法完成操作，因为系统资源已耗尽，请尝试关闭一些应用程序并重试 | 释放不必要的系统资源，例如关闭不必要的应用程序、清理临时文件、释放内存 |  |
| 说明： | | | | | |

## 维护处理过程表

系统出错时，将调用维护处理过程对错误进行处理，有关维护处理过程的各项内容由维护处理过程表进行描述。

当系统有多个子系统(模块)组成时，每个子系统分别使用一张维护处理过程表进行描述。维护处理过程表的格式如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：1  子系统英文名称：user\_manage  子系统中文名称：用户管理 | | | | | | |
| 错误编号 | 处理过程 | 处理过程 | 处理功能 | 入口参数 | 出口参数 | 备注 |
| 英文名称 | 中文名称 |
| 101 | reconnect | 网络重连 | 自动重新连接网络 | 500 | 500/200 |  |
| 101 | off-line | 离线模式 | 切换到离线模式处理 | 500 | 200 |  |
| 102 | rewrite | 撤销重写 | 让用户纠正错误并重新输入数据 | 600 | 200 |  |
| 102 | help | 指出错误 | 给用户正确输入例子 | 600 | 正确输出 |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：2  子系统英文名称：expan\_recore  子系统中文名称：消费记录 | | | | | | |
| 错误编号 | 处理过程 | 处理过程 | 处理功能 | 入口参数 | 出口参数 | 备注 |
| 英文名称 | 中文名称 |
| 201 | reconnect | 网络重连 | 自动重新连接网络 | 500 | 500/200 |  |
| 201 | off-line | 离线模式 | 切换到离线模式处理 | 500 | 200 |  |
| 202 | rewrite | 撤销重写 | 让用户纠正错误并重新输入数据 | 600 | 200 |  |
| 202 | help | 指出错误 | 给用户正确输入例子 | 600 | 正确输出 |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：3  子系统英文名称：expan\_analyse  子系统中文名称：消费分析 | | | | | | |
| 错误编号 | 处理过程 | 处理过程 | 处理功能 | 入口参数 | 出口参数 | 备注 |
| 英文名称 | 中文名称 |
| 301 | reconnect | 网络重连 | 自动重新连接网络 | 500 | 500/200 |  |
| 301 | off-line | 离线模式 | 切换到离线模式处理 | 500 | 200 |  |
| 302 | rewrite | 撤销重写 | 让用户纠正错误并重新输入数据 | 600 | 200 |  |
| 302 | help | 指出错误 | 给用户正确输入例子 | 600 | 正确输出 |  |
| 说明： | | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 子系统编号：4  子系统英文名称：data\_manage  子系统中文名称：数据管理 | | | | | | |
| 错误编号 | 处理过程 | 处理过程 | 处理功能 | 入口参数 | 出口参数 | 备注 |
| 英文名称 | 中文名称 |
| 401 | reconnect | 网络重连 | 自动重新连接网络 | 500 | 500/200 |  |
| 401 | off-line | 离线模式 | 切换到离线模式处理 | 500 | 200 |  |
| 402 | release | 释放资源 | 释放不必要的系统资源 | 700 | 200 |  |
| 说明： | | | | | | |

# 数据库设计

1. 数据库需求分析：

账号方面需要存储用户输入的用户名和密码，并记录注册登录时间设备ID和类型等，账单方面需要记录用户输入的交易的日期、金额、类别和描述。

1. 数据库结构设计：

账号数据表：存储在Account表，有int类型主键，需要存储字符串类型的用户名、密码（由用户输入）、注册时间、登录时间、设备ID类型及登录时间（调用函数获取）。

账单数据表：存储在Transaction表，有自增的主键和联系Account表的外键，存储字符串类型的变量记录交易的日期、金额、类别和描述，都由用户输入。

两表通过Transaction表中的外键和Account表中的账号相关联。

1. 数据库安全设计：

实现数据备份和恢复机制，以防止数据丢失或损坏。

定期进行数据备份，并将备份数据存储在安全的位置。

# 词汇表

账户：记录资金流动的账户，包括现金账户、银行账户等。

备份：数据备份是指将重要的数据复制到另一个存储介质上，以防止数据丢失或损坏。

财务报表：反映财务状况和经营成果的报表，包括资产负债表、利润表等。费用：指企业在生产经营过程中所发生的、用于取得收益的支出，包括成本、费用和损失。

过账：将已记录在凭证上的会计分录转移到总账中的过程。

汇率：不同货币之间的兑换比率。

收入：企业在生产经营过程中所获得的、用于增加净资产的各种资金或非资金性收益

记账：将企业的各项经济业务按照一定的规则和程序记录在会计账簿中的过程。

科目：会计核算的基本单位，反映企业各项经济业务的资金量和性质。

流水号：每个会计凭证所对应的唯一标识符。

密码：用于验证用户身份的机密字符组合。

年度结转：将一个会计年度的损益结余转入下一年度的过程。

欠款：指企业欠他人的货款或服务费等。

凭证：会计核算的基本单元，用于记录各种经济业务的账务处理情况。

清单：列明某一特定范围内的全部内容的单据或表格。

日记账：最基本的会计账簿，记录所有的业务往来情况。

税金：企业依法应当缴纳的各种税金和附加费用。