物业收费系统分析

Jason5Lee

# 目录

[物业收费系统分析 1](#_Toc57792689)

[目录 1](#_Toc57792690)

[1 需求分析 2](#_Toc57792691)

[1.1 问题陈述 2](#_Toc57792692)

[1.2 用例图 2](#_Toc57792693)

[1.3 用例规约 2](#_Toc57792694)

[1.3.1 登录 2](#_Toc57792695)

[1.3.2 登出 3](#_Toc57792696)

[1.3.3 修改密码 3](#_Toc57792697)

[1.3.4 查看账单 4](#_Toc57792698)

[1.3.5 查看已缴费项 4](#_Toc57792699)

[1.3.6 查询账单 4](#_Toc57792700)

[1.3.7 确认缴费 5](#_Toc57792701)

[1.3.8 创建收费项 5](#_Toc57792702)

[1.3.9 编辑收费项 6](#_Toc57792703)

[1.3.10 删除缴费项 7](#_Toc57792704)

[1.3.11 查询个人 7](#_Toc57792705)

[1.4 补充规约 8](#_Toc57792706)

[1.4.1 可靠性 8](#_Toc57792707)

[1.4.2 安全性 8](#_Toc57792708)

[1.4.3 可用性 8](#_Toc57792709)

[1.4.4 缴费存档 8](#_Toc57792710)

[1.5 术语表 8](#_Toc57792711)

[2 架构设计 9](#_Toc57792712)

[2.1 架构描述 9](#_Toc57792713)

[2.2 架构图 9](#_Toc57792714)

[2.3 关键抽象 9](#_Toc57792715)

[3 用例分析 10](#_Toc57792716)

[3.1 补充用例规约 10](#_Toc57792717)

[3.2 类的析取 10](#_Toc57792718)

[3.2.1 登录 10](#_Toc57792719)

[3.2.2 修改密码 10](#_Toc57792720)

[3.2.3 查看账单 11](#_Toc57792721)

[3.3 类的设计 11](#_Toc57792722)

[4 子系统及其接口设计 12](#_Toc57792723)

[4.1 确定设计类 12](#_Toc57792724)

[4.2 定义子系统 12](#_Toc57792725)

[5 部件设计 12](#_Toc57792726)

[5.1 分析并发需求 12](#_Toc57792727)

[5.2 针对缴纳账单时的解决方案 12](#_Toc57792728)

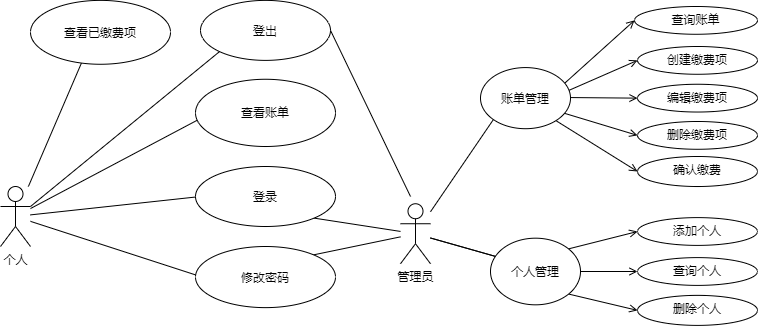
[5.3 针对个人操作时可能被删除的解决方案 12](#_Toc57792729)

# 需求分析

## 问题陈述

物业收费是日常生活中非常常见的一个问题。物业会对个人开具收费项，个人未缴纳的收费项构成账单。当个人缴纳账单的总收费后，物业可以标记该账单已缴纳，其所有收费项进入已缴纳状态，将不会出现在账单中。本系统可以管理物业的个人信息、缴费项。物业可以创建、编辑、删除缴费项，

## 用例图



## 用例规约

### 登录

1. **简要说明**

个人和管理员需要登录后方可进行自己的工作；

1. **参与者**

个人或管理员；

1. **事件流**

用例开始于用户点击“登录”按钮；

1. 系统显示登录页面，用户可以选择“个人和管理员”；
2. 若用户选择“个人”，显示输入身份证号和密码；
3. 若用户选择“管理员”，显示输入管理员密码；
4. 当用户点击“登录”后，查询输入信息和密码是否匹配；若不匹配，显示“登录失败，请检查身份证号和密码是否正确”/ “登录失败，请检查管理员密码是否正确”；若匹配，显示“登录成功”并跳转到用户主页；
5. **前置条件**

用户点击登录按钮；

1. **后置条件**

如果本用例成功，用户进入登录状态，否则系统状态不变。

### 登出

1. **简要说明**

个人和管理员登录后可登出，退出登录状态；

1. **参与者**

个人或管理员；

1. **事件流**

用例开始于用户点击“登出”按钮；

1. 登出后系统进入未登录状态，返回到主页；
2. **前置条件**

用户点击登出按钮；

1. **后置条件**

如果本用例成功，用户未登录状态，否则系统状态不变；

### 修改密码

1. **简要说明**

用户登录后可以修改自己的密码；

1. **参与者**

个人或管理员；

1. **事件流**
2. **基本事件流**

用例开始于用户点击“修改密码”按钮；

1. 系统显示修改密码页面，包括新密码、确认密码、确认按钮；
2. 用户填写密码；
3. 用户填写确认密码；
4. 用户点击“确认修改”；

D1. 确认密码和新密码不一致；

1. 系统跳转到用户首页；
2. **后备事件流**

D1. 确认密码和新密码不一致；

当用户所填写新密码和确认密码不一致时，系统提示“两次输入密码不一致.“；返回基本事件流第一步；

1. **前置条件**

用户点击登录按钮；

1. **后置条件**

如果本用例成功，用户密码修改，否则系统状态不变；

### 查看账单

1. **简要说明**

账单为个人所有未缴费的缴费项集合，个人可以查看自己的账单；

1. **参与者**

个人；

1. **事件流**
2. **基本事件流**

用例开始于用户点击“查看账单”按钮；

1. 系统显示账单，为表格，包括“收费名”、“费用”列；
2. **前置条件**

用户点击查看账单按钮；

1. **后置条件**

该用例不改变系统状态；

### 查看已缴费项

1. **简要说明**

个人可以查看自己的已缴费的收费项；

1. **参与者**

个人；

1. **事件流**
2. **基本事件流**

用例开始于用户点击“查看已缴费项”按钮；

1. 系统显示已缴费项，为表格，包括“收费名”、“费用”列；
2. **前置条件**

用户点击查看已缴费项按钮

1. **后置条件**

该用例不改变系统状态

### 查询账单

1. **简要说明**

管理员可以查询指定用户的账单，并对其进行管理操作；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员点击“账单管理”按钮；

1. 系统显示“账单个人身份证号“、一个输入框和”查询“按钮；
2. 管理员填写要查询的身份证号后点击“查询”按钮；

B1. 输入为空；

B2. 输入内容不是合法的身份证号；

B3. 输入的身份证号在数据库中不存在；

1. 系统进入查询界面，显示“[身份证号] 的账单如下“；下面显示总费用、创建收费项按钮和一个表格；表格有”收费名“、”费用“列，对应一个收费项，每一列都有一个”编辑“和”删除“按钮；
2. **后备实践流**

B1. 提示“请填写此字段“

B2. 提示”身份证号不合法“

B3. 提示“找不到用户”

1. **前置条件**

管理员点击“账单管理”按钮

1. **后置条件**

本用例不改变系统状态；

### 确认缴费

1. **简要说明**

管理员在查看个人账单时，若个人向组织缴纳了该账单费用，管理员可确认该账单缴费，此时账单内所有缴费项变成已缴费；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员在“查询账单”界面点击“确认缴费”按钮；

1. 系统显示“确认已收到费用 [账单总费用]？”和一个“确认”、“返回”按钮；

A1. 用户点击“返回”按钮；

1. 用户点击确认按钮，系统确认缴费后界面返回到查询账单页面；
2. **后备事件流**

A1. 用户点击“返回”按钮，系统不做任何修改，界面直接回到查询账单页面；

1. **前置条件**

管理员在“查询账单”界面点击“确认缴费”按钮；

1. **后置条件**

如果本用例成功，则账单内缴费项进入已缴纳状态，否则系统状态不变；

### 创建收费项

1. **简要说明**

管理员可以创建收费项；为了创建收费项，管理员需要提供收费名和费用；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员在查询收费界面上点击“创建收费项”；

1. 系统显示“身份证号”和不可修改的当前账单收费对象身份证号、收费名和输入框、费用和默认为0的输入框、创建收费项按钮和返回；

A1. 用户点击“返回”按钮；

1. 用户输入收费名和费用后点击“创建收费项”按钮；

B1. 收费名为空；

B2. 费用为空；

1. 收费成功创建后界面返回到查询账单；
2. **后备事件流**

A1. 用户点击返回按钮，界面直接回到查询账单界面；

B1. 若创建收费项时收费名为空，系统在收费名输入框上显示“请填写此字段”；

B2. 若创建收费项时费用为空，系统在费用输入框上显示“该项不可为空”；

1. **前置条件**

管理员在查询收费界面上点击“创建收费项”

1. **后置条件**

如果本用例成功，收费项将在系统中创建，否则系统状态不变；

### 编辑收费项

1. **简要说明**

管理员可以编辑未支付收费项，修改收费名和费用；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员在“查询账单”界面点击对应收费项的“编辑”；

1. 系统显示“身份证号”和不可修改的当前账单收费对象身份证号、收费名和初始原收费名输入框、费用和初始为原费用的输入框、修改按钮和返回；

A1. 用户点击“返回”按钮；

1. 用户输入收费名和费用后点击“修改”按钮；

B1. 收费名为空；

B2. 费用为空；

1. 收费成功创建后界面返回到查询账单；
2. **后备事件流**

A1. 用户点击返回按钮，界面直接回到查询账单界面；

B1. 若创建收费项时收费名为空，系统在收费名输入框上显示“请填写此字段”；

B2. 若创建收费项时费用为空，系统在费用输入框上显示“该项不可为空”；

1. **前置条件**

管理员在“查询账单”界面点击对应收费项的“编辑”；

1. **后置条件**

如果本用例成功，收费项将在系统中被修改，否则系统状态不变；

### 删除缴费项

1. **简要说明**

管理员可以删除未支付收费项；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员在“查询账单”界面点击对应收费项的“删除”；

1. 系统显示“确认删除如下收费项？”、收费项的信息、“确认删除”按钮和返回按钮；

A1. 用户点击“返回”按钮；

1. 用户点击“确认删除”按钮后回到查询账单界面；
2. **后备事件流**

A1. 用户点击返回按钮，界面直接回到查询账单界面；

1. **前置条件**

管理员在“查询账单”界面点击对应收费项的“删除”；

1. **后置条件**

如果本用例成功，收费项将在系统中被删除，否则系统状态不变；

### 查询个人

1. **简要说明**

管理员可以查询个人，并进行创建、删除操作；

1. **参与者**

管理员；

1. **事件流**
2. **基本事件流**

用例开始于管理员在主页界面点击“个人管理”；

1. 系统显示“个人管理”、“创建”按钮和个人信息表格，表格有“身份证号”、“姓名”、“住址”列和删除按钮；
2. **前置条件**

管理员在主页点击“查询账单”按钮；

1. **后置条件**

本用例成功不改变系统状态；

## 补充规约

### 可靠性

本系统可以保证7\*24小时连续运行，一周维护时间不会超过2小时；

### 安全性

本系使用框架提供的防止over-posting和跨站访问攻击的机制；

密码以密文存储在数据库中；

使用https协议；

### 可用性

本系统目标群体为物业管理员和住户，页面简单易理解、易操作；

### 缴费存档

所有已缴纳的收费项会存档在数据库中，任何身份均不可修改；

## 术语表

|  |  |
| --- | --- |
| **术语** | **含义** |
| 个人 | 缴纳的个体，信息包括姓名、身份证号、住址和密码； |
| 管理员 | 代表物业收费方；具有管理个人和收费的权限； |
| 收费项 | 一个需要个人缴费的项目；信息包括id、缴费对象个人身份证号、收费项名、是否已缴纳、费用； |
| 收费项名 | 收费的名称，必要时根据具体业务的收费周期填写决定时间，比如“2020年7月水费”； |
| 账单 | 表示同一个人呢未缴费收费项的列表；账单包含缴费对象个人身份证号、收费项列表、账单费用； |
| 账单费用 | 账单所有收费项总收费的总和； |
| 已缴费项 | 已缴纳的收费项，作为存档储存在数据库中，任何身份均不可修改； |

# 架构设计

## 架构描述

本系统基于ASP.NET Core MVC框架，采用MVC（Model View Controller）三层架构。该架构的使用实现了应用程序的分层管理，简化了后续对程序的修改和 扩展，并且使程序某一部分的重复利用成为可能。

View层又称视图层，是用户看到并与之交互的页面。在本系统中，视图 层由众多HTML文件组成，它们负责获取用户的输入及显示控制层处理的结果。

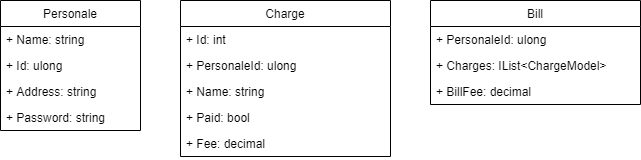
Controller层又称控制层，它收到来自视图层的请求并对实体类进行增 删改查。当单击Web页面中的超链接和发送HTML表单时，控制层接收请求并 处理请求，然后确定用哪个视图来显示处理返回的数据。

## 架构图



## 关键抽象

经过对术语表分析，本系统有四个实体类，分别为个人类、管理员类、组织类、收费项类和账单类。个人类包含个人的一切信息，包括姓名、身份证号、住址和密码；管理员类包含管理员的所有信息，包括管理员密码；组织类包含组织的一切信息，包括组织名、组织id；收费项类包含收费项的一切信息，包括缴费对象id、收费对象id、收费项、是否已缴纳、基础收费、截止日期和日滞纳金；账单包括缴费对象id、收费项列表、创建日期、滞纳金列表、账单收费；



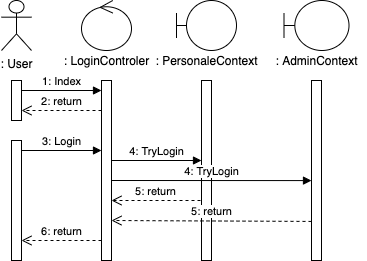
# 用例分析

## 补充用例规约

经检查， 本项目组发现用例规约比较完善，暂时无需补充。

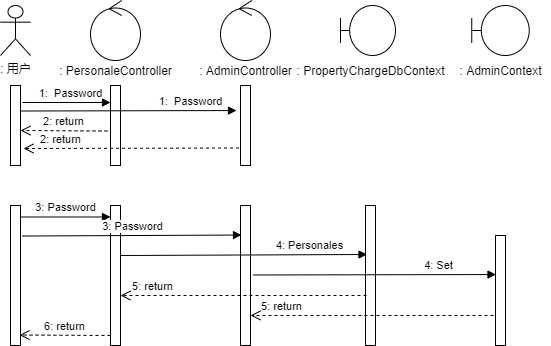
## 类的析取

### 登录



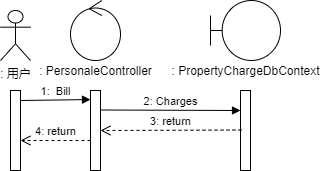
用户点击主页Home上的“登录”按钮，进入Login页面。系统调用 LoginController控制类的Index()方法，得到登录页面；当用户点击登录后，调用LoginController的Login()方法，该方法根据登录类型，调用边界类PersonaleContext或AdminContext的TryLogin方法，Context类会在数据库中查询，返回是否登录成功；若登录成功，设置登录SessionId并返回跳转到Home，否则返回登录页面并显示登录错误信息。

### 修改密码



用户在登录后点击“修改密码”按钮，根据其身份从 PersonaleController或AdminController获得修改密码页面；用户点击确认修改密码按钮后，调用PersonaleController或AdminController的Password方法，若两次输入密码一致，根据用户身份从数据库边界类PropertyChargeDbContext或管理员信息边界类AdminContext中修改密码；若两次输入密码不一致，返回修改密码页面并显示错误信息；

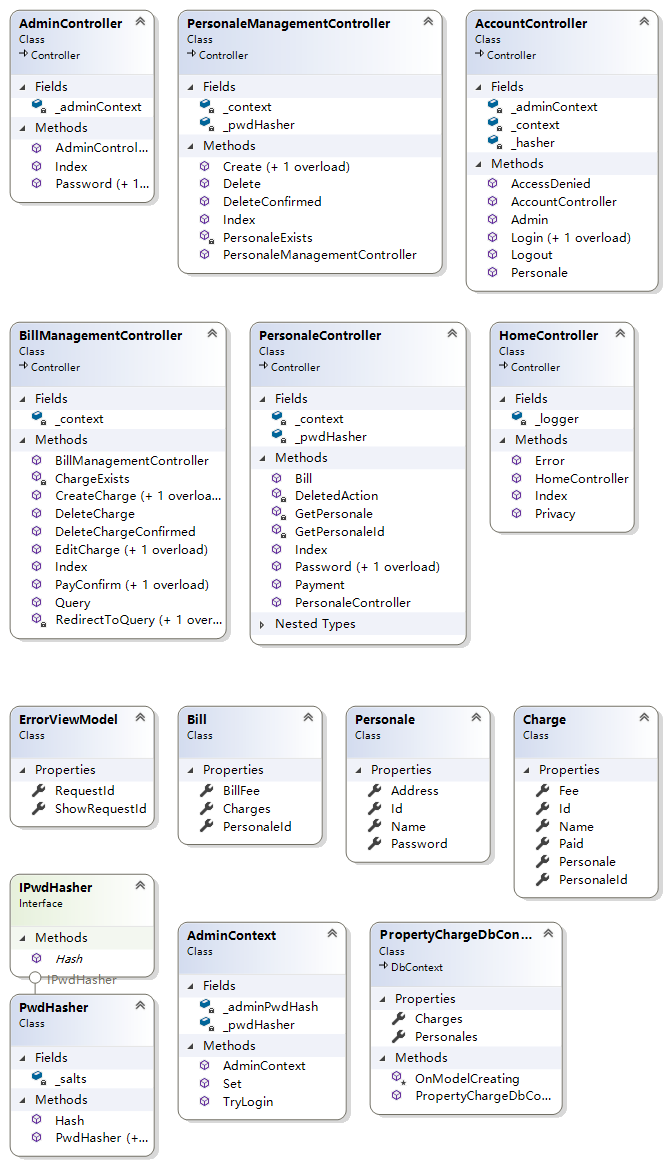
### 查看账单



个人在查看账单时，调用PersonaleController的Bill方法， BillController从数据库边界类PropertyChargeDbContext中检索信息，组织成页面返回给用户。

## 类的设计

经过分析，如下为系统类设计；



# 子系统及其接口设计

## 确定设计类

本项目组对第三章的分析类进行了分析与检查，以确定其是否能成为设计类。经过分析发现，第三章中所有分析类均为单逻辑，不需要进行类的分解或合并，因此不做修改，设计类如第三章所示；

## 定义子系统

经过本项目组分析，本系统无需进行子系统设计及其接口设计。

# 部件设计

## 分析并发需求

本系统在运行时会产生一些并发需求：

1. 个人在缴纳账单时，会有新的收费项产生；
2. 个人在操作时，可能被管理员删除；

## 针对缴纳账单时的解决方案

个人在缴纳账单的过程中，可能会有新的收费项产生。为了解决这个问题，当管理员查询账单后，用户缴纳该账单费用，管理员选择确认缴纳，此时系统只将管理员查询到账单的收费项标记为已缴费，新产生的仍然是未缴费；

## 针对个人操作时可能被删除的解决方案

当个人在登录状态被删除时，其所有操作都会导致登出并返回主页。此时用户无法再以相同个人身份登录。