**小区物业管理系统**

分析与设计

**目录**

[第1章 需求分析 4](#_Toc456260574)

[1.1小区物业管理系统问题陈述 4](#_Toc456260575)

[1.2 小区物业管理系统用例析取 5](#_Toc456260576)

[1.3小区物业管理系统用例规约 5](#_Toc456260577)

[1.3.1 业主管理 5](#_Toc456260578)

[1.3.2 提交报修 7](#_Toc456260579)

[1.3.3 报修管理 9](#_Toc456260580)

[1.3.4 登录 11](#_Toc456260581)

[1.3.5 个人信息管理 11](#_Toc456260582)

[1.3.6 查看报修 11](#_Toc456260583)

[1.3.7 提交投诉 11](#_Toc456260584)

[1.3.8 查看投诉 11](#_Toc456260585)

[1.3.9 查看缴费信息 11](#_Toc456260586)

[1.3.10 查看公示信息 11](#_Toc456260587)

[1.3.11 注册 11](#_Toc456260588)

[1.3.12 投诉管理 11](#_Toc456260589)

[1.3.13 收费管理 12](#_Toc456260590)

[1.3.14 公示信息管理 14](#_Toc456260591)

[1.4 补充规约 16](#_Toc456260592)

[1.4.1 目标 16](#_Toc456260593)

[1.4.2 范围 16](#_Toc456260594)

[1.4.3 功能 16](#_Toc456260595)

[1.4.4 可用性 16](#_Toc456260596)

[1.4.5 可靠性 17](#_Toc456260597)

[1.4.6 性能 17](#_Toc456260598)

[1.4.7 可支持性 17](#_Toc456260599)

[1.4.8 安全性 17](#_Toc456260600)

[1.4.9 设计约束 17](#_Toc456260601)

[1.5 术语表 17](#_Toc456260602)

[第2章 小区物业管理系统架构设计 20](#_Toc456260603)

[2.1系统框架 20](#_Toc456260604)

[2.2系统关键抽象 20](#_Toc456260605)

[2.3 用例分析 22](#_Toc456260606)

[2.3.1注册用例分析 22](#_Toc456260607)

[2.3.2登陆用例分析 22](#_Toc456260608)

[2.3.3个人信息管理用例分析 23](#_Toc456260609)

[2.3.4修改密码用例分析 24](#_Toc456260610)

[2.3.5业主管理用例分析 25](#_Toc456260611)

[2.3.6投诉管理用例分析 25](#_Toc456260612)

[2.3.7收费管理用例分析 26](#_Toc456260613)

[2.3.8公示信息管理用例分析 26](#_Toc456260614)

[2.3.9报修管理用例分析 27](#_Toc456260615)

[2.3.10提交报修用例分析 27](#_Toc456260616)

[2.3.11查看报修用例分析 28](#_Toc456260617)

[2.3.12提交投诉用例分析 28](#_Toc456260618)

[2.3.13查看投诉用例分析 29](#_Toc456260619)

[2.3.14查看缴费用例分析 29](#_Toc456260620)

[2.3.15查看公示信息用例分析 30](#_Toc456260621)

[第3章 小区物业管理系统子系统及其接口设计 31](#_Toc456260622)

[3.1小区物业管理系统架构图 31](#_Toc456260623)

[3.2小区物业管理系统子系统划分及其接口设计 31](#_Toc456260624)

[3.2.1用户子系统设计 32](#_Toc456260625)

[3.2.2管理员子系统设计 34](#_Toc456260626)

# 第1章 需求分析

## 1.1小区物业管理系统问题陈述

小区物业管理系统是物业部门用来对小区物业进行管理的系统，具体包括小区的业主信息、收费信息、维修信息等的综合管理。

小区物业管理系统可以帮助物业部门更有效率地管理小区信息和解决小区事务，提升小区的管理水平，使小区的日常管理更加方便，从而使业主更好地享受物业服务。

该系统设定两个角色，管理员和业主。

业主的主要功能是注册、个人信息管理、提交报修、查看报修、提交投诉、查看投诉、查看缴费清单以及查看公示信息。在进行注册时，业主填写账号密码、业主名、电话、电子邮件以及住址，提交后系统将更新给管理员进行核实、添加。业主登录后可以查看并修改个人信息，但是一些信息例如账号、业主名以及住址是不能修改的，只能修改电话、电子邮件、密码等，提交后系统将更新保存。在进行报修或投诉时，业主填写需要报修或投诉的内容，提交后系统将更新保存，之后业主可以查看自己报修或投诉内容的更新状态，管理人员正在处理或者已处理。业主可以在缴费清单里查看个人需要缴费的信息。业主还可以查看小区的公示信息，包括所有业主的姓名、电话和电子邮件，物业的通知等，业主还可以根据关键字搜索相关信息。

管理员的功能是业主管理、报修管理、投诉管理、收费管理以及公示信息管理。在进行业主管理时，管理员可以对申请注册的业主的信息进行核实，然后添加该账号，使该账号拥有业主的权限。同时管理员可以根据业主名查找相关业主的信息，对业主的账号进行管理，删除已搬出本小区的业主的账号。管理员还可以对业主的报修或投诉信息进行管理，更改报修或投诉的信息状态并更新给相关业主。对于收费管理，管理员在填写完业主名、收费时间、收费项目以及收费金额，提交后系统会自动检测业主名是否存在，如果不存在系统将不更新，否则将更新给相关业主。管理员可以对公示信息进行管理，增加、修改或删除业主和管理员的被公示的信息，新建公示信息，修改或删除已存在的公示信息，提交后系统将更新保存。

## 1.2 小区物业管理系统用例析取

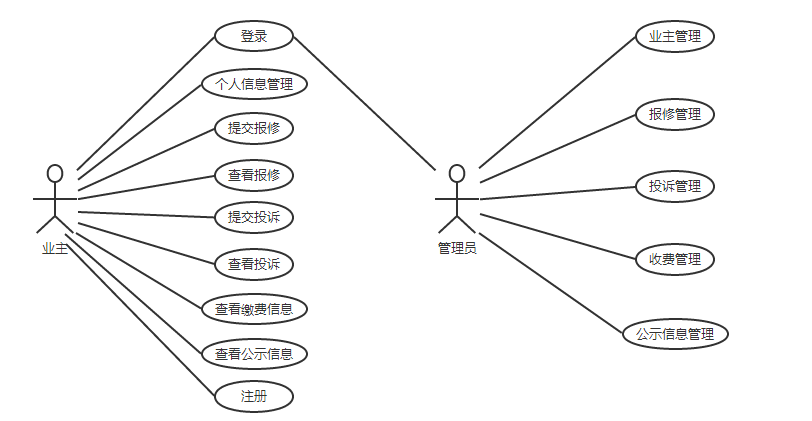


图 1：小区物业管理系统用例图

## 1.3小区物业管理系统用例规约

### 1.3.1 业主管理

**1） 简要说明**

本用例描述了管理员对业主的账号信息管理，包括查看已存在的账号信息，删除已搬出的业主的账号，并且对申请注册的业主信息进行核实，核实后添加该账号到系统，使该账号拥有业主权限。

**2） 事件流**

1. **基本事件流**

用例开始于管理员已登录系统，并选择业主管理对业主的账号信息进行管理

1. 系统要求管理员指出要执行的操作（添加业主或管理业主）
2. 一旦管理员提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“添加业主”，添加业主子事件流将被执行

如果选择的是“管理业主”，管理业主子事件流将被执行

1. 添加业主
   1. 系统将所有申请注册的信息以列表的形式显示出来
   2. 管理员对某一条注册信息进行核实，包括业主名、住址、电话和电子邮件，然后处理该注册信息

A1：删除注册信息

A2：添加账号

1. 管理业主
   1. 系统将所有业主的信息包括业主名、住址、电话、电子邮件、账号密码等以列表的形式显示出来
   2. 管理员对某一条业主信息进行处理

A3：修改业主信息

A4：删除业主信息

1. **后备事件流**

A1: 删除注册信息

当管理员在对申请注册的信息进行核实时，发现业主信息不符实，将该注册信息删除，不允许该账号注册。

A2: 添加账号

当管理员在对申请注册的信息进行核实时，如果业主信息是准确的，将该账号添加到系统，使该账号的业主拥有业主权限。

A3: 修改业主信息

管理员有权限对业主的信息就行修改。当业主搬到本小区的其它住房后，管理员可以更改该业主的住址信息而非删除该业主信息让其重新注册；当业主的账号密码被他人修改时，管理员可以修改该业主的密码帮其找回原账号。

A4: 删除业主信息

如果某个业主搬离本小区，管理员可以将其原本的账号删除，取消其业主权限。

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前管理员已登录系统。

**5） 后置条件**

如果用例成功，系统将新添加账号使其拥有业主权限，并且系统更新状态；若失败，系统状态不改变。

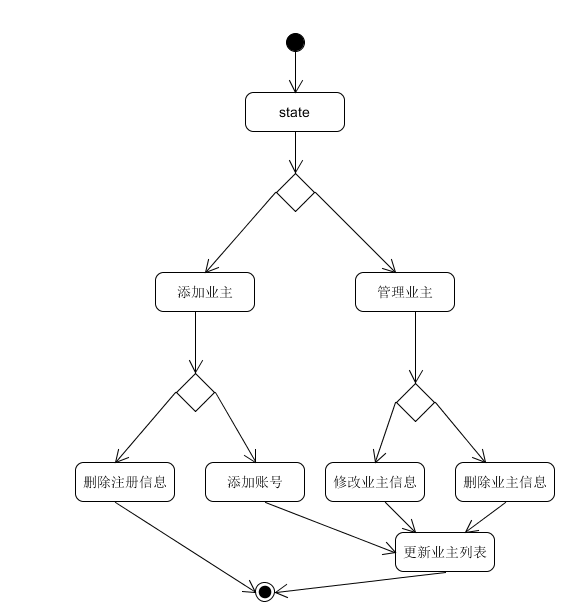


图 2：业主管理活动图

### 1.3.2 提交报修

**1） 简要说明**

本用例描述了业主对报修信息的提交，业主需要填写要报修的内容，提交后系统添加上业主名、电话、电子邮件、住址、提交时间后，进行状态的更新。

**2） 事件流**

1. **基本事件流**

用例开始于业主已登录系统，并且进行报修

1. 系统请求进行报修，显示出一个文本框给业主填写报修内容
2. 业主填完后提交给系统
3. 系统自动添加上业主名、电话、电子邮件、住址和提交时间后更新状态
4. **后备事件流**

无

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前业主已登录系统

**5） 后置条件**

如果本用例成功，系统将该报修信息更新到管理员的报修管理列表以及业主的报修信息上。若失败，系统状态不改变。



图 3：提交报修活动图

### 1.3.3 报修管理

**1） 简要说明**

本用例描述管理员对业主的报修信息进行管理，能够在所报修的项目被处理后改变报修信息的处理状态，并由系统更新给所报业主。

**2） 事件流**

1. **基本事件流**

用例开始于管理员已登录，并选择报修管理查看报修信息

1. 系统请求查看报修信息，将所有的报修信息以列表的形式显示出来
2. 管理员将某一条报修信息的状态修改为已受理
3. 管理员将某一条报修信息删除
4. **后备事件流**

无

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前管理员已登录系统

**5） 后置条件**

如果本用例成功，系统将更新状态，用户可以通过查看报修查看修改后的报修信息状态；若失败，系统状态不改变。



图 4：报修管理活动图

### 1.3.4 登录

### 1.3.5 个人信息管理

### 1.3.6 查看报修

### 1.3.7 提交投诉

### 1.3.8 查看投诉

### 1.3.9 查看缴费信息

### 1.3.10 查看公示信息

### 1.3.11 注册

### 1.3.12 投诉管理

**1） 简要说明**

本用例描述管理员对业主的投诉进行管理，能够在投诉的信息被处理后，系统将更新投诉的处理状态。

1. **事件流**

**i 基本事件流**

用例开始于管理员已登录，并选择投诉管理查看业主的投诉信息。

1. 系统请求查看投诉信息，将所有的投诉信息以列表的形式显示出来
2. 管理员点击选择某一条未处理的投诉信息
3. 管理员在点击完某一条未处理的投诉信息后在点击更新状态，未处理状态变为已处理状态。

**ii 后备事件流**

无

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前管理员已登录系统

**5） 后置条件**

如果本用例成功，系统将更新状态，用户可以通过查看投诉列表，查看修改后的投诉信息状态；若失败，系统状态不改变。

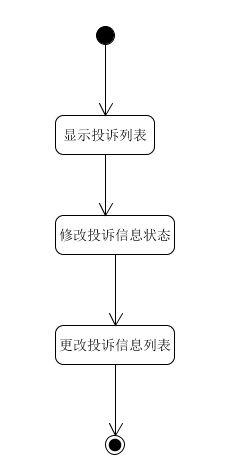


图 12：投诉管理活动图

### 1.3.13 收费管理

**1）简要说明**

本用例描述了管理员对业主的收费进行管理，包括添加收费创建订单和查看业主的缴费情况。

**2）事件流**

**i 基本事件流**

用例开始于管理员已登录系统，并选择收费管理，对业主的收费进行管理。

1. 系统要求管理员指出要执行的操作（添加收费或查看收费）
2. 一旦管理员提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“添加收费”，添加收费子事件流将被执行

如果选择的是“查看收费”，查看收费子事件流将被执行

1. 添加收费

系统将提供一个收费订单，管理员把包括用户名，业主名、住址、身份证，联系方式，邮箱，水费，电费，维修费，管理费，缴费月份进行填写，并提交到系统。

1. 查看收费

系统将所有业主的信息包括业主姓名，缴费日期，水费，电费，维修费，管理费，缴费总额，缴费状态等以列表的形式显示出来，供管理员查看。

**ii 后备事件流**

无

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前管理员已登录系统。

**5） 后置条件**

如果用例成功，系统将把新的收费订单提交给系统，添加到收费列表里，更新收费列表，支持管理员查看。

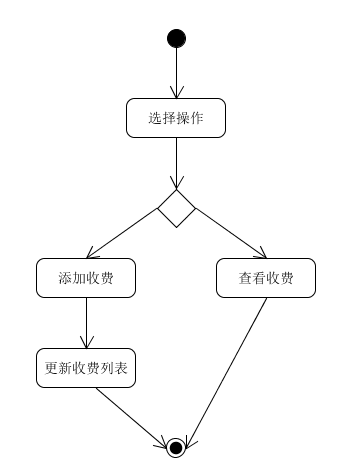


图13.收费管理活动图

### 1.3.14 公示信息管理

**1）简要说明**

本用例描述了管理员对公示信息进行管理，包括公示信息管理和发布公示信息。

**2）事件流**

**i 基本事件流**

用例开始于管理员已登录系统，并选择公示信息管理，对公示信息进行管理。

1. 系统要求管理员指出要执行的操作（公示信息管理或发布公示信息）
2. 一旦管理员提供了所需要的信息，以下的一条子事件流将被执行

如果选择的是“公示信息管理”，公示信息管理子事件流将被执行

如果选择的是“发布公示信息”，发布公示信息子事件流将被执行

1. 公示信息管理

系统将提供公示信息的列表，管理员点击其中一条公示信息，可以将其进行删除管理

1. 发布公示信息

系统将提交一个表单，包括公示编号，公示标题跟公示内容，管理员将其填写并提交到系统。

**ii 后备事件流**

无

**3） 特殊需求**

无

**4） 前置条件**

本用例开始前管理员已登录系统。

**5） 后置条件**

如果用例成功，系统将把公示信息列表展现给管理员，支持管理员删除管理，同时管理员可以发布公示信息，更新系统的公示信息列表。

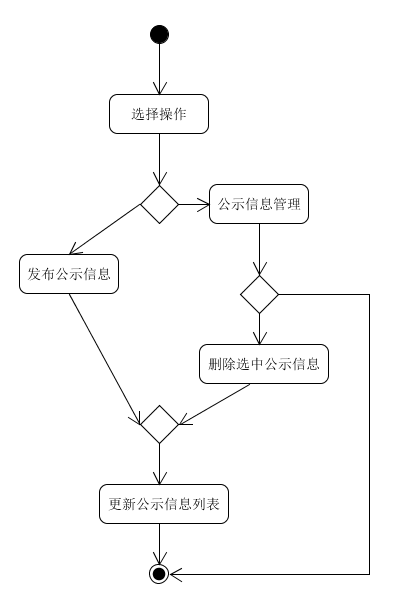


图13.公示信息管理活动图

## 1.4 补充规约

### 1.4.1 目标

本部分定义了小区物业管理系统的非功能性需求，补充了在用例规约中难以定义的系统需求。

### 1.4.2 范围

本补充规约适用于小区物业管理系统，定义了本系统的非功能性需求，包括功能、可靠性、可用性、可支持性等。

### 1.4.3 功能

1. 多人在线：本系统支持管理员和多名业主同时进行操作。
2. 投诉处理提示：业主登录系统后，若其已提交的投诉被管理员查看并作出相应反馈，则系统会给出提示。
3. 公示信息更新提示：业主登录系统后，若管理员进行了公示信息的更新，则系统会给出提示。
4. 缴费提示：管理员发出缴费通知后，相关业主会受到消息提示。
5. 新报修/投诉信息提示：若有业主发出了新的报修或投诉，则管理员会收到提示，避免延误。
6. 业主登录历史记录：系统可记载业主一个月之内的登录历史。

### 1.4.4 可用性

1. 无相关网站管理员经验的管理人员学会使用该系统应不超过6小时。
2. 注册业主学会使用该系统应不超过4小时。

### 1.4.5 可靠性

本系统应保证每天24小时均能够使用，每3个月进行一次系统维护。保证业主、管理员能够行使其权限。

### 1.4.6 性能

无。

### 1.4.7 可支持性

无。

### 1.4.8 安全性

1. 业主和管理员的账号与密码需进行加密后再保存。
2. 管理员权限和业主权限需进行严格的界定，防止出现跨权限操作的恶意行为。
3. 一旦发生账号被盗取事件，能够快速冻结账号防止事态恶化，并找回账号。

### 1.4.9 设计约束

无。

## 1.5 术语表

本部分对本系统设计和开发过程中出现的关键概念进行了定义。

表1 小区物业管理系统术语表

|  |  |
| --- | --- |
| 术语 | 说明 |
| 小区物业管理系统/系统 | 物业部门管理员用来对小区物业进行管理的系统，具体包括小区的业主信息、收费信息、维修信息等的综合管理，业主也可通过账号密码登录该系统行使业主权限。该系统即为本次设计开发的主体，下文所提“系统”均为本系统。 |
| 管理员 | 物业部门管理员，由系统设定，不可注册。拥有本系统的管理权限，可管理系统中的业主信息、收费信息、维修信息、公示信息等。 |
| 业主 | 注册成功后使用本系统行使业主权限的人。 |
| 业主名 | 业主注册时填写提交的真实姓名，一经注册，不可更改。 |
| 账号 | 业主注册所使用的身份证号码，唯一标识业主身份，一经注册，不可更改。 |
| 住址 | 业主在该小区的住址，一经注册，不可由业主自主更改，若搬至该小区内其他地址，可联系管理员进行更改。 |
| 注册 | 业主申请行使本系统的业主权限，需提交注册信息。 |
| 业主权限 | 拥有本系统账号的业主可行使的权限，包括个人信息管理、提交报修、查看报修、提交投诉、查看投诉、查看缴费清单以及查看公示信息 |
| 注册信息 | 业主进行注册时提交的个人信息，供管理员进行核实，包括账号、业主名、住址、电话和电子邮件。 |
| 报修信息 | 业主填写的报修内容与系统自动添加的业主信息共同构成报修信息。 |
| 报修管理列表 | 管理员可查看，包括小区内所有已提交的报修的信息和状态。 |
| 个人信息管理 | 已注册的业主可对个人信息进行查看和修改，但账号、业主名、住址不能修改，只能修改电话、邮箱等内容。 |
| 提交报修 | 业主使用账号登陆后，可在该模块填写需要保修的内容并提交，系统会进行更新。 |
| 查看报修 | 业主可登陆账号查看已提交的报修的状态。 |
| 提交投诉 | 业主使用账号登陆后，在该模块填写投诉内容并提交，系统会进行更新。 |
| 查看投诉 | 业主可登陆账号查看已提交的投诉和其他业主的投诉。 |
| 查看缴费清单 | 业主使用账号登录后，在该模块可查看自己的缴费清单。 |
| 业主管理 | 仅限管理员使用该权限，包括查看已存在的账号信息，删除已搬出的业主的账号，并且对申请注册的业主信息进行核实，核实后添加该账号到系统，使该账号拥有业主权限。 |
| 报修管理 | 仅限管理员使用该权限。管理员对业主的报修信息进行管理，能够在所报修的项目被处理后改变报修信息的处理状态，并由系统更新给所报业主 |
| 投诉管理 | 仅限管理员使用该权限。管理员对业主的投诉信息进行管理，能够及时给予反馈。 |
| 收费管理 | 仅限管理员使用该权限。管理员在填写完业主名、收费时间、收费项目以及收费金额，提交后系统会自动检测业主名是否存在，如果不存在系统将不更新，否则将更新给相关业主。 |
| 公示信息管理 | 仅限管理员使用该权限。管理员增加、修改或删除公示信息，公示信息可由管理员和业主查看。 |

# 第2章 小区物业管理系统架构设计

## 2.1系统框架

本系统采用MVC三层架构的系统设计框架，这三层分别为：表示层，控制层以及实体层。该三个分层结构具有明晰的依赖关系，表示层依赖于控制层，控制层调用实体层。



图 5：小区物业管理系统包依赖图

* 表示层

表示层是用户与系统的交互界面，负责显示用户信息，系统功能列表，功能操作界面和系统操作后得到的结果，并且负责获取用户请求消息。本系统的表示层决定采用MFC，即微软基础类库提供的诸多windows API函数实现。

* 控制层

控制层负责执行整个系统的业务操作，控制整个系统的运行。首先系统的控制层会接收用户请求消息，然后调用实体层数据，接着负责执行业务操作，最后将操作的结果传输到表示层显示给用户。

* 实体层

实体层是对数据进行封装，同时包含数据的处理方法，有直接访问数据的权限。

## 2.2系统关键抽象

系统关键抽象即系统实体类图。本项目组对本系统进行实体类的析取，得到系统关键抽象。如图所示，包括ReadPeople, WritePeople, ReadMessage, ReadPayment, WriteMessage, WritePayment, User7个数据类。

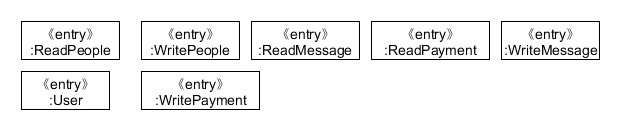


图 6：小区物业管理系统关键抽象

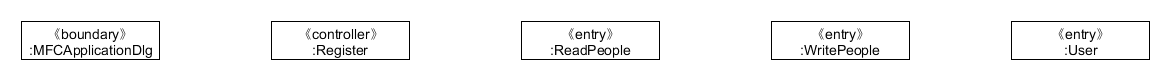
## 2.3 用例分析

### 2.3.1注册用例分析

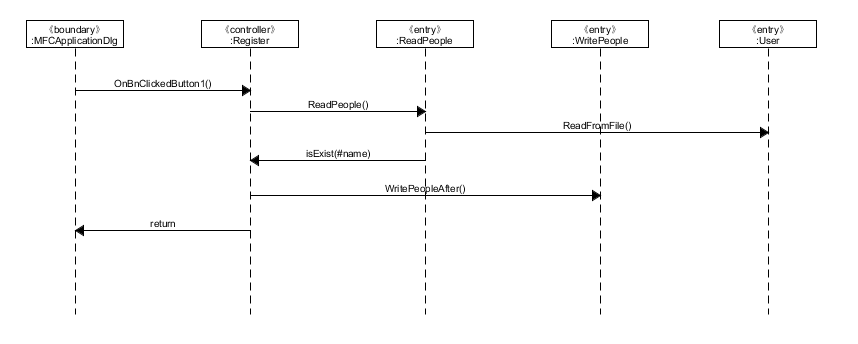
1. 注册用例功能描述

业主或者管理员在该系统进行注册身份。

2）注册用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

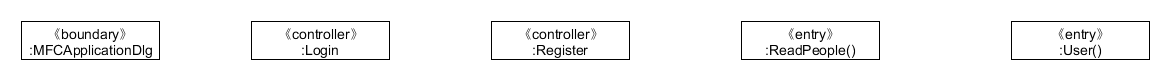


### 2.3.2登陆用例分析

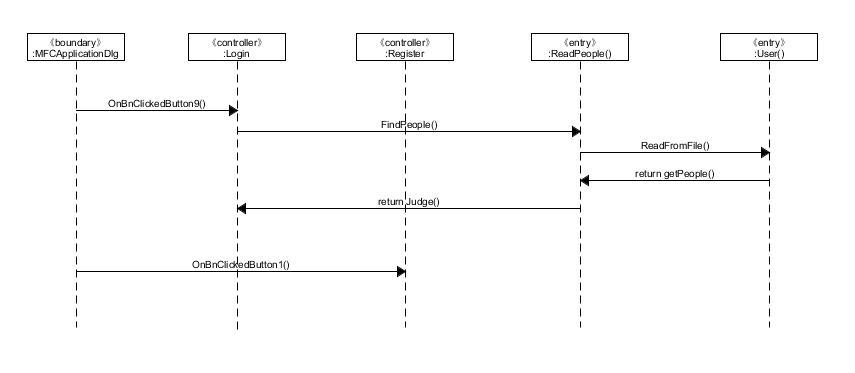
1）登陆用例功能描述

业主或者管理员根据注册信息登陆系统。

2）注册用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



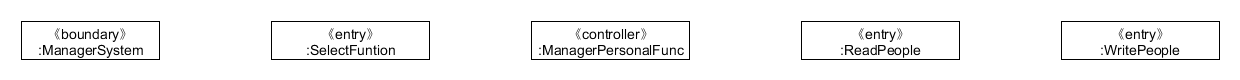
### 2.3.3个人信息管理用例分析

1）个人信息管理用例功能描述

业主或者系统管理员对自己的信息进行管理。

1. 个人信息管理用例类的析取

管理员个人信息管理

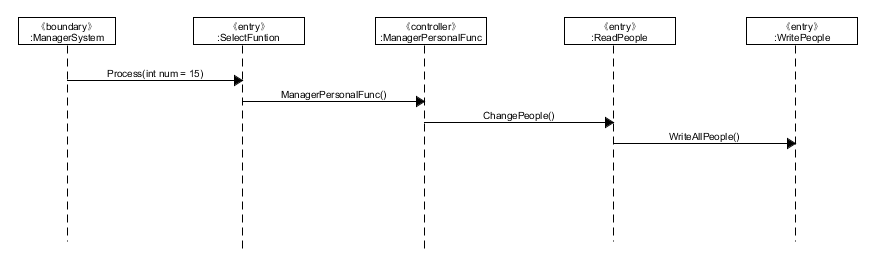


业主个人信息管理

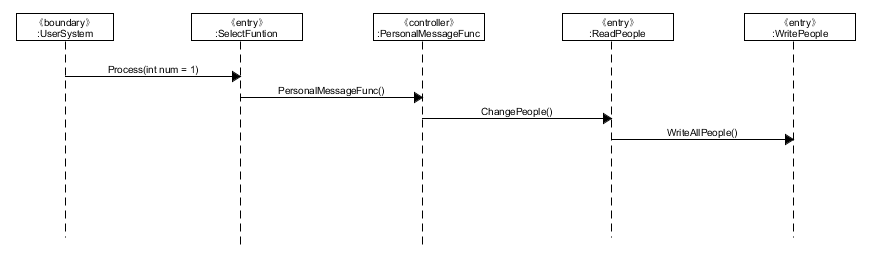


为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

管理员个人信息管理



业主个人信息管理



### 2.3.4修改密码用例分析

1）修改密码用例功能描述

业主或者系统管理员对自己登陆系统的密码进行修改。

2）修改密码用例类的析取

管理员修改密码

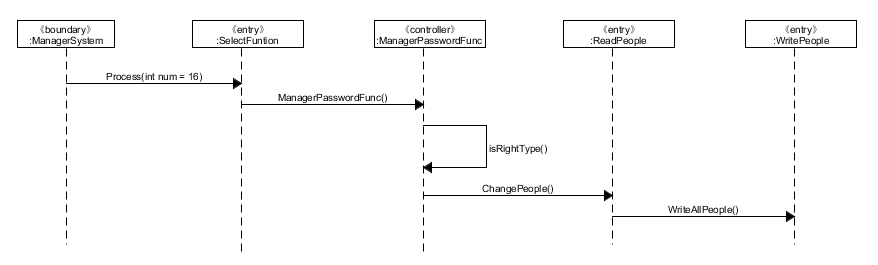


业主个人修改密码

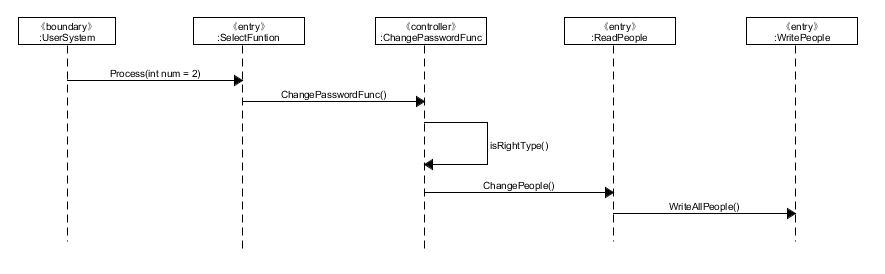


为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

管理员修改密码



业主修改密码



### 2.3.5业主管理用例分析

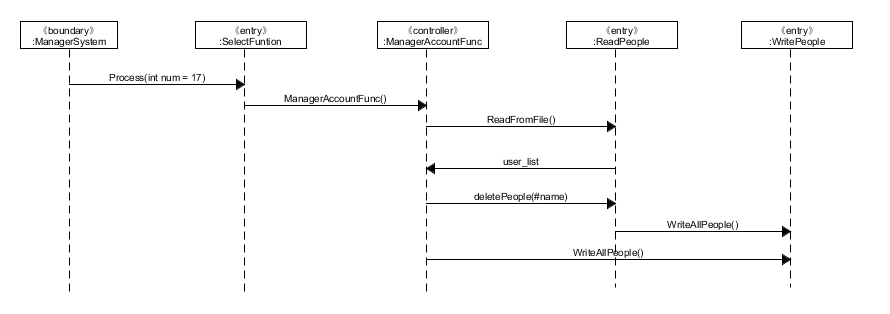
1）业主管理用例功能描述

管理员对业主的信息进行管理

2）业主管理用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.6投诉管理用例分析

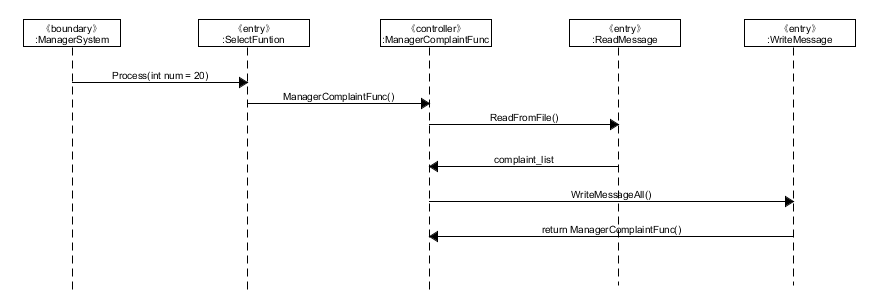
1）投诉管理用例功能描述

管理员对业主的信息进行管理

2）投诉管理用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.7收费管理用例分析

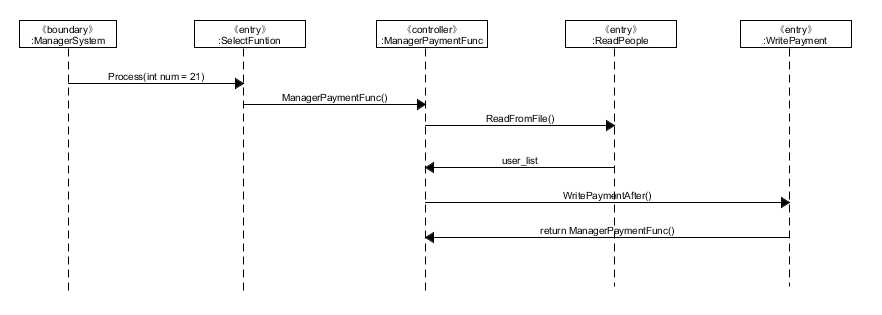
1）收费管理用例功能描述

管理员对业主的收费进行管理，添加业主收费信息

2）收费管理用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

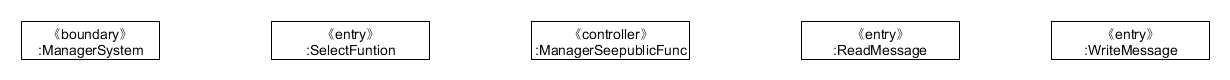


### 2.3.8公示信息管理用例分析

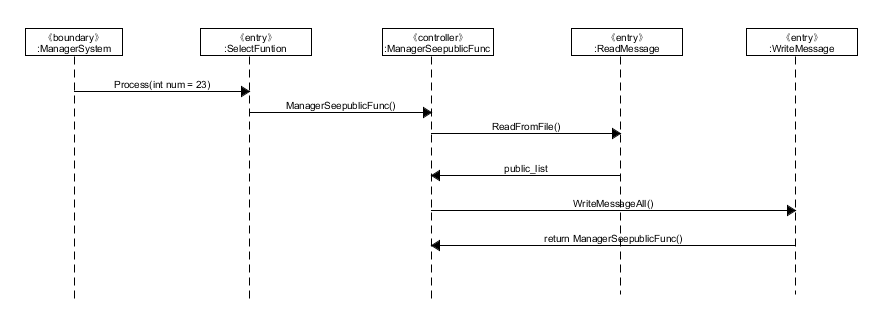
1）公示信息管理用例功能描述

管理员添加公示信息，对公示信息进行管理

2）公示信息管理用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.9报修管理用例分析

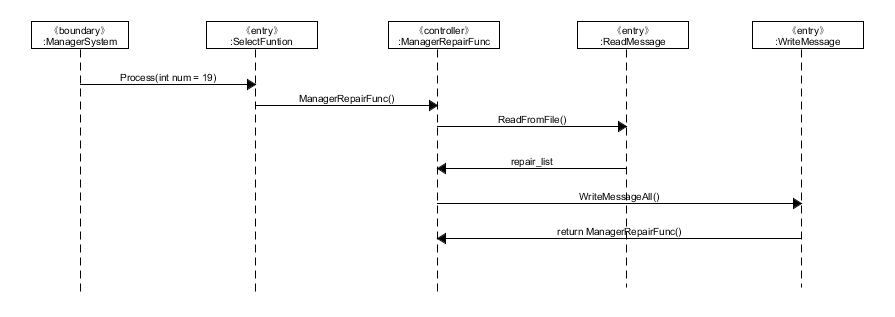
1）报修管理用例功能描述

管理员添加公示信息，对公示信息进行管理

2）报修管理用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

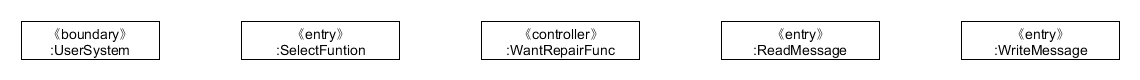


### 2.3.10提交报修用例分析

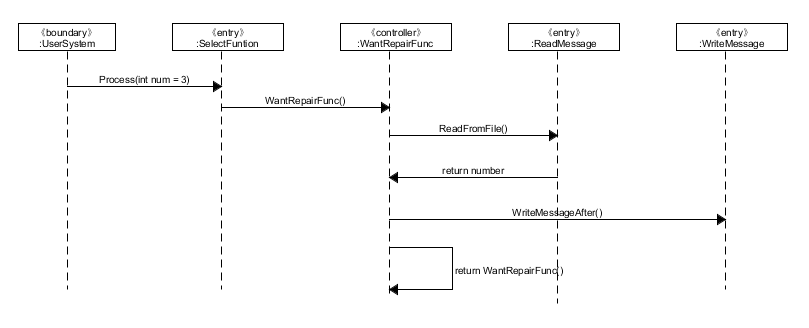
1）提交报修用例功能描述

业主向系统提交报修信息

2）提交报修用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。

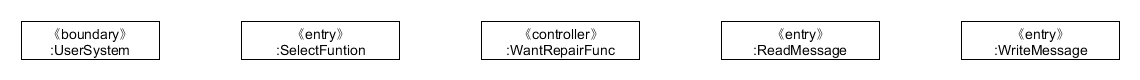


### 2.3.11查看报修用例分析

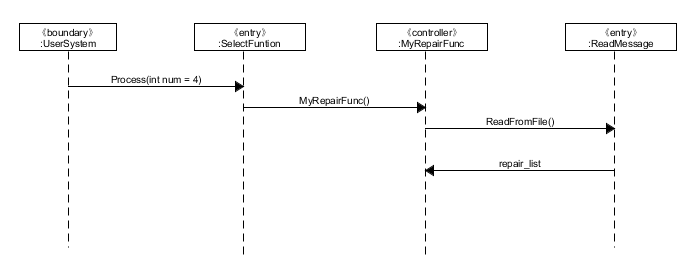
1）查看报修用例功能描述

业主在系统里查看自己的报修信息

2）查看报修用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.12提交投诉用例分析

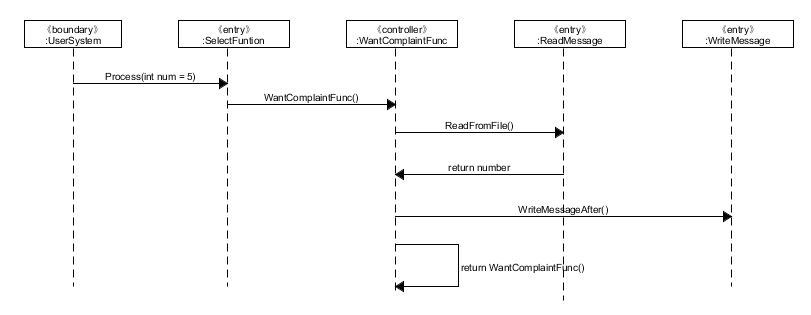
1）提交投诉用例功能描述

业主向系统提交自己的投诉信息

2）提交投诉用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.13查看投诉用例分析

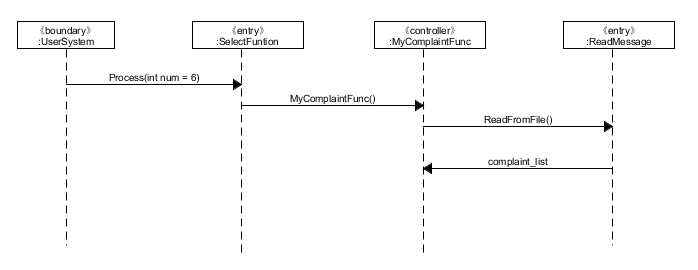
1）查看投诉用例功能描述

业主在系统里查看自己的投诉信息

2）查看投诉用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.14查看缴费用例分析

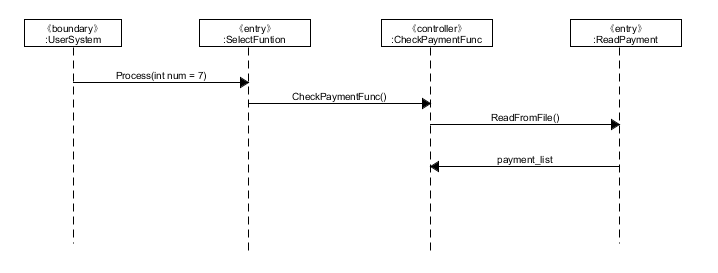
1）查看缴费用例功能描述

业主在系统中查看缴费信息

2）查看缴费用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



### 2.3.15查看公示信息用例分析

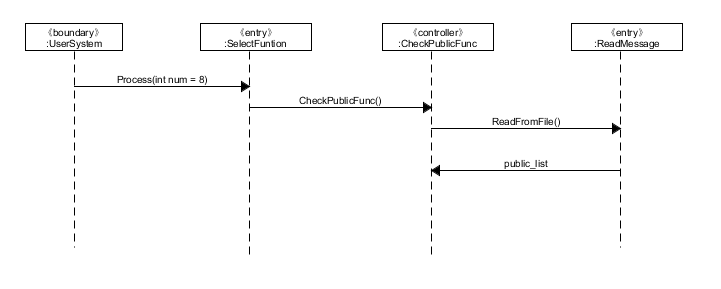
1）查看公示信息用例功能描述

业主在系统中查看管理员发布的公示信息

2）查看公示信息用例类的析取



为了进一步描述本用例的消息传递的情况，绘制本用例的时序图，如图所示。



# 第3章 小区物业管理系统子系统及其接口设计

## 3.1小区物业管理系统架构图

对于所有的设计类，我们按照它们在系统架构的三层中所处的位置打包成了View﹑Controller和Entity三个Package。则系统架构图如下所示。

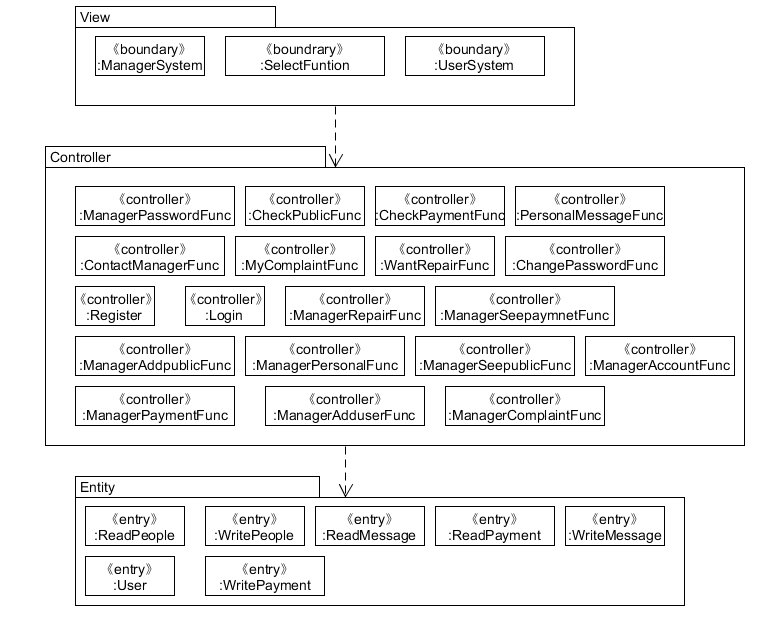


图17：小区物业管理系统架构图

## 3.2小区物业管理系统子系统划分及其接口设计

在小区物业管理系统架构的基础上，本项目组将该系统划分为2个子系统，分别是用户子系统和管理员子系统。每一个子系统都基于MVC三层架构，而且每个子系统都对应相应的业务。截止目前，本系统已经完成子系统划分，接下来将进行接口设计。

进入子系统的接口调用如下所示：

if (myLogin.Judge(login\_name, login\_password, login\_identity)) {

//登录成功

if (login\_identity == 0) {

//用户登录，进入用户子系统界面

EndDialog(IDCANCEL);

UserSystem myDialog;

myDialog.DoModal();

}

else{

//管理员登录，进入管理员子系统界面

EndDialog(IDCANCEL);

ManagerSystem myDialog;

myDialog.DoModal();

}

}

根据登陆的账号，密码跟身份，验证以后调用用户或者管理员系统界面，进入用户或者管理员子系统

### 3.2.1用户子系统设计

用户子系统下有各类业务调用程序接口：

#include "PersonalMessageFunc.h"

#include "ChangePasswordFunc.h"

#include "WantRepairFunc.h"

#include "MyRepairFunc.h"

#include "WantComplaintFunc.h"

#include "MyComplaintFunc.h"

#include "CheckPaymentFunc.h"

#include "CheckPublicFunc.h"

#include "ContactManagerFunc.h"

if (num == 1) {

//点击个人信息功能

v = new PersonalMessageFunc();

v->Create(IDD\_FORMVIEW, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 2) {

//点击修改密码功能

v = new ChangePasswordFunc;

v->Create(IDD\_FORMVIEW1, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 3) {

//点击我要报修功能

v = new WantRepairFunc;

v->Create(IDD\_FORMVIEW2, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 4) {

//点击我的报修功能

v = new MyRepairFunc;

v->Create(IDD\_FORMVIEW3, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 5) {

//点击我要投诉功能

v = new WantComplaintFunc;

v->Create(IDD\_FORMVIEW4, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 6) {

//点击我的投诉功能

v = new MyComplaintFunc;

v->Create(IDD\_FORMVIEW5, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 7) {

v = new CheckPaymentFunc;

v->Create(IDD\_FORMVIEW6, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 8) {

v = new CheckPublicFunc;

v->Create(IDD\_FORMVIEW7, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 9) {

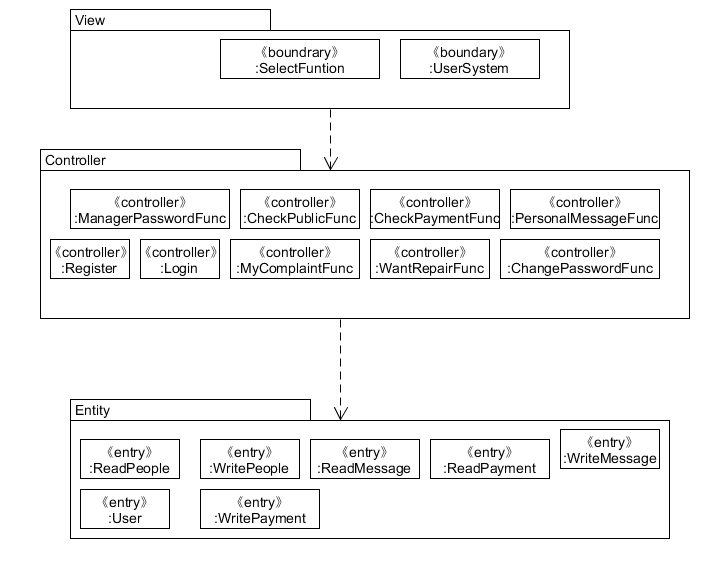
v = new ContactManagerFunc;

v->Create(IDD\_FORMVIEW8, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

用户子系统的架构图如下所示：



### 3.2.2管理员子系统设计

管理员子系统下有各类业务调用程序接口：

#include "ManagerPersonalFunc.h"

#include "ManagerPasswordFunc.h"

#include "ManagerAccountFunc.h"

#include "ManagerAdduserFunc.h"

#include "ManagerRepairFunc.h"

#include "ManagerComplaintFunc.h"

#include "ManagerPaymentFunc.h"

#include "ManagerSeepaymnetFunc.h"

#include "ManagerSeepublicFunc.h"

#include "ManagerAddpublicFunc.h"

else if (num == 15) {

v = new ManagerPersonalFunc;

v->Create(IDD\_FORMVIEW9, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 16) {

v = new ManagerPasswordFunc;

v->Create(IDD\_FORMVIEW10, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 17) {

v = new ManagerAccountFunc;

v->Create(IDD\_FORMVIEW11, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 18) {

v = new ManagerAdduserFunc;

v->Create(IDD\_FORMVIEW12, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 19) {

v = new ManagerRepairFunc;

v->Create(IDD\_FORMVIEW13, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 20) {

v = new ManagerComplaintFunc;

v->Create(IDD\_FORMVIEW14, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 21) {

v = new ManagerPaymentFunc;

v->Create(IDD\_FORMVIEW15, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 22) {

v = new ManagerSeepaymnetFunc;

v->Create(IDD\_FORMVIEW16, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 23) {

v = new ManagerSeepublicFunc;

v->Create(IDD\_FORMVIEW17, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

else if (num == 24) {

v = new ManagerAddpublicFunc;

v->Create(IDD\_FORMVIEW18, GetActiveWindow());

v->ShowWindow(SW\_SHOWNORMAL);

}

管理员子系统架构图如下所示：

