****

**餐厅预订平台**

**项目组成员：梁智昊 15331187**

**林 淼 15331202**

**廖颖泓 15331191**

**刘 阳 15331218**

**刘俊君 15331214**

**明友芬 15331242**

**林泽云 15331207**

**指导老师： 衣 杨**

目录

**[一、需求分析 3](#_Toc13309)**

[1.1问题描述 3](#_Toc15023)

[1.2用例图 10](#_Toc22674)

[1.3用例归约 11](#_Toc27502)

[1.4术语表 11](#_Toc6012)

[1.5补充归约 12](#_Toc9951)

**[二、架构设计 12](#_Toc14787)**

[2.1架构描述 12](#_Toc2780)

[2.2架构图 12](#_Toc21362)

[2.3关键抽象 12](#_Toc31354)

**[三、用例分析 12](#_Toc9657)**

**[四、类与子系统设计 12](#_Toc6891)**

[4.1确定类的设计 12](#_Toc3765)

[4.2子系统划分 12](#_Toc20649)

[4.3子系统及其接口设计 12](#_Toc11395)

**[五、运行时架构设计 12](#_Toc1726)**

[5.1分析本系统的并发需求 12](#_Toc30182)

[5.2识别出相应的进程和线程 12](#_Toc6485)

[5.3描述相应的进程和线程 12](#_Toc23807)

# 一、需求分析

## 1.1问题描述

### 项目目的

本项目将建立一个网站，网站名为“餐厅订座平台”。建立此网站主要目的在于：

1. 顾客提前预订座位节省时间精力；
2. 顾客在订座的同时还可以选择订餐；
3. 餐厅通过订座情况提前准备。

### 1.1.2平台规定

1. 订座时间分为3个时间段：上午（9:00-11:00）中午（12:00-14:00）晚上（18:00-20:00）；
2. 餐厅可选座位为40个；
3. 顾客可自由选择任何座位，包括拼座；
4. 顾客在预定座位的时候可以选择订餐，也可以不订餐，到了店里再点餐；
5. 顾客只能当天预订座位，逾期平台将自动取消其所定座位。

### 1.1.3用户分类

使用该系统的用户分为两类：顾客，管理员。管理员使用权限大于顾客，体现为如下功能：

顾客可使用的功能如下：

1. 点击注册功能，输入唯一标记的登录手机号、登录密码即可成功注册，以上所有输入信息将转化为条目被添加到网页自带的**顾客信息列表**中，并由管理员进行管理；
2. 注册后顾客可使用“预订座位”、“取消订座”、“预定餐品”三种功能。进入“预订/取消”界面，选择座位，确定订座时间后点击“提交”，平台将自动生成“订单”，顾客返回网页主页之后点击“查看订单”即可查看订座情况；预定座位之后，顾客可进入“点餐”界面，选择要预定的菜品，点击“提交”，平台将自动生成“订单”，顾客返回网页主页之后点击“查看订单”即可查看顾客的订座情况和点餐情况。
3. 任何未注册的顾客不具有“预订座位”、”“取消订座”、“预定餐品”和“查看订单”4种功能。未注册顾客可以访问网页首页信息。

管理员可使用的功能如下：

（1）管理员点击网页首页“管理员”，输入员工id以及唯一的登录密码即可进入管员模式；

（2）在管理员模式下，管理员可使用“预订座位”和“取消订座”两种功能标记餐厅内的座位状态。使用该功能之前不受座位2种状态的影响；

（3）管理员点击“查看订单”使用第3种功能，输入任何顾客的登录名即可查看该顾客的订单信息，包括：座位标号、订座时间以及顾客所预定的菜品。

### 1.1.4 网页界面说明

该网页分为以下7个界面：

1. 首页
2. 注册界面
3. 登录界面
4. 管理员界面
5. 预订/取消座位界面
6. 订餐界面
7. 查看订单界面
8. 顾客信息界面

各个界面的功能说明如下：

1. 首页

如图1.1 所示，首页顶端的内容包括：首页标志、登录按钮、注册按钮、管理员按钮、预订/取消按钮、查看订单按钮、餐厅全貌图片、餐厅地址信息、餐厅优惠活动、餐厅菜色推荐、剩余座位提醒、联系方式。

其中所有为按钮的标志点击之后都可进入相应的特定页面，例如点击登录按钮，顾客将进入登录界面。

其中剩余座位提醒将以3分钟为间隔自动更新剩余可选座位数量，用户必须刷新网页才能显示最新剩余座位数。

首页可被任何用户访问，包括未注册的顾客，但是未注册顾客能点击的按钮只有“注册”和“登录”按钮。已注册顾客可点击除“管理员”之外的全部按钮。管理员可点击首页的全部按钮。



订座/取消/订餐

联系方式

菜色推荐

优惠活动

餐厅地址

餐厅全貌

剩余座位数：

xx

查看订单

欢迎使用餐厅订座系统

登录

注册

管理员

首页

图1.1 首页

1. 注册界面

如图1.2所示，未注册用户输入登录用户名，密码，再确认（密码），电话4种信息后，点击最下面的“确定”按钮即可注册成功。



注册

确认

电话

再确认

密码

用户名

图1.2 注册界面

1. 登录界面

如图1.3所示，登录界面要求已注册用户输入用户名和密码，输入后点击确认即返回首页使用其他功能。



确认

密码

用户名

登录

图1.3 登录界面

1. 管理员界面

如图1.4所示，在进入管理员界面之前，平台也要求管理员进入登录界面输入用户名和密码，系统认证通过后即可进入管理员界面。

界面包括：管理员界面标记（左上角），中间3种按钮，分别是查看顾客信息按钮、预订/取消按钮、查看订单按钮。点击相应的按钮进入不同的界面，例如点击查看顾客信息按钮即进入顾客信息界面。



管理员

查看订单

预订/取消

查看顾客信息

图1.4 管理员界面

1. 预订/取消座位界面

如图1.5所示，界面显示整个餐厅内40个座位的位置，右边显示时间订座的可选时段，顾客一次性选择多个座位，最多可选40个，选择之后点击右边“确定时间”按钮选择时间，最后点击界面下方的“预订”和“取消”两种按钮来选择订座还是取消订座。



40

38 39

36 37

34 35

33

32

30 31

28 29

26 27

25

23 24

21 22

19 20

15 16 17 18

11 12 13 14

7 8 9 10

6

5

4

3

2

1

取消

预订

确定时间

晚上：18:00-20:00

中午：12:00-14:00

上午：9:00-11:00

用户名

预订/取消

图1.5 预订/取消座位界面

1. 订餐界面

如图1.6所示，用户在预定座位完成之后，点击“订餐”即可进入到如下图所示的订餐界面：



图1.6 预定餐品界面

客户可以根据菜品简介，选择自己需要的菜品，并提交。

1. 查看订单界面

如图1.7所示，顾客进入之后，即可查看当日的所有预订座位标号。

而管理员进入之后，无需输入用户名，页面显示除图中“用户名”之外的全部信息。管理员将看到3个时段所预订的所有座位号。



晚上

中午

上午

座位号

座位号

座位号

用户名

查看订单

图1.6 查看订单界面

1. 顾客信息界面

如图1.7所示，管理员输入顾客的用户名之后，显示用户的联系电话以及顾客在当日3个特定时间段内的订座座位标号。



座位号

上午

中午

晚上

联系电话

用户名

座位号

座位号

图1.7 顾客信息界面

## 1.2用例图



图 1.8 用例图

## 1.3用例归约

### 1.3.1 登录

#### （1）简要描述

登录即要求顾客输入登录手机号、登录密码。管理员登陆则使用特定的管理员账号。该平台会从后台的登录手机号条目中检查该手机号和密码是否匹配，如果匹配成功即可使用该平台的订座/取消等功能。

#### （2）参与者

顾客、管理员

#### （3）场景描述

顾客打开本平台网页，点击“登录”按钮，进入登录页面。输入登录手机号和登录密码。平台使用后台的顾客信息条目核对登录手机号和登录密码是否匹配，如果匹配，页面返回首页，顾客将被允许使用预订/取消、查看订单功能。如果不匹配，页面提示“登录失败！用户名或密码错误”；

管理员打开平台网页，点击“管理员”按钮，进入登录页面。输入唯一管理员用户名和唯一的密码、平台使用后台的管理员条目检查管理员用户名和密码是否匹配，如果匹配，页面进入“管理员界面”，管理员可以使用预订/取消座位、查看订单、查看顾客信息3种功能。

#### （4）前置条件

顾客和管理员要打开本平台网页，并且已经注册过登录用户名和登录密码。

#### （5）后置条件

顾客和管理员登录失败会有窗口提示，登录成功会进入新页面。

#### （6）事件流

#### 1

图 1.8 登录事件流

### 1.3.2 预订

#### （1）简要描述

预订座位即在顾客成功登陆了本平台网页以后，选择系统中尚未被预订的座位进行预订。当顾客选中的座位确定后，顾客可以进行预订菜品。

#### （2）参与者

顾客

#### （3）场景描述

顾客成功登录了本网页平台以后，在平台的主页面能够看到预定座位和预定菜品的按钮。顾客可以根据需要选择不同座位数的桌子，顾客选择桌子可以点击预定菜品按钮进行订餐，或者只预定座位而不提前订餐。顾客在预定菜品前需要先预定座位，否则会提示先预定座位。

#### （4）前置条件

顾客成功登陆网页平台、选择的座位未被预订

#### （5）后置条件

可预定的座位数量减少，更新数据库状态。

#### （6）事件流



图1-9预定座位事件流

### 1.3.3 取消座位

#### （1）简要描述

本用例描述顾客或者管理员对已选座位的取消过程：

顾客取消订位：当顾客已经成功的完成了餐桌的预定了，但是因为顾客自身的原因想要取消订位的时候就可以选择取消订位，在顾客取消之后在后台数据库和取消座位界面都会有相应的更改；

管理员取消订位：当餐厅出现什么零时性的状况不能为顾客提供餐饮服务的时候，管理员可以取消任何顾客的订座，并且在后台数据库实时的更新对餐桌空位的更改。同时在取消座位界面也会实时的显示出来。

#### （2）参与者

顾客，管理员

#### （3）场景描述

取消座位分为顾客取消和管理员取消两种情形：

顾客取消：顾客已经预定好了餐桌位置，但是顾客想要取消预定。如果此时顾客已经退出了系统，那么顾客就需要重新打开平台网页以一般模式登陆进去，如果顾客没有退出登录的话就不需要重新登录。顾客点击平台首页上的“订座/取消”按钮，进入“订座/取消界面”，选中顾客已经预定的座位之后再点击“取消”，就可以在后台数据库更改顾客的订位信息，取消订位，并且返回到取消了订位后的“预定/取消座位界面”，将顾客已经取消的座位显示为空。完成操作之后顾客可以选择重新预定座位，或者退出登录。

管理员取消：管理员打开平台网页以管理员模式登陆进去，点击平台上的“订座/取消”按钮，进入“订座/取消界面”，选中需要强行取消的顾客订单，再点击“取消”，就可以在后台数据库中取消顾客的订位，并且返回到取消了订位后的“预定/取消座位界面”，将顾客已经取消的座位显示为空。完成操作之后即可选择退出登录。

#### （4）前置条件

顾客已经成功订座或者有需要管理员强行取消的订座。

#### （5）后置条件

后台数据库会实时的更新顾客和管理员对订单的更改，并且“订座/取消界面”也会实时的显示出更改后的界面。

#### （6）事件流



图1.10取消订位活动图

### 1.3.4 查看订单

#### 简要描述

顾客成功登录的情况下，可以查看当前订单和历史订单，订单显示有效时间、有效地点、预定时间等相关信息。管理员在成功登录的情况下，可以查看餐桌各个时间段的预约情况，默认显示当前时间段餐桌的预约情况。

#### 参与者

顾客，管理员

#### 场景描述

对于顾客，在登录的情况下可以查看自己的订单，订单情况分为两种，一种是已完成（即历史订单），一种是进行中（当前有效订单），每一条订单信息包含有效时间段、有效地点、预定时间等相关信息。

对于管理员，在登录的情况下可以查看餐桌的预约情况，进入查看界面后，默认显示的是当前时间段的餐桌预约情况，已经预约的餐桌显示黑色，未预约的餐桌显示浅色，并且显示当前已经预约的桌号列表和未被预约的桌号列表，桌号列表按照所容纳人数升序显示。当管理员选择不同的时间段时，界面会显示所选时间段餐桌的预约情况。

#### 前置条件

顾客和管理员只有在已经注册并且成功登录的情况下才能够查看订单信息。

#### 后置条件

当数据库中的餐桌预约情况发生更改时，显示的订单信息实时发生改变。

#### 事件流



图1.11查看订单活动图

### 1.3.5 注册

#### （1）简要描述

注册即要求顾客输入登录手机号、登录密码。该平台会从后台的已有的用户名条目中检查用户输入的用户名是否存在，如果存在则拒绝用户以此手机号来注册，如果该手机号没有使用过则可成功注册。

#### （2）参与者

顾客

#### （3）场景描述

顾客打开本平台网页，点击“注册”按钮，进入注册页面。用户需要输入新的登录手机号、登录密码、确认密码。平台使用后台的顾客信息条目核对用户输入的手机号是否存在，如果存在，页面弹出窗口，页面提示“手机号已存在”。用户输入密码和确认密码，如果两者不相同，则页面弹出窗口，页面提示“密码和密码确认不一致！”如果没有错误发生，则提示注册成功并返回首页。

#### （4）前置条件

顾客成功打开首页。

#### （5）后置条件

顾客和管理员注册失败会有窗口提示，登录成功会进入新页面。注册成功后，后台数据库存入用户信息。

#### （6）事件流



图 1.11 注册活动图

### 1.3.6 管理顾客信息

#### （1）简要描述

顾客信息维护即根据系统已注册的所有用户信息，对部分用户进行必要的管理员操作。具体操作为添加未注册的新用户、删除数据库中已注册用户、更新已注册用户信息。

#### （2）参与者

管理员

#### （3）场景描述

管理员进入顾客管理页面，对顾客信息进行添加、删除和修改。用户登录系统，系统验证用户管理员身份。如果身份验证成功，用户以管理员身份登录成功。如果身份验证失败，用户页面弹出窗口，页面提示“非管理员无法登录”，返回登录界面。用户登录成功后，系统列出用户表中所有注册用户的用户名以及选座信息，用户选择选择所需要进行的操作，如果需要添加新用户，系统首先检查该用户是否存在，如果不存在，则系统向出具看写入新的用户信息，如果存在，则系统无法对数据库进行操作。如果需要删除列表中的旧用户，系统弹出对话框，确认是否删除，得到确认后，系统从数据库中删除已选中的用户。如果需要修改列表中的旧用户，用户输入修改后的用户信息，系统检查输入的信息是否合法，检查通过后系统将从数据库中删除旧用户的信息并添加新的用户替代该用户。

#### （4）前置条件

本用例开始前用户登录餐厅管理员账号

#### （5）后置条件

数据库完成对应条目的添加、删除、更新

#### （6）事件流



图 1.13 维护顾客信息活动图

## 1.4术语表

1. 顾客

属于平台的使用者之一，已经注册的顾客可以使用登录、预订/取消座位、查看订单功能，未注册的顾客无以上功能，只有查看网页信息的权利。

1. 管理员

属于平台的使用者之一，管理员可以使用登录、预订/取消座位，查看顾客信息，查看订单功能，权限大于顾客。

1. 顾客信息条目

保存在平台后端的一个条目列表，每个顾客信息条目包含——顾客的登录手机号、顾客的登录密码、顾客预订的所以桌号和时间段）。

1. 管理员信息条目

保存在平台后端的一个条目，包含一个管理员登录用户名和一共管理员登录密码。

1. 座位

一共有40个相同的座位，每个座位有两种状态——被预订、未被预订。顾客和管理员可以通过使用预订/取消功能来改变任何一个座位的状态，并且管理员的权限大于顾客，即管理员可以强制改变顾客所选的座位的状态。

1. 订单

保存在平台后端的一个条目列表，每个订单条目包含——顾客的登录用户名、顾客的电话、顾客在本日所预订的3个时段所有座位的标号（1-40）。

## 1.5补充规约

1. 目标

本补充规约描述了该平台的非功能需求。

1. 范围

本补充规约适用于“餐厅订座平台”网站。

1. 参考

无

1. 可行性

本平台为网络平台，需要在网络环境中运行。

1. 可靠性

本平台开放的时间为：8:00-22:00

1. 安全性
2. 未注册的顾客只能查看网页首页信息；
3. 只有已注册和管理员才能适用预订/取消座位等功能；
4. 管理员的权限最高，可强制改变顾客的选择；
5. 顾客不可以改变其他顾客的任何信息，包括用户名、密码、预订座位。
6. 可支持性
7. 本平台支持中文界面；
8. 本平台支持windows操作系统。

# 二、架构设计

## 2.1架构描述

整个系统平台的架构采用MVC架构，分为模型层（M）、视图层（V）和控制器层（C）。MVC架构设计将业务逻辑、数据和界面显示分离开来。把相应的数据模型封装起来，通过视图层与用户进行交互，并且通过控制器层、实现层与层之间的逻辑控制。这样可以使得模型设计、界面设计、逻辑设计分离开来，而开发人员不用在改进和个性化界面及用户交互的同时重新编写业务逻辑。使得整个系统平台的开发更加清晰明了，更加模块化，从而提高开发效率。

### 2.1.1模型层

本系统平台的数据模型分为三部分：

（1）第一部分为系统平台的用户数据模型。其中包括了平台管理员、顾客两种子模型，有各自不同的权限和数据类型。平台管理员模型包括了管理员ID、联系方式、邮箱、密码等信息，平台顾客模型包括了顾客登录手机号、密码等信息。

（2）第二部分为系统平台的座位数据模型。其中包括了座位ID、是否被预定等信息。

（3）第三部分为系统平台的订单数据模型。其中包括了订单ID、订单关联顾客ID、订单关联座位ID及数量、订单生成时间、订单是否已完成等信息。

### 2.1.2视图层

本系统平台的视图层由以下几个页面视图组成：

（1）平台首页

（2）注册界面

（3）登陆界面

（4）用户个人信息页面

（5）座位预定页面

（6）查看订单页面

### 2.1.3控制器层

本系统平台的控制器层主要包括四个部分：

（1）第一部分为用户注册的逻辑控制。用户注册时需要提交手机号码、用户密码。控制器从视图层的表单中获得这些信息以后，作相应的验证处理后将这些数据存储到数据库中。并且控制相应的页面跳转。

（2）第二部分为用户登陆的逻辑控制。用户登录时需要输入手机号、密码、验证码。控制器从视图层提交的表单中获取到这些信息以后，作相应的验证处理后更改数据库的状态。并且控制相应的页面跳转。

（3）第三部分为座位预定的逻辑控制。用户在系统平台首页选择了心仪的座位后选择提交预订订单。控制器从视图层提交的表单中获取提交订单的用户ID、座位ID、预定时间后，判断该座位满足预订后生成相应的订单ID并且把数据存储在数据库中。把对应的座位对应的时间段修改为不可预订以后，控制相应的页面跳转。

（4）第四部分为用户订单取消的逻辑控制。用户在查看了自己的所有订单后以及管理员查看了某个订单后，可以选择取消对应的订单。控制器从视图层提交的表单中获取取消订单的用户ID、座位ID、预定时间后，把数据存储在数据库中。把对应的座位对应的时间段修改为可预订以后，控制相应的页面跳转。

（5）第五部分为用户个人信息修改的逻辑控制。用户在个人信息视图中可以选择修改部分个人信息，比如用户名、密码等。完成修改后控制器修改在数据库中对应的数据。

## 2.2架构图



图2-1 架构图

## 2.3关键抽象

按照不同的模块分类主要包含以下类：

（1）用户模块：User（抽象类）、Administrator（管理员）、Customer（顾客）

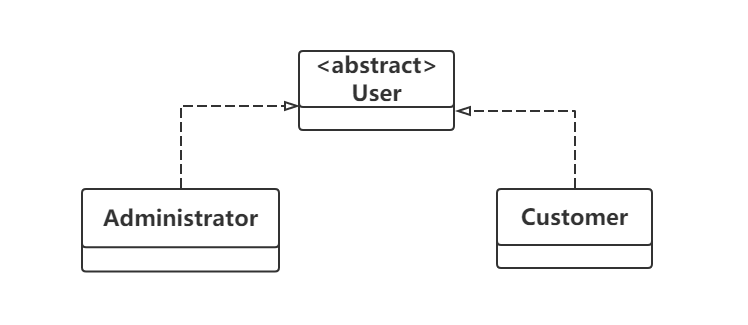


图2-2 用户模块

（2）座位模块：Seat



图2-3 座位模块

（3）订单模块：Order



图2-4 订单模块

（4）餐品模块：Meal



图2-4 餐品模块

# 用例分析

## 3.1补充用例规约

经检查，本项目用例规约比较完善，暂无补充。

## 3.2类的析取

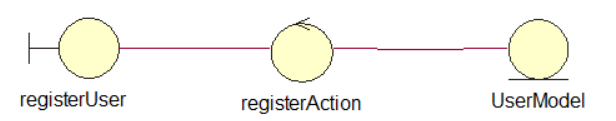
### 3.2.1注册用例析取

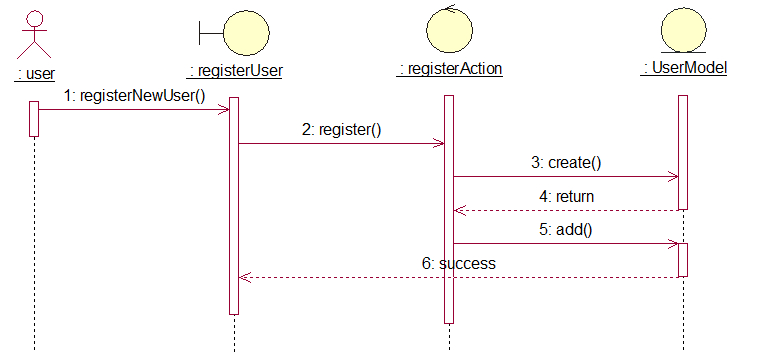
根据第一章系统需求分析中的用例规约，经分析可得该用例中的三种类。

边界类： registerUser。registerUser为注册用户信息填写页面。

控制类：registerAction。registerAction控制类负责处理关于注册的相关操作，对用户填写的信息进行操作，包括手机号码、密码是否合法以及是否存在的验证。

实体类：userModel。userModel实体类表示用户实体，主要存储用户的相关数据信息，包括密码、手机号码。





时序图的简单描述如下：

用户点击注册按钮，系统调用registerUser上的registerNewUser()方法，该方法以post方式将用户数据传给registerAction控制类的register()方法。register ()方法获取到post传过来的用户信息，调用UserModel实体类的create()方法创建用户对象，并进行自动验证和自动完成。自动验证表示对用户输入的信息进行验证，判断是否为合法数据。自动完成表示将数据插入数据库时会对部分数据进行处理，比如设置默认值。若创建失败，返回错误信息。若创建成功，再调用UserModel实体类的add()方法保存新活动到数据库。最后把反馈信息传给边界类。

### 3.2.2登录用例析取

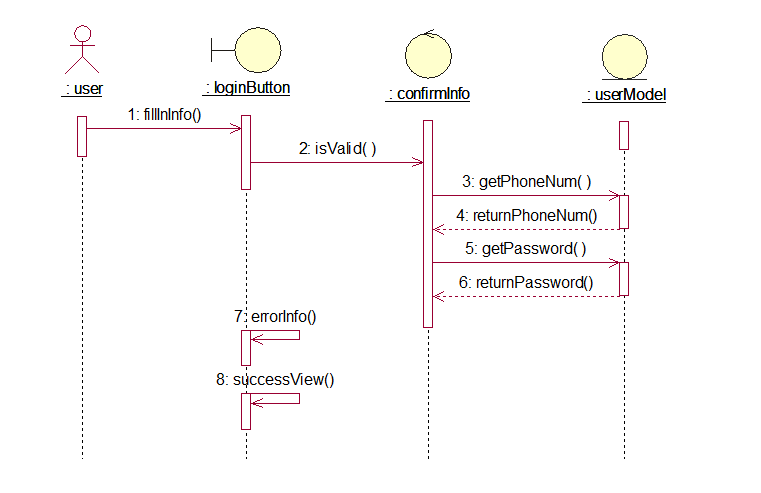
根据1.3.1中的用例规约，分析得到该用例的边界类、控制类和实体类如下：

边界类：loginAutton，用户点击登录按钮进入用户登录信息填写页面。

控制类：confirmInfo，负责对用户填写的信息进行操作，包括手机号码和密码是否合法以及是否存在的验证。

实体类：userModel，主要存储用户的相关数据信息，包括手机号码和密码。





时序图的简单描述如下：

用户首先点击登录按钮，调用loginAutton类的fillInInfo()方法填写用户信息，然后将信息传给控制类，控制类confirmInfo获取边界类传过来的信息，控制类得到相应的信息后调用isValid()函数对用户信息进行验证。在验证的过程中，控制类需要根据用户的手机号码到实体类userModel类中获取相关的用户信息并进行验证。如果验证成功，跳转到相应的用户界面；如果用户信息错误或者是用户不存在，返回错误信息给边界类。

### 3.2.3 订座用例析取

根据1.3.2中的用例规约，分析得到该用例的边界类、控制类和实体类如下：

边界类：loginButton，用户点击登录按钮进入用户登录信息填写页面；

bookingButton，顾客点击预定座位按钮；

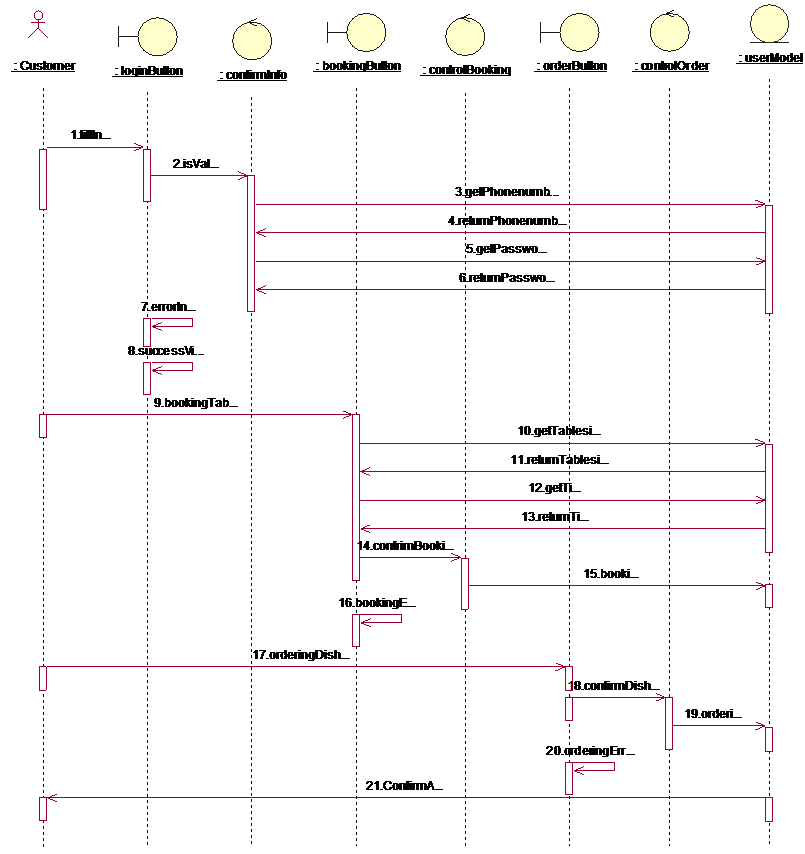
orderButton，顾客点击预定菜品按钮。

控制类：confirmInfo，负责对用户填写的信息进行操作，包括用户名、密码、手机号码是否合法以及是否存在的验证；

controlBooking，负责对用户填写的座位预订信息进行操作，包括桌子大小和用餐时间的确定；

controlOrder，负责对用户填写的菜品预订信息进行操作，包括菜品内容的确定；

实体类：userModel，主要存储用户的相关数据信息，包括用户名、密码、手机号码。



时序图的简单描述如下：

用户首先点击登录按钮，调用loginAutton类的fillInInfo()方法填写用户信息，然后将信息传给控制类，控制类confirmInfo获取边界类传过来的信息，控制类得到相应的信息后调用isValid()函数对用户信息进行验证。在验证的过程中，控制类需要根据用户的手机号码到实体类userModel类中获取相关的用户信息并进行验证。如果验证成功，跳转到相应的用户界面；如果用户信息错误或者是用户不存在，返回错误信息给边界类。然后用户调用bookingAutton类的bookingTable()方法进行座位预订。在预订的过程中，控制类需要根据用户的需要的桌子大小和用餐时间进行确认。如果确认成功，跳转到相应的用户界面；如果确认信息有冲突，返回错误信息给边界类。之后用户调用orderingAutton类的orderingDishes()方法进行座位预订。在预订的过程中，控制类需要根据用户的需要的桌子大小和用餐时间进行确认。如果确认成功，跳转到相应的用户界面；如果确认信息有冲突，返回错误信息给边界类。

### 3.2.4取消订座用例析取

根据1.3.3中的用例规约，分析得到该用例的边界类、控制类和实体类如下：

边界类：loginButton，用户点击登录按钮进入用户登录信息填写页面；

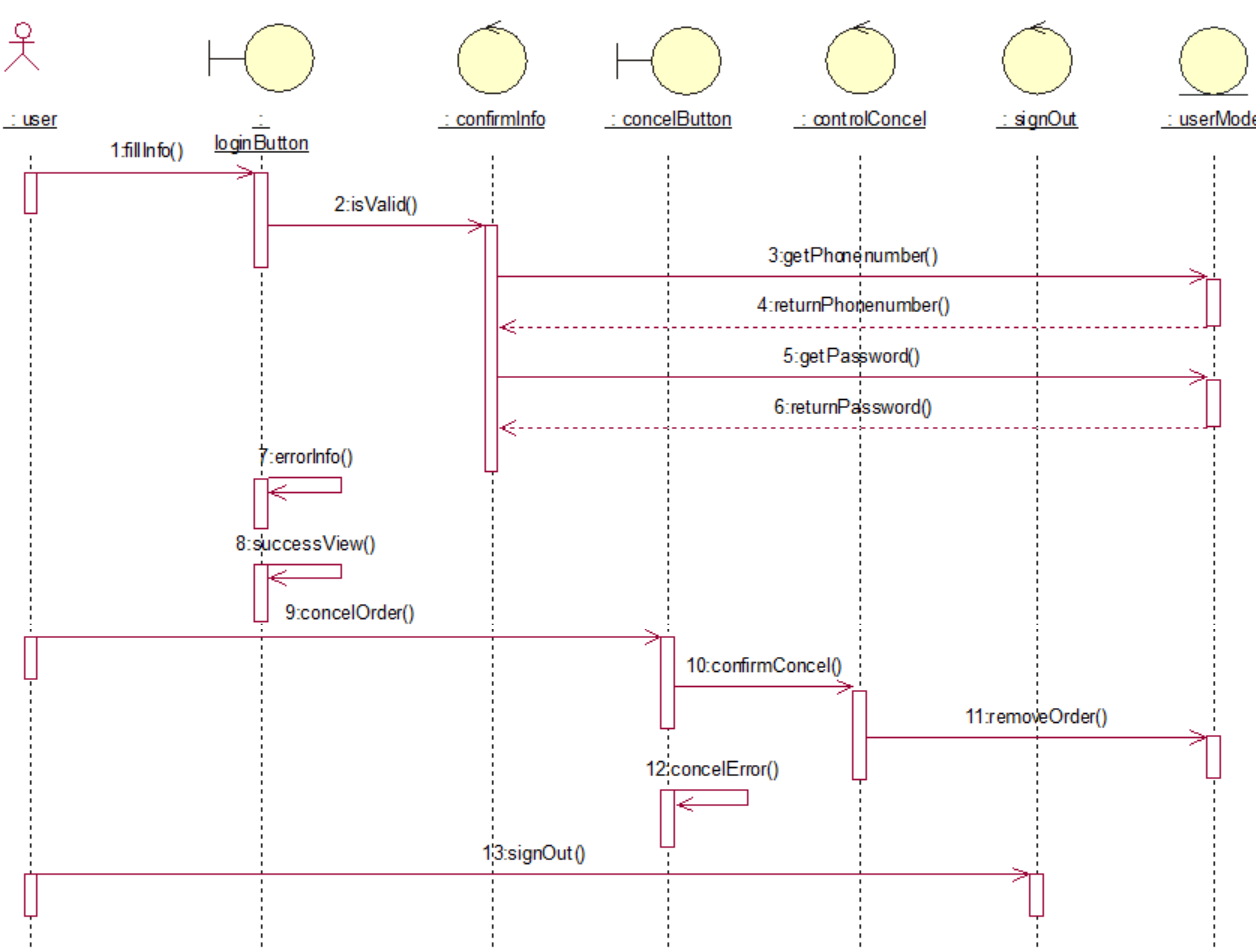
cancelButton，顾客点击取消订单按钮，取消某一个订单；

控制类：confirmInfo，负责对用户填写的信息进行操作，包括用户名、密码、手机号 码是否合法以及是否存在的验证；

controlConcel，控制订单是否成功取消；

signOut，控制用户退出当前系统。

实体类：userModel，主要存储用户相关数据与用户订单相关的信息。



时序图的简单描述如下：

用户首先点击登录按钮，调用loginAutton类的fillInInfo()方法填写用户信息，然后将信息传给控制类，控制类confirmInfo获取边界类传过来的信息，控制类得到相应的信息后调用isValid()函数对用户信息进行验证。在验证的过程中，控制类需要根据用户的手机号码到实体类userModel类中获取相关的用户信息并进行验证。如果验证成功，跳转到相应的用户界面；如果用户信息错误或者是用户不存在，返回错误信息给边界类；

顾客点击取消订单按钮，调用cancelButton类的cancelOrder()方法来取消订单，控制类controlCancel获取边界类传来的信息，调用confirmCancel()函数对订单进行取消，如果订单取消成功，调用uerModel类的removeOrder()方法将数据库的订单信息删除；如果订单取消失败，返回到取消订单界面；如果用户点击退出按钮，调用signOut()方法，退出登录。

### 3.2.5 管理顾客信息用例析取

根据1.3.3中的用例规约，分析得到该用例的边界类、控制类和实体类如下：

边界类：AdminUI：管理员界面，用于显示顾客的信息，提供添加用户、删除用户、修改用户信息功能；

控制类：Add：用于添加新用户

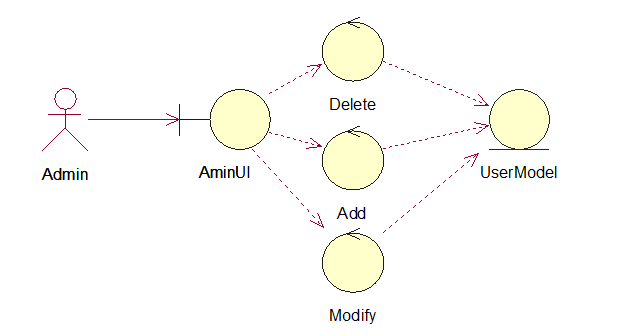


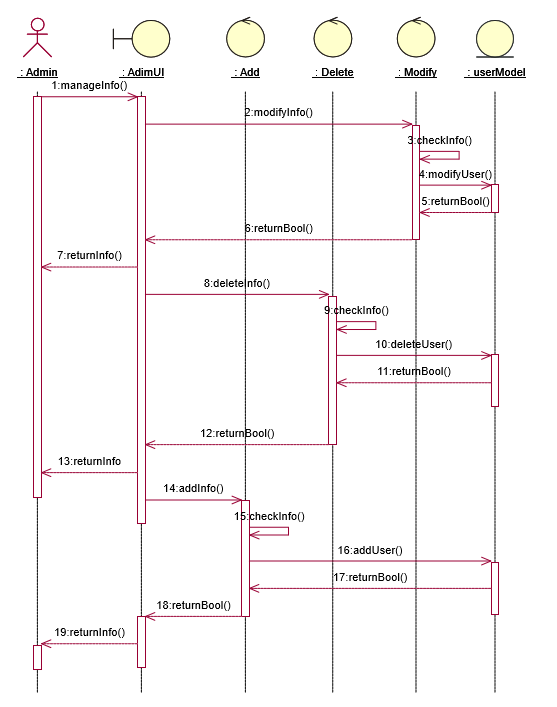
图 xx 管理顾客信息用例分析

图xx为该用例的时序图，简要描述如下：

管理员进入了管理员界面之后，可以考到所有的顾客信息，包括顾客的手机号，顾客的订单情况以及顾客的到店时间。

管理员可以通过管理员界面使用方法manageInfo()来添加新用户，删除已有用户或者修改用户信息。

例如管理员先使用实体类AdimUI的modifyInfo()方法来修改用户信息，控制类Modify首先检查修改的信息是否合法，调用自身的checkInfo()方法，再调用自己的方法modifyUser()方法来修改用户信息，并传递信息到实体类userModel中，该实体类修改用户信息之后返回一个布尔值给控制类Modify表示修改成功与否，控制类Modify也返回给边界类AdminUI一个布尔值显示出修改成功与否，接着边界类AdminUI将成功与否这个信息表达给管理员Admin。



图xx 管理顾客信息时序图

### 3.2.6 查看订单用例析取

根据1.3.4中的用例规约，分析得到该用力的边界类、控制类和实体类如下：

边界类：userUI， 用户页面，显示有订位，订餐，查看订单等信息的页面；

控制类： viewOrder，viewOrder控制类负责用户点击查看订单按钮后显示订单信息；

实体类：userModel，主要存储用户相关数据与用户订单相关的信息。

时序图的简要描述如下：

用户进入系统后，点击用户界面页上的“查看订单”按钮，系统调用userUI上的viewOrder()方法，该方法以get方式把请求提交给viewOrder控制类的view()方法。view()方法将调用userModel实体类的getOrder()方法获取全部订单信息，并用return()方法返回到viewOrder控制类。另外，控制类还有一个selectTime()方法，调用该方法将以get方式把时间参数time传给userModel实体类的getOrder()方法获取特定时间的订单信息，也用return()方法返回到viewOrder控制类。



# 四、类与子系统设计

## 4.1确定类的设计

## 4.2子系统划分

## 4.3子系统及其接口设计

# 五、运行时架构设计

## 5.1分析本系统的并发需求

## 5.2识别出相应的进程和线程

## 5.3描述相应的进程和线程