



酒店管理系统

组长&主讲人：14130110009 苏靖淞

PPT制作组员分工：

task4 应用建模：14130110011 魏哲 14130110013 陈奇冰
14130110014 林杜 14130110016 李帅

task5 web应用架构设计：14130110100 肖杰

task6 web应用设计：14130110006 彭硕 14130110060 张振宇

目录

1

项目简介

2

Web应用建模

3

Web应用架构设计

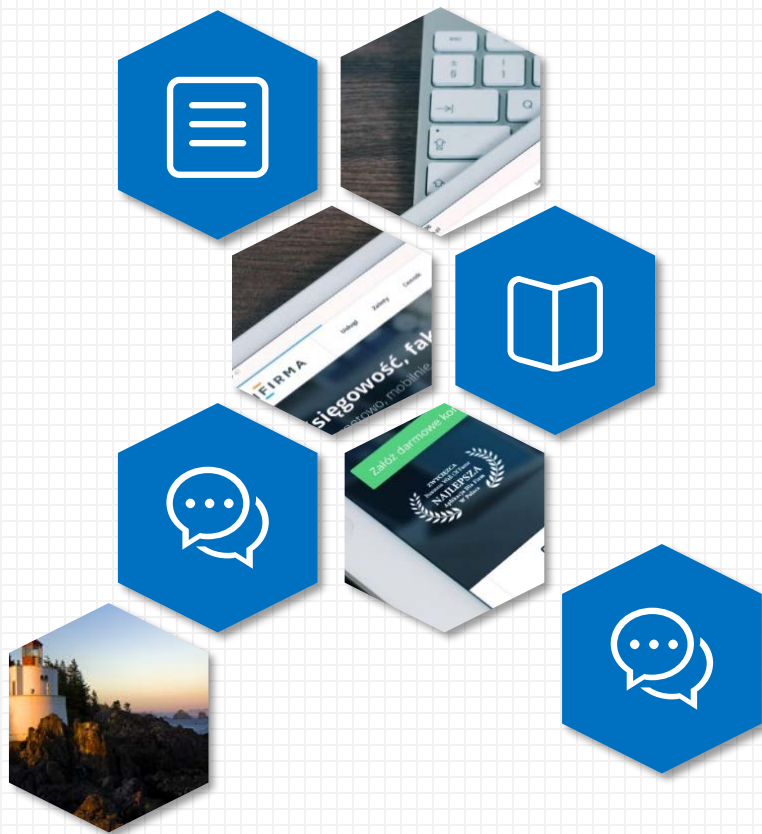
4

Web应用设计

前言

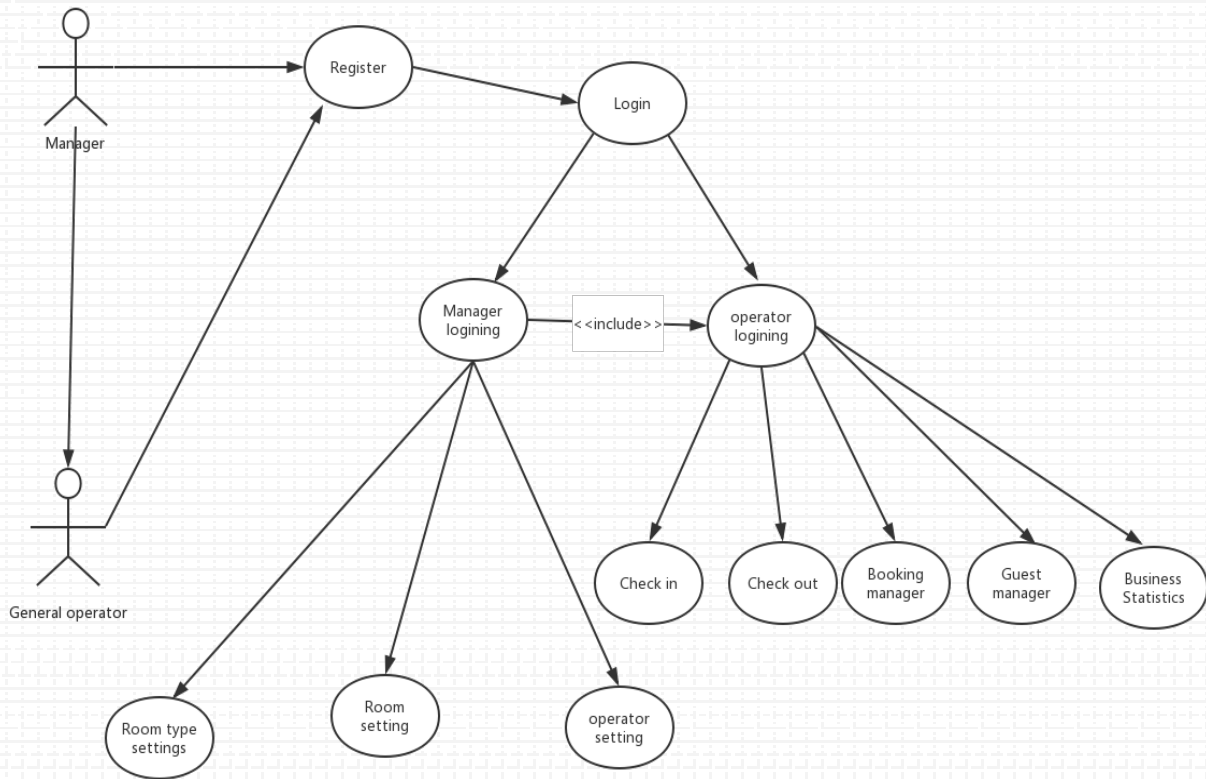
项目简介

本项目在基于b/s架构的基础上设计了一套酒店管理系统，项目集成了多种功能，如：顾客查询，登记，入住，费用结算等等。该方案采用Java Web技术同时利用了SQL Server数据库，以Eclipse作为平台，以B/S作为基础，开发了具有多层架构的。酒店管理系统主要分为前台和后台两个部分，其中前台包括五个模块供所有工作人员使用：入住登记，结账，预定，客户信息管理，业务统计。实现了酒店前台人员的常用管理功能。其中后台管理系统包括客房类型，客房设置，操作员设置模块。实现了客房的详细管理和操作人员的操作管理。利用该系统，酒店人员可以在前台实现办公自动化，同时系统提供后台数据分析，可以为酒店领导决策层提供相应经营参考，会员管理功能的实现会大大提高酒店服务水平，可以为酒店提高竞争力。



功能需求建模

静态建模



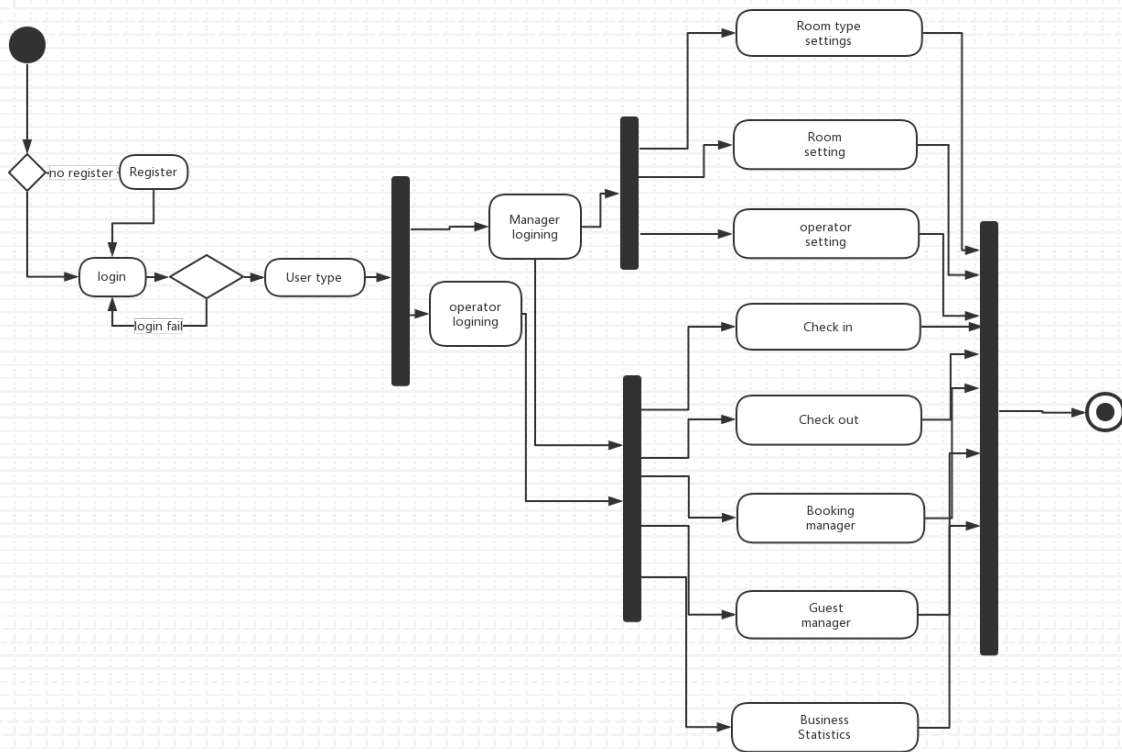
主要用户:

Manager
General operator

主要功能:

Register,
Login,
Check in,
Check out,
Booking manager,
Guest manager,
Business Statistics,
Room type settings,
Room setting,
operator setting

动态建模



简要流程:

如图所示，用户进入系统中先注册一个账号，然后登录，系统会判定账号密码的正确性，如果正确则进入下一步，如果不正确则返回登录错误。登陆后再判定用户类型是Manager还是operator，如果是operator则可以使用Check in，Check out，Booking manager，Guest manager，Business Statistic功能。如果是Manager，则除了operator的功能外，还可以使用Room type settings，Room setting，operator setting。

内容建模

静态建模

客户信息类（Guest）：封装了所有的客户信息。

客房类型信息类（RoomCatalog）：封装了所有的客房类型信息。

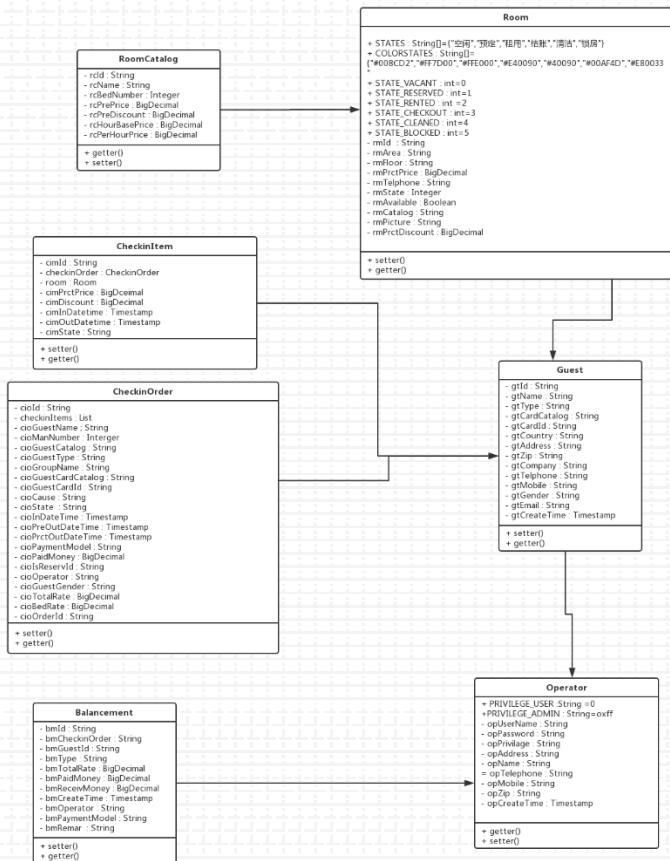
客房信息类（Room）：封装了所有的客房信息，包括客房状态数组、客房状态颜色数组等等。

操作员信息类（Operator）：封装了所有的操作员信息，类中定义了两个变量来表示操作员的两种权限。

入住登记信息类（CheckinItem）：封装了所有的客户入住登记信息。

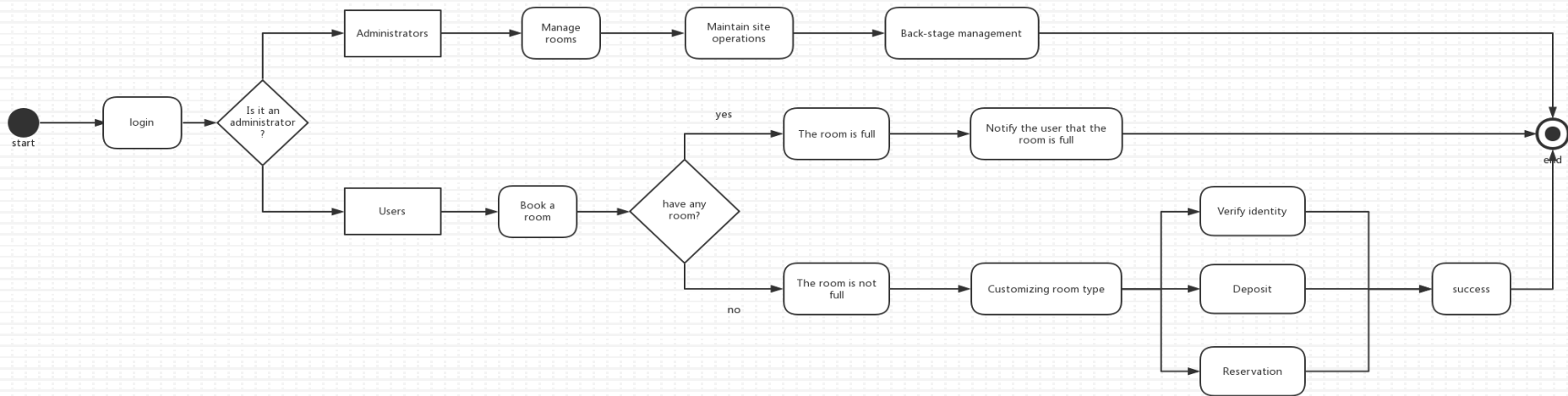
入住登记订单类（CheckinOrder）：封装了所有入住登记情况，以及订单信息。

结账信息类（Balancement）：主要保存客户的结账信息，包括客户编号、结账类型、结算金额等等信息。



内容建模

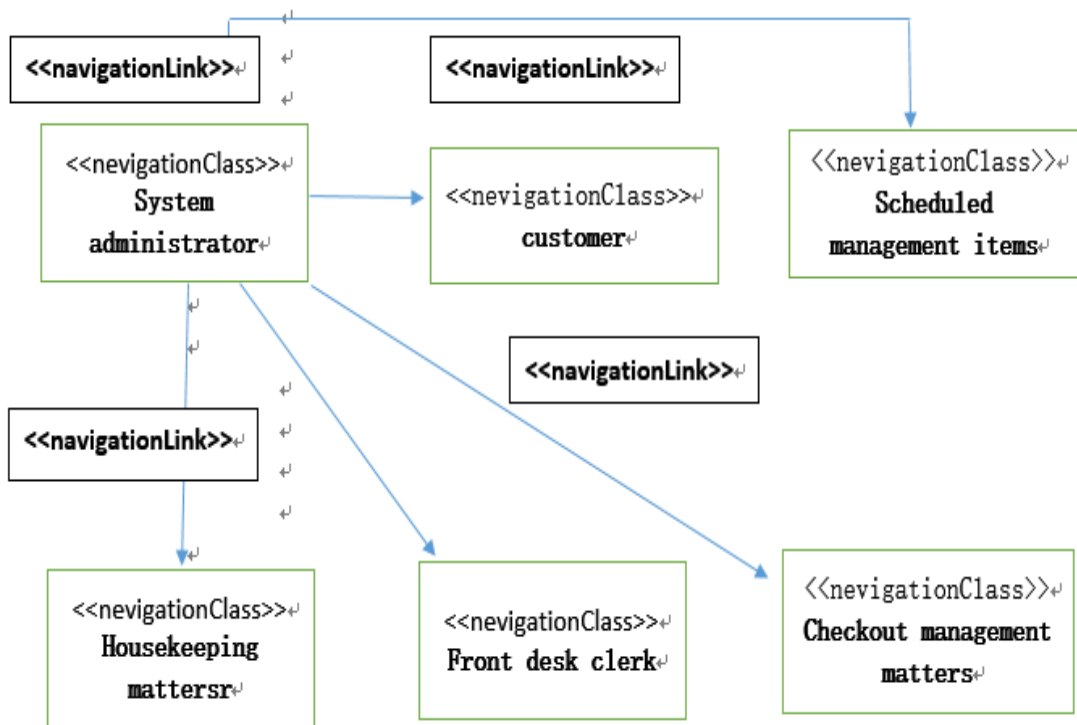
动态建模



状态图描述：从登陆开始会由操作员的权限不同进行分支，一支为管理员权限。管理员会进入管理系统并进行相关的高权限操作。一支为普通用户。普通进入系统后会进入预定房间状态，然后进行判断是否有空房存在。如果房间已满，则提示普通用户房间状态。如果房间未满，则进入房间类型选择界面，并进行后续的身份验证、支付押金、提交后台数据库预定成功。

超文本建模

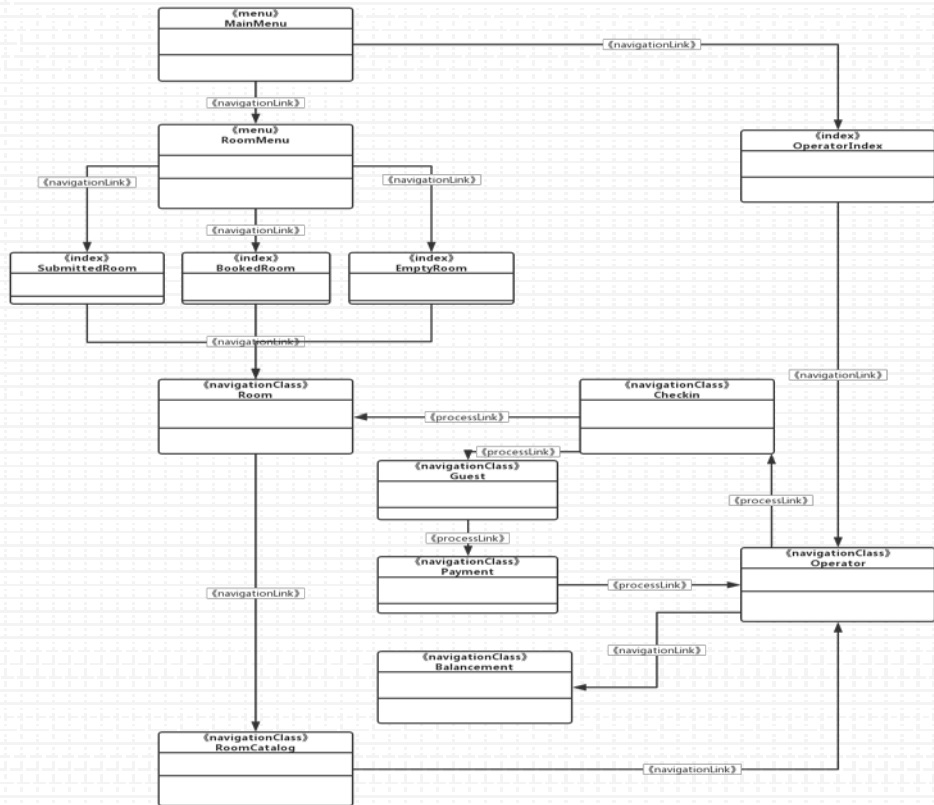
静态建模



静态超文本建模以内容建模为基础，主要强调超文本的结构，超文本模型模型同时被当作内容模型的一种特定视图因此成为导航视图。本例中以系统管理员身份登录后，可以进入房间管理页面及订房管理页面并查看顾客及前台职员信息，并有权进入管理界面对房间进行调度。以满足订房，退房等需求。

超文本建模

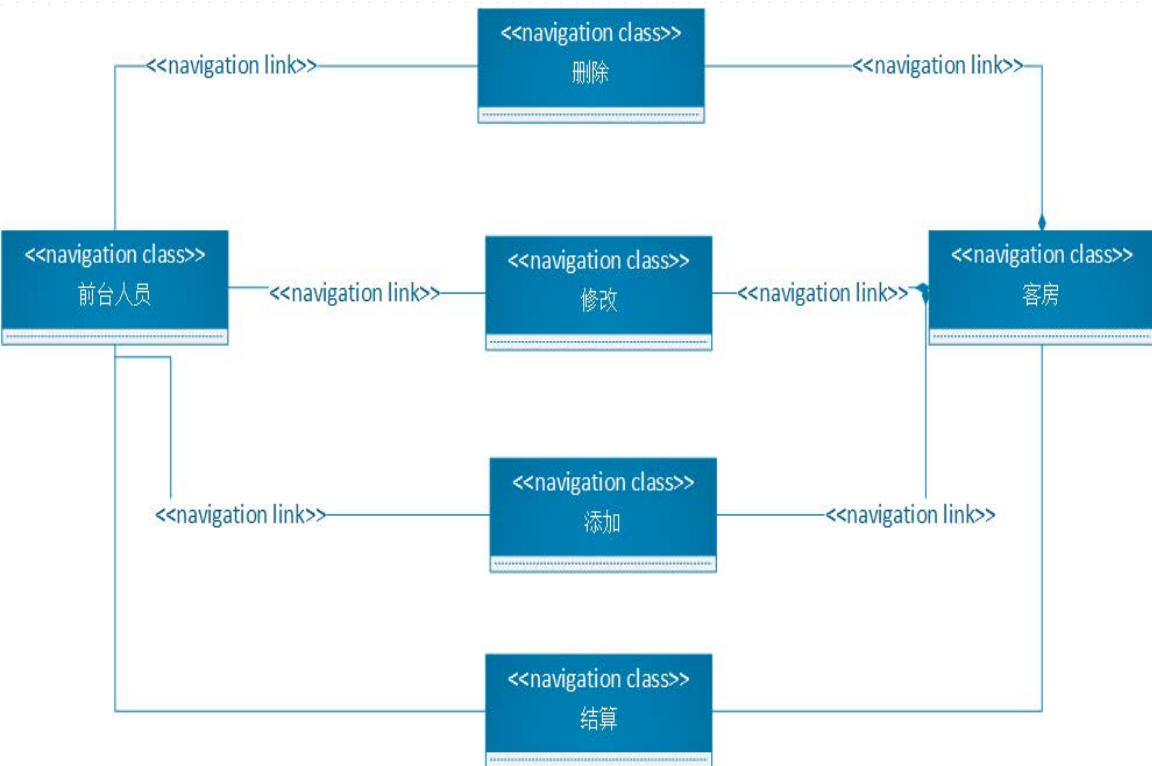
动态建模



动态超文本建模主要表现如何通过导航定位到节点，本例中我们以主菜单为入口，进入操作员注册及登录界面，若已经是系统许可拥有权限的操作员，则进入房间管理菜单，房间被分为已登记入住，已预订，及空房，进一步选择返回房间信息，操作员还可以进入登记入住界面，记录顾客信息，并获取顾客信息及房间信息生成支付信息，最后生成周期性的收支总结。

适应性建模

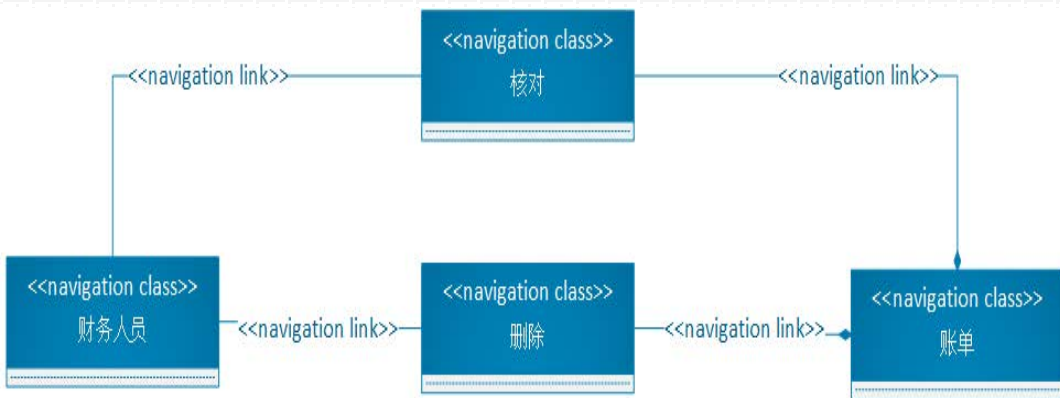
静态建模



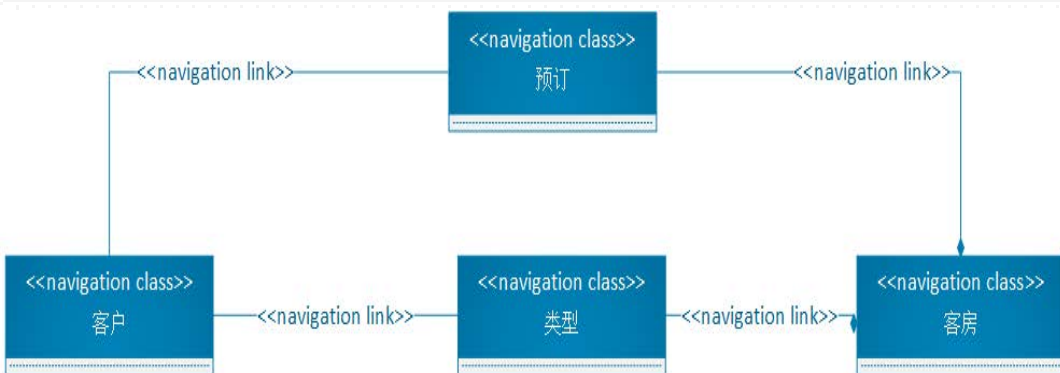
根据系统的使用范围以及其使用的上下文特性和不同特征不同需求，提供给用户合适的展示。

针对系统的直接使用者，即前台接待人员。每次客户前来消费，无论是订房或者订餐，客户都不会直接使用系统，直接使用系统的是前台接待人员。前台接待人员可以添加客房、修改客房信息、增添消费信息和根据客人消费情况进行结算。

适应性建模



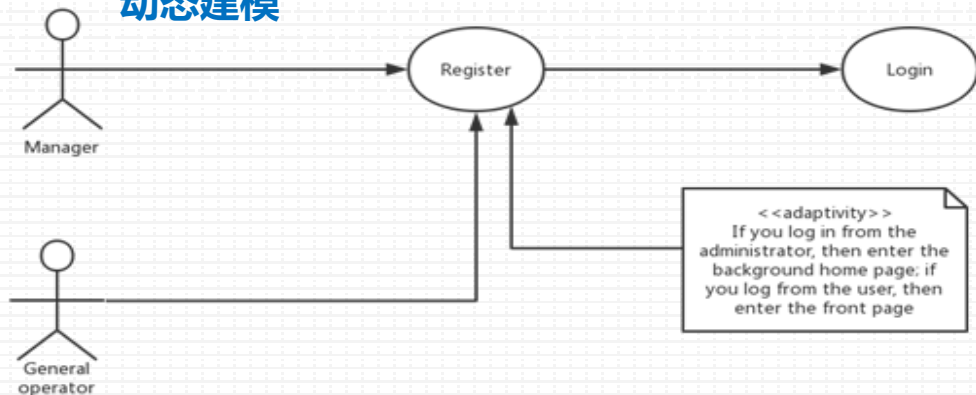
针对系统的使用者，即财务人员，他们的主要工作就是根据客户消费记录，以及收银记录与实际的收银进行核对，以及删除过期的消费记录。



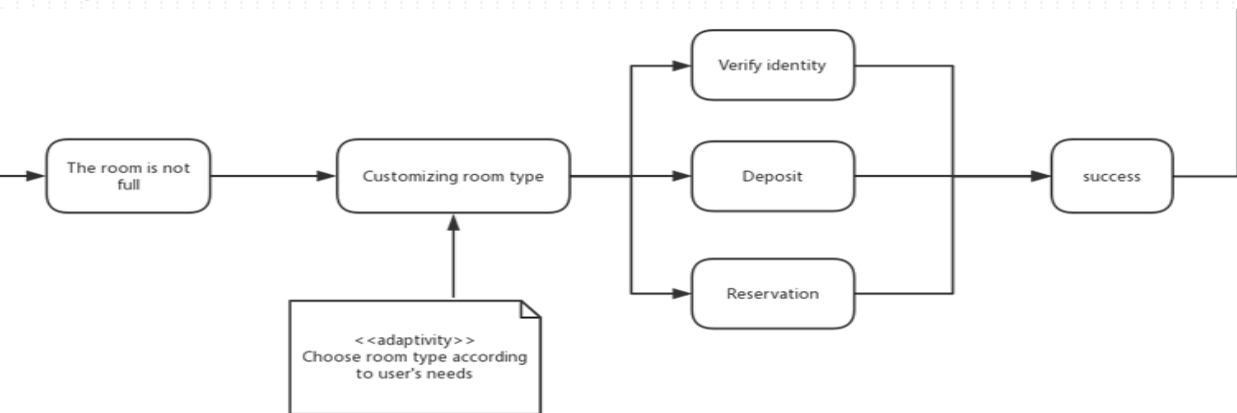
针对客户来讲就是简单的预订客房功能。

适应性建模

动态建模



用户进行登录时，当选择普通用户登录时，信息验证成功后进入前台首页进行相关操作；当选择管理员登录时，信息验证成功后进入后台首页进行相关操作。



当选择客房时，可以根据用户的需求和兴趣进行选择。

B/S架构分层设计

Browser/Server结构是Web兴起后的一种网络结构模式，用户不再依赖于单一的通过下载客户应用程序来访问信息。

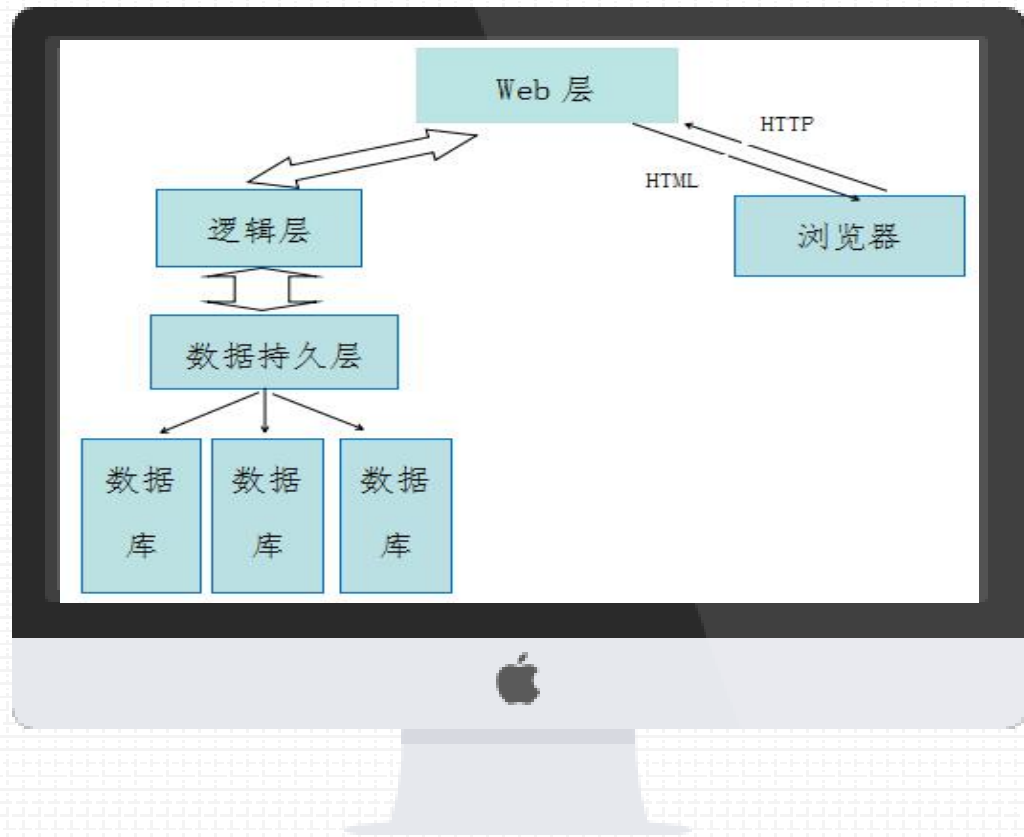
客户端即浏览器主要负责人机交互。

WEB 层主要负责对客户端应用程序的集中管理。对请求做出响应。

逻辑层主要负责应用逻辑中的集中管理即事务处理。

数据持久层位于数据库之上隐藏数据读取和操纵中的所有数据访问代码细节完全抽象出开发小组建立应用程序时使用的数据物理细节。

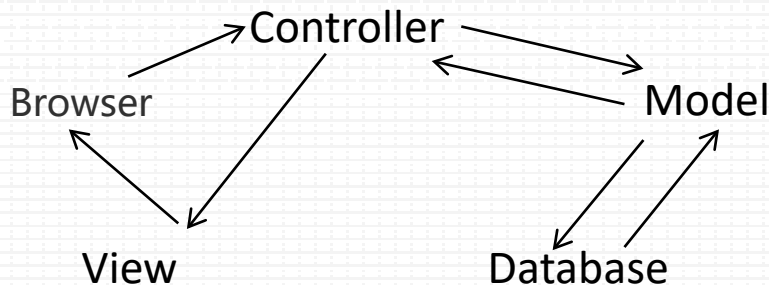
数据库则主要负责数据的存储等等。



基于MVC模式的开发

模型-视图-控制器

按照模型层、视图控制进行分离从而使得整系统责任明确、接口清晰，加快了开发过程。



模型层用来处理业务逻辑模型是指业务逻辑的处理，接受视图请求，并返回最终得结果。

视图层主要是用来展现户所需数据，是和用户进行交互的界面，采用HTML/Jsp来实现。

控制器接收用户的请求，将模型与视图进行匹配，完成用户的请求

优点：

- 耦合性低
- 重用性高
- 生命周期成本低
- 部署快
- 可维护性高
- 有利软件工程化管理

缺点：

- 视图与控制器间的过于紧密的连接
- 视图对模型数据的低效率访问

首页展示总体结构--综合框架型

该页面采用[综合框架型](#)，即结合上下框架和左右框架。

在上下框架中，上页空间较下页空间占小部分。上页标明酒店管理系统标志，使用户第一眼就能看到。下页为用户管理和获取客户住房信息的具体功能。使用上下框架可以让页面看起来一目了然，易于辨认。

下页使用左右框架，最左侧为主菜单，用户可以点击不同的功能：预定管理，客人信息，业务统计和历史档案。中间部分显示房间状态，最右侧显示入住登记信息。当用户点击主菜单的功能时，中间和右侧部分会变为相应功能显示的部分。

使用左右框架分为左中右部分，可以充分利用空间资源，也使功能划分更加明朗直观。

