**<SJTU>**

# <牙科医院管理系统>
# 软件架构文档

版本 **<1.0>**

# 修订历史记录

| 日期 | 版本 | 说明 | 作者 |
|---|---|---|---|
| <日/月/年> | <x.x> | <详细信息> | <姓名> |
| 9/4/2022 | 1.0 | 架构文档 | 全体成员 |
| | | | |
| | | | |

# 目录

# 软件架构文档

## 1. 简介

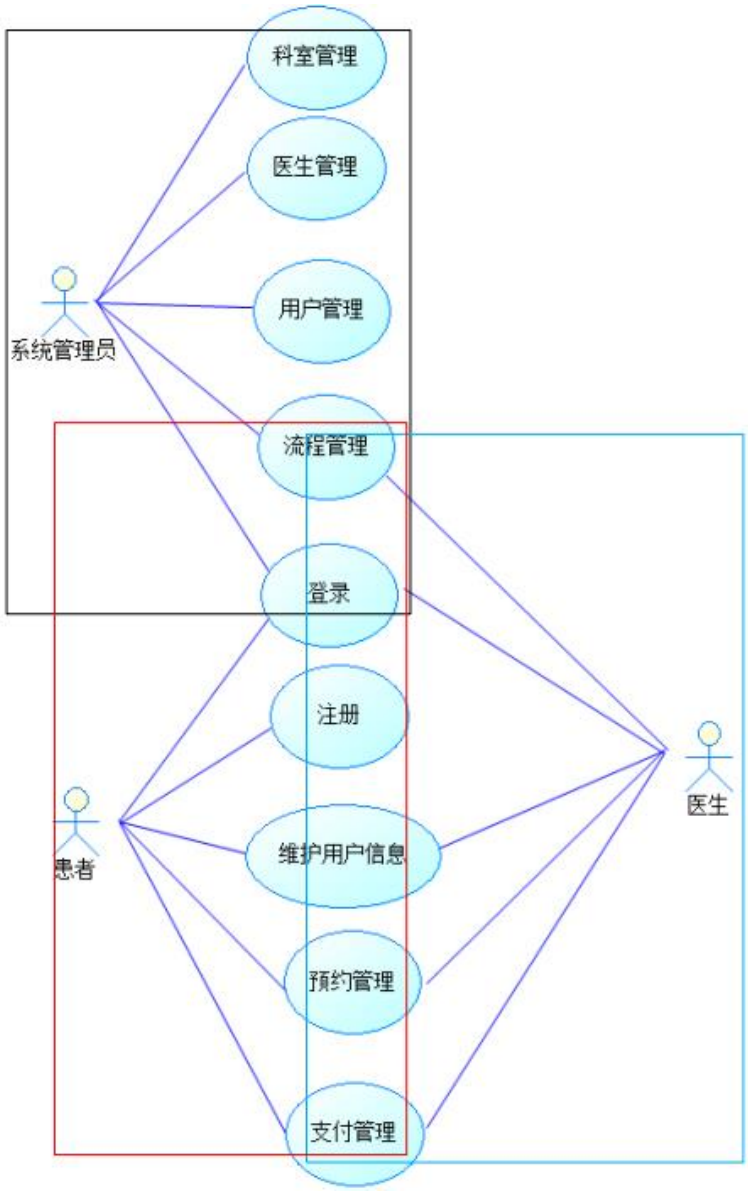
### 1.1 目的

本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

### 1.2 参考资料

《软件工程原理》

## 2. 用例视图



# 3. 逻辑视图

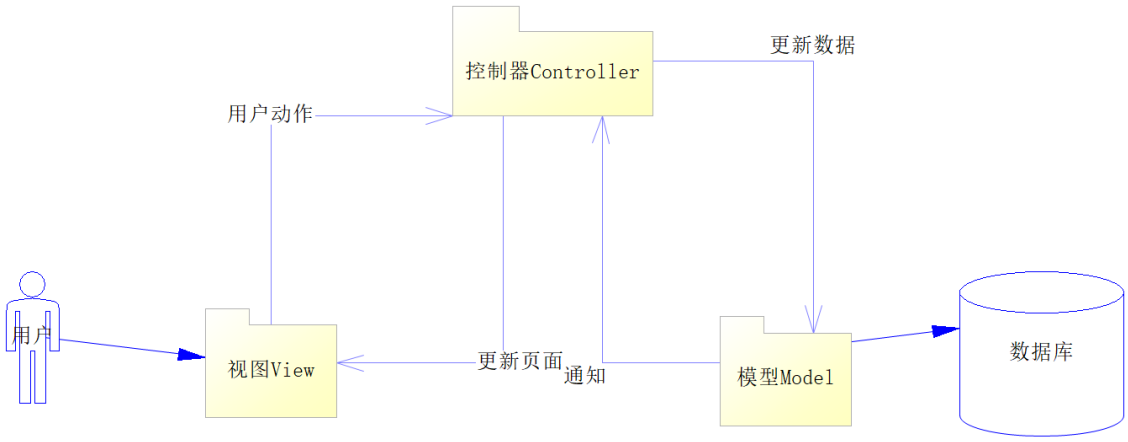
## 3.1 概述

牙科医院管理系统采用 MVC 为主要逻辑架构，将整个系统分为视图 View、控制器 Controller 和模型 Model 三个部分。

视图 View 部分主要负责在浏览器中给不同用户呈现内容信息和系统各项页面功能的交互界面，同时负责将用户的输入信息发送给控制器。

控制器 Controller 部分主要接受用户输入、请求，调用后端、数据库获得响应，并反馈给用户界面，由视图部分依据信息更新内容。

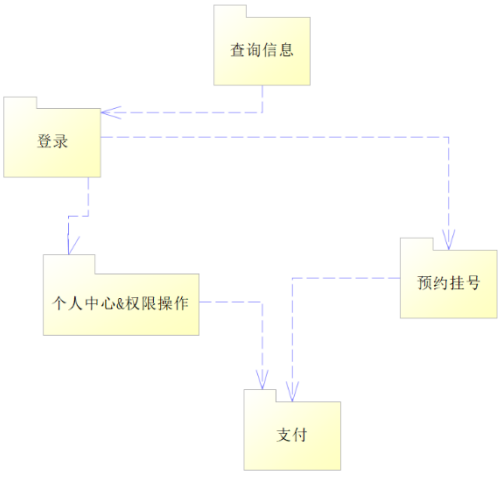
模型 Model 部分处理控制器请求，通过对数据库进行操作产生独立于视图的数据，并返回给控制器。




系统 MVC 架构图

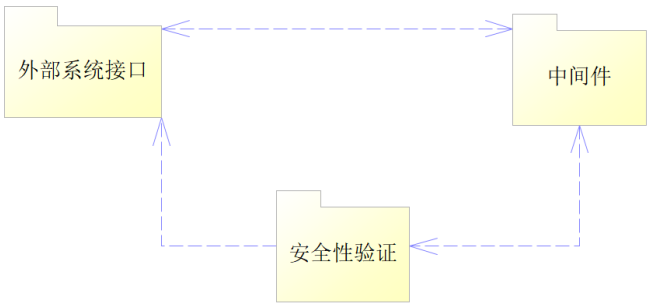
## 3.2 在构架方面具有重要意义的设计包

### 3.2.1 视图 View




包括系统核心页面功能界面预约挂号、查询信息、支付、个人中心操作。

3.2.2 控制器 Controller




通过调用中间件 java.lang、 java.rmi、java.amt、java.sql、java.io 等、调用外部系统如支付系统实现相关功能，并对信息进行安全性验证。

3.2.3 模型 Model

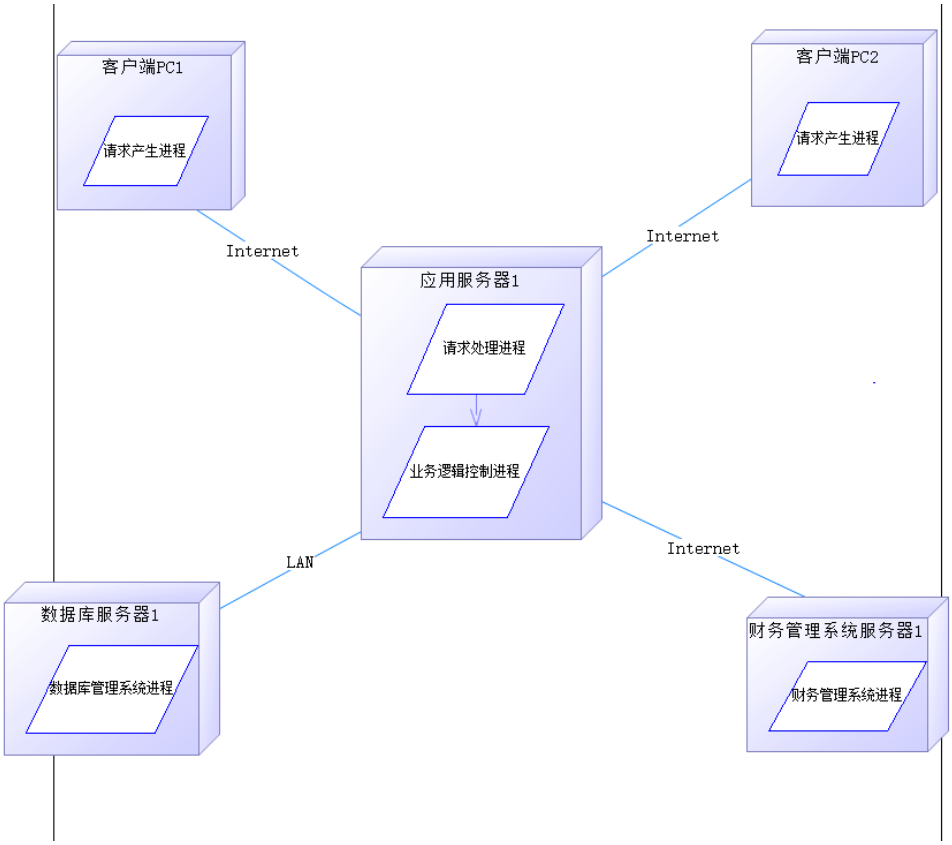

管理系统中存储的数据和业务逻辑，存储实体类数据包括医生信息、患者信息、管理员信息、订单表、排班表、黑名单等信息。

## 4. 物理视图

牙科医院预约管理系统主要采用的 B/S 架构，因为是需要建立在广域网上，面向各种各样的用户群，地域分散，但是便于构件的个别更换，实现系统的无缝升级，将系统的维护开销减到最小。

考虑到开发和测试阶段与实际运行阶段的侧重点的不同，我们画了两个不同阶段的部署视图，便于抓住重点，提高效率。

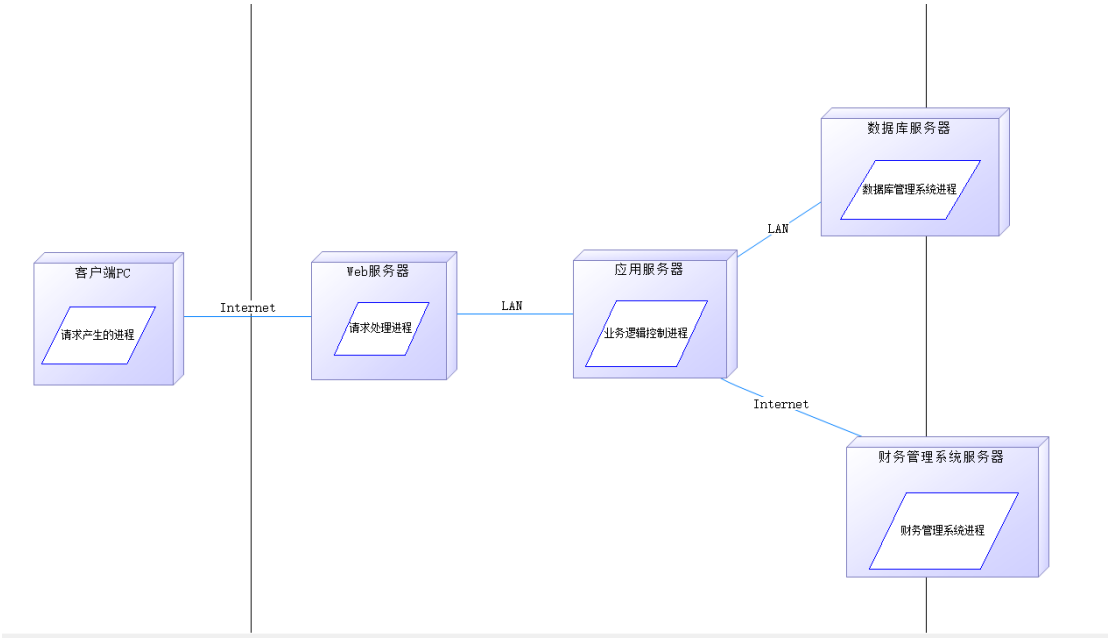
下图是开发和测试阶段的部署视图：




由于开发和测试阶段主要是对正确性的测试，因此并发量有限，请求处理进程和业务逻辑控制进程部署在同一台应用服务器上，此时重点是正确实现我们所需要的各种功能。数据库管理系统可以和 web 层以及应用层共同运行在同一台物理机器上。此外我们为支付系统提供了一个接口，方便最后的时候把支付系统链接上去。

下图是实际使用的时候的部署视图：

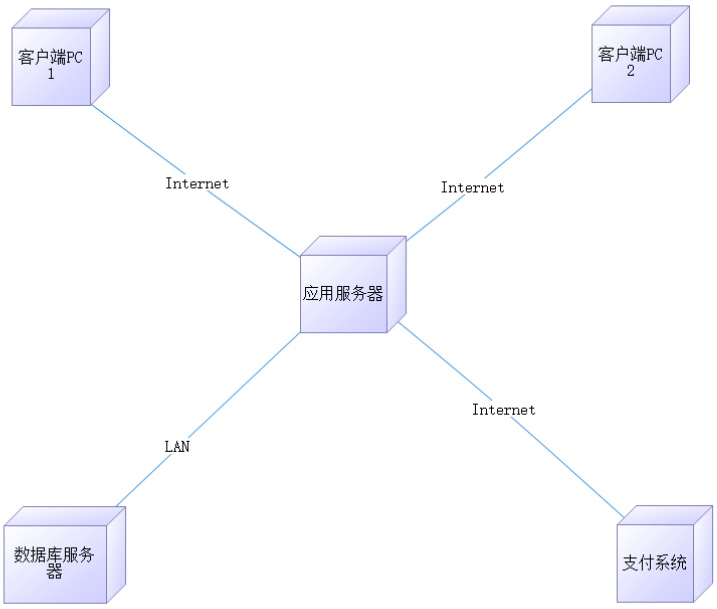
考虑到在实际使用的时候潜在的用户并发量可能很大，当系统的浏览请求增多时，Web 服务器容易成为系统瓶颈，因此，将请求处理进程和业务逻辑控制进程分离开，让他们各自在独立的物理机器上运行，同时当系统的并发访问量过大时，可以创建多个 web 服务器、web 服务器或数据库服务器的实例，将他们构成集群以提供强大的并发能力。同时我们保留了对与支付系统的接口。

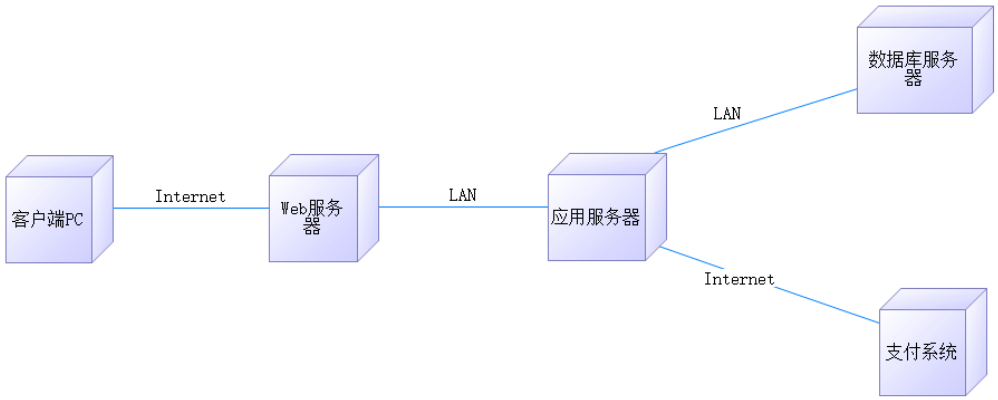



## 5. 文字阐述

总的来说，牙科医院管理系统的物理架构选用 B/S 架构

第一个图用于开发和测试，考虑到最开始搭建系统和测试系统正确性时并发访问量不是很大，故将请求处理进程和业务逻辑控制进程都部署到同一台服务器上，第二个图用于实际的使用过程中，面向 Web 用户，其潜在的并发访问量会很大，当系统的浏览请求增多时，Web 服务器容易成为系统瓶颈，因此，将请求处理进程和业务逻辑控制进程隔离开，让他们各自在独立的物理机器上运行，如果并发访问量实在过大，还可以创建多个 Web 服务器，应用服务器或数据库服务器的实例，构成集群以提供更强的并发能力和可靠性。




数据库服务器
LAN
客户端PC
Internet
Web服务器
LAN
应用服务器
Internet
支付系统