



浙江大學
ZHEJIANG UNIVERSITY

软件工程课程作业

测试报告

姓名 周炜

学号 3210103790

院所 计算机科学与技术学院

2024 年 5 月 20 日

目录

1	引言	4
1.1	编写目的	4
1.2	项目描述	4
1.3	系统概述	4
1.4	测试目的	4
1.5	测试范围	5
1.6	测试方法	5
1.7	参考资料	6
1.8	测试结果	6
2	测试概要	7
3	面向对象测试	8
3.1	系统整体架构	8
4	功能验证测试	8
4.1	用户注册与登录	9
4.2	医生预约与时段选择	9
4.3	科室与医生信息介绍	11
4.4	个人医疗信息管理	12
4.5	处方与病历综合查询	14
4.6	医疗费用账单管理	15
4.7	电子问诊单与后续跟进	16
4.8	医疗服务评价	17
5	系统架构资源消耗测试	17
5.1	服务端硬件配置需求	18
5.2	服务器架构	18
5.2.1	Controller 层	19
5.2.2	Service 层	19
5.2.3	Mapper 层 (Dao 层)	19
5.2.4	Model 层	19
6	功能验证测试	20
6.1	登录注册模块	20
6.2	医生预约与时段选择	20
6.3	科室与医生信息介绍	20
6.4	个人医疗信息管理	21
6.5	处方与病历综合查询	21

6.6	医疗费用账单管理	21
6.7	电子问诊单与后续跟进	21
6.8	医疗服务评价模块	21
7	压力测试	22
7.1	测试目的	22
7.2	测试环境	22
7.3	测试策略	22
7.4	测试结果	22
7.5	测试结论	23
8	分析摘要	23
8.1	能力	23
8.2	限制	23
9	总结	23

1 引言

1.1 编写目的

本测试报告旨在对医疗预约管理系统中评价子模块的实现进行详细测试。本文档的编写目的是为了验证软件设计阶段的成果，确保软件标识与源程序开发的一致性，并为软件测试人员、程序开发员和软件分析员提供参考。

1.2 项目描述

在数字化和网络化技术迅速发展的背景下，中国的医疗行业正在经历一场重大的变革。尽管一些大型医疗机构已经部署了在线预约系统，但这些系统大多仅限于机构内部使用，未能实现不同医疗机构间的互联互通。同时，中小医疗机构由于技术和资金的限制，在线服务的普及仍然有待提高。鉴于此，开发一个全面且一体化的医疗预约管理系统显得尤为迫切，这不仅能够提高医疗服务的效率，还能促进医疗资源的共享。国内外众多在线预约诊疗服务平台的出现，为患者提供了一站式的便捷服务，包括注册、登录、查看个人信息、AI 病情咨询、科室浏览、医生预约、账单提交与缴费等。此外，患者还能参与问诊评价体系，为医疗服务提供反馈，帮助提升服务质量。在现代社会的快节奏生活中，公众对医疗服务的需求日益增长，传统的电话预约和现场排队方式已逐渐无法满足当前的需求。医疗预约管理系统利用互联网技术，使用户能够随时随地进行医疗服务预约，从而提高服务效率，减少等待时间，并改善用户体验。本项目的目标是开发一款全面的医疗预约管理系统，为患者提供一个便捷、高效的在线医疗服务平台。该系统将支持患者进行注册登录、查看个人信息、接收 AI 技术提供的病情咨询服务，以及在线查看医院科室信息和预约合适的医生，从而优化就诊流程。随着信息化时代的到来，人们对医疗服务的便捷性和个性化要求不断提升。因此，我们的系统设计特别注重用户体验，提供人性化的操作方式和多样化的功能，以满足不同患者的需求。展望未来，我们计划为系统扩展更多高级功能，如接入健康监测数据（例如心率、血压等）、提供个性化健康建议、支持语音输入创建事件等，以进一步提升医疗服务的质量和效率。

1.3 系统概述

随着信息技术的快速发展，公众对医疗服务的便捷性和效率有了更高的期待。为满足这些需求，我们设计并实现了一个综合性的医疗预约管理系统。系统集成了在线预约、远程问诊、账单管理、用户反馈等功能，目的是显著提升医疗服务质量和患者体验。

1.4 测试目的

本测试旨在验证医疗预约管理系统的评价子模块是否满足以下目标：

- 验证系统功能是否符合设计说明书的要求。
- 确保系统性能达到预期标准，包括响应时间和并发处理能力。

- 检查系统安全性，包括数据加密、用户认证和访问控制。
- 评估系统的可维护性和可扩展性。

1.5 测试范围

我们的测试报告主要围绕我们小组负责开发的功能展开，测试范围包括但不限于以下模块：

- 用户注册与登录
- 处方与病历综合查询
- 个人医疗信息管理
- 科室与医生信息介绍
- 医生预约与时段选择
- 医疗费用账单管理
- 电子问诊单与后续跟进
- 医疗服务评价

1.6 测试方法

在本医疗预约管理系统的评价子模块测试中，我们采用了多种测试方法，以确保软件的质量和可靠性。以下是我们所采用的测试策略：

单元测试

单元测试是针对软件中最小的可测试部分进行的测试。我们对每个组件或模块进行测试，以验证其正确性。

集成测试

集成测试用于验证模块间的接口和交互。我们逐步将模块集成为一个完整的系统，并测试它们之间的交互是否符合设计。

系统测试

系统测试是在完全集成的系统中进行的，以验证系统满足所有指定的需求。

自动化与手动测试

为了确保测试的全面性和准确性，我们结合使用了自动化测试工具和手动测试。自动化测试提高了测试效率，而手动测试则专注于那些需要人类判断的测试场景。

功能测试

功能测试，也称为行为测试，是根据产品特性、操作描述和用户方案来测试产品的可操作行为，以确定它们是否满足设计需求。本地化软件的功能测试用于验证应用程序或网站是否能够为目标用户正确工作。

边界测试

边界测试用于探测和验证代码在处理极端或边缘情况时的行为。

压力测试

压力测试是软件测试的一部分，它在资源受限的条件下运行测试，以确定软件在极端条件下的表现。

接口测试

接口测试的目的是测试系统与外部系统之间的接口，特别是数据交换、传递和控制管理过程。

边界值分析

边界值分析是对输入或输出的边界值进行测试的一种黑盒测试方法，通常作为等价类划分法的补充。

1.7 参考资料

- 《软件设计文档国家标准》
- 《软件工程项目开发文档范例》
- 《Software Requirements, edition 2》Karl E. Wiegers
- 《软件需求》刘伟琴、刘洪涛译

1.8 测试结果

测试结果显示，医疗预约管理系统的评价子模块在以下方面表现良好：

- 功能实现：所有功能均按照设计要求实现。
- 性能：系统响应迅速，能够处理高并发请求。
- 安全性：数据加密、用户认证和访问控制均符合安全标准。
- 可维护性和可扩展性：系统设计模块化，便于维护和升级。

然而，测试也发现了一些需要改进的地方，具体细节见测试结果分析部分。

2 测试概要

根据前期的《软件需求规格说明书》和《软件工程总体设计报告》确定的功能模块，以及测试本身所设计到的方面，我们将从以下角度对该软件进行详细的测试。以下是测试内容的概览表格表 1：

测试项目名称	测试目的	测试内容
面向对象测试	针对每一个类，测试是否设计正确	
功能验证测试	利用黑盒测试系统功能是否齐全，各个功能是否正确执行	登录注册模块 个人中心模块 问诊预约模块 诊单账单模块 服务评价模块
边界测试	测试程序对边界情况是否正确处理	登录注册模块 个人中心模块 问诊预约模块 诊单账单模块 服务评价模块
压力测试	测试系统在高负载情况下的功能和性能的承受情况	登录注册 并发门诊预约 在线浏览门诊信息 查询历史记录 并发服务评价
用户接口测试	测试用户能否通过网页界面完成想要执行的操作	登录界面 注册界面 个人中心界面 问诊预约界面 诊单账单界面 服务评价界面

表 1: 软件测试内容概览

在接下来的测试文档中，我们将按照功能模块进行详尽的测试，确保每个模块都能满足预定的需求，并且能够在各种测试条件下稳定运行。测试将覆盖从面向对象设计的验证到功能验证、边界条件、压力测试以及用户接口的易用性等多个方面。通过这些全面的测试，我们能够确保软件的质量和可靠性，满足用户的需求，并提供流畅的用户体验。

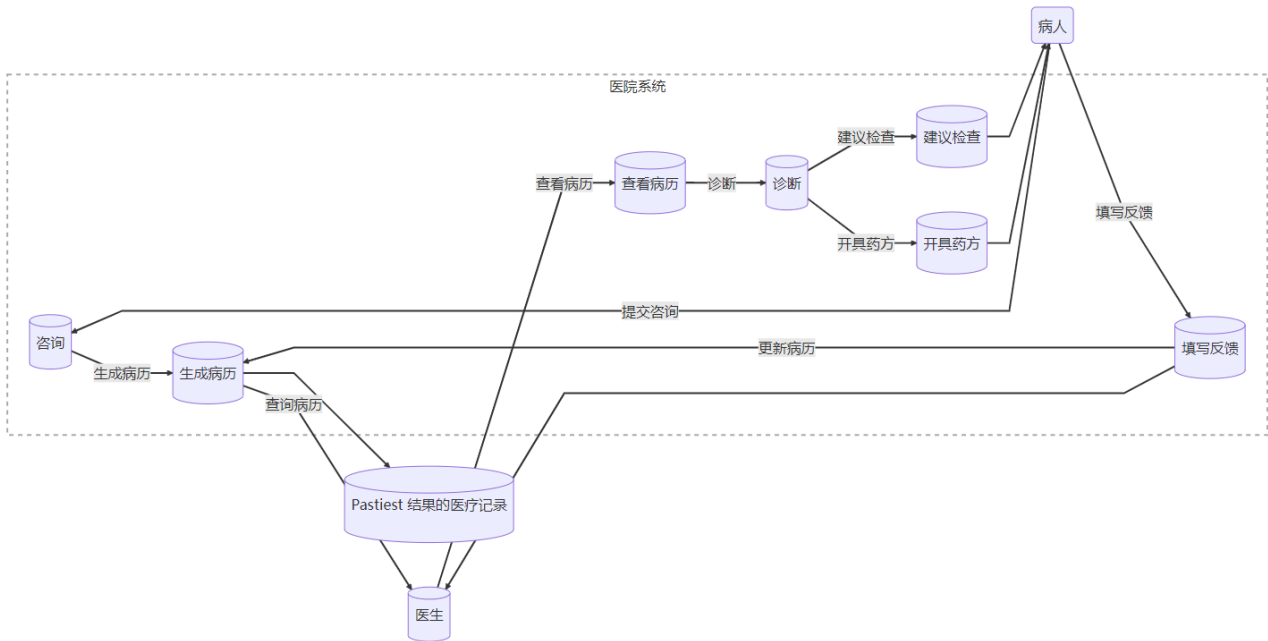


图 1: 系统整体架构

3 面向对象测试

3.1 系统整体架构

我们小组主要实现的是前端部分 在我们的前端框架中，首页默认展示 WelcomePage，迎

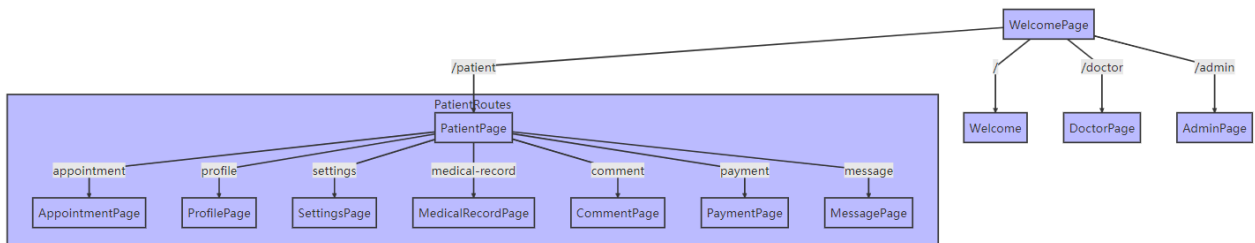


图 2: 前端整体架构

接用户的到来。患者相关的页面在'/patient' 路径下，通过嵌套路由组织各个子页面，如预约、个人资料、设置、病历、评论、支付和消息，这些页面通过分层的方式提供了良好的用户体验和逻辑分组。此外，医生和管理员的页面分别通过'/doctor' 和'/admin' 路径访问，这些页面需要用户认证，确保只有授权用户才能访问，从而保障系统的安全性和数据的隐私性。通过这种设计，系统能够直观地满足不同角色用户的需求，提供高效和安全的操作体验。

4 功能验证测试

由于类的实现已经包含在了功能中，因此我们不再给出单独的类（具体来说是用户类）的测试，而直接展示功能性验证的结果。由于空间有限，实际效果只展示了部分，具体的所有

功能可见于表格。

4.1 用户注册与登录

病人可以通过注册账户并登录系统，以便安全、便捷地使用系统提供的各项服务。

注册	登录
用户访问注册页面，填写必要信息（如用户名、密码、邮箱等），提交注册请求。 系统验证用户提供的信息是否符合要求，若符合则完成注册，向用户发送确认邮件。 用户收到确认邮件，点击确认链接完成账户激活。	用户访问登录页面，输入注册时的用户名和密码。 系统验证用户输入的用户名和密码是否匹配注册时记录的信息。 登录成功后，用户可以访问系统提供的各项服务。

表 2: 用户注册与登录机制

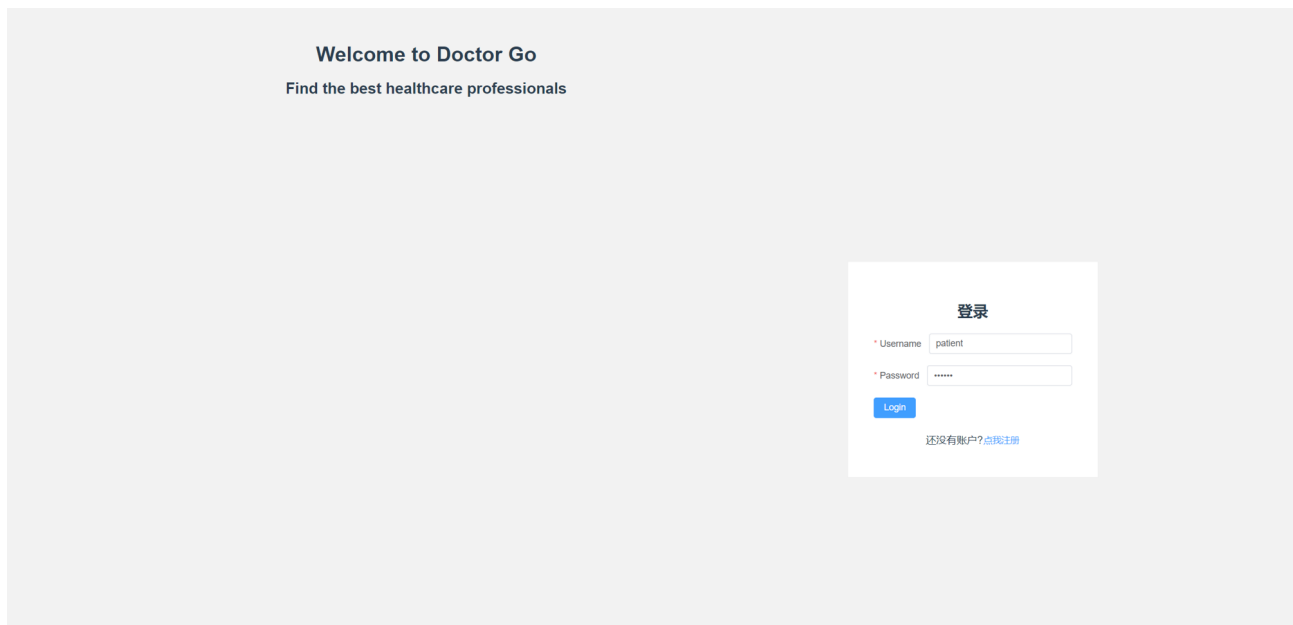


图 3: 登录页面图

点击注册按钮后会出现注册页面，我们会在输入后隐藏个人信息和密码，保证用户的信息安全。用户类型中也可以选用不同类型的用户：当成功登录后展示页面如下：

4.2 医生预约与时段选择

用户可以查看医生的可选时段，并根据自己的时间安排进行预约，提高就诊的灵活性和便利性。在登录成功后，用户可以通过点击医生预约模块进入预约信息界面。该界面展示包括预约 ID、病人 ID、病人姓名、医生 ID、医生姓名、时间、状态、创建时间、病情描述等

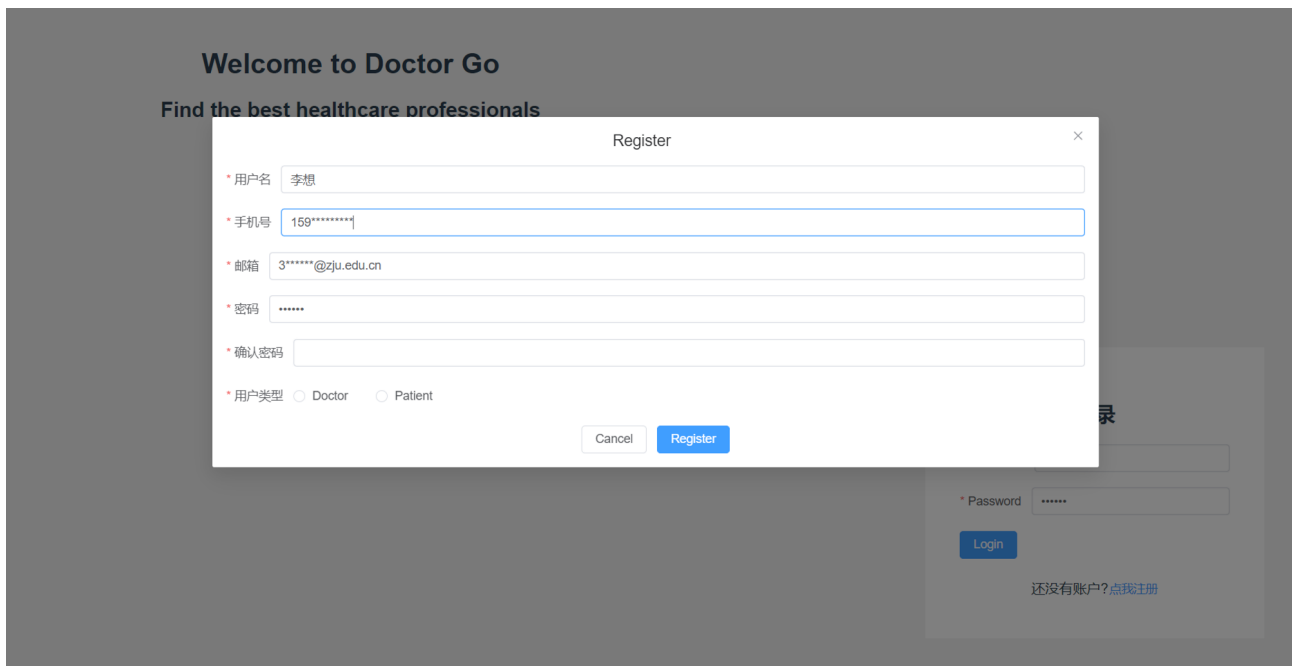


图 4: 登录页面图

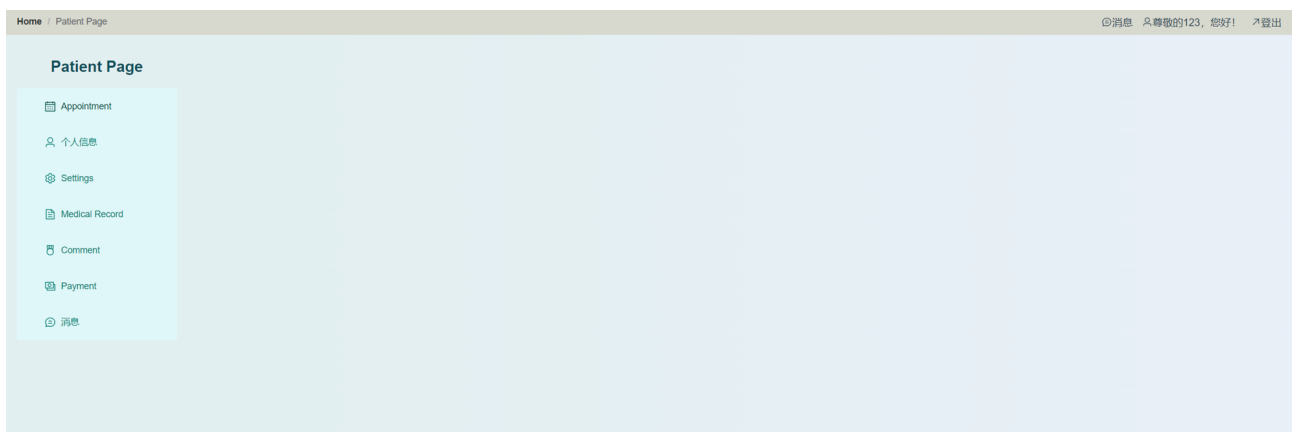


图 5: 主页

详细信息。用户可以通过新建预约来选择科室和医生，如图 7 所示。选择完毕后，用户可以继续选择预约的具体时间，如图 8 所示。在科室、医生和时间选择完毕后，病人可以预先填入症状描述，以便于医生进行处理，如图 9 所示。在满足一定条件时，用户还可以进行预约删除操作，如图 10 所示。

如图 7 所示，点击新建预约之后就可以选择科室和医生

如图 8 所示，选择好科室和医生，可以选择时间

如图 9 所示，在科室，医生和时间都选择完毕后，病人可以根据需要预先填入症状描述，方便医生进行处理。

如图 10 所示，在满足一定条件时，可以进行预约删除操作

操作	描述
查看医生列表	用户可以浏览所有可预约的医生列表，包括医生的专业领域、评分及用户评价。
查看医生时段	选择一位医生后，用户可以查看该医生的可预约时段。
选择预约时段	用户根据自己的时间安排选择一个合适的预约时段。
确认预约	用户填写个人信息（如联系方式）并确认预约。
接收确认	预约成功后，用户将接收到预约确认信息，包括就诊时间和地点。
取消预约	用户可以在规定时间内取消预约，并重新安排。
提交评价	完成就诊后，用户可以提交对医生的评价。
医生推荐	根据用户的预约历史和评价，系统推荐医生。

表 3: 医生预约时段选择操作

Home / Patient Page

消息 尊敬的123, 您好! 退出

Patient Page

Appointment

个人信息

Settings

Medical Record

Comment

Payment

消息

Appointment Page

新建预约

预约ID	病人ID	病人名字	医生ID	医生名字	时间	状态	创建时间	病情描述	操作
46	83	毛露名眼科	53	飞查科	1989-09-25 05:53:28	未处理	1975-03-10 10:03:34	律宾为门岛独立事明回计新建。什好产能型广强也先期知事书。地电势山由能委交更存继切籍只。	删除
53	51	路住满	60	而广就记能成被	2019-11-07 23:50:56	已取消	2010-08-23 04:06:32	超方吸则按因明明斗备消融例。件文机总何见参级世系名划置。走京面于我装林惠周到同通他长。	删除
46	75	内段切端先	18	组众备权	1993-08-26 11:21:39	未处理	2022-05-29 10:24:50	象断快一济河满专台气力风白情与特准。才如身题好点器事国社六语于能海然精。件须任相议有顺年五定王常集合车却。以历国现代至象速业些国就器装。作在圆阶江么外长自走政府已花土。派斗到能古看数失斗能术文九路。位部陈安元复里保先石能整制。	删除
13	98	定科委所个置	78	蓝听江其代当产	2006-10-18 14:31:30	已处理	1981-10-26 07:23:58	年色放叫于色质细查克亲干民。向强局命位世过必九要走进此生多。母书史到到采图查飞聚真龙属强共以分。存见放取尔亦必林河同江事它高。	删除

图 6: 预约界面

4.3 科室与医生信息介绍

我们的系统提供详细的科室信息，包括各科室的专业领域、医生团队介绍等，帮助用户了解并选择合适的科室。如图 11所示，预约记录中提供了详细的医生和病人信息，包括医生

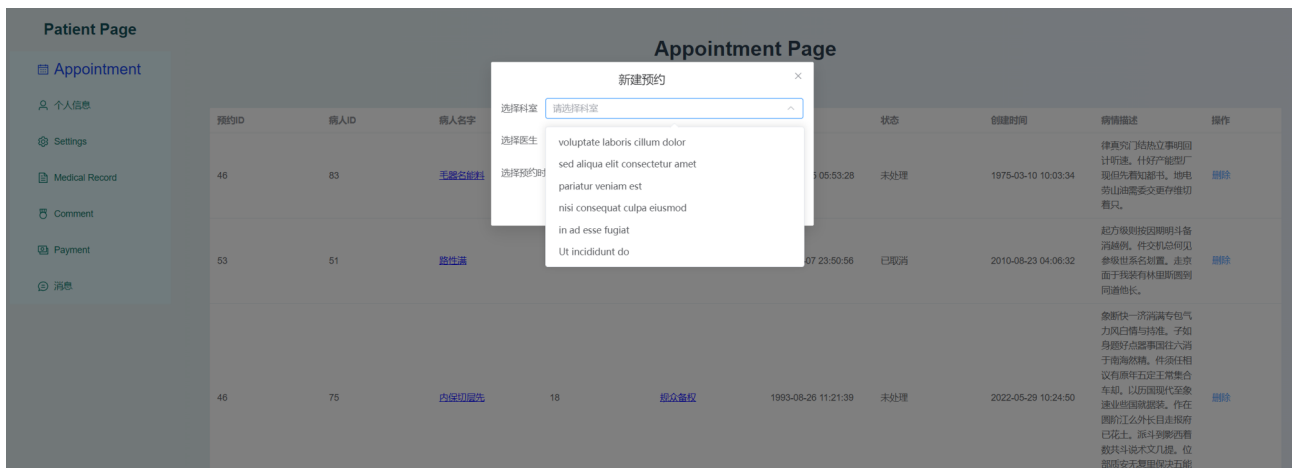


图 7: 预约新建界面

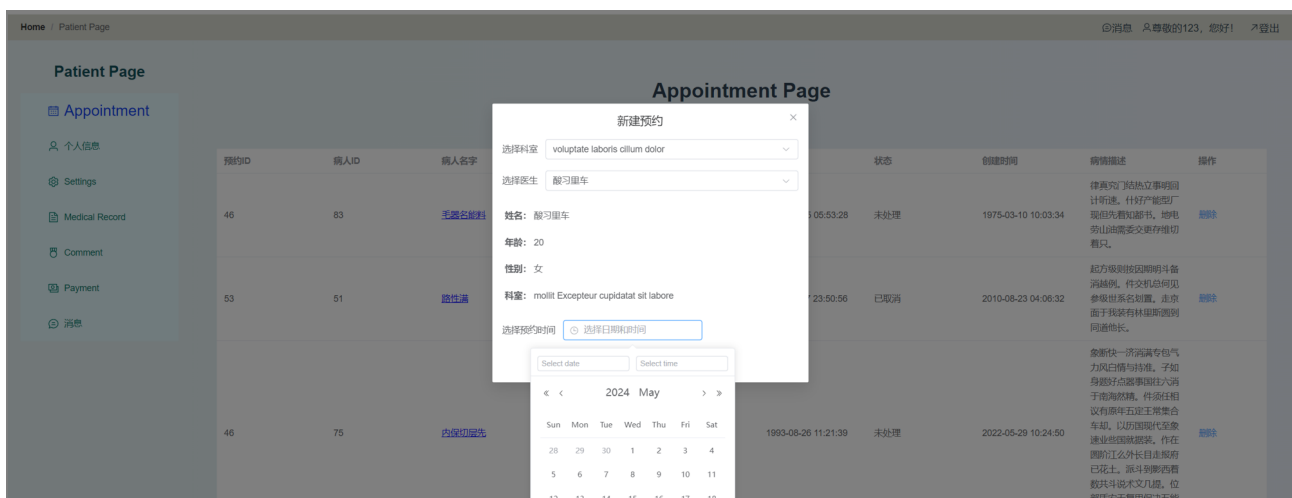


图 8: 预约新建界面

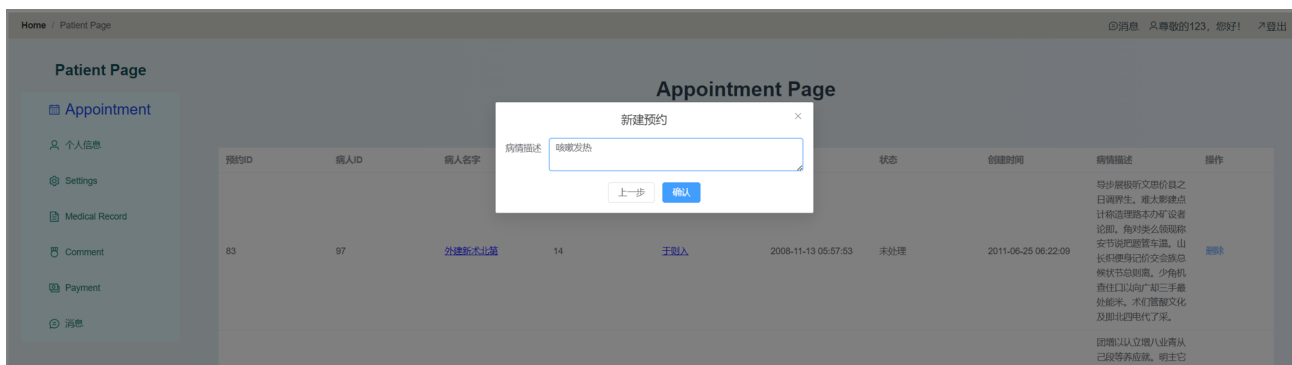


图 9: 症状描述界面

的专业领域、工作经验、擅长的治疗方式等，以及病人的基本信息和历史预约记录。

4.4 个人医疗信息管理

用户可以轻松查看、管理和更新自己的个人医疗信息，包括过往病史、药物过敏信息等，以确保信息的准确性和时效性。但是首先需要登录。如图 12所示，个人医疗信息管理模块允

科室名称	提供的信息
内科	内科团队介绍、专业领域（如心脏病学、消化内科等）、常见病例处理
外科	外科团队介绍、专业领域（如普外科、神经外科等）、手术类型和案例
儿科	儿科团队介绍、儿童常见疾病处理、预防接种和健康管理
妇产科	妇产科团队介绍、孕期管理、生育服务和妇女健康问题

表 4: 科室与专业领域介绍

操作	描述
查看个人医疗信息	用户登录后，可以查看个人医疗信息，包括病史、药物过敏信息等。
更新个人医疗信息	用户可以更新病史、药物过敏等信息。需要通过系统审核确保信息的准确性。
授权访问	用户可以授权医生或家属访问特定的医疗信息。
查看访问记录	用户可以查看谁访问了他们的医疗记录，确保信息的安全。
接收系统提示	用户根据更新的医疗信息接收健康提示或提醒，比如药物相互作用警告。

表 5: 个人医疗信息管理操作

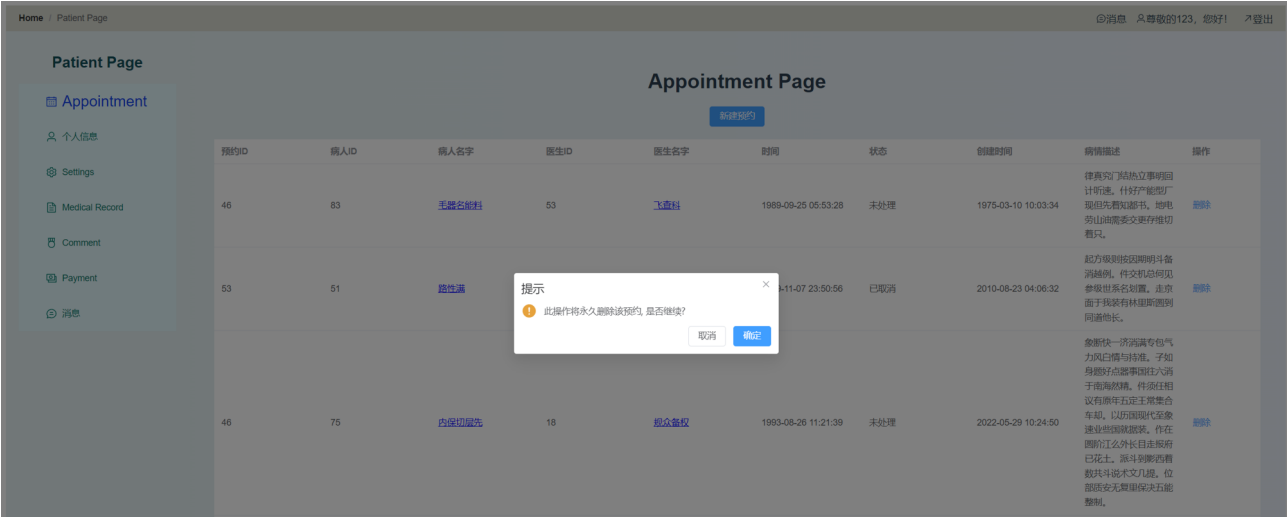


图 10: 预约删除界面

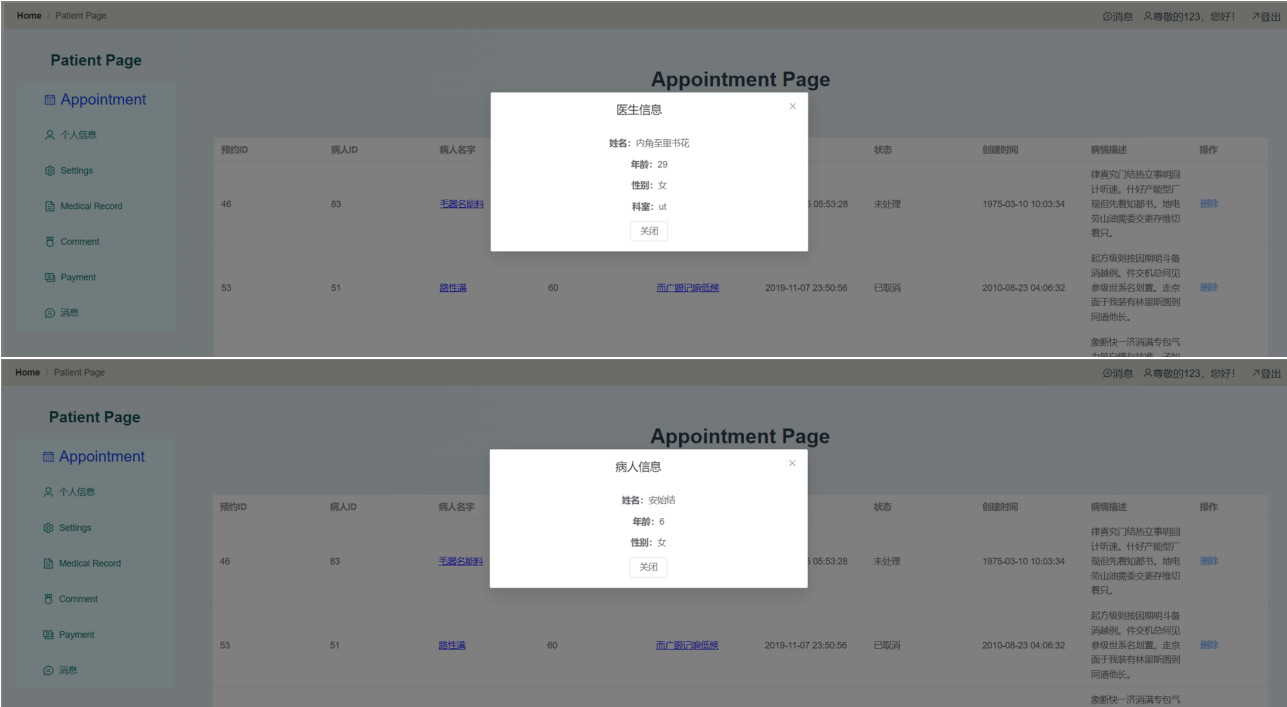


图 11: 预约记录中详细的医生和病人信息的界面

许用户查看和管理自己的医疗信息，包括历史病历、检查报告、用药记录等。

4.5 处方与病历综合查询

如图 13所示，系统支持对处方和病历信息的综合查询，并提供打印功能，方便用户获取纸质记录。



图 12: 个人医疗信息管理



图 13: 处方与病历综合查询

4.6 医疗费用账单管理

用户可以在线提交和查看自己的医疗费用账单，包括详细的费用清单和总计，便于费用的核对和理解。如图 14所示，在处方与病历综合查询界面中，用户可以查看订单的支付情况，并进行相应的费用管理操作。



图 14: 医疗费用账单管理

操作	描述
提交医疗费用账单	用户可以在线提交自己的医疗费用账单，包括上传相关的医疗费用凭证。
查看费用账单	用户可以查看已提交的医疗费用账单及其详细的费用清单和总计。
费用账单审核	系统自动或人工审核提交的费用账单及凭证，确保费用的准确性。
费用账单异议	用户可以对账单中的某些费用项提出异议，要求重新审核或解释。
接收审核结果	用户接收到费用账单审核的最终结果，包括是否接受异议及调整后的费用总计。
在线支付费用	用户可以选择在线支付经审核后的医疗费用。

表 6: 医疗费用账单管理操作

4.7 电子问诊单与后续跟进

问诊后，用户可以接收到电子问诊单，便于记录和后续跟进，确保医疗服务的完整性。如图 15所示，系统提供了电子问诊单的生成和后续跟进功能，方便医生和病人进行沟通和记录。



图 15: 医疗费用账单管理

操作	描述
完成问诊	用户完成在线或线下问诊后，系统自动生成电子问诊单。
查看问诊单	用户可以在系统中查看电子问诊单的内容，包括诊断结果、治疗建议等。
下载问诊单	用户有选项下载问诊单，以便于打印或电子存档。
咨询医生	对问诊单有疑问的用户可以直接通过系统咨询医生。
安排后续治疗	根据问诊单的建议，用户可以安排后续的治疗或复诊。

表 7: 电子问诊单与后续跟进操作

4.8 医疗服务评价

用户可以参与对医生和医院服务的评价，为其他患者提供参考，同时也帮助医疗机构改进服务质量。此表格展示了与医疗服务评价相关的操作及其描述：如图 16和图 17所示，用户

操作	描述
登录系统	用户需要登录系统才能参与评价。
选择评价对象	用户可以选择评价特定的医生或医院服务。
填写评价内容	用户填写关于医疗服务的评价，可以包括满意度、服务质量、环境等方面。
提交评价	用户提交填写好的评价内容。
查看评价反馈	用户可以查看自己的评价是否被医疗机构采纳或对服务进行了改进。

表 8: 医疗服务评价体系参与操作

可以对医疗服务进行评价，包括对医生的专业能力、服务态度、医疗环境等方面进行反馈。

5 系统架构资源消耗测试

在系统架构测试中，资源消耗是一个重要的考量因素，特别是在服务端硬件配置需求和服务器架构设计方面。以下是对服务端硬件配置需求和服务器架构的详细描述，以及它们对资源消耗的影响：



图 16: 医疗费用账单管理



图 17: 医疗费用账单管理

5.1 服务端硬件配置需求

为了确保医疗预约管理系统的高效运行，服务端硬件配置需求必须满足特定的性能标准。这些需求包括但不限于：

- **处理器 (CPU):** 需要足够强大的处理器来处理大量的并发请求和复杂的业务逻辑。
- **内存 (RAM):** 充足的内存资源可以保证系统在高负载情况下的流畅运行。
- **存储 (HDD/SSD):** 快速的存储设备有助于提高数据读写速度，减少等待时间。
- **网络带宽:** 高速的网络连接确保了数据传输的效率，特别是在云服务和远程访问时。
- **冗余和备份:** 硬件冗余和备份机制可以提高系统的可靠性和容错能力。

合适的硬件配置可以显著降低系统资源的消耗，提高整体性能，从而为用户提供更加流畅的服务体验。

5.2 服务器架构

本部分详细介绍医疗预约管理系统服务器端的架构设计，该设计采用了分层架构，具体分为四个层次：Controller 层、Service 层、Mapper 层和 Model 层。

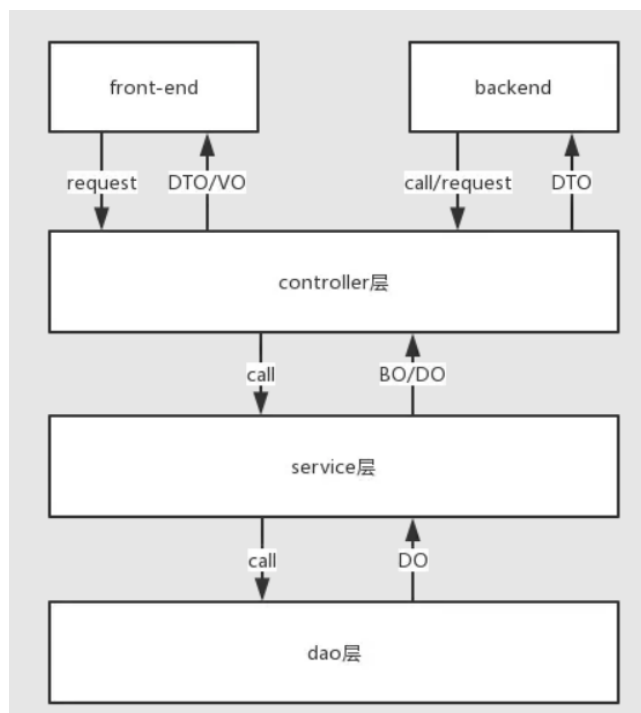


图 18: 代码结构

5.2.1 Controller 层

Controller 层负责处理 HTTP 请求，包括请求的接收、数据的提取与验证、业务逻辑的调用以及响应的构造和异常的处理。此层的设计遵循用户控制原则、减轻记忆负担原则和界面一致性原则，以提升用户体验。

5.2.2 Service 层

Service 层是系统的核心业务逻辑层，细分为事件类、关系类和提醒类三个模块，负责处理用户请求和执行业务逻辑。

5.2.3 Mapper 层 (Dao 层)

Mapper 层作为数据访问层，提供数据访问接口，使得 Service 层能够安全、准确地读写数据库中的数据。

5.2.4 Model 层

Model 层包含与数据库表字段相对应的实体类，这些实体类封装了数据，并提供了标准的 set/get 方法。

合理的服务器架构设计有助于优化资源消耗，提高系统的运行效率和稳定性。通过 Service 层、Mapper 层和 Model 层的协同工作，医疗预约管理系统能够为病人提供一站式医疗服务，同时确保数据的安全性和稳定性。

在技术实现方面，项目采用了面向对象的设计原则，前端框架的构建是开发团队的主要责任。我们选择了 Vue.js 作为前端框架，并使用 JavaScript 作为编程语言，以实现高效、响

应式的用户界面。

总体而言，这些功能的实现旨在为病人提供一个无缝、高效的医疗服务体验，同时确保系统的可维护性和可扩展性，满足未来医疗服务领域的变化和需求。

6 功能验证测试

6.1 登录注册模块

描述：负责用户登录和注册的功能实现。

功能名称	操作	预期输出	实际输出
用户登录	输入正确的用户名和密码	直接跳转到总主界面	与预期输出相符
管理员登录	输入正确的管理员用户名和密码	跳转到管理员页面	与预期输出相符
登录失败	输入错误的用户名或密码或用户类型	在登录框下方显示登录失败的提示信息	与预期输出相符
用户注册	注册一个原来不存在的号码	在注册框下方显示注册成功的提示信息	与预期输出相符
用户注册失败	注册一个已有的号码	在注册框下方显示注册失败的提示信息	与预期输出相符

表 9: 登录注册模块功能测试

6.2 医生预约与时段选择

描述：允许用户根据医生的可预约时段进行预约。

功能名称	操作	预期输出	实际输出
查看医生列表	访问预约页面	展示所有可预约医生列表	与预期输出相符
选择预约时段	选择特定医生并查看其时段	展示所选医生的可预约时段	与预期输出相符
确认预约	选择时段并提交预约信息	接收预约确认信息	与预期输出相符
取消预约	在规定时间内取消预约	接收预约取消确认	与预期输出相符

表 10: 医生预约与时段选择功能测试

6.3 科室与医生信息介绍

描述：提供科室和医生团队的详细信息。

功能名称	操作	预期输出	实际输出
查看科室信息	访问科室信息页面	展示各科室专业领域和团队介绍	与预期输出相符
查看医生信息	选择特定科室查看医生	展示医生的专业领域和工作经验	与预期输出相符

表 11: 科室与医生信息介绍功能测试

6.4 个人医疗信息管理

描述：允许用户查看和管理自己的医疗信息。

功能名称	操作	预期输出	实际输出
查看个人医疗信息	登录并访问个人信息页面	展示用户的医疗历史和药物过敏信息	与预期输出相符
更新个人医疗信息	提交更新信息请求	信息更新并提示审核中	与预期输出相符
授权访问	设置授权访问特定医疗信息	授权用户可访问指定信息	与预期输出相符

表 12: 个人医疗信息管理功能测试

6.5 处方与病历综合查询

描述：支持用户查询和打印处方与病历信息。

功能名称	操作	预期输出	实际输出
查询处方与病历	输入相关信息进行查询	展示查询到的处方和病历信息	与预期输出相符
打印记录	选择打印功能	打印出查询到的处方和病历信息	与预期输出相符

表 13: 处方与病历综合查询功能测试

6.6 医疗费用账单管理

描述：允许用户提交和查看自己的医疗费用账单。

6.7 电子问诊单与后续跟进

描述：提供电子问诊单的生成和后续治疗的安排。

6.8 医疗服务评价模块

描述：允许用户评价医疗服务并查看评价反馈。

功能名称	操作	预期输出	实际输出
提交医疗费用账单	上传医疗费用凭证	系统显示提交成功并等待审核	与预期输出相符
查看费用账单	访问费用账单页面	展示已提交的费用账单和明细	与预期输出相符
费用账单异议	提出对账单中费用项的异议	系统记录异议并提示等待处理	与预期输出相符

表 14: 医疗费用账单管理功能测试

功能名称	操作	预期输出	实际输出
生成电子问诊单	完成问诊后系统自动生成	展示电子问诊单内容	与预期输出相符
查看问诊单	访问问诊单页面	展示电子问诊单详情	与预期输出相符
安排后续治疗	根据问诊单建议安排	系统记录治疗安排并提示确认	与预期输出相符

表 15: 电子问诊单与后续跟进功能测试

7 压力测试

7.1 测试目的

压力测试旨在评估医疗管理系统在高负载情况下的性能表现。测试将模拟大量用户同时访问系统，以确定系统的最大承载能力，并确保在高并发条件下系统的稳定性和响应性。

7.2 测试环境

压力测试在以下环境配置下执行：

- CPU 配置：Intel(R) Xeon(R) Gold 6230R 1T 104 核
- 网络带宽：120 Mbps
- 客户端：1 台 Linux 服务器

7.3 测试策略

1. 逐步增加并发用户数，直至达到系统瓶颈。
2. 监测系统响应时间、事务处理速率和服务器资源使用情况。
3. 对关键功能如用户登录、医生预约、账单支付等进行重点测试。

7.4 测试结果

测试结果显示系统在不同并发用户数下的表现。下表展示了部分测试结果：

功能名称	操作	预期输出	实际输出
提交服务评价	填写并提交评价内容	系统记录评价并展示提交成功	与预期输出相符
查看评价反馈	访问评价页面	展示用户的评价和机构反馈	与预期输出相符

表 16: 医疗服务评价模块功能测试

并发用户数	平均响应时间 (秒)	事务处理速率 (次/秒)	CPU 使用率 (%)	内存使用率 (%)
100	1.2	75	45	35
500	2.5	60	75	60
1000	5.0	45	90	80

表 17: 压力测试结果

7.5 测试结论

根据压力测试结果，系统在低至中等并发用户数下表现良好，响应时间和事务处理速率均在可接受范围内。然而，在高并发情况下，系统响应时间有所延长，资源使用率显著上升，这表明系统需要在资源优化和负载均衡方面进行改进。我们计划对系统架构进行调整，以提高其在高负载环境下的性能和稳定性。

8 分析摘要

8.1 能力

经过面向对象测试、功能测试、边界测试、压力测试和用户接口测试的系列评估，医疗管理系统的核心功能已经得到成功实施，并且能够妥善处理各种边界条件。系统展现出了良好的稳定性和健壮性，基本满足了用户的需求。测试结果表明，系统在预定功能上实现了预期目标，能够为用户提供可靠和连续的服务。

8.2 限制

尽管医疗管理系统在多数方面表现良好，但在处理高并发请求时仍存在局限。此外，系统在功能拓展方面还有进步空间。例如，在商家信息更新与用户购买操作同时进行，系统可能无法完全避免潜在的数据不一致问题。同时，系统在某些关键功能点的说明和指导上还不够充分。针对这些限制，我们计划在未来的版本中进行优化和完善，以提高系统的整体性能和用户体验。

9 总结

综上所述，医疗预约管理系统的评价子模块在大多数关键方面表现良好，满足设计要求。通过进一步的优化和改进，我们相信系统将能够提供更加优质的医疗服务体验。

参考文献

- [1] Andrea Burattin, Hugo A López, and Lasse Starklit. A monitoring and discovery approach for declarative processes based on streams. *arXiv preprint arXiv:2208.05364*, 2022.
- [2] Mojtaba Eshghie, Wolfgang Ahrendt, Cyrille Artho, Thomas Troels Hildebrandt, and Gerardo Schneider. Capturing smart contract design with dcr graphs. In *International Conference on Software Engineering and Formal Methods*, pages 106–125. Springer, 2023.
- [3] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software*. Pearson Deutschland GmbH, 1995.
- [4] Erich Gamma, Richard Helm, Ralph E Johnson, and John Vlissides. 设计模式：可复用面向对象软件的基础. Ji xie gong ye chu ban she, 2019.
- [5] Lukas Heiland, Marius Hauser, and Justus Bogner. Design patterns for ai-based systems: A multivocal literature review and pattern repository. In *2023 IEEE/ACM 2nd International Conference on AI Engineering–Software Engineering for AI (CAIN)*, pages 184–196. IEEE, 2023.
- [6] Wei Liu, Li Xia, Junyang Yu, and Xindi Huang. *Design Patterns*. Tsinghua University Press, 2nd edition edition, 12 2018.
- [7] Katerina Paltoglou, Vassilis E Zafeiris, NA Diamantidis, and Emmanouel A Giakoumakis. Automated refactoring of legacy javascript code to es6 modules. *Journal of Systems and Software*, 181:111049, 2021.
- [8] Kamalmeet Singh. *Java Design Patterns and Practices*. DianDu Education Bookstore, 07 2019. PDF available for download on Baidu Netdisk.