



软件工程进度汇报

预 约 系 统 小 程 序

报告人：胡茵

CONTENTS 目录

01. 项目进度

02. 版本管理

03. 编程标准

04. 质量保证

PAERT 01

项目进度

项目目标

ENTER THE TITLE

项目目标

实现通过小程序实现研讨室预约，教学楼预约，实验楼预约，在V2版本中还将上线体育场预约，图书馆预约，在V3版本中还将包括桑浦山校区的研讨室、教学楼、实验楼等预约。

研讨室预约

- 有4人、6人、10人间
- 最长预约时间两个小时
- 用户与用户的预约时间间隔至少半个小时以上

教学楼预约

- 一学期五次，前五次可以随便预约
- 人数限制（大中小教室）

实验室预约

- 实验类型（医学类，化学类，物理类）

需求分析

预约系统

项目功能模块

用户（学生、老师（老师权限更高））

登录、个人中心、修改密码

公告 查看公告

留言 添加留言

预约教学楼

提前结束教室的使用

刷卡签到

增删改查

人数

次数限制（如一学期五次，前五次可以随便预约）若本学期预约次数已满但是又特别紧急，则可向管理员提交申请并附加申请理由

实验楼

增删改查

刷卡签到

提前结束实验室的使用

实验室类型（如医学类，化学类，物理类）

研讨室

增删改查

刷卡签到

提前结束研讨室的使用

人数

次数限制（一天之内只能约一次）

时间限制（最长三个半小时）

可能不要？

体育场

增删改查

提前结束体育场的使用

预约羽毛球

收费

篮球

可能不要？

图书馆预约

预约入馆

随机分配座位

提前结束图书馆的使用

人数限制（如500人）

公告

显示公告信息

查看公告详细信息

添加公告、修改公告、删除公告

留言管理

查看留言

回复留言

删除留言

是否允许将留言展示给用户

黑名单管理（预约三次即可进黑名单，可设置黑名单有效期，设置下一位教室使用优先级）

用户管理

管理员可以查询账号（不可以新增、删除学生账号）

管理员可以更改用户的预约记录和预约的记录

加入黑名单

移出黑名单

会议室管理

查看可预约类型

取消预约、通知用户

批准预约（用户第五次之后的预约需要管理员进行审核）

添加会议室

删除会议室

修改会议室

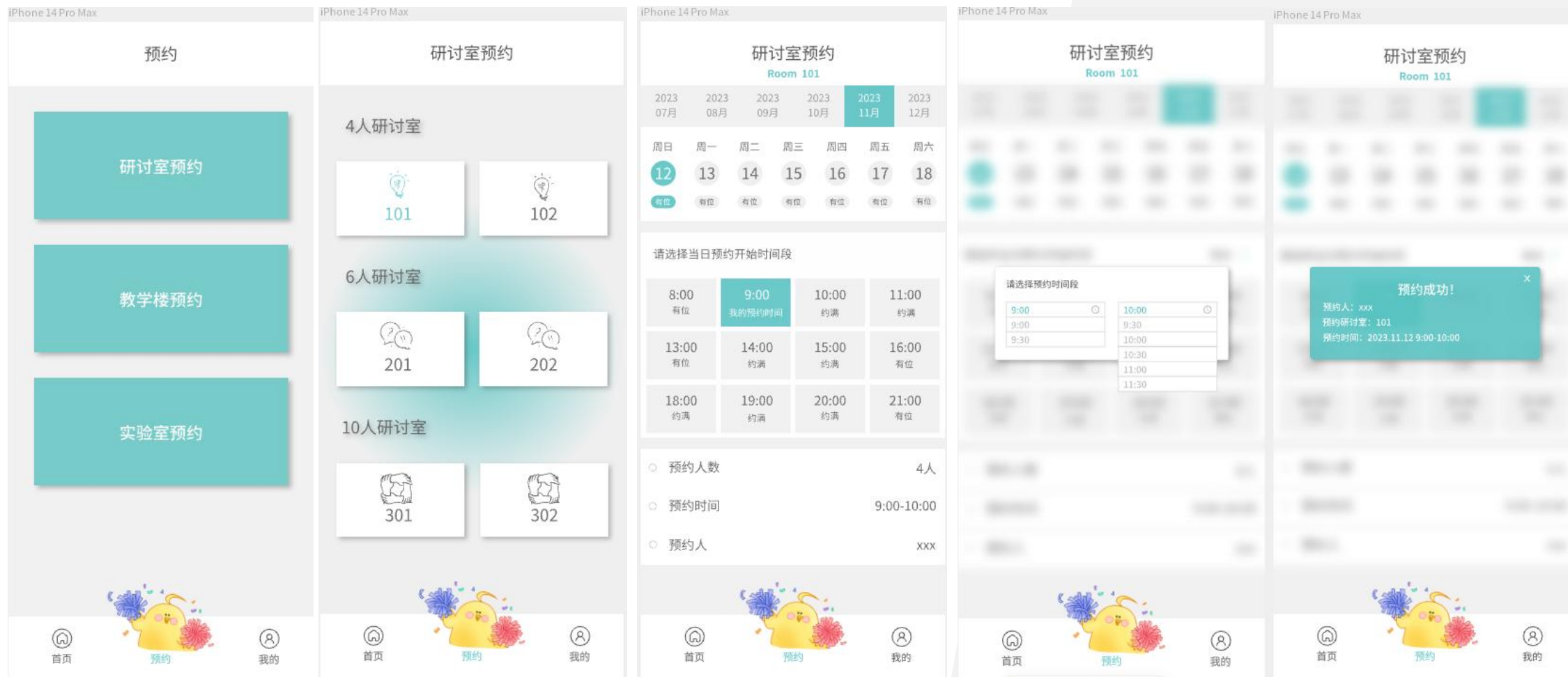
查看会议室信息（教室号、容纳的人数、服务时间）

若刷卡签到不成功，即叫号问题，可以留言反馈，管理员修改预约记录

U I



UI



U I



前端进度



所设计到的技术：uniapp+vue2
目前完成了大部分用户端的页面制作，接口还在制作中

HBuilderX：这是DCloud推出的一款IDE，使用它编写的代码可以运行在微信开发者工具。
微信开发者工具：微信官方提供的针对微信小程序的开发工具

后端进度

开发环境：

1. jdk: 11
2. 依赖管理工具: maven 3.6.1
3. 数据库: mysql
4. 开发平台: IntelliJ IDEA 2023.2.3
5. 版本控制工具: github
6. 操作系统: windows

所涉及到的技术：

1. springboot框架
2. mysql数据库
3. mybatis操作数据库
4. maven项目管理

1. 全部后端成员均已完成开发环境配置。

2. 已将mysql数据库部署到云服务器

3. 本项目已设计好的接口有5个：

- 登录
- 查询所有公告
- 查询我的所有预约
- 是否可以预约
- 删除预约。

后端已完成登录、查询所有公告、查询我的所有预约这3个接口的代码编写以及代码整合

正在尝试把已整合好的代码部署到云服务器上，并与前端登录部分进行联调测试。另外两个接口的代码正在编写中。

版本管理

使用Git管理代码的版本，跟踪变更历史

<https://github.com/hudadaya/Wechat-appointment-mini-program>

Git是一种分布式版本控制系统，用于跟踪文件的变化并协作开发。它是由Linus Torvalds在2005年创建的，用于管理Linux内核开发，现已成为最流行和广泛使用的版本控制系统之一。



前端编程标准

01

Vue组件规范

.vue文件，包括 `<template>`，`<script>`，`<style>` 等部分，保持每个组件的简洁性和可读性

02

命名规范

变量、函数、组件名使用驼峰命名法，例如 `myFunctionName`。

03

代码风格

遵循一致的缩进、空格和换行风格，保持代码整洁易读

04

数据交互

使用Axios或类似的库进行数据交互，保持统一的API调用方式和错误处理机制

05

状态管理

使用Vuex，确保状态管理的清晰性，将状态分离、统一管理，避免在组件之间传递无关的状态数据。



后端编程标准

01

RESTful API设计

遵循RESTful API设计原则，使用合适的HTTP方法（GET、POST、PUT、DELETE等）和状态码进行API设计。

02

命名和代码结构

遵循Java的命名规范，使用驼峰式命名法，类、方法、变量名应当具有描述性。合理的包结构和模块化设计，使代码易于扩展和维护。

03

异常处理

实现合适的异常处理机制，对于异常情况返回合适的HTTP状态码和错误信息，保证接口的健壮性

04

数据库操作

MyBatis持久化框架，避免直接操作数据库

05

安全性

考虑数据安全性，实施适当的用户认证和授权机制，保护敏感信息



质量保证

测试策略和执行

- 单元测试：针对某个模块测试
- 集成测试：各个模块之间的交互和集成
- 系统测试：对整个系统进行端到端的测试

自动化测试

- 自动化单元测试：
使用测试框架（如JUnit、Jest等）编写自动化测试，加速反馈周期

代码审查和静态分析

- 代码审查：
开展代码审查会话，让团队成员检查代码，识别潜在的问题和改进机会

文档和培训

- 技术文档：
编写清晰、准确的技术文档，帮助团队理解系统设计和实现细节。






项目分工

组长：胡茵

UI：张雨馨 张文秀

前端：赵嘉萱 张文秀 张雨馨 李晓行 胡茵

后端：丁妍婷 刘文龙 廖乐乐 马乙佳 廖锴俊 胡茵



谢谢大家