

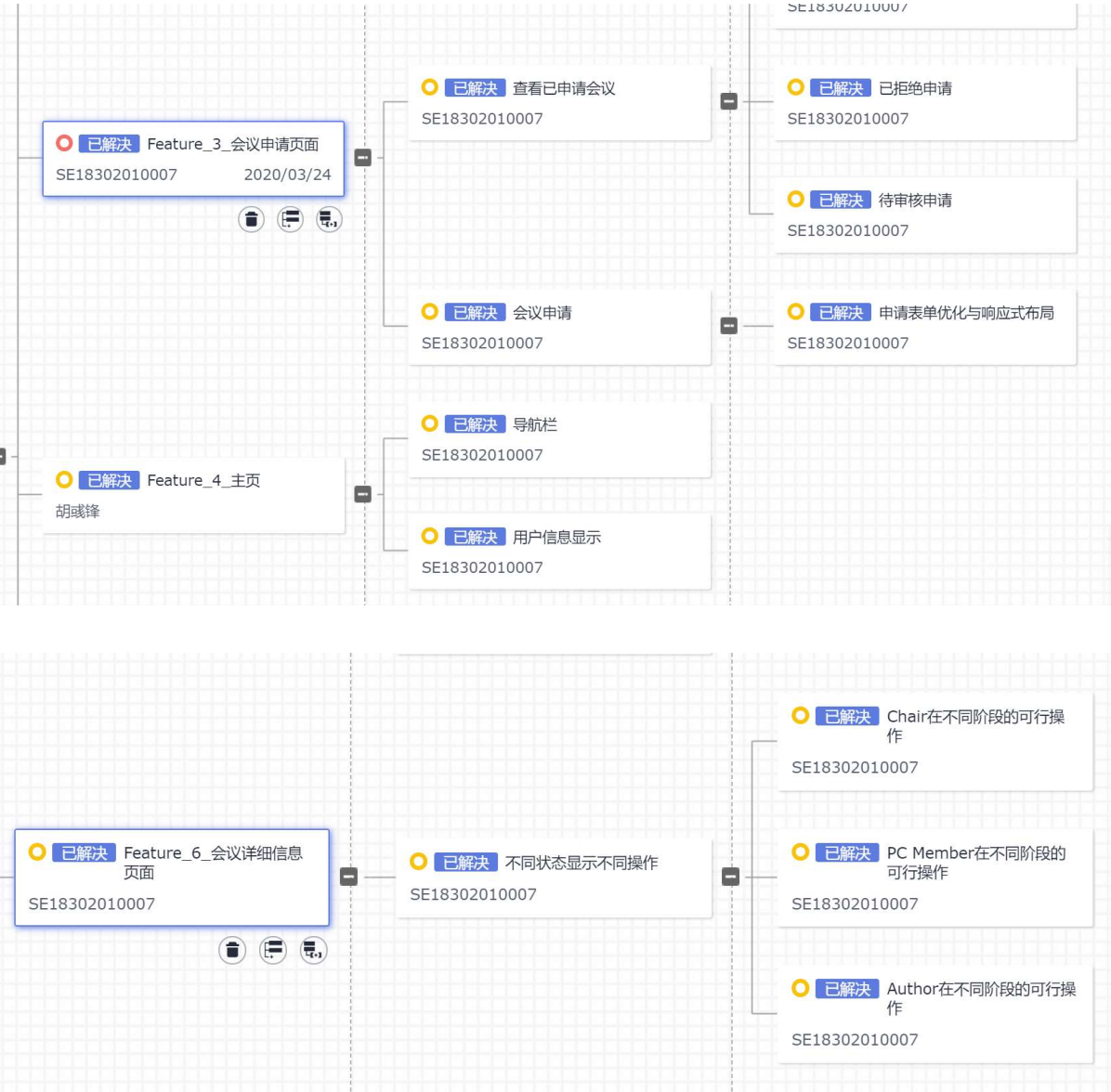
Lab3 会议组织与投稿 小组实验报告

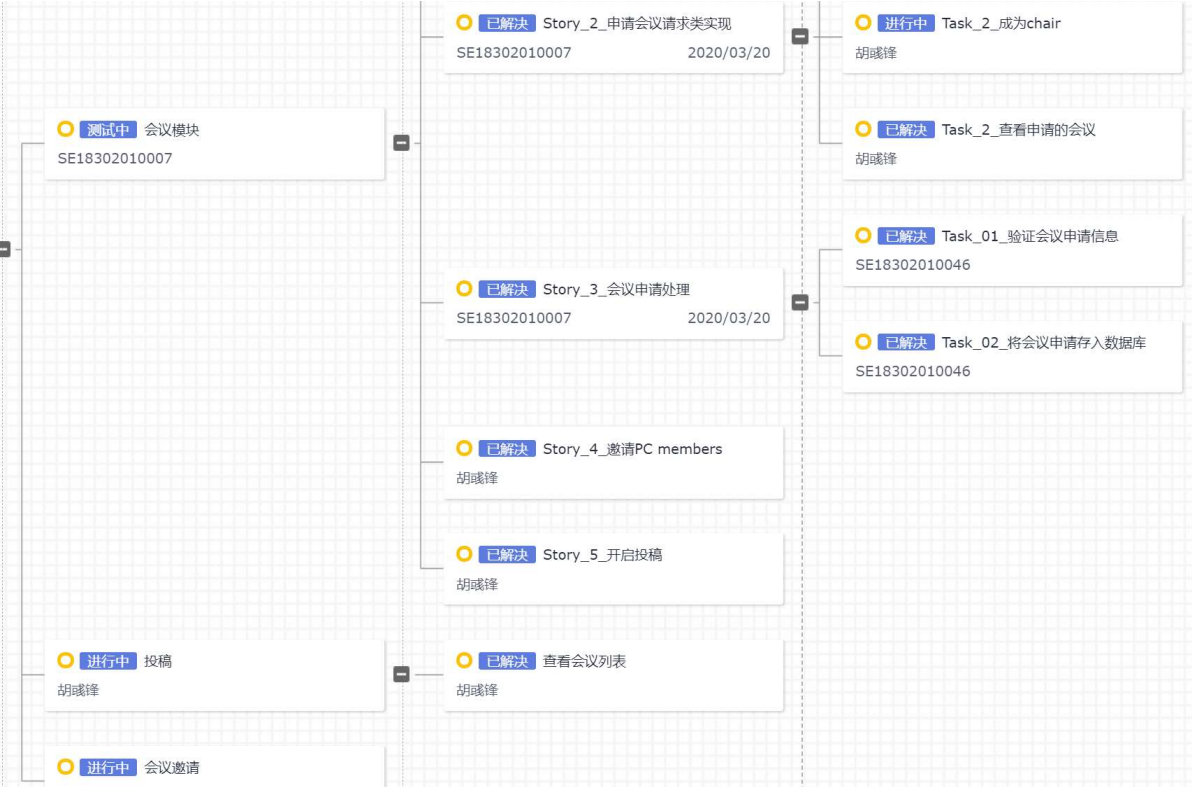
小组成员： 潘星宇 胡彧锋 颜华

一、DevCloud 项目规划

(一) 项目需求规划

通过需求规划功能对功能点进行划分



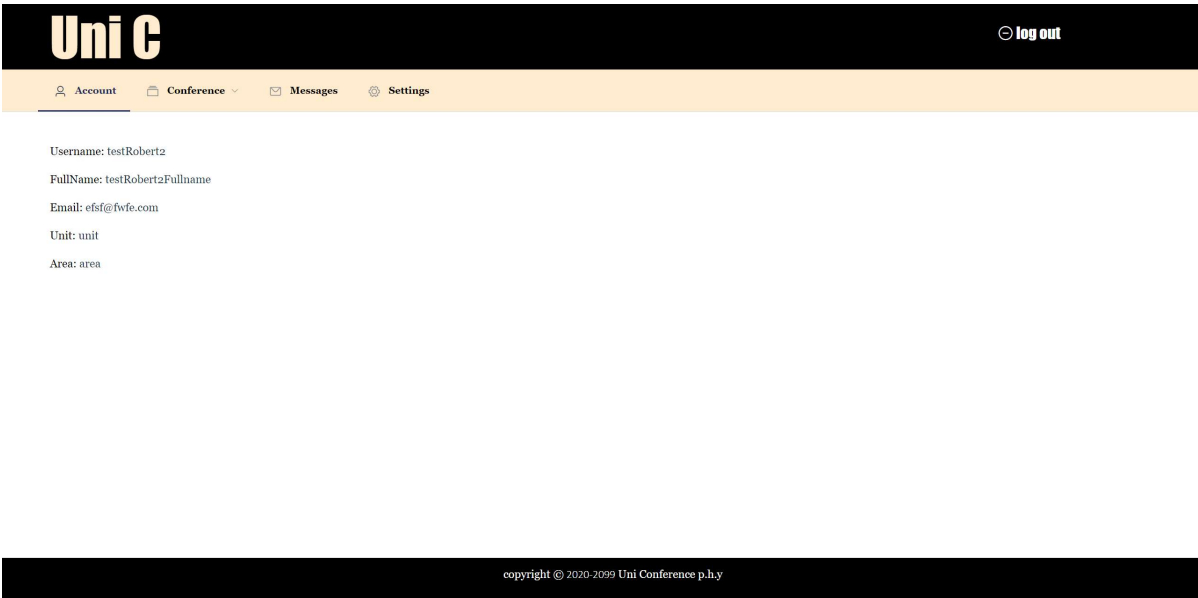


(二) 代码检查

任务	所属门禁	问题			最近一次检查	操作
<div><div>H</div><div>HardChair-Frontend</div><div>2020软件工程_UniChair_胡晓峰</div></div>	Passed	0	0	26	提交触发 检查于 2020-04-13 23:16:34	...
<div><div>H</div><div>HardChair-Backend</div><div>2020软件工程_UniChair_胡晓峰</div></div>	Passed	0	0	7	提交触发 检查于 2020-04-13 23:32:46	...

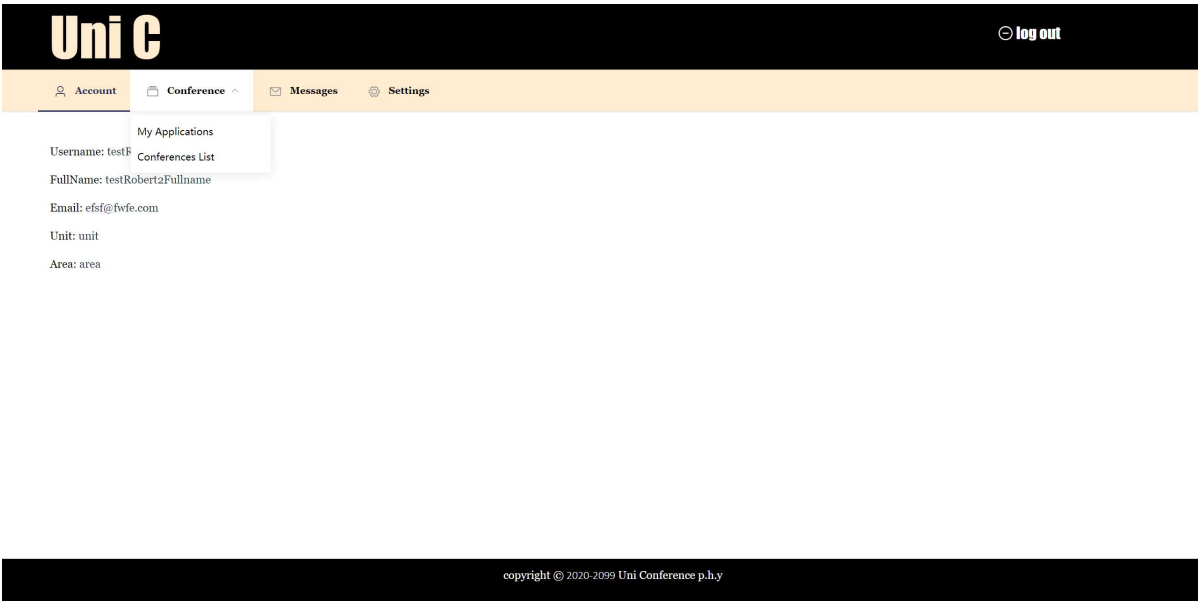
二、项目成果概况

• 系统主界面：



此界面为用户登录后跳转至的界面，用户可以在此界面得到账户基本信息，后续拟添加**修改密码**，完善个人信息等功能。

• 导航栏：

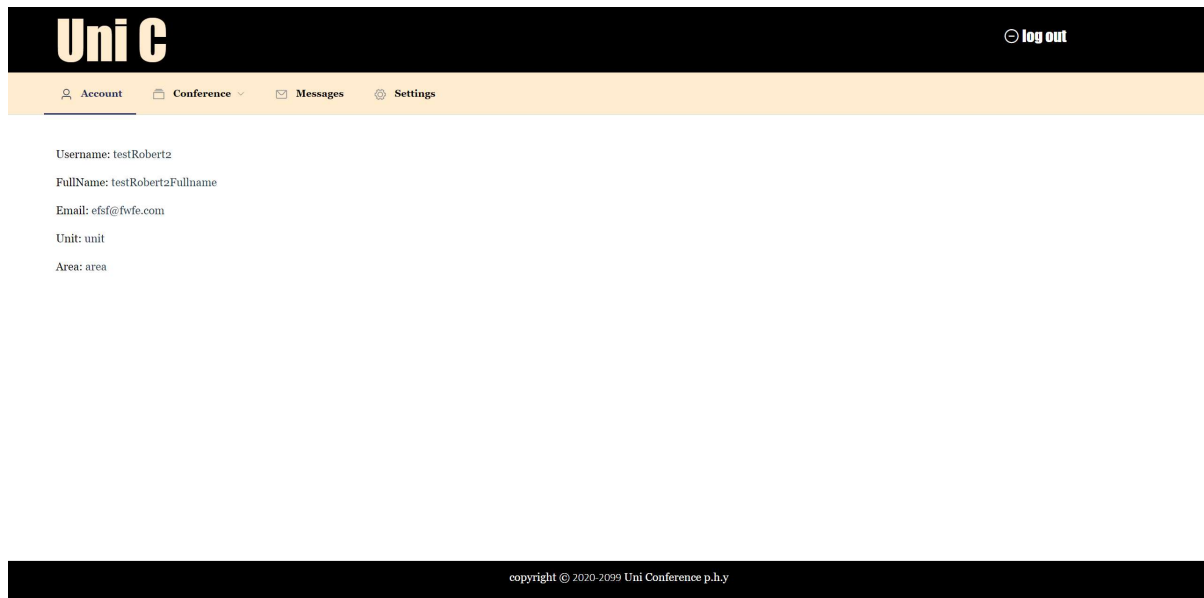


导航栏分为四部分：①**Account** 将跳转至系统主界面，显示用户基本信息
②**Conference** 下 **My Application** 将显示该用户所申请的所有会议；**Conference** 下 **Conference List** 将显示所有通过管理员审核的会议 ③**Messages** 将跳转至类似消息查看界面，本次 LAB 中将对收到的 PC Member Invitation 进行处理，未来该模块的功能还可增加，处理各类不同的消息，诸如 paper 评审结果等 ④**Settings** 是保留板块，还未实现

· 会议申请界面

My Application 集成了会议申请**查看**和**新增**的功能。进入**NEW APPLICATION**板块则可以发起新的会议申请。重新排布了表单的布局，以及修改了单行表单的样式，使得整个页面更加整洁。

· 用户系统界面



用户登陆成功后将会进入主界面，在后续功能逐渐完善之后，用户将可以在此界面查看自己的个人信息，查看自己创建的会议，作为投稿人加入的会议，作为审稿人加入的会议等等。此时浏览器已经接收到了登录时从后端返回的 **token**，并存放在 **COOKIE** 中。目前，用户系统主界面可以通过点击界面中央的 **Set up Meeting** 按钮跳转到会议申请界面。

Uni C

log out

Account

Conference

Messages

Settings

My Applications

PASSED

PENDING

REJECTED

ALL

NEW APPLICATION

Abbr :

SEC

FullName :

2020 Software Engineering Conference

HeldPlace :

Shanghai

HeldDate :

2020-04-24

Submission Date :

2020-04-25

Submission deadline :

2020-04-26

Release Date :

2020-04-30

Introduction :

This is a conference for all computer scientists together to discuss the problems in software engineering. When writing codes, it's really easy to get dirty words out of you mouth. So let's write articles to discuss about this question.

Submit

copyright © 2020-2099 Uni Conference p.h.y

进入 **ALL** 板块则可以看到自己发起的所有会议申请，以及每个申请的状态（**PENDING, PASS, REJECT**）。

Uni C

log out

Account

Conference

Messages

Settings

My Applications

PASSED

PENDING

REJECTED

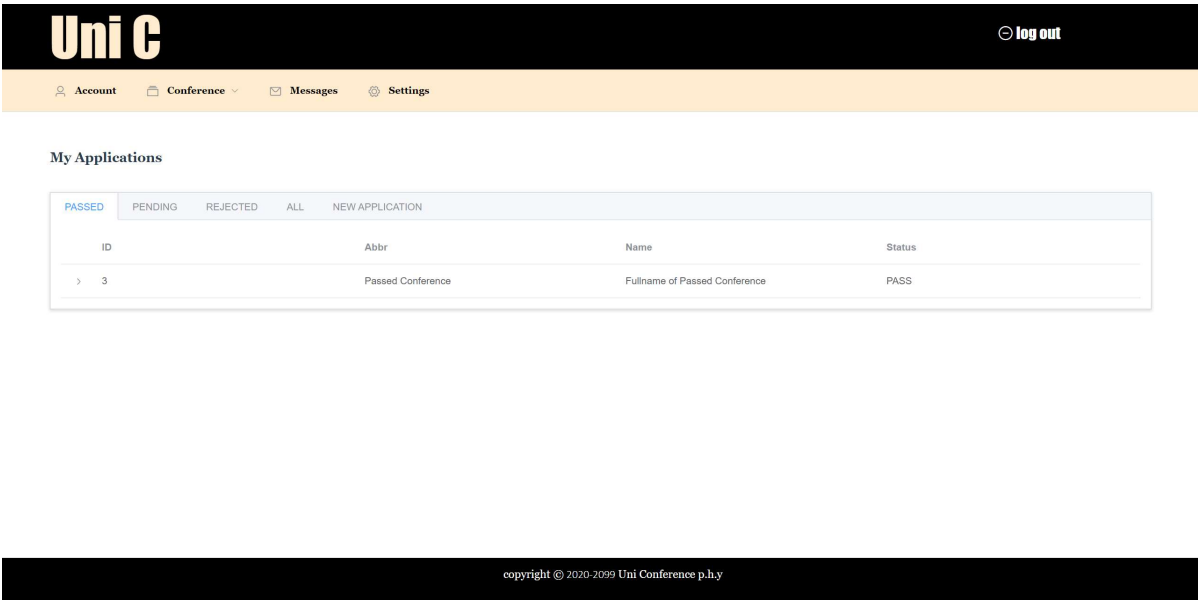
ALL

NEW APPLICATION

ID	Abbr	Name	Status
> 4	Pending Conference	Fullname of Pending Conference	PENDING
> 3	Passed Conference	Fullname of Passed Conference	PASS
> 2	Rejected Conference	Fullname of Rejected Conference	REJECT

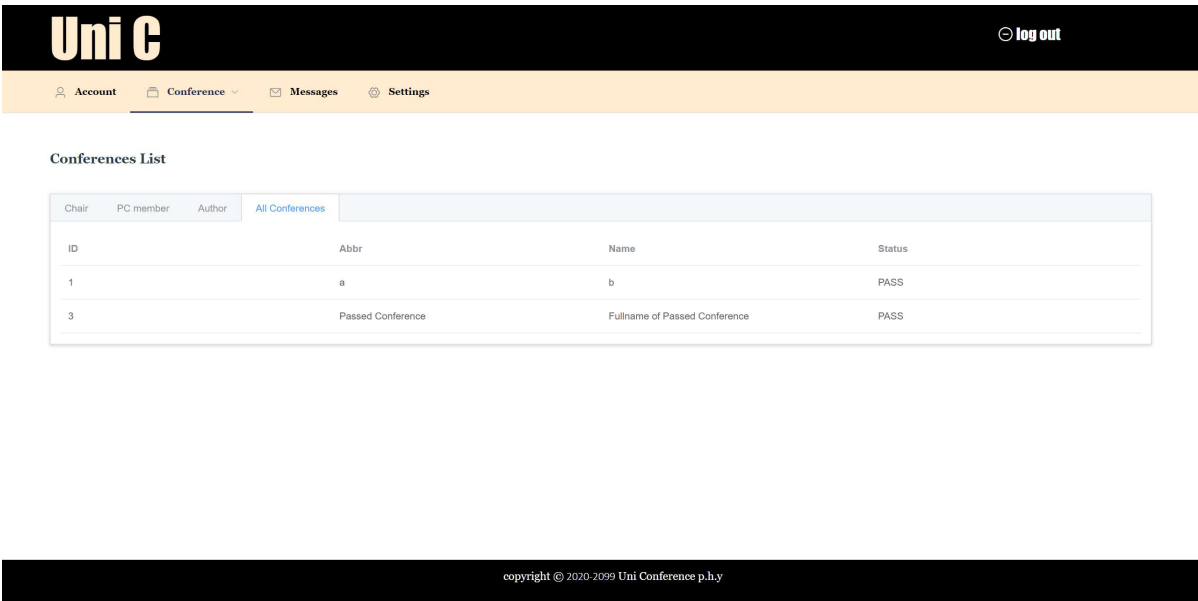
copyright © 2020-2099 Uni Conference p.h.y

进入 **PASSED, PENDING, REJECTED** 板块则可以查看某一状态的所有会议申请，下图为 **PASSED** 示例。



• 会议列表界面

Conference List 用来处理与已通过（PASSED）会议相关的事务。进入 **ALL Conferences** 板块可以查看到所有已通过（PASSED）会议的缩略信息。单击表格中的一行，则可以进入到对应会议的详细信息界面。



• 会议详细信息界面

Conference Detail 会显示一个会议的详细信息，包括全称（Fullname of Passed Conference），简称（Passed Conference），会议主席，会议所有 PC Member，会议全部或部分 Author（至多显示 5 位提交论文的作者），简介，会议阶段

(preparation(准备中), contribution(投稿中), reviewing(审稿中), grading(终评中) and ending(审稿结束)) 等等会议详细信息。

Uni C

log out

AccountConferenceMessagesSettings

Fullname of Passed Conference(Passed Conference)

Chair: testRobert2

PC Member:

Author:

Introduction:
None

Stage:

HeldPlace:

HeldDate:

Submission Date:

Submission Deadline:

ReleaseDate:

Preparation

None

2020-04-12

2020-04-16

2020-04-18

2020-04-28

You are not an author of this conference.

Submit paper now

copyright © 2020-2099 Uni Conference p.h.y

针对不同用户不同的角色，可进行的操作也会不同。对于 Chair，可以通过 Move to next Stage 使会议进入下一个阶段。同时在 Reviewing 阶段之前可以通过 Invite PC Member 邀请其他成员成为该会议的 PC Member. 对于 PC Member，可以在 Reviewing 阶段进入论文评审界面（未实现）。对于已经提交过论文的 Author 可以查看自己已提交的论文（未实现）或者在 Contribution 阶段点击 Submit More 提交新的论文。对于与其他用户则可以在 Contribution 阶段提交论文，第一次提交成功后则自动成为 Author 之一。

Uni C

log out

AccountConferenceMessagesSettings

2020 Software Engineering Conference(2020 SEC)

Chair: AIRuleTheWorld

PC Member:

Author:

Introduction:
This is a conference to discuss difficult problems when writing codes with a large group of people.

Stage:

HeldPlace:

HeldDate:

Submission Date:

Submission Deadline:

ReleaseDate:

Preparation

Shanghai

2020-04-14

2020-04-18

2020-04-19

2020-04-24

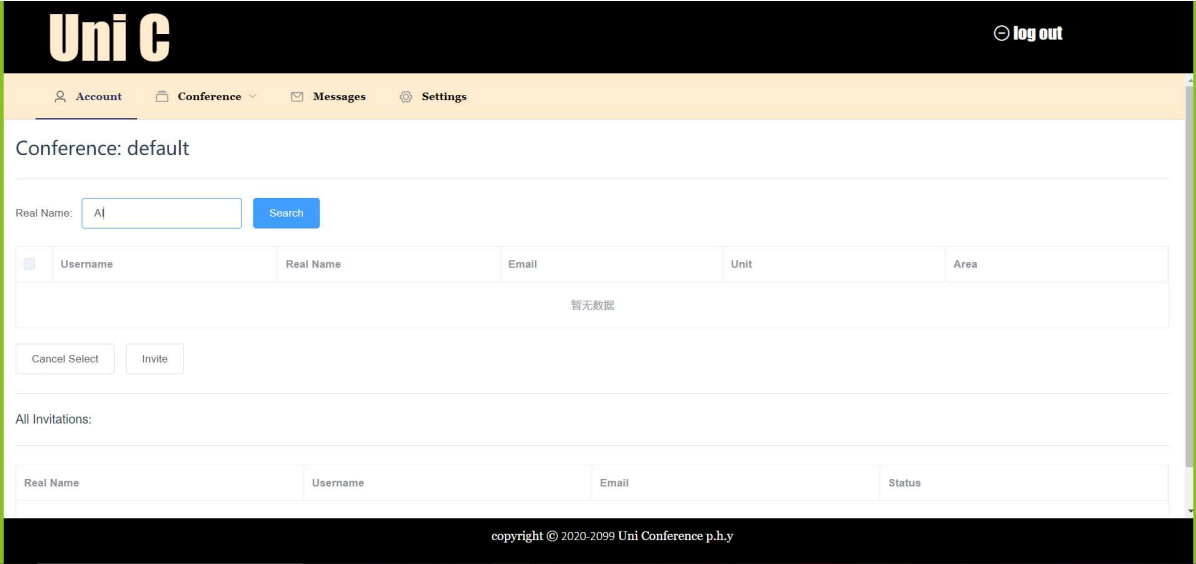
You are the chair.

Invite PC Member

Move to next Stage

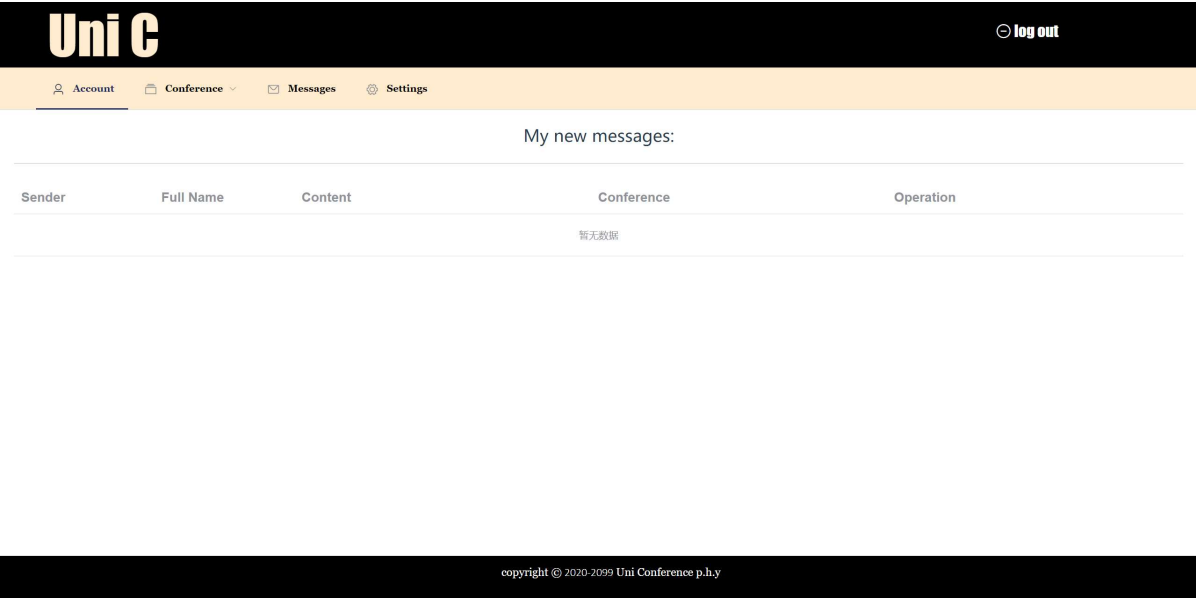
copyright © 2020-2099 Uni Conference p.h.y

• PC Member 邀请发送界面



通过搜索用户发送请求

• Message 界面



三、组员任务分配情况

胡彧锋（后端）：

用户请求：

- 获取自己申请的会议列表（通过、被拒绝、待审核）
- 获取通过的全部会议列表，以投稿
- 提交投稿

- 查看自己被邀请的会议
- 查看自己被邀请的会议

Chair 请求:

- 改变会议阶段
- 搜索用户列表
- 邀请 PC members
- 查看自己邀请的会议

潘星宇（前端）：

导航栏

系统主界面

- 用户信息请求

会议申请界面

- 会议申请查看
- 新增会议申请

会议列表界面

- 所有已通过会议查看

会议详细信息界面

- 根据用户身份与会议状态提供可以进行的操作

颜华（前端）：

论文上传界面

- 只能上传 PDF

PC Member 邀请界面

- 用户搜索

- 邀请发送

Message 界面

- 查看 PC Member 邀请

四、实验过程记录及问题解决

阶段一：Lab2 回顾阶段（3.24 - 4.1）

对 Lab2 中未优化的代码进行完善，修改一些页面的 CSS 样式，对站点风格进行重构

阶段二：任务分配（4.1）

在对整个项目需求有了基本的了解之后，我们对小组内成员的任务进行了分配，初步确定开发流程和时间节点。

阶段三：前后规范接口（4.1 - 4.4）

对目前所预见的到的前后端的接口进行规范，建立接口文档。

阶段四：前后端分离开发（4.4 - 4.9）

在统一后的接口下对前后端进行分别开发，前端以实现页面布局和样式为主

阶段五：本地测试以及 DEBUG（4.10 - 4.12）

前端已经完成基本布局的实现，后端完成基本数据返回的实现，同时前后端能通过规范的 AJAX 接口请求进行数据交互。因此在本地针对系统逻辑进行测试和 DEBUG。

若干问题：

****问题 1：** **admin 在 PASS 一个申请后可以继续 REJECT 该申请。

****原因：** **admin 在操作后没有立刻刷新页面，因此操作后的会议还留在 PENDING 栏目中，因此还可以继续进行 REJECT 操作

****解决：** **已解决，解决方法：操作后立即更新 PENDING 列表

****问题 2: ****用户主界面刷新后用户具体信息丢失

****原因: ****用户主界面通过本地缓存的方式读取信息，刷新后本地缓存丢失，没有再次通过 AJAX 向后端请求用户具体信息，因此用户具体信息丢失

****解决: ****每次加载页面时都通过 AJAX 请求来得到所有的用户信息

****问题 3: ****会议注册时，前后端均未对会议开始时间，投稿截止时间，评审结果发布时间进行合理性检查

****解决: ****考虑到会议开始时间，投稿截止时间等应该随着 Chair 实际推进会议 Stage 的时间来进行更新，因此留待后续需求进一步完善后再行设计相应逻辑。

****问题 4: ****会议详细信息界面，fullName 重复的用户会错误地成为 Chair

****解决: ****判定用户是否为 Chair 应该通过 User Id 来判断，不能通过 fullName 来判断

阶段六：云端部署（4.13）

本地 DEBUG 之后，将前后端分别部署到云服务器上。部署顺利。

与 LAB2 相似，LAB2 时已采取自定义模板对前端进行编译构建。

阶段七：云端测试（4.13）

通过 IP (<http://114.115.246.37:80/index>) 访问登录界面，登录后进入系统。测试后与本地行为一致，admin 会议申请审核，Chair 邀请 PC Member, 用户投稿等功能实现。

五、小组成员实验总结

胡彧锋

本次 lab 我转为负责后端部分。这是一次对 Spring 学习和使用的尝试。

- 前期学习：
- 学会了前后端分离的写法：

- 传统做法：后台运行 Java 代码，生成全部的 html 代码，接着通过 http 协议把 html 代码传输到浏览器，问题：
 1. 用户等待时间长
 2. 前后端开发效率低
- 前后端分离：先准备一个不包含数据的 html，把它传给浏览器，然后再通过 Ajax 技术，仅仅从服务器获取“纯数据”，然后把纯数据显示在 html 上，好处：
 1. 即便是后台数据库比较花时间，但是用户体验也比前面的方式好
 2. 后端只提供数据，所以前后端开发耦合度降低了很多，整体开发效率可以得到较大提高
- 学会了单页面应用：
 - 单页面应用指只有一个主页面的应用，浏览器一开始就要加载所有必须的 html, js, ss, 单页面的页面跳转仅刷新局部资源，多应用于 pc 端
 - 多页面就是指一个页面中有多个页面，页面跳转时是整页刷新
 - 单页面优点：
 1. 用户体验好，快，内容改变不需要重新加载整个页面
 2. 没有页面之间的切换，就不会出现“白屏现象”
 - 单页面缺点：
 1. 首次加载耗时比较多；
 2. 不利于 SEO
 3. 不可以用导航实现前进后退效果；
 4. 页面复杂度高

前端，我开始了对于 Spring Boot 方面知识的恶补。

• 开发过程

- Spring magic***

• 合作感想

万恶的 merge! ☹️

潘星宇

本次 lab 我转为负责前端部分。这是一次对 VUE 学习和使用的尝试。

- **前期学习：**

前端，我开始了对于 VUE 和 el-element 方面知识的恶补。包括

- **开发过程**

在本次 lab 的开发中，我认识到了规范接口的重要性。在项目越来越大，页面越来越多，请求越来越多，越来越复杂的时候。就需要对这些请求的接口事先做好分类和规范。这样后期花在统一变量名这种糟心事情上的时间就几乎不需要了。同时对于前后端的分离开发效率也有很大的提升。

随后我认识到了 VUE 框架相对于原生 JS 的强大之处，以更加精简的形式和代码完成了许多重要的功能。在脱离后端的情况下也能分离页头页脚。同时可以根据各种不同的情况动态的改变页面内容和元素 CSS 样式，v-系列非常简单好用，有点类似 JSP 的感觉。

- **合作感想**

zoom 会议依旧是开发日常。本次 LAB 相较于上次更加合理地使用远程仓库进行版本控制，养成了开发过后就上传远程仓库进行版本控制的好习惯，并通过 commit 信息良好地反映了每一次提交所做出的修改。

颜华

由于上次Lab 2我是写后端的，所以这次就写前端，熟悉一下前端的开发模式和工具



- **开发工具**

- Vue.js: <https://cn.vuejs.org/v2/guide>

主要学习了 Vue.js 的基本语法，组件基础，路由和储存。

- Element UI: <https://element.eleme.io/#/zh-CN/component/installation>

之前学习过 bootstrap 之类的前端框架，所以还是很好理解诸如此类的 UI 框架，基本都是要用的时候去查文档。但是使用这些框架的时候也会产生

莫名其妙的 bug，很大一部分原因就是框架元素自身的在其内部定义的模式。

- 浏览器控制台

我觉得浏览器 (chrome) 自带的控制台应该是前端网页开发最为重要的工具了，不仅可以查看元素的布局，CSS 样式，还可以查看报错信息，请求内容以及数据的存储。几乎绝大部分的 debug 调试都是借助于工具台来进行的

- 困难与挑战

- 页面的跳转与数据传递：Vue 使用 router 来控制各个页面，与之前 html 和 js 的跳转还是有很大不同的
- 表单提交附加文件：刚开始被 element UI 里的 Upload 控件迷惑了，最后还是用 html 原生的 input 输入框指定 type="file"完成的

- 总结：

做完这个 lab 后，对 Vue+springBoot 的前后端分离开发有了初步了解，深刻体会到了接口约定的作用。这种工程化、规范化的开发模式我认为在较大型的项目开发中还是很有必要的。