# 校园时间管理工具-系统设计报告

目录

[校园时间管理工具-系统设计报告 1](#_Toc1992)

[1. 引言 2](#_Toc12347)

[1.1编写目的 2](#_Toc21926)

[1.2 背景 2](#_Toc15115)

[1.3定义 2](#_Toc13342)

[1.4　参考资料 3](#_Toc3728)

[2. 界面设计 3](#_Toc10730)

[2.1用户登录 3](#_Toc9408)

[2.2用户注册 4](#_Toc9919)

[2.3 DashBoard 5](#_Toc32172)

[2.4 任务列表 6](#_Toc1241)

[2.5团队成员 7](#_Toc9060)

[3. 事务逻辑设计 7](#_Toc25537)

[3.1 用户类相关逻辑 7](#_Toc26141)

[3.2 团队相关 8](#_Toc3682)

[3.3任务相关 9](#_Toc23162)

[3.4标签管理 10](#_Toc28328)

[3.5 管理员相关 10](#_Toc13095)

[4.数据库设计 11](#_Toc25410)

[4.1用户信息数据库 11](#_Toc6173)

[4.2团队信息数据库 11](#_Toc12344)

[4.3 标签信息数据库 12](#_Toc29009)

[4.4 任务数据库 12](#_Toc18808)

[4.5 反馈数据库 12](#_Toc1951)

[4.6 评论数据库 13](#_Toc28296)

[4.7 申请数据库 13](#_Toc13860)

[4.8 组数据库 13](#_Toc2297)

# 引言

## 1.1编写目的

经过面向对象分析阶段，得到了系统的类图，细化的属性表和操作表，以及三层精化类图；该报告处于设计阶段，任务是给出各个类每层分解的实现方案，即界面显示方式，事物处理逻辑、数据库设计，不同任务的程序员可以根据此进行项目的实现。

目标读者：软件开发人员、项目评审人员、软件测试人员。

## 1.2 背景

待开发软件系统名称：校园时间管理工具；

本项目的任务提出者：软件工程课程设计组；

本项目的任务开发者：许羽晟；

用户：需求时间管理的师生；

实现该软件的计算机网络：任意装有对应环境的服务器设备；

## 1.3定义

用户输入：用户输入的用户名及密码；

用户信息即用户类别：存放在数据库表中，用此查询用户的信息和所属的类别；

管理员用户信息管理：管理员可以对注册用户和注册请求进行相应的管理；

用户基本信息：用户具有姓名、学号、学院、专业等基础信息；

团队创建：用户通过一系列操作创建一个新的任务团队；

任务信息：任务信息的总数据库，有相关权限的人可以对任务进行查看、完成、评论，任务的发起者可以对自己的任务进行一系列的修改；

任务输入：用户输入的任务描述、截止日期、完成期限、是否需要反馈等基本信息；

出错提示：当用户名和用户密码不匹配、任务描述、评论、反馈没有通过本地的校验机制时，会给予一些友善的提示，并要求用户重新输入；

显示界面：用户和计算机系统进行交互的显示渠道；

标签创建：用户通过一些操作和输入创建一个新的任务标签；

任务评论：用户可以对他的任务或是其余他能看看见的任务进行评论；

任务反馈：用户可以对他完成的任务进行一些反馈，由发起人审核反馈结果

## 1.4　参考资料

|  |  |  |
| --- | --- | --- |
| 编号 | 参考资料 | 作者 |
| 1 | 《软件工程（第三版）》，高等教育出版社，2012 | 齐治昌等 |
| 2 | 基于JAVA的博客系统　系统设计报告 | 佚名 |
| 3 | 《软件工程》PPT | 南航软工课程组 |

# 界面设计原型图

约定：目前该项目仍然处于work in progress阶段，具体的界面图还未设计出来，现在的设计也不能代表最终设计，目前仅通过原型制作工具大致的制作出一个表现出每个模块位置的图，具体的细节会在之后细化。(草稿)

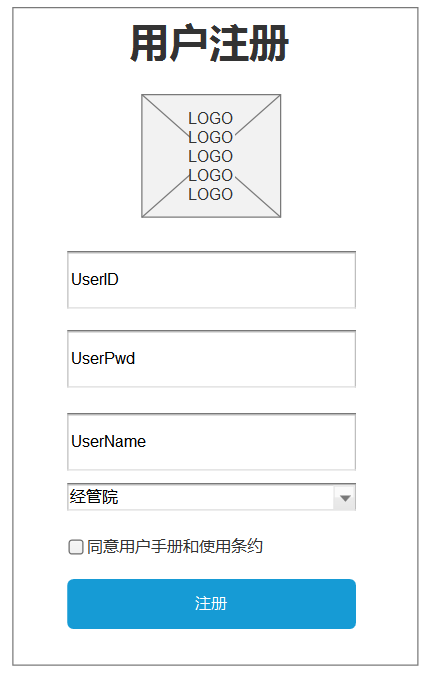
采用**Axure**进行界面原型的开发，这样设计界面可以有更直观的看到效果吗，为将来的开发提供更好的协助。

目前项目基本完成，请在第三大点中查看实际设计图(2020.12)

## 2.1用户登录



## 2.2用户注册



## 2.3 DashBoard



## 2.4 任务列表



## 2.5团队成员



## 2.6 详情&评论



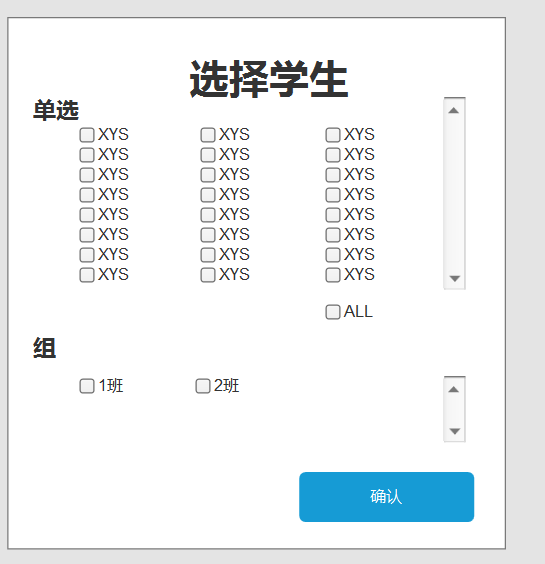
## 2.7 创建团队



## 2.8 发送任务



## 2.9 选择学生



## 2.10 提交



## 2.11 审核



## 2.12 标签管理



## 2.13 添加标签



# 事务逻辑设计

约定：代码采用伪代码进行描，不代表真实代码，而是希望能够用这些伪代码描述清楚我们的所有事物逻辑的行为和涉及到的不同类型数据之间的交互，仅代表该软件中事务的处理逻辑。同时为了便于突出重点，可能会忽视一些细枝末节的操作(如合法性检查)和优先级不高的方法

Input:泛指输入，来自前端的表单元素

Fetch:自动获取的表单项，无需用户填写，例如当前登陆账号，由vuex和js管理

XXXInfo:相应结构体的简写方法，用来让事物逻辑更清晰

## 3.1 用户类相关逻辑

1. 用户登录

Input: UserID,UserPassword

Query UserID,UserPassword From UserDataBase

If Result = Success

GoTo DashBoard

Else

Prompt Error

1. 用户注册

Input: UserID,UserPassWord,UserCollege,UserName

Query UserID From UserDataBase

If Havn’t registered

Write Those Data Into RequestDataBase

Insert New Record Into UserDataBase Set UserID = UserID

Update userType To Waited Where UserID = UserID

Else

Prompt Error

1. 信息修改

Input:NewUserCollege,newUserName,newUserPassWord

Fetch:UserID

If Only Password modified

Update UserInfo Where UserID=UserID

Else

Update NewUserPassWord Where UserID=UserID

Insert New Record RequestDataBase Set newUserName=newUserName,newUserCollege=newUserCollege

## 3.2 团队相关

1. 新建团队

Input:TeamName,TeamDescription

Query TeamName From TeamDataBase

If Result = Exist

Prompt Error

Else

Insert New Record TeamDataBase

1. 删除团队

Input:

Fetch:TeamID

Delete From TeamDataBase Where TeamID=TeamID

1. 设置标签页

Input:TagID

Fetch:TeamID

Update TagID In TeamDataBase Where TagID=TagID

1. 加入团队

Input:quickJoinCode

Fetch:UserID

Query From TeamDataBase Where quickJoinCode = quickJoinCode

## 3.3任务相关

1. 发送任务

Input:StudentID[] GroupID[],TaskName,TaskDescription,TaskDeadLine,requireReview

Fetch:TeamID

Let StudentArray = []

For Each Student In StudentID

StudentArray.UniqueJoin(Student)

For Each ID in GroupID

StudentList[] = QueryListFromGroupDataBase(ID)

For Each Student In StudentList

StudentArray.UniqueJoin(Student)

WriteTaskIntoTaskDataBase(StudentList,TaskName,TaskDescription,TaskDeadLine,requireReview)

1. 检查任务

Input:

Fetch:TaskID,StudentID

Query ReviewerID From ReviewerDataBase Where ID=TaskID + StudentID

If Result = NULL

Prompt No Report

Else

Show Report

1. 审查任务

Input: ReviewContent,Accept

Fetch:ReviewID

Update ReviewDataBase Where ReviewID=ReviewID Set ReviewContent=ReviewContent,Accept=Accept

## 3.4标签管理

1. 新建标签

Input:TagName,TagColor,TagDescription

Fetch:TagID

Insert NewRecord In TagDataBase.TagInfo Set TagInfo = TagInfo

1. 新建标签页

Input:TagName,TagDesciprtion

Fetch:

Insert NewRecoted In TagDataBase Set TagInfo = TagInfo

1. 修改标签

Input:TagName,TagColor,TagDescription

Fetch:TagIndex

Update Tag In TagDataBase.TagInfo Where TagIndex=TagIndex Set TagInfo = TagInfo

## 3.5 管理员相关

1. 审核用户请求

Input:RequestInfo,Accept

Fetch:AdminID

UserInfo = Query From UserDataBase Where UserID = RequestInfo.UserID

If Accept

Update UserDataBase Where UserId = RequestInfo.UserID Set UserInfo = RequestInfo

Else

Update UserDataBase Where UserID = RequestInfo.UserID Set UserInfo.Type = USERACCOUNTSTATE.Rejected

1. 修改用户信息

Input:ModifiedUserInfo

Fetch:AdminID

Update UserDataBase Where UserID=ModifiedUserInfo.UserID Set UserInfo = ModifiedUserInfo

## 3.6 统计相关

1. 统计用户完成率情况

Input:

Fetch:TaskID

TaskInfos = Query From TaskDataBase Where TaskID=TaskID

Finish = 0

Total = |TaskInfos|

For Task In TaskInfos

Finish = Finish + (Task.Finish == True)

FinishRate = Finish/Total

DrawStatisticsBar(TaskInfos)

1. 统计团队总体任务数量

Input:  
Fetch:TeamID

Tasks = Query \* From TaskDataBase Where TeamID=TeamID

Total = |Tasks|

## 3.7评论相关

1.评论

Input:Comment

Fetch:TaskID,User

Insert Comment Into CommentDataBase Where TaskID = TaskID Set Comment=Comment,User=User,Likes = 0

1. 删除评论

Input:Comment

Fetch:TaskID

Delete Comment From CommentDataBase Where TaskID=TaskID

1. 点赞

Input:

Fetch:CommentID,UserID

AlreadyLiked = Query From CommentDataBase Where CommentID=CommentID

if(AlreadyLiked)

--Likes

LikeUser.remove(UserID)

Else

++Likes

LikeUser.add(UserID)

AlreadyLiked ^= AlreadyLiked

Update CommentDataBase Where CommentID=CommentID Set CommentInfo = CommentInfo;

# 4.数据库设计

约定：以下内容不代表最终定稿，仅代表根据目前需求识别下的数据库设计，内容可能会被修改，这边只写出键名、类型和该键的描述，而非真实的数据库文件，这样可以更方便届时项目的开发，以及便于他人理解数据库组织的结构

字符串拼接：即指将数组拼接成字符串存储入数据库

## 4.1用户信息数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 唯一 |
| userAccount | Text | 存储用户的账号信息 |
| password | Text | 存储用户的密码凭据 |
| tasks | Text | 存储用户的人物信息 |
| userType | Int | 存储用户的类别 |
| userGroups | Text | 存储用户的所有团队信息 |
| classes | Text | 存储用户的班级信息 |
| requestId | Int | 信息修改请求ID |
| userName | Text | 用户的姓名 |

## 4.2团队信息数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储团队的标识，唯一 |
| groupName | Text | 存储团队的名字 |
| creatorId | Text | 项目的发起人 |
| classList | Int | 存储项目的班级ID |
| memberList | Text | 存储项目的所有学生ID，字符串拼接 |
| TaskList | Text | 存储所有的任务ID，字符串拼接 |
| linkCode | Int | 快速加入团队的邀请码 |
| groupDesc | Text | 存储团队的描述信息 |

## 4.3 标签信息数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| tagID | Int,主键 | 存储标签页的标识，唯一 |
| schemeName | TEXT | 里面的所有标签的ID号 |
| tagList | Text | 这套标签的列表 |

## 4.4 任务数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储任务的标识，唯一 |
| description | Text | 存储任务的描述信息 |
| deadline | Date | 存储任务的截止日期 |
| userList | Text | 存储所有参与者ID，字符串拼接 |
| statesList | Text | 存储所有完成的ID，字符串拼接 |
| needFeedback | Boolean | 存储是否需要反馈 |
| feedbackID | Text | 存储所有的Review ID，字符串拼接 |
| taskTitle | Text | 任务标题 |
| commentsID | Text | 存储所有的评论 |
| schemeID | Text | 配色方案 |
| taskSender | Text | 发起人 |

## 4.5 反馈数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储反馈的标识，唯一 |
| reviewTime | Text | 存储反馈的时间 |
| senderID | Int | 存储发送者的ID |
| feedbackContent | Int | 存储反馈的内容 |
| status | Text | 存储反馈结果 |
| feedbackTime | Time | 存储反馈的时间 |

## 4.6 评论数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储评论的标识，唯一 |
| comment | Text | 存储评论的内容 |
| commenterID | Int | 存储评论者的ID |
| commentDate | Time | 存储评论时间 |

## 4.7 申请数据库

|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储申请的标识，唯一 |
| Request\_info | Text | 申请，存储方式ID\_PWD\_Name\_College，字符串拼接 |
| Request\_type | Text | 申请类型 |

## 4.8 组数据库

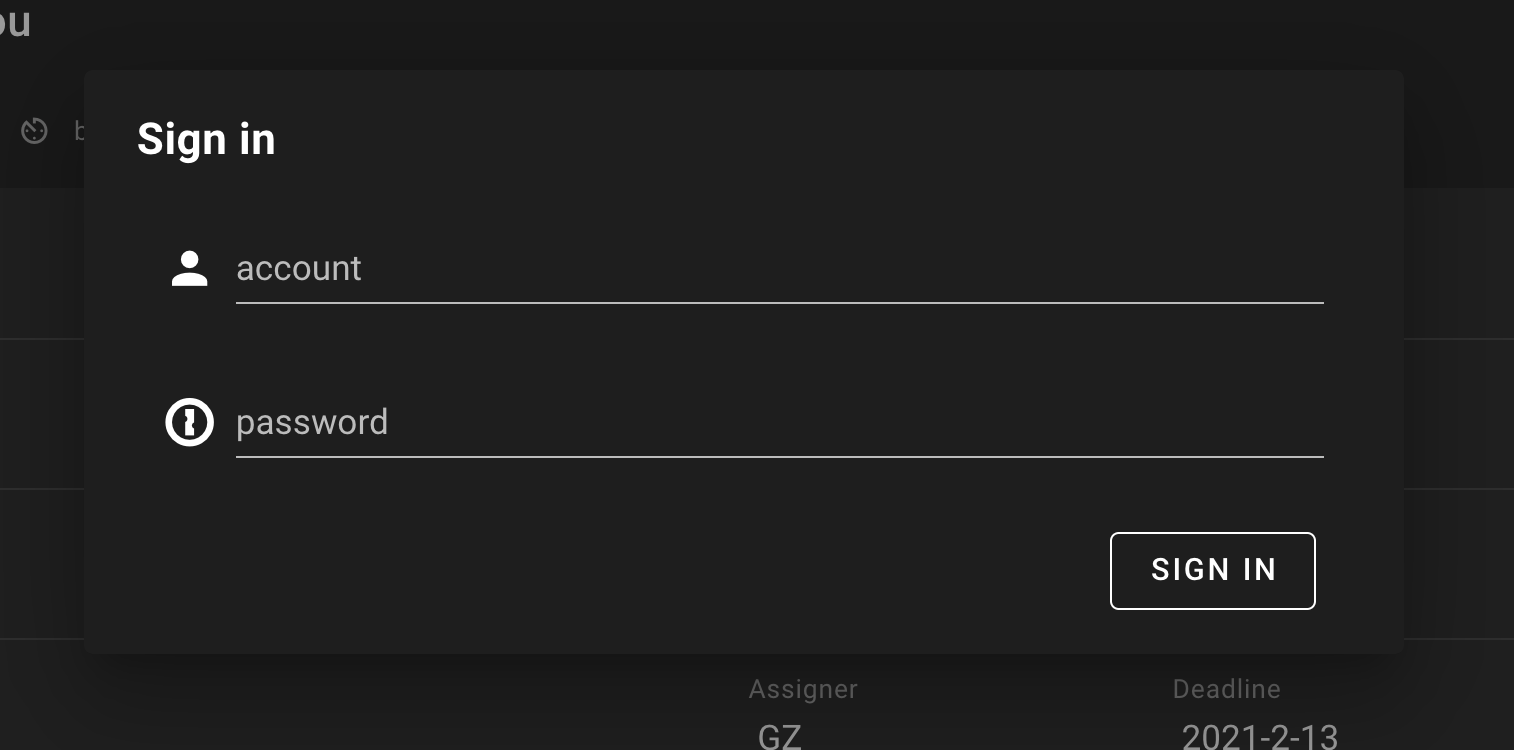
|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储申请的标识，唯一 |
| studentList | Text | 学生ID，字符串拼接 |
| className | Text | 班级名 |
| teacherList | Text | 教师ID |

## 4.9 头像数据库

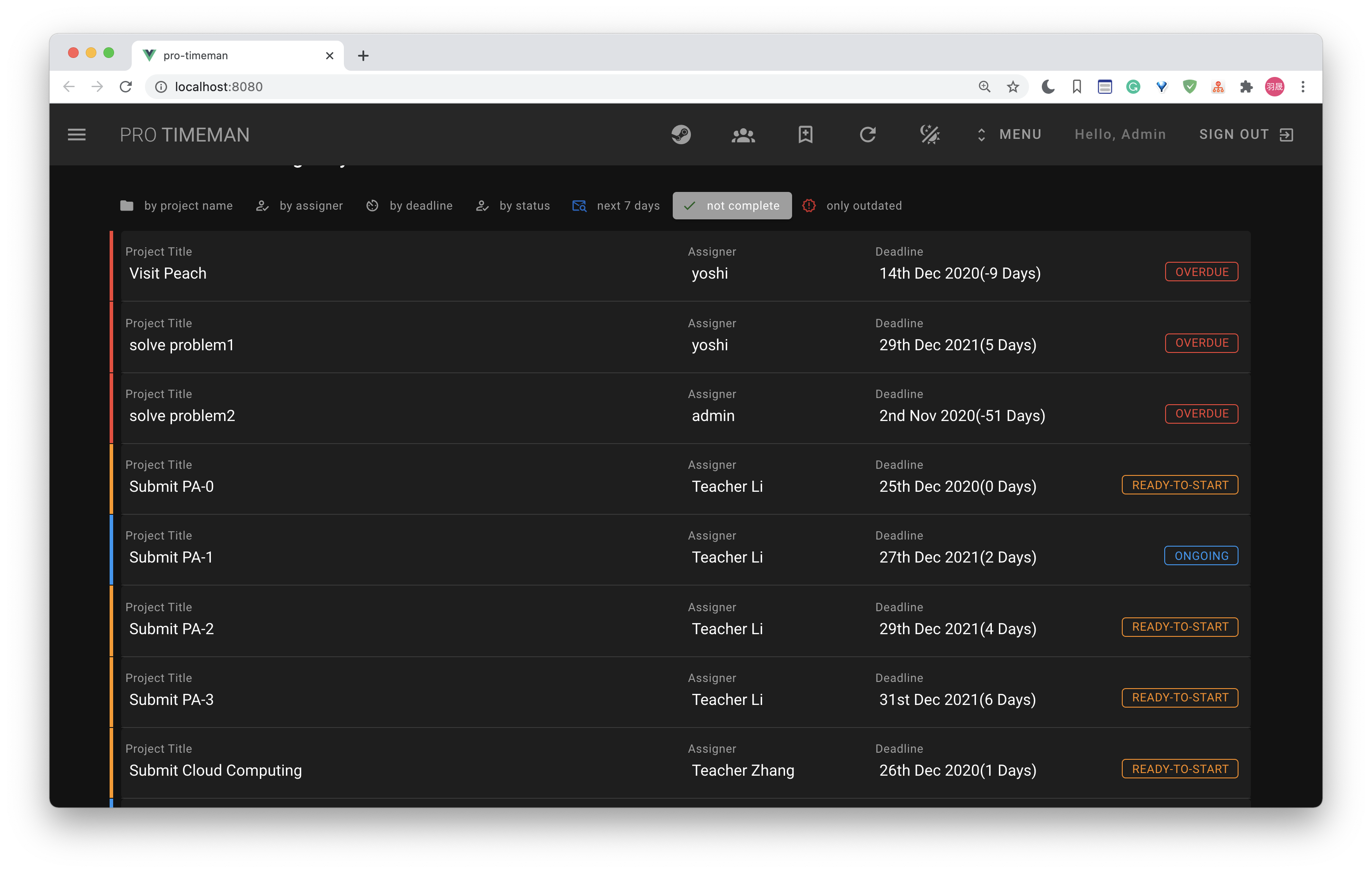
|  |  |  |
| --- | --- | --- |
| 键名 | 类型 | 描述 |
| ID | Int,主键 | 存储用户的标识，唯一 |
| URI | Text | 存储头像资源路径 |

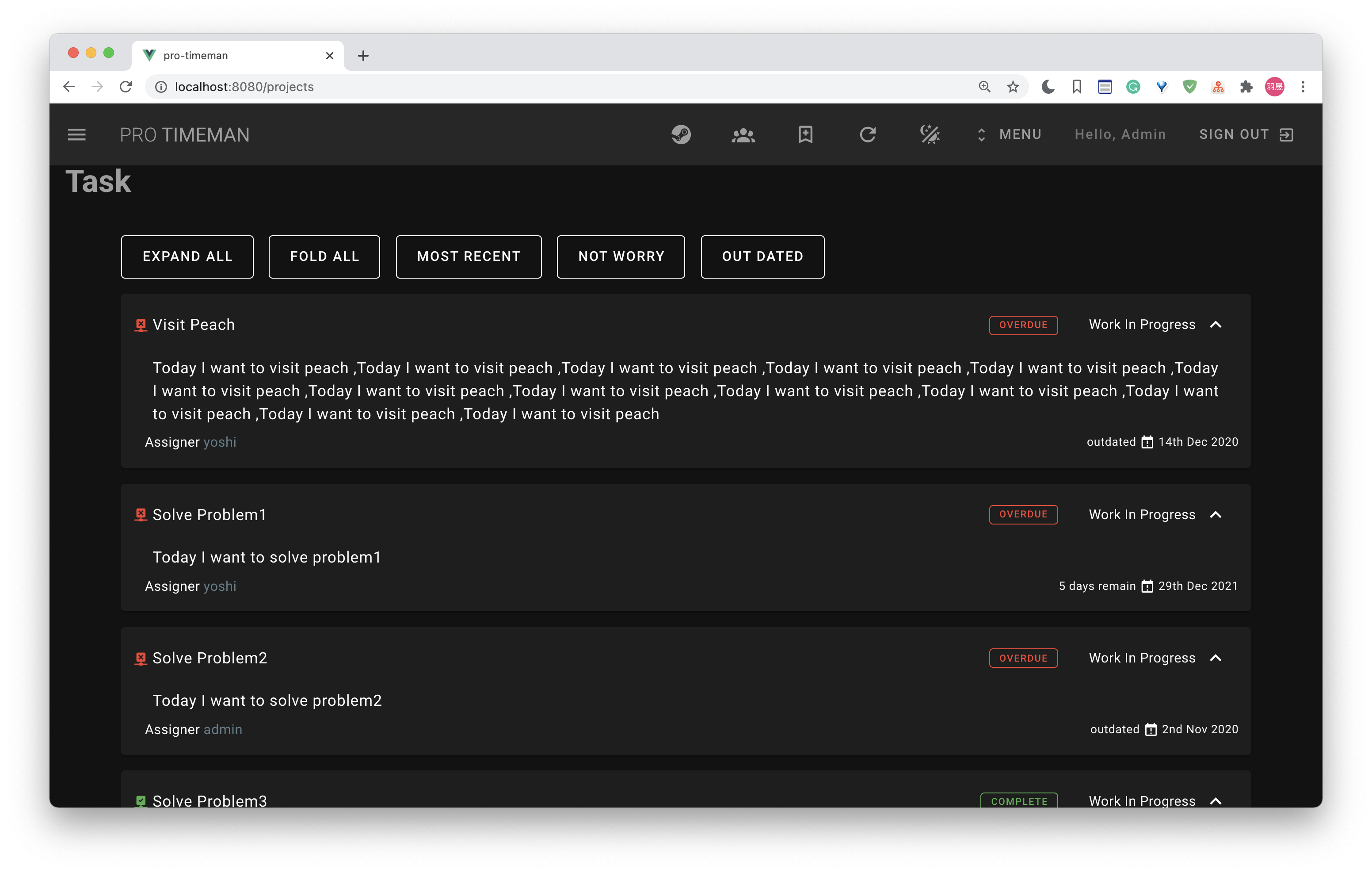
# 界面设计实际图

## 5.1用户登录

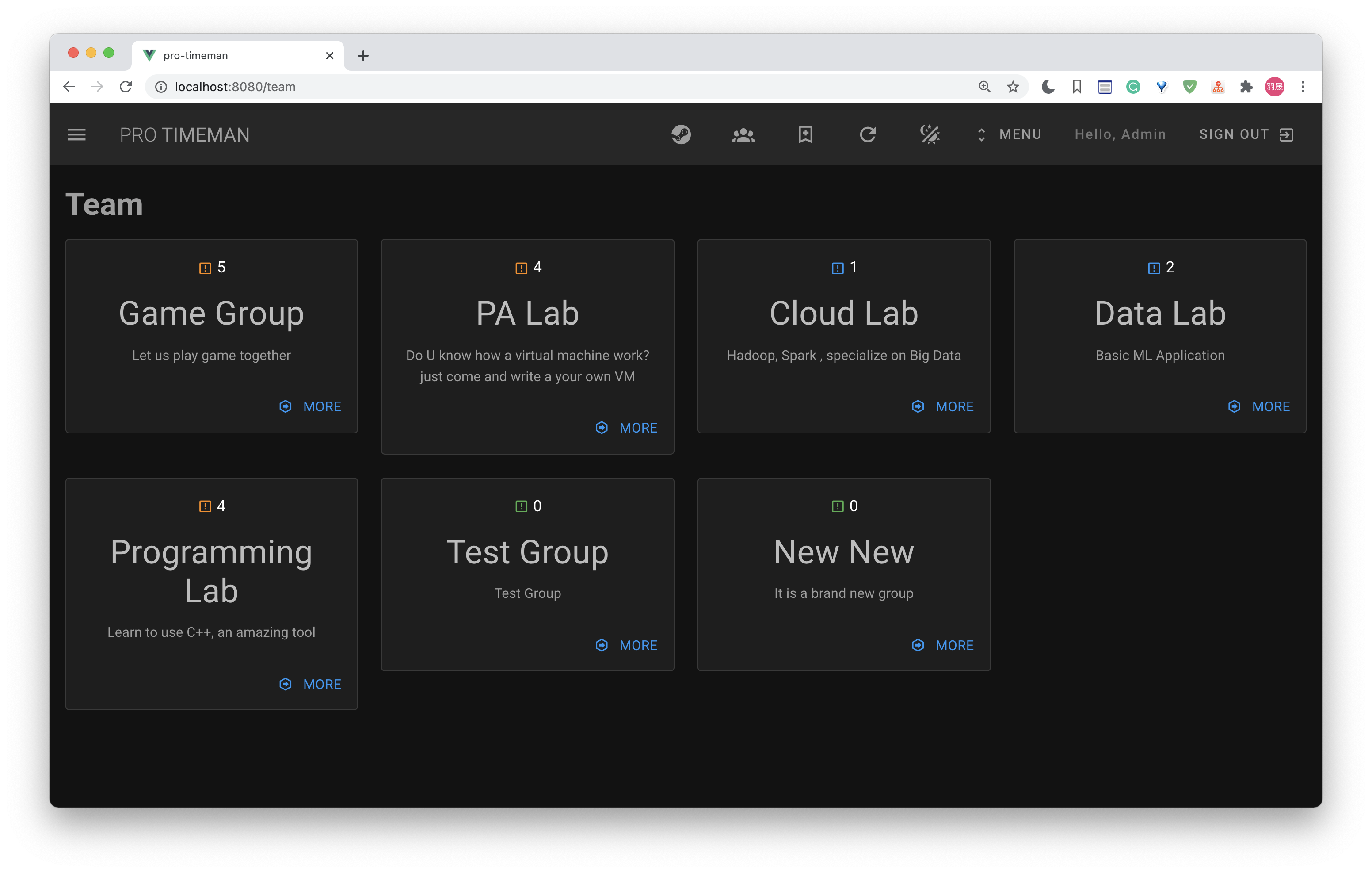


## 5.2 DashBoard

5.3 任务列表

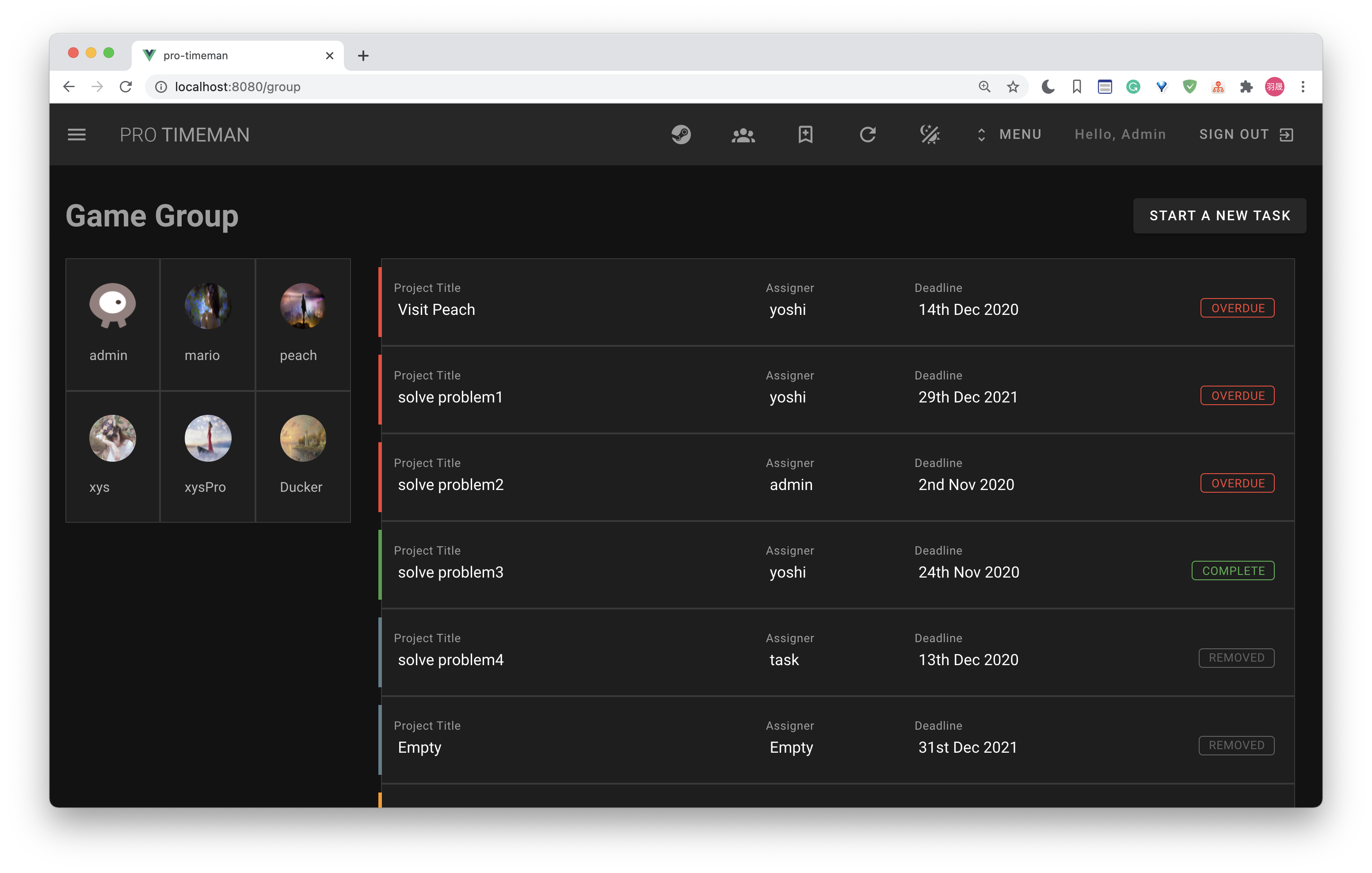


## 5.4 团队面板

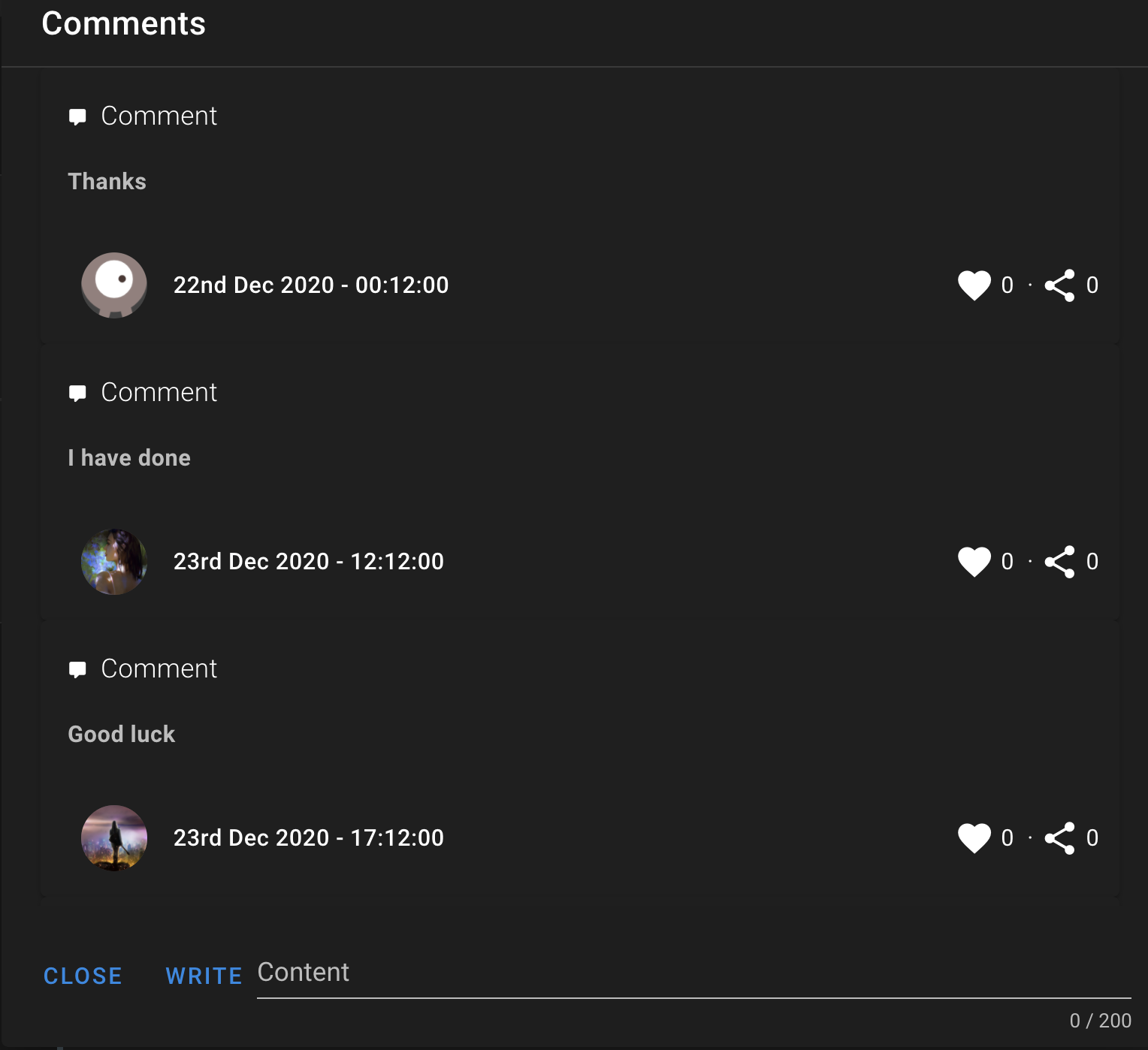


## 5.5 团队详情

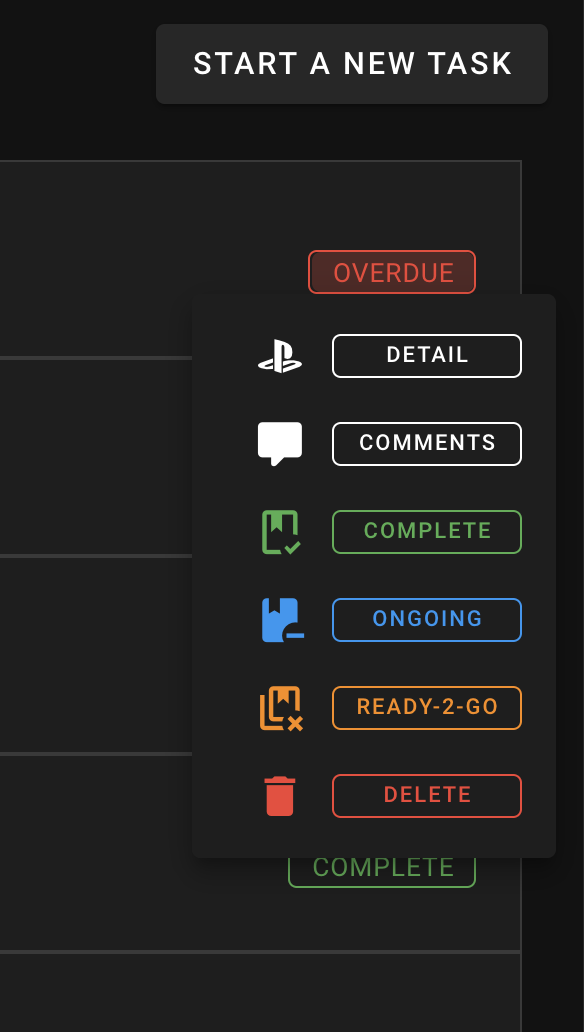
左边参与用户，右边任务列表



## 5.6 评论



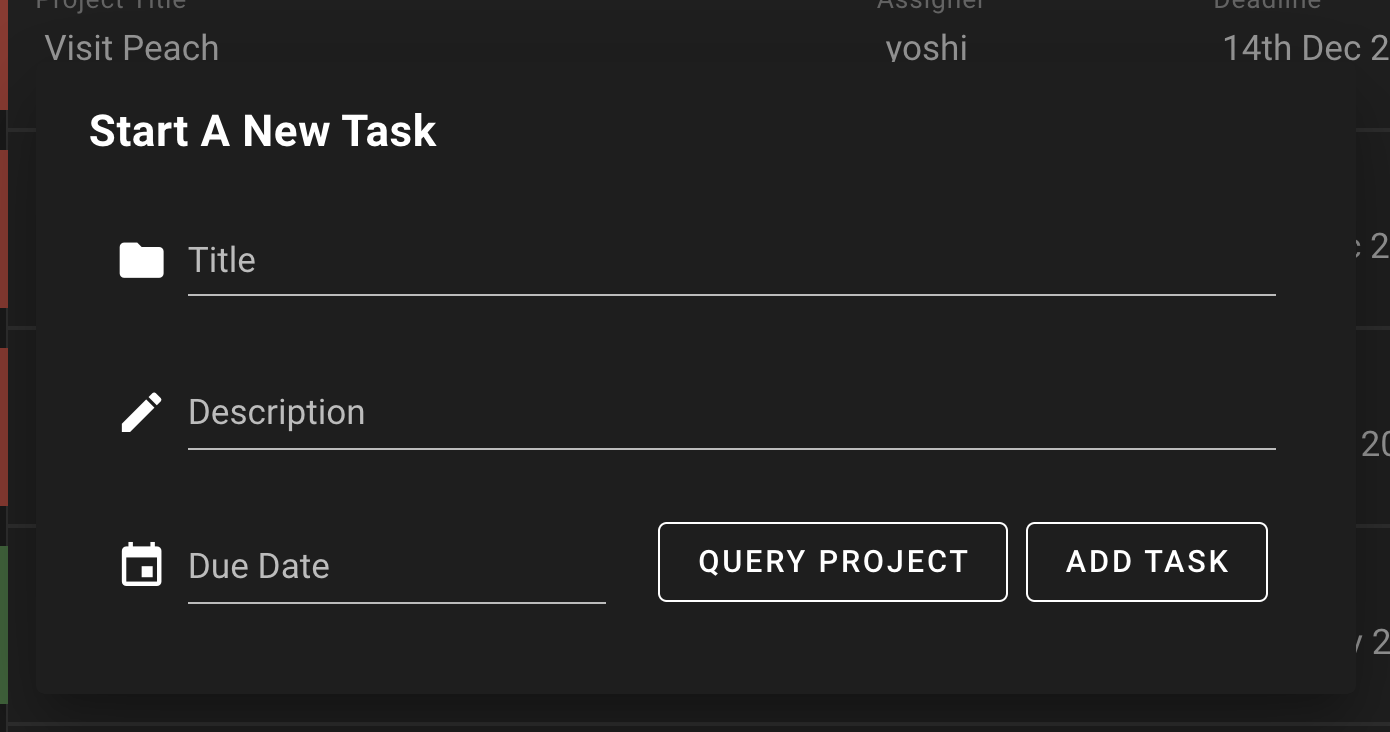
## 5.7 修改任务状态&菜单



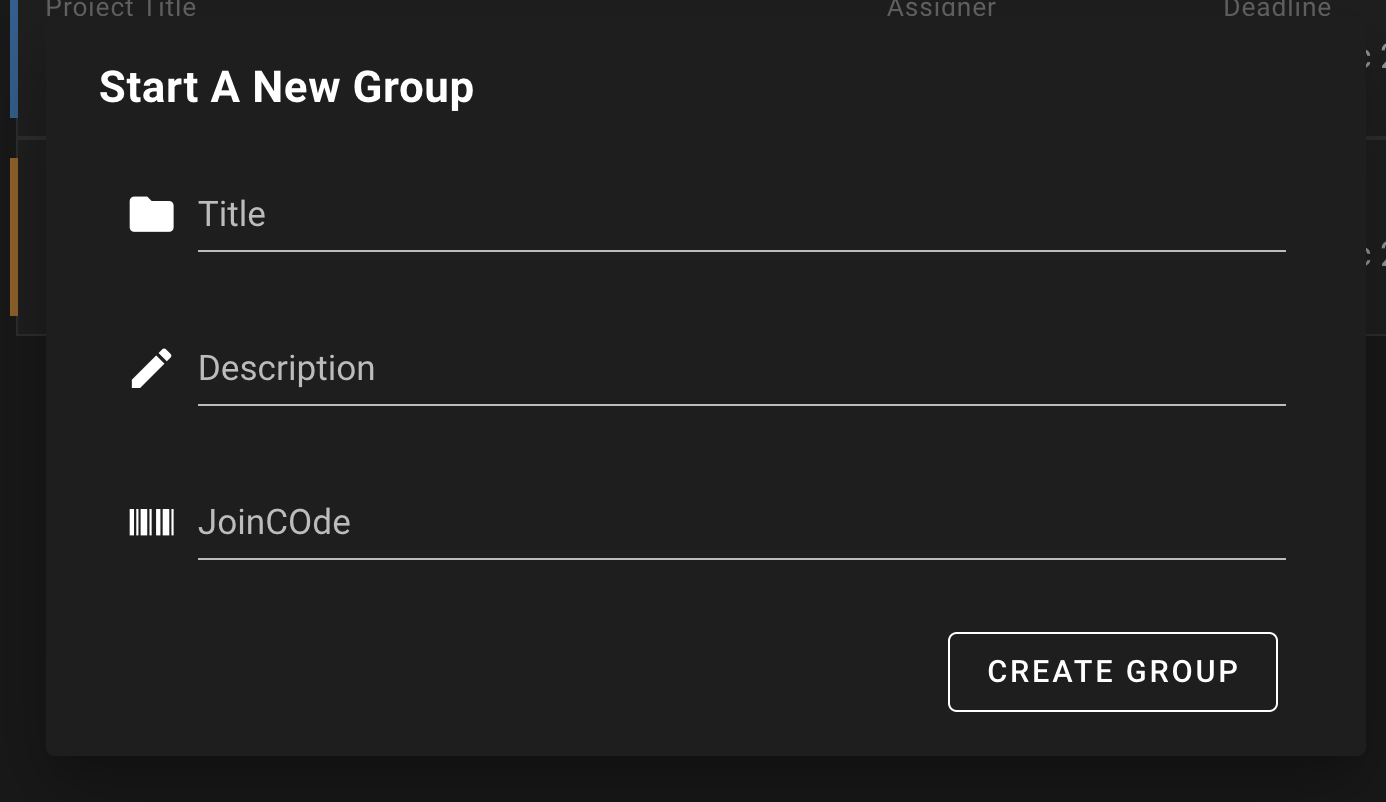
## 5.8 任务小组成员完成详情



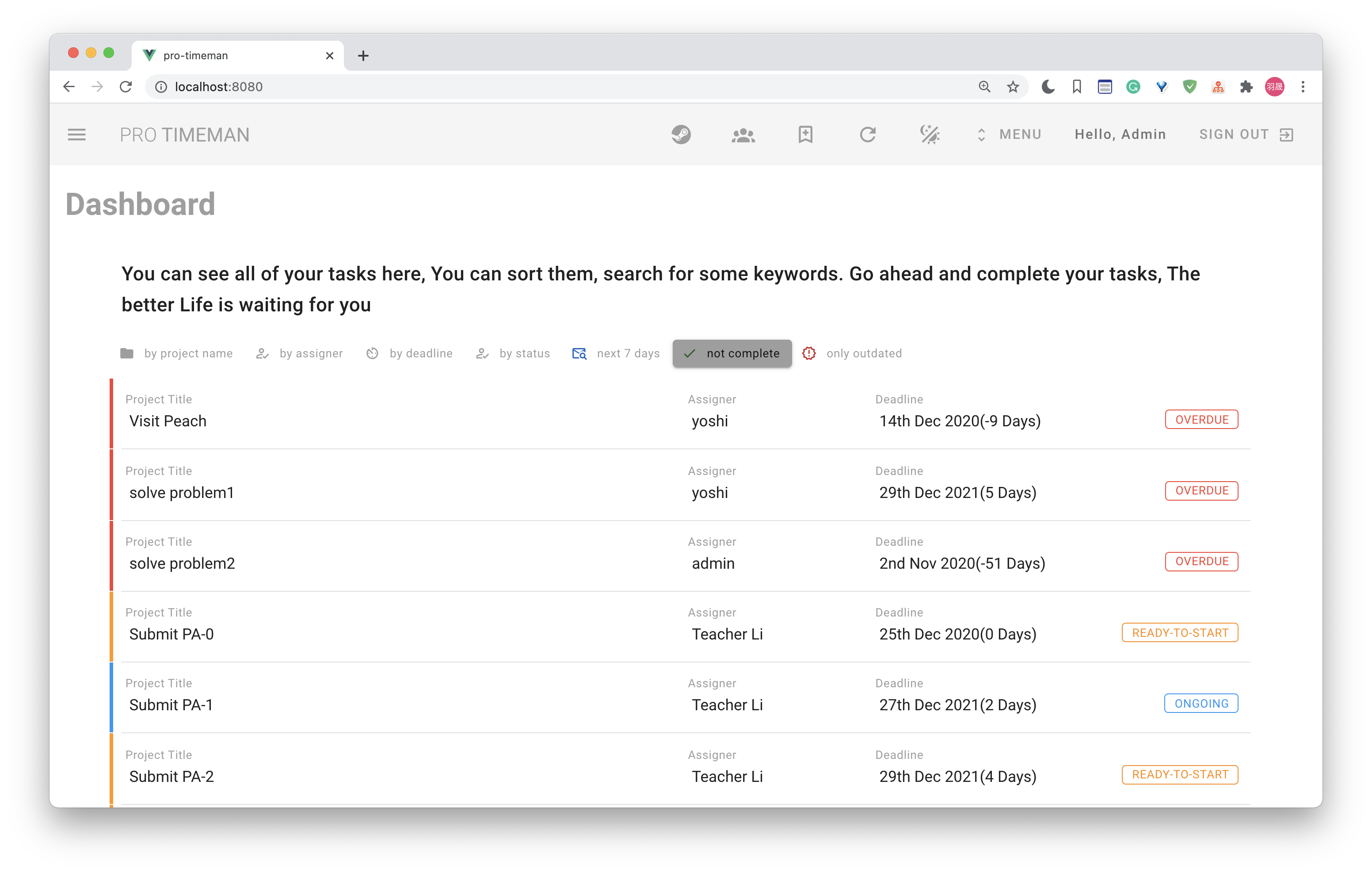
## 5.9 新建任务



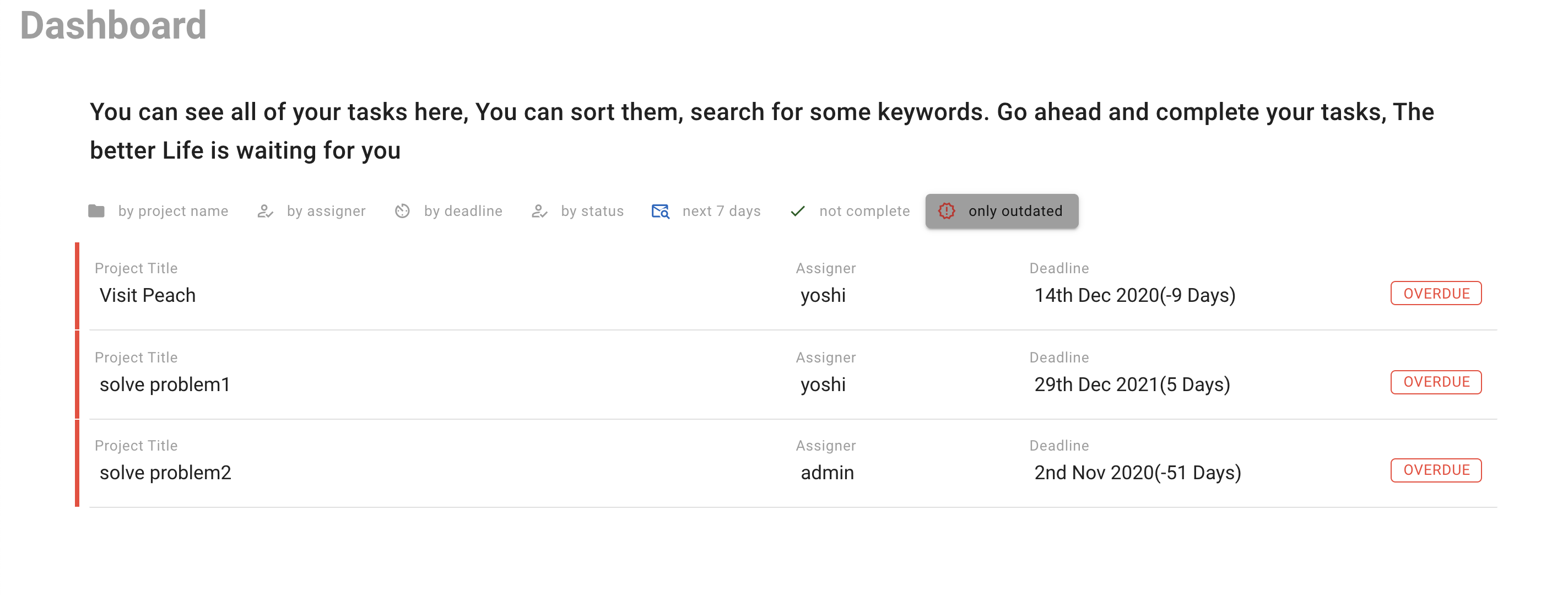
## 5.10 新建团队



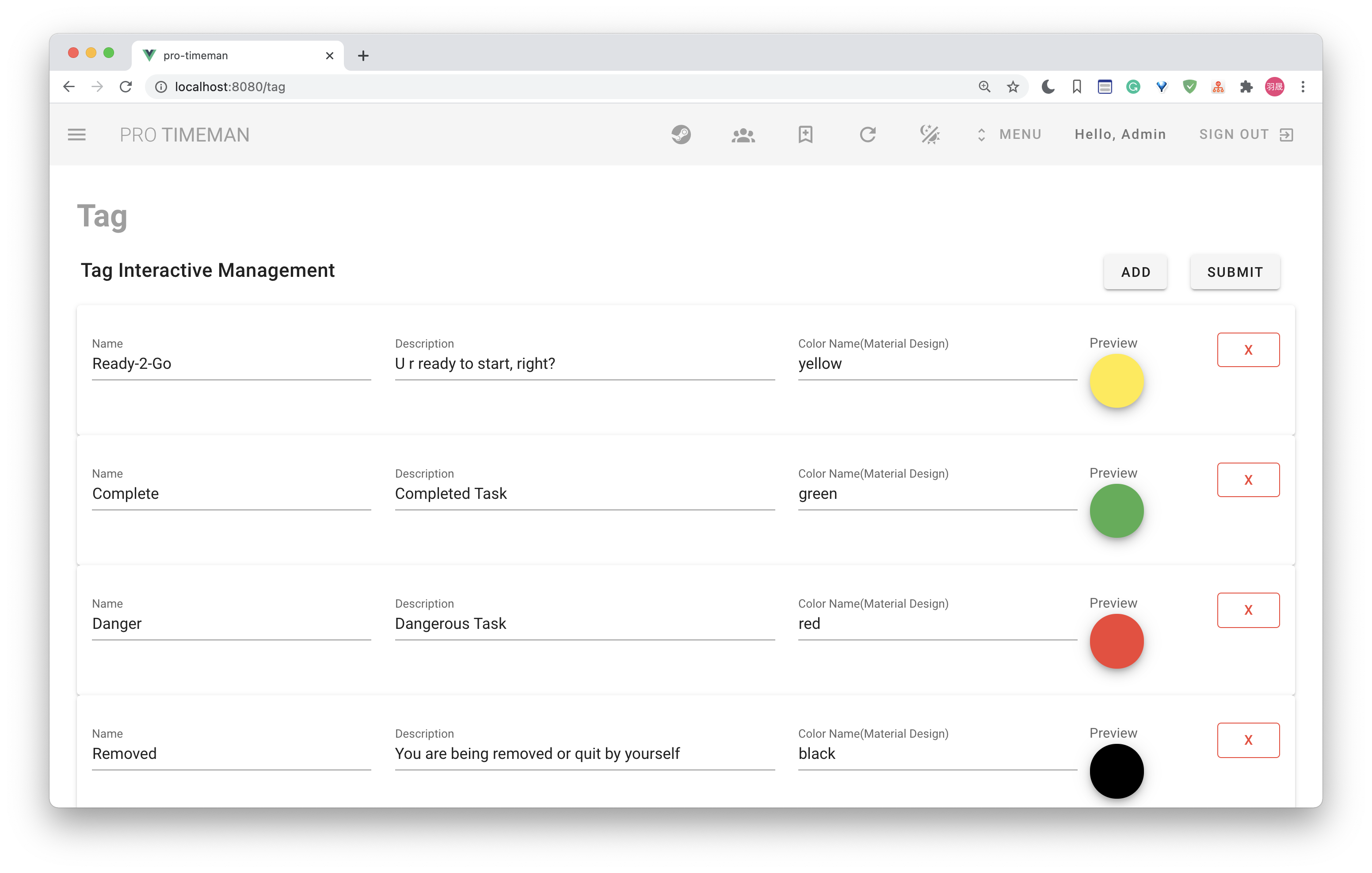
## 5.11 白天/黑夜切换



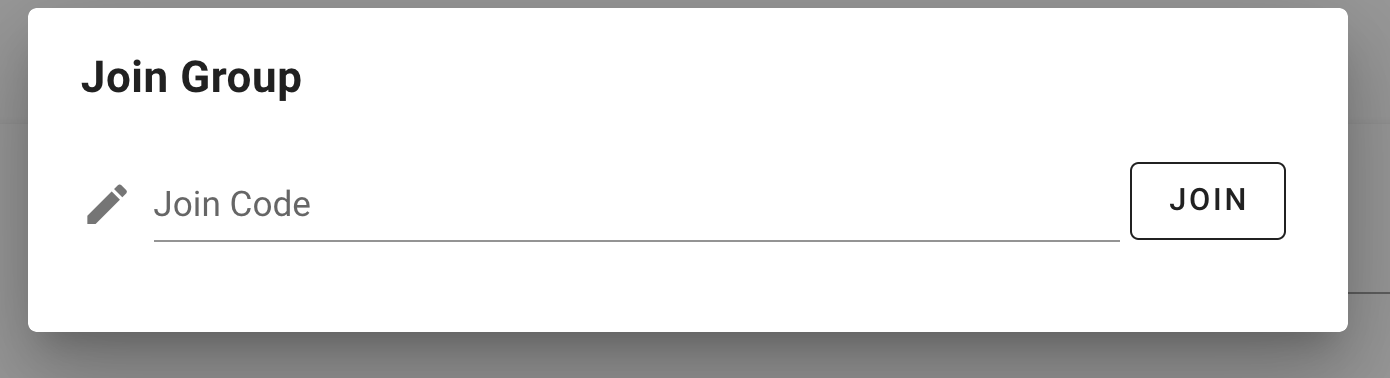
## 5.12功能筛选



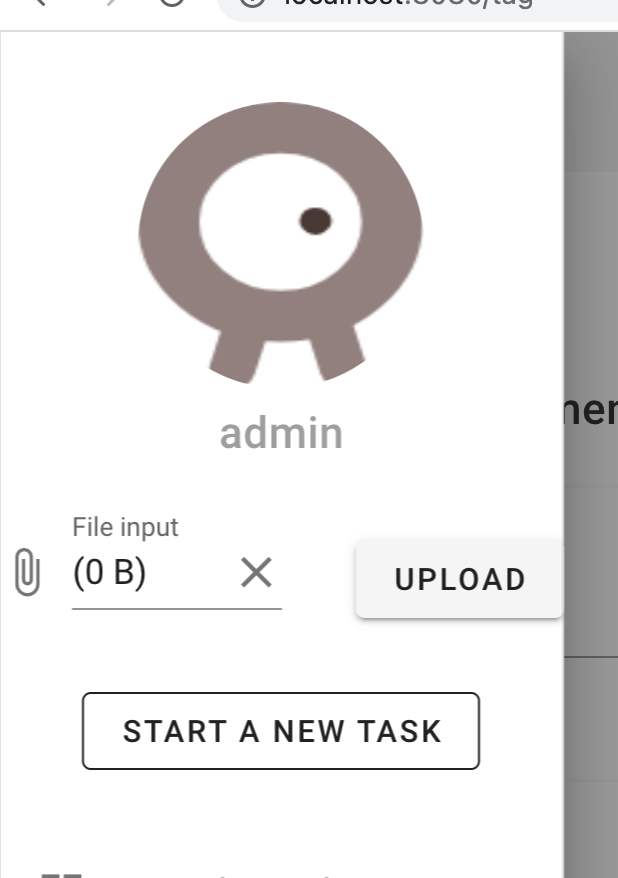
## 5.13 标签修改



## 5.14 加入团队



## 5.15 上传头像



## 菜单栏（抽屉菜单 & 弹出菜单）

