

算法竞赛开赛提醒聚合站

软件需求说明书

目 录

1. 简介	1
1.1 目的	1
1.2 背景	1
1.3 概述	1
1.4 定义	1
2. 综合描述	1
2.1 产品前景	2
2.2 产品功能	2
2.3 用户特征	2
2.4 一般性限制	2
2.5 假设和依赖	3
2.6 参考文献	3
3. 详细需求	3
3.1 功能需求	3
3.2 功能用例图	7
3.3 用例说明	11
3.4 外部接口需求	14
3.5 性能需求	14
3.6 质量属性	14
3.7 其他需求	15

1. 简介

1.1 目的

为明确算法竞赛开赛提醒聚合站软件，统筹安排项目规划与进度，组织软件开发与测试，撰写本文档。本文档供项目经理、设计人员、开发人员参考。

1.2 背景

通过对现行主流的算法竞赛网站（包括 LeetCode、AtCoder、CodeForces、洛谷、牛客网等）的充分调研，注意到各算法竞赛网站间存在信息壁垒，互相之间的赛事活动完全不互通。一个算法竞赛爱好者往往需要频繁、手动地往返各大网站间以留意当前竞赛开展情况，十分不便。而根据本组调研，目前暂无较方便的算法竞赛聚合网站及赛事提醒工具。因此，我们希望有一个一站式平台，以整合所有算法竞赛网站的相关信息，并为用户提供订阅、推荐及推送提醒服务。通过这一平台，让算法竞赛爱好者们可以更加快速、便捷、及时地获知各种竞赛的消息。

1.3 概述

算法竞赛开赛提醒聚合站应为一个整合不同算法竞赛平台信息的门户网站。用户可通过浏览器访问网站并进行交互。服务端同时还应支持浏览器、邮箱和 QQ 对用户进行赛事相关信息推送。

1.4 定义

术语	定义
【普通用户】	一经系统注册，即成为【普通用户】。拥有登录、在设置中绑定 QQ、邮箱、接收推送，以及订阅的感兴趣的比赛等权限
【管理员】	除具有【普通用户】的权限外，还额外具有浏览/更改用户信息、封禁/删除用户、添加/修正/删除赛事信息的权限
【订阅】	用户可根据自己的喜好选择接收何种类型的赛事信息； 用户在个人中心里选择接收某类赛事信息的行为称为【订阅】
【预约】	用户可根据自己的喜好选择准备参加哪些赛事； 用户在主页里选择参与某次比赛的行为称为【预约】
【日历】	网站主页以日历的形式展示各大算法站的竞赛开赛信息
【爬虫】	指网络爬虫，网站信息需通过【爬虫】定时自动爬取，保存在服务器并同步在网站上
【推送】	当服务器检测到用户【订阅】的赛事更新或用户【预约】的赛事即将开始时，通过浏览器、邮件、QQ 等方式提醒用户注意

2. 综合描述

本网站旨在打造一个算法竞赛信息的聚合平台，为广大计算机行业爱好者和从业者提供准确及时的算法竞赛信息，减轻相关人员收集相关竞赛资讯的精力，并鼓励号召更多的人了解算法竞赛并参与

到其中，为相关人员提供便利，进一步扩大算法比赛的影响力。为此，我们需要建设一个算法竞赛开赛提醒聚合站，该一站式平台的主页将以日历的形式展示各大算法网站（如 LeetCode，洛谷，牛客网等）的竞赛开赛信息，同时可以对用户订阅的比赛信息以 QQ 或邮箱等方式进行推送，支持用户自定义订阅需求。本网站力求在功能上实用可靠，功能完备，方便快捷，且在设计上体贴人性，风格简约大气。

2.1 产品前景

本网站是对现行主流算法竞赛网站的算法竞赛信息的聚合站，主要以爬虫的方式获取主流大型算法竞赛站的竞赛信息并进行汇总。由于各算法竞赛网站之间存在消息壁垒，故设计此网站以对各个网站的算法比赛信息进行汇总，并以订阅的方式，方便用户快捷方便的进行查看。本网站主要需以爬虫的方式或对接算法竞赛网站 API，以获取其算法比赛消息。将比赛信息汇总制作成日历，方便用户进行查看。

2.2 产品功能

本产品的主要功能包括：

- 账号管理：用户注册，用户登录和登出，用户信息管理，密码找回和修改。
- 日历信息管理：日历信息收集与自动更新，历史日历管理，日历浏览方式筛选，日历详情查看
- 订阅预约管理：自定义订阅设置，用户订阅赛事，用户预约赛事。
- 推送管理：推送方式设置，推送订阅赛事信息
- 智能推荐系统：用户偏好设置，赛事智能推荐
- 讨论区版块：讨论区主题管理，用户发帖与浏览，用户评论与回复
- 比赛信息上传：用户上传比赛信息（网站未收录），赛事信息审核，赛事公开

2.3 用户特征

产品主要面向算法学习者、Oier、程序员及准程序员。开发人员应结合产品主要受众的用户面貌设计较好的交互 UI；服务端运维人员应具有基本的数据库、网络知识，能对用户反馈进行及时合理的回应。

2.4 一般性限制

- 至少支持 Windows 7 及之后版本的操作系统。
- 本产品为浏览器访问，至少支持在最新版 Edge/Chrome/Firefox 上正常显示。
- 主要支持中国地区使用，采用简体中文为界面语言。
- 服务器要求：自行架设服务器，至少支持 5000 名用户注册，至少支持 50 名用户同时在线，至少支持 100 名用户同时接收推送。
- 至少需要支持的算法网站：洛谷、LeetCode、牛客网、Atcoder、CodeForces。
- 每日 6:00 至次日 4:00 需能够提供稳定的服务。
- 至少保证最基础的用户信息安全。

- 至少需要支持的推送方式：浏览器、邮箱、QQ。注册时可以使用第三方手机验证码 api 或邮箱的方式进行验证注册。
- 服务端需要使用可维护的主流编程语言开发。为便于验收，请尽可能使用 c++/java/python 编写。
- 要求使用 MySQL 等主流数据库。为便于验收，请尽可能使用 MySQL 数据库。

2.5 假设和依赖

使用爬虫时需首先浏览该网站的 robots.txt，只爬取各个算法网站中允许爬取的算法竞赛信息，不能违反法律法规。

2.6 参考文献

- [1] 清华大学软件需求规格说明 <http://canvas.tongji.edu.cn/courses/53736/files>
 [2] 开源软件需求文档模板 <https://github.com/eynol/software-document-boilerplate>
 [3] 计算机软件产品开发文件编制指南(GB 8567-1988)

3. 详细需求

3.1 功能需求

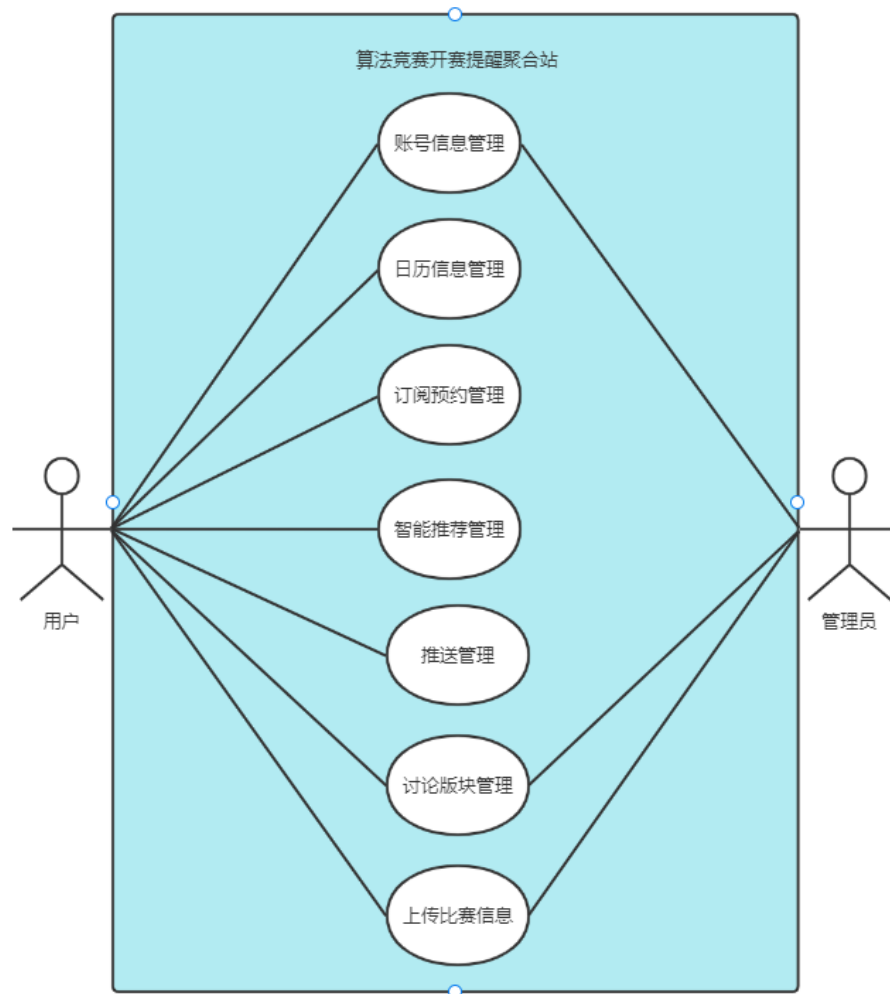
功能模块		输入	处理操作	输出	使用角色
账号信息管理	注册	用户账号、密码和手机号或邮箱信息，若成功注册还需输入验证码	检查密码是否符合安全性需求，账号是否已被注册；若满足条件则发送验证码，验证码正确则完成注册。	向用户邮箱/手机号发送验证码。 注册成功将用户信息存入数据库并提示注册成功，否则提供错误信息。	未注册用户
	登录	账号、密码	判断账号密码是否匹配，账号权限是否有效，均满足条件则用户成功登录。	输出欢迎信息，进入用户首页	注册用户
	登出	点击“登出”	清除当前登录用户的信息	进入登录界面	用户
	忘记密码	提供账号和注册用的手机号/邮箱。若有效还需通过重置密码链接输入两次需更改的密码	针对输入的账号和注册用的手机号/邮箱，检测有效性。 检测输入的新密码是否符合要求。	向注册手机号/邮箱发送重置链接。 将更改成功的新密码存入数据库	用户

	修改密码	输入正确的旧密码，输入两次相同的新密码	检查旧密码是否正确，新密码是否符合要求	提示更改成功，将新密码存入数据库	用户
	用户权限变更	点击“禁用”或“激活”	禁用或激活普通用户的账户使用权限	将用户账户的权限变更存入数据库	管理员
	用户信息变更	编辑用户信息	更新用户信息，并向用户发送提示	将用户信息的变更存入数据库，提示用户信息已变更	管理员
日历信息管理	日历浏览	全部比赛信息，登录用户订阅列表，登录用户预约列表，推荐列表	将各赛事按时间展示在日历时间表中，将用户订阅赛事、预约赛事、推荐赛事、未订阅赛事分别以不同颜色标注	日历界面及赛事日程表	用户
	定时更新	系统时间及用户设置，各网站信息	更新首页日历显示内容	日历界面更新及赛事内容更新	系统
	详情查看	点击日历界面的赛事框	打开一个对话框，展示赛事具体信息，及订阅按钮	赛事详情	用户
	浏览筛选	选择的时间段	显示该时间段内的日历信息	筛选时段内的日历信息	用户
订阅预约管理	订阅赛事	在用户设置里选择赛事，点击“订阅”	将该赛事归入用户已订阅赛事中	提示订阅成功，将该赛事存入用户订阅赛事中，在日历中展示颜色改变	用户
	取消订阅	在用户设置里选择已订阅赛事，点击取消订阅	将该赛事从该用户已订阅赛事中移除	提示成功取消，将该赛事从该用户已订阅赛事中移除，在日历中展示颜色改变	用户
	查看订阅	在用户设置里“查看订阅赛事”	获取用户订阅赛事，展示	输出用户订阅赛事	用户
	预约赛事	在日历中点开相应赛事的详情，点击“预约”	将该赛事归入用户已预约赛事中	提示预约成功，将该赛事存入用户预约赛事中，在日历中展示颜色改变	用户
	取消预约	点开已预约赛事详情，点击“取消预约”	将该赛事从该用户已预约赛事中移除	提示成功取消，将该赛事从该用户已预约赛事中移除，在日历中展示颜色改变	用户

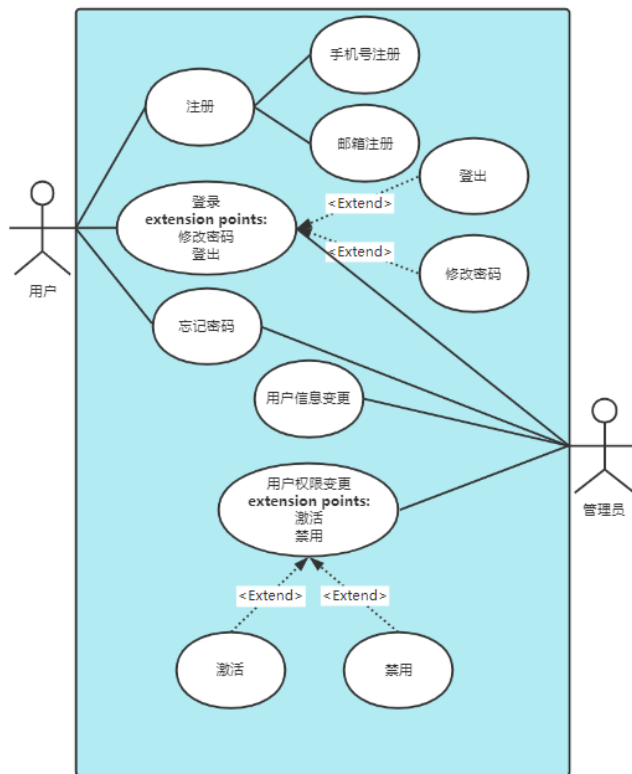
	查看预约	在用户设置中点击“查看预约赛事”	获取用户预约赛事，展示	输出用户预约赛事	用户
智能推荐系统	偏好设置	在用户设置中填写比赛偏好相关信息	保存用户输入信息	将用户偏好设置存入数据库	用户
	智能推荐	用户偏好设置结果	根据设置推荐相应赛事	符合用户偏好的推荐赛事，以特殊颜色展示在日历中	系统
推送管理	推送设置	自定义推送时间(如预约赛事开赛前 3 天等)，推送方式（浏览器、邮箱等）	保存用户的偏好设置	将设置内容存入数据库，提示设置成功	用户
	推送信息	用户的偏好设置，检测到赛事更新、推荐赛事或预约赛事开始	根据用户的推送设置将赛事更消息通过相应渠道在相应时间间隔内推送	向用户通过设置的渠道发送相应推送	系统
讨论版块	选择版块	初始含有“问题反馈”、“学术交流”、“聊天灌水”三个版块，用户点击一个版块	进入该讨论版块，展示该版块各讨论内容	选中的讨论区界面及该讨论区内容	用户，管理员
	进入帖子	点击相应帖子	进入该帖，展示该帖内容及评论	该帖内容及评论内容	用户，管理员
	创建帖子	在版块下点击“创建帖子”，输入主题和内容	创建一个新帖子加入该版块下并展示	将新帖子及内容存入数据库，界面上展示新帖子内容	用户，管理员
	删除帖子	普通用户在自己创建的帖子下点击“删除”；管理员可在任意帖子下点击“删除”	判断删除权限，如果无误则删除该帖	从数据库中删除该帖，在当前界面中移除该帖内容	用户，管理员
	帖子置顶	管理员选择相应帖子，点击“置顶”	将该帖标注为“置顶”，显示在该板块的顶部	将帖子“置顶”状态改动存入数据库，将该帖显示在该板块的顶部	管理员
	评论回复	进入帖子，直接输入回复内容或点击已有的某评论输入回复内容	将评论内容及回复对象保存并输出	将该评论在此帖子下展示	用户

上传 比赛 信息	自行 添加 赛事 信息	点击“添加赛事”，逐项输入赛事信息或在智能输入文本框中按格式输入整个赛事信息	保存用户添加的赛事各项信息或智能识别解析文本框中的各项信息并保存	将用户自行添加的赛事信息存入数据库，并同系统其它赛事一样展示在该登录用户的日历中	用户
	公开 赛事 信息	用户自行添加赛事后点击“公开”	将用户添加的赛事信息上传，交由管理员审核	提示已上传，将该赛事信息放入待审核队列中	用户
	审核	待审核赛事	管理员审核赛事信息的真实性和准确性，审核通过后可公开至全站，不通过则拒绝公开请求	审核通过将赛事信息加入全站赛事中，同系统其它赛事一样展示在全站日历中，不通过则向申请公开的用户发送通知“审核未过”	管理员

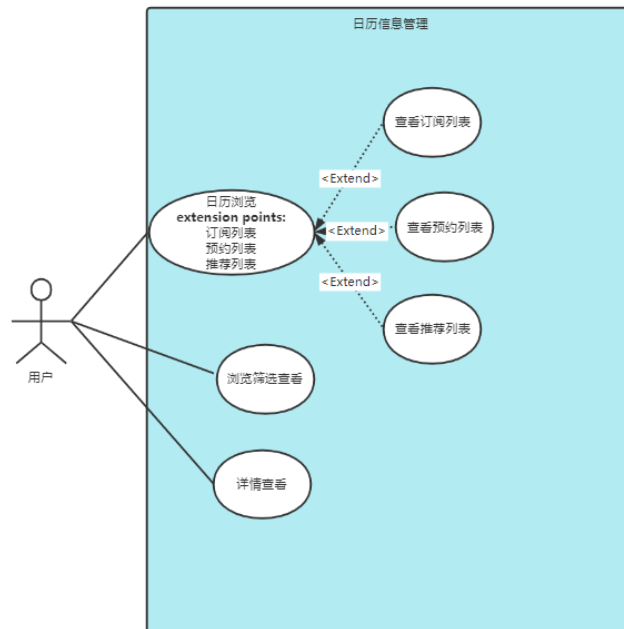
3.2 功能用例图



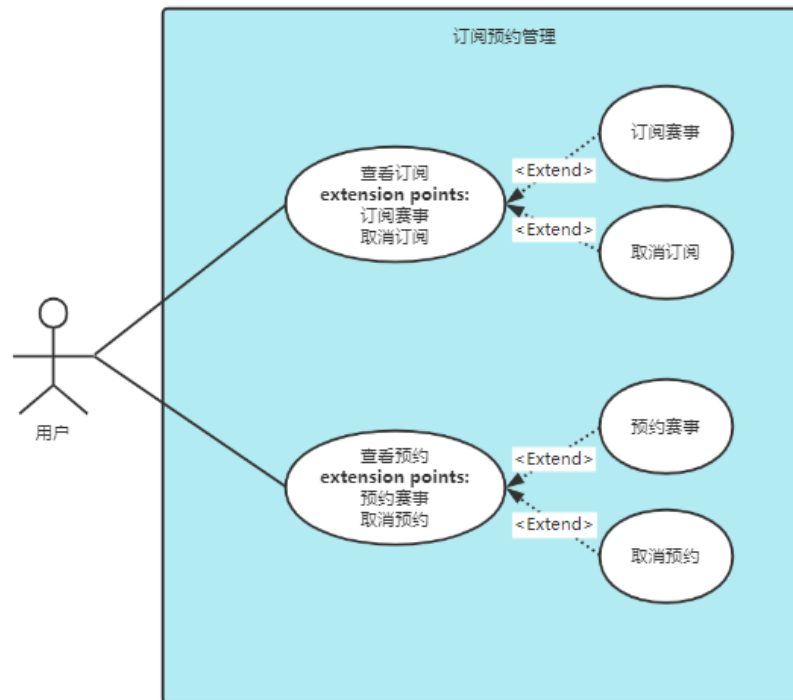
3.2.1 算法竞赛开赛提醒聚合站总用例图



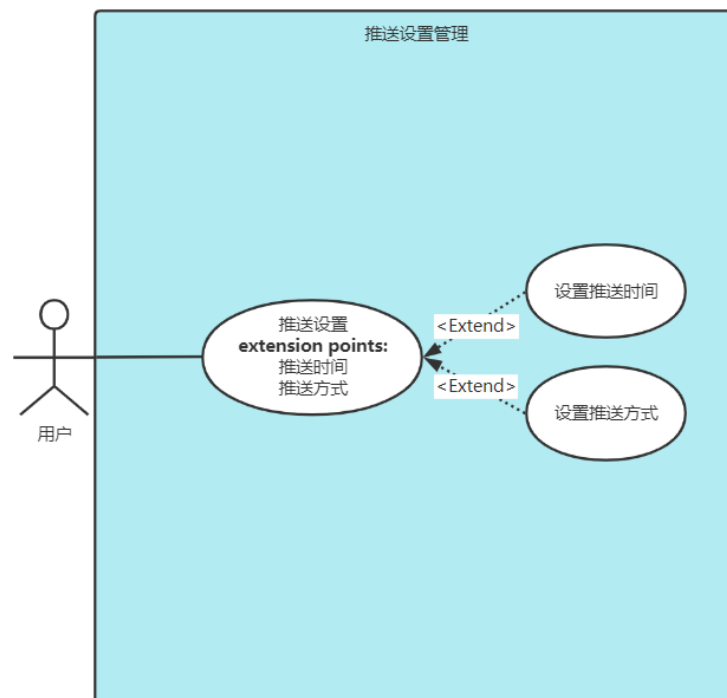
3.2.2 账号信息管理用例图



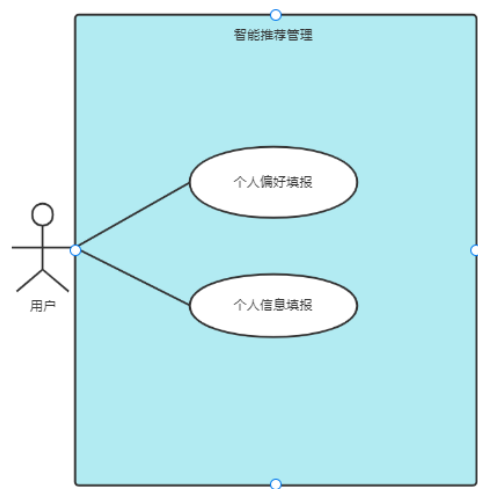
3.2.3 日历浏览用例图



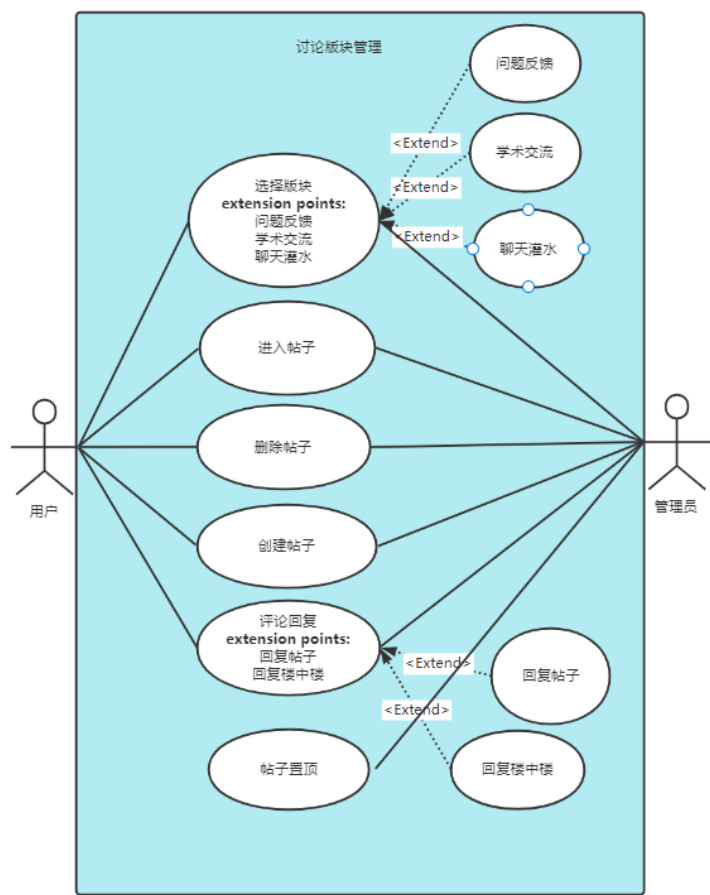
3.2.4 订阅预约管理用例图



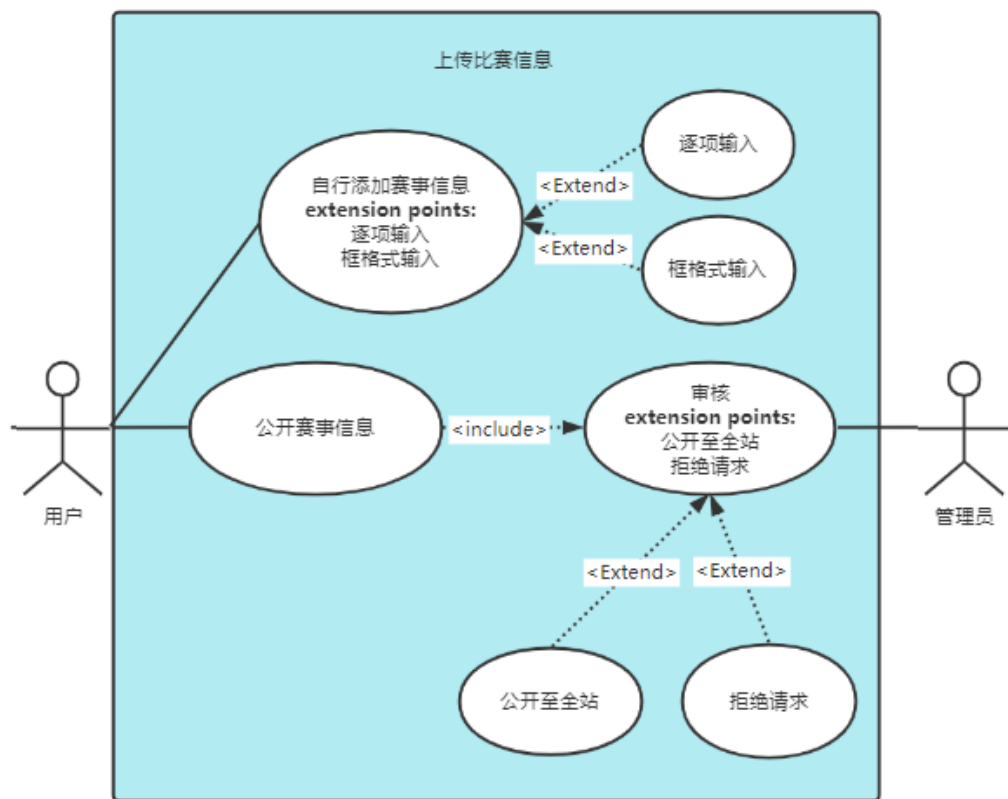
3.2.5 推送预约管理用例图



3.2.6 推荐管理用例图



3.2.7 讨论版块用例图



3.2.8 上传比赛信息用例图

3.3 用例说明

账号信息管理

- 前置条件：网站可以正常访问，且可以建立数据库链接，并且能够防止直接改变网页链接从而进入用户界面
- 事件流：
 - ◆ 用户访问网页时，应用读取 cookie 获取已登录信息，若已登录则直接跳转至用户界面
 - ◆ 未登录状况，应用给出登录界面，包括注册键，忘记密码，用户名密码输入等，可选择加入验证码等式进行进一步验证
 - ◆ 用户选择注册后进入注册页面，需输入包括用户名，密码，验证方式(包括手机号验证和邮箱验证)在内的一系列信息，应用须以较为明显的方式表达对密码等要素的限制条件，如包括什么种类的字符等
 - ◆ 用户完成注册后跳转至登录界面
 - ◆ 在登录界面，用户点击登录时，应用检测页面的用户名，密码等需求信息，防注入
 - ◆ 应用根据给出信息在数据库进行比对，成功则转至登录，失败则给出提示

- ◆ 用户选择忘记密码时转至忘记密码界面，用户需给出账号和联系方式等验证信息
- ◆ 给出验证信息后用户提供新密码并确认，之后直接转移至 用户界面
- ◆ 在用户界面，用户可进入设置界面，选择登出和修改密码
- ◆ 修改密码需用户给出原密码，联系方式等验证信息
- ◆ 对于用户管理员，当其登入后，能够得到非管理员用户的信息变更
- ◆ 用户管理员也可对非管理员用户的登录权限进行修改，并将这部分修改存入数据库中
- 后置条件：网页保持用户进行的诸如登录状态，修改密码等修改并存储

日历浏览

- 前置条件：系统可获取各个网站的事件信息并存储
- 事件流：
 - ◆ 应用定时对系统日历进行更新，在对应的日期将对应的事件填入或消除
 - ◆ 用户登入后查看日历(日历即为主页面)
 - ◆ 在主界面，用户可通过指定筛选条件或输入合法的筛选条件对日历中的比赛进行筛选
 - ◆ 在初次登陆时可弹出一个提示框，用户可以点击选择自己喜好的赛事类型或是难度类型等(可参照牛客网),日后系统根据用户此选择做推荐
 - ◆ 用户也可在主界面对自己之前选择过的喜欢进行修改
- 后置条件：无

用户预约管理

- 前置条件：用户已登录
- 事件流：
 - ◆ 用户进入网站主页
 - ◆ 在主页面中，无筛选条件情况下，所有赛事都会显示，其中用户已预约赛事和用户未预约以颜色方式进行区分
 - ◆ 用户点击日历赛事可查看赛事的详细信息，并且选择对未预约的赛事进行预约，或对已预约的赛事取消预约
- 后置条件：无

用户订阅管理

- 前置条件：用户已登录
- 事件流：
 - ◆ 用户在主页面选择进入订阅界面
 - ◆ 页面显示各个网站，并且辅以提供网站相关介绍，说明用户是否订阅此网站
 - ◆ 用户点击网站栏弹窗提示是否订阅
 - ◆ 用户选择后系统保存
- 后置条件：无

用户推送预约管理

- 前置条件：用户选择开启预约推送
- 事件流：
 - ◆ 用户打开推送设置
 - ◆ 用户对推送时间安排进行设置，包括固定时间段和自定义时间段
 - ◆ 用户对推送方式安排进行设置，从系统可用的几种方法中进行选择
 - ◆ 用户确认刚刚进行的全部设置
 - ◆ 应用检查用户的设置可行性
- 后置条件：系统保存用户修改后的设置

系统推送预约管理

- 前置条件：有用户开启了订阅推送
- 事件流：
 - ◆ 检测和获取能取得的赛事信息并进行存储
 - ◆ 检测所有开启了订阅推送的用户设置
 - ◆ 用户定时获得系统根据赛事信息和用户推送时间设置推送的消息
- 后置条件：无

推荐管理

- 前置条件：用户已登录
- 事件流：
 - ◆ 用户在主界面可看见自己的喜好类型
 - ◆ 用户可在主界面对自己的喜好类型进行修改
 - ◆ 用户可获得经由系统根据用户自己填写的喜好类型智能筛选的赛事信息
- 后置条件：数据库保存用户填报的喜好类型

讨论模块管理

- 前置条件：用户已登录
- 事件流：
 - ◆ 讨论模块分为 1、问题反馈，2、学术交流，3、聊天灌水，三个分类
 - ◆ 用户登录后从主界面选择进入讨论模块，可看见讨论度较高的帖子
 - ◆ 用户也可选择进入何种分类然后细看该类帖子
 - ◆ 用户可在进入分类后选择创建帖子
 - ◆ 在讨论主模块或分类中，用户可点击进入帖子中浏览详情
 - ◆ 用户既可直接在帖子最后一层新发言，也可在帖子中已有的楼层进行回复，产生楼中楼
 - ◆ 用户也可在讨论模块中进入自己曾经发过的帖子，并选择进行删除帖子
 - ◆ 管理员可以查找所有帖子（搜索要素包括时间，帖子名，发帖用户等）进行删除

- ◆ 管理员可选择帖子将其在所属分类中置顶
- 后置条件：无

比赛上传管理

- 前置条件：用户已登录
- 事件流：
 - 用户在主界面点击上传比赛按键，跳转至上传界面
 - 用户在上传界面可选择问卷式逐项输入上传，也可选择直接按照框页面上上传
 - 用户输入完赛事信息后提交上传请求，待管理员审核
 - 管理员审核后若通过则允许上传，不通过则返回失败记录和理由
 - 用户上传比赛时，可选择公开赛事信息
 - 管理员审核通过后将此用户同意公开的赛事信息公开，使得全站用户都可进行订阅浏览
- 后置条件：数据库存储经由管理员审核通过的赛事信息

3.4 外部接口需求

- 用户接口：具有可以与数据库系统交互的接口，可将系统中注册的用户账号存储在数据库中，方便管理员进行操作。
- 通信接口：支持 Chrome、Firefox、Edge 等浏览器，使用 Http/IP 网络传输协议。

3.5 性能需求

- 至少支持 5000 名用户注册
- 至少支持 50 名用户同时在线
- 至少支持 100 名用户同时接收推送
- 服务器至少存在 8GB 的空间用以存储信息
- 用户更改设置应做到及时更新（网站及时提示，QQ 机器人及时回话），避免用户错过相关信息推送
- 数据同步操作过于频繁可以暂时屏蔽，避免服务端与用户交互响应时间过长
- 用户接受推送的时间与其预设时间误差不得超过 1%
- 支持在最新版 Edge/Chrome/Firefox 上正常显示

3.6 质量属性

- 可用性：需要保证服务器在每天 6:00 至次日 4:00 始终能够提供稳定的服务。
- 可扩展性：当需要扩展新功能时，开发人员仅需要完成：
 1. 添加相关程序文件
 2. 增加接口

3. 修改用户使用界面

- 安全性：网站应预防可能的 SQL 注入；非开发人员及维护人员无法查看及修改源程序；需要用户登陆才能提供服务，未登录则仅能查看登录界面及网站主页，不能通过输入路由直接访问用户空间
- 可靠性：考虑到在节假日可能会产生拥挤，要求系统应在 95% 以上的时间无故障执行，发生故障后能快速定位故障，并及时恢复服务
- 互操作性：允许用户在不同的系统环境中使用不同的浏览器访问网站
- 可维护性：网站运行过程中出现的 warning 及 error 应保存在一个维护人员可查看的文件中
- 可重用性：明确模块和类的功能，使类或模块的内容尽可能的简单纯粹，便于后续测试、维护及扩展
- 可测试性：能够对任意一个功能模块单独测试而不需要其他功能模块的支持
- 易用性：网站应附带简单的教程说明

3.7 其他需求

- 后台升级测试、功能扩展不影响正常服务使用
- 提供详细的文档，以便后期系统维护