**需求分析规约**

**修订历史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编写日期 | 说明 | 作者 | 评审时间 | 评审人 |
| 2021/11/27 | 完成引言的编写工作 | 蒋汶霖 | 11.27 | 吴渭 |
| 2021/11/29 | 完成三个子系统的建模工作 | 吴渭/李林洲/何子硕 | 11.30 | 蒋汶霖 |
| 2021/11/30 | 用户子系统迭代完善 | 吴渭 | 12.1 | 蒋汶霖 |
| 2021/11/30 | 动态子系统添加新用例 | 李林洲 | 12.1 | 吴渭 |
| 2021/12/1 | 非功能性需求编写 | 蒋汶霖 | 12.1 | 何子硕 |
| 2021/12/2 | 招聘子系统进行完善 | 何子硕 | 12.3 | 蒋汶霖 |

1. **引言**

**1.1 背景**

随着近几年来国内创业热潮的兴起，许多新中小企业应运而生。招聘雇主数量稳步增长，人才需求持续旺盛。企业雇主倾向于通过一种高效的方式尽快招揽与岗位职责匹配的人才。

本项目名称为职业发展社交平台。是一款专注互联网招聘求职、让职场hr与求职者在线聊天、加快面试的网络招聘平台。采用去中介化的模式，针对互联网行业为企业和求职者搭建起高效沟通、信息对等的招聘平台，提升找工作的效率，缩短企业招聘时长。同时，通过精准的数据算法提高人岗匹配效率。

本项目由同济大学软件学院4名开发者组成团队，以自购2-4台服务器为部署中心对外提供api接口服务，前端页面访问，数据库存储服务等，以互联网为载体通过浏览器进行访问。

用户在登录过程中可以通过第三方方式进行登录，简化注册流程提升用户体验，考虑对接包括微博、微信、QQ在内的主流社交平台。同时后期将会开发为付费用户提供的高级功能属性，因此还可能对接微信支付，支付宝支付，云闪付支付等线上支付手段。

**1.2 参考资料**

参考前端原型：领英 (中国) <https://cn.linkedin.com/>

参考需求分析文档：BOSS直聘产品分析报告http://www.woshipm.com/evaluating/1089776.html

参考需求建模书籍：SoftwareEngineering-A Practitioner's Approach.8thEdition.RogerPressman.2014

参考国家标准：计算机软件需求规格说明规范GB/T 9385-2008

**1.3 假定和约束**

|  |  |
| --- | --- |
| 经费限制 | 3台服务器租赁费用<1000/年 |
| 开发期限 | 10.29-12.30（教学周第八周周五-第十七周周四） |
| 设备条件 | 四台工作电脑 |
| 用户资料准备 | 从数据超市中获取数据，或采用python从现有招聘网站中爬取数据 |
| 交流问题 | 采用飞书团队和qq微信线上交流与线下交流结合 |
| 编程语言 | 后端Java，前端采用vue脚手架和elementUI |
| 运行环境 | 服务端JVM和node.js，客户端使用浏览器访问 |
| 人力资源 | 2名前端人员，2名后端人员 |

**1.4 用户的特点**

企业用户特点：企业雇主多集中在一线城市大型企业，也存在个别小微企业。其希望有一种高效的方式尽快招揽与岗位职责匹配的人才，并能与求职者在线聊天，快速了解求职者是否与岗位匹配。

求职用户特点：产品的目标用户年龄段为20-49岁，主要集中在20-29和30-39这两个年龄段，整体以青年人为主体，求职需求旺盛。

数据库维护人员：负责管理和维护数据库服务器，包括备份系统数据、恢复数据库系统、产生用户信息表，并为信息表授权、监视系统运行状况，及时处理系统错误、保证系统数据安全等。

预期使用频度：日活跃普通用户：10万+；日活跃企业用户1万+。

1. **功能需求**

**2.1 系统范围**

系统开发意图：开发一个求职者和企业高效沟通、用户之间分享求职经验、互通有无的招聘平台。

应用目标：求职者可以通过本平台进行发表动态、浏览招聘信息等基本活动，并与企业取得联系、投递简历；企业可以同本平台发布招聘信息或动态，并与求职者取得联系。

作用范围：希望通过互联网找到心仪工作的求职者、希望通过互联网招聘到所需人才的企业。

**2.2 系统体系结构**

以图+文本结合的方式描述系统的总体架构，如，前端、前后端接口及后端组成的系统体系架构，其中的后端可以进一步抽象其层次结构。

Iteration1:

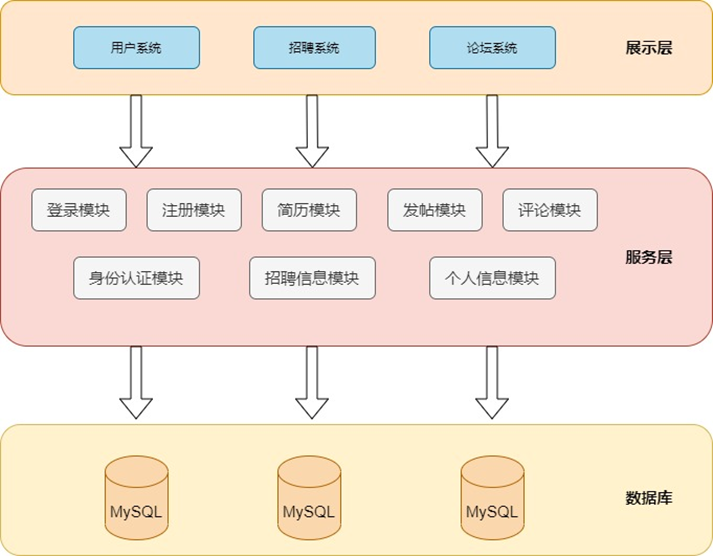
本项⽬拟采⽤springcloud框架开发，基于⾯向服务架构和分布式的思想，将本项⽬纵向切⽚为三⼤

⼦系统⸺⽤⼾⼦系统、招聘⼦系统以及论坛⼦系统。

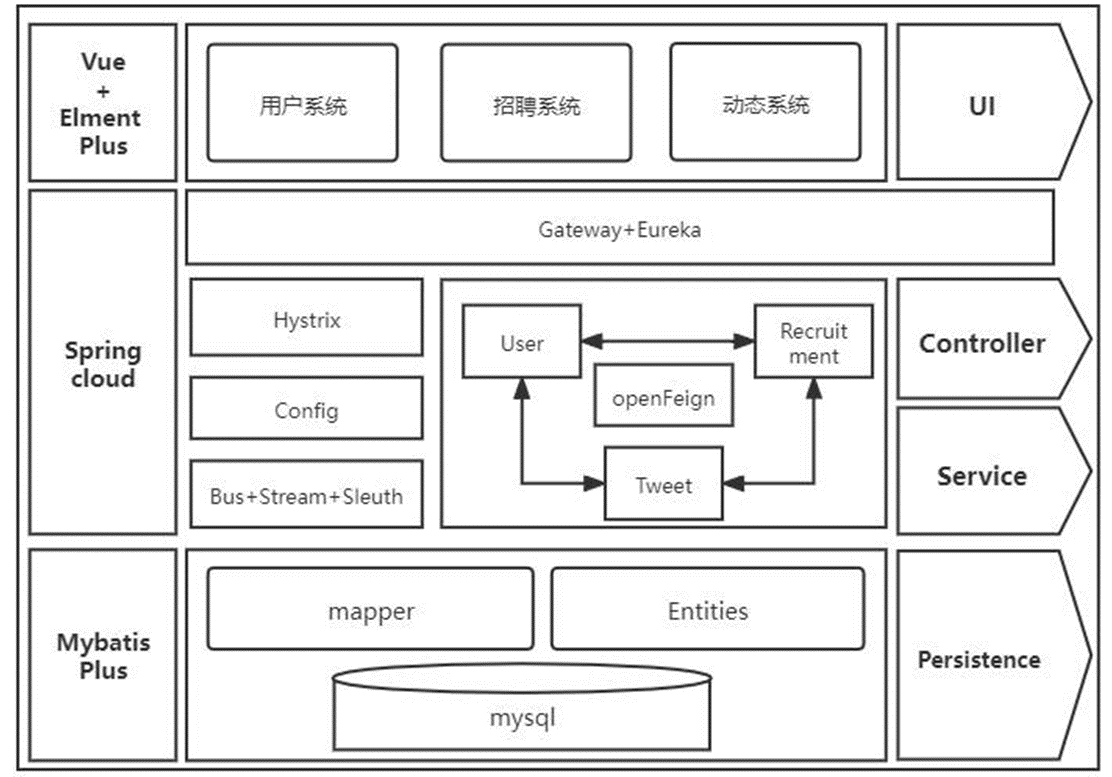
在横向分解上，我们的项⽬在传统的三层架构上做了进⼀步划分，分为：数据持久层(DAO)，业务逻

辑层，对外接⼝层，服务层以及表⽰层。其中对外接⼝层负责在分布式系统中各个主机对外提供服务

调⽤的API，服务层拟采⽤Eureka为服务发现框架，负责服务的注册与治理。



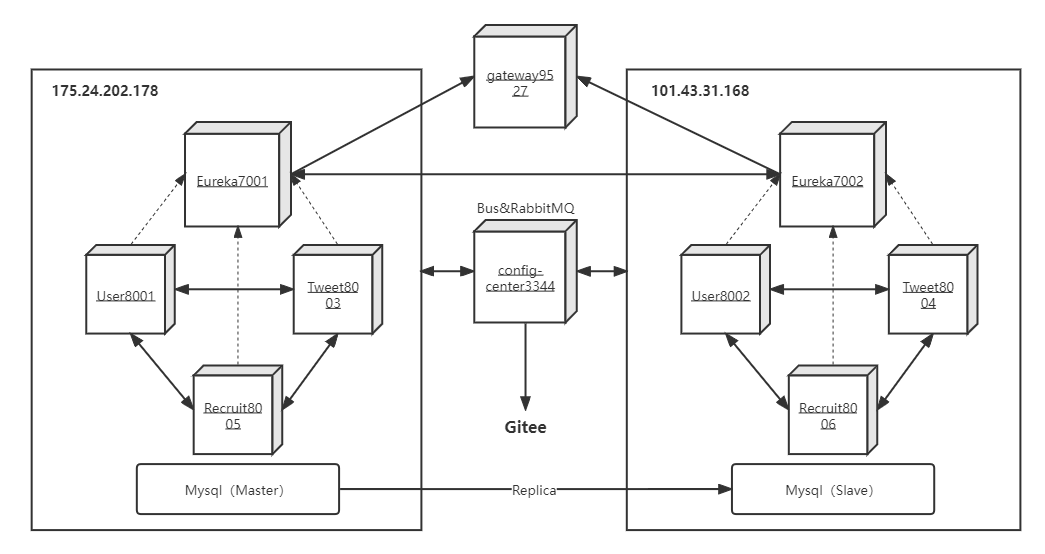
iteration2:



**横向架构**

在横向上，我们的项目架构仍分为传统的三层架构：

* 数据库持久层：Mysql数据库，采用Mybatis-plus作为ORM框架
* 服务层：按照业务逻辑分为三大模块：用户模块（User），动态模块（Tweet）和招聘模块（Recruitment）
* 表示层：采用Vue响应式框架，辅以Element组件库



**纵向架构**

在纵向上基于spring cloud搭建微服务架构：

* Eureka负责服务注册与治理
* 分布式部署，在101.43.31.168与175.24.202.178两台服务器上部署所有模块，通过两个Eureka相互注册，构成多节点高可用的服务集群
* OpenFeign实现服务间相互调用（自带负载均衡）
* Hystrix实现服务降级与服务熔断
* Config分布式配置中心，从Gitee仓库中实时爬取配置信息
* Bus消息总线和RabbitMQ消息队列，实现配置信息的动态刷新
* Gateway网关，对前端请求做token鉴权与日志记录，转发到服务模块的同时启用轮询负载均衡

**数据架构**

在两台服务器都启动了Mysql服务，采用**Mysql Replication**主从复制策略，以实现数据库层面的多节点高可用集群

* **主从复制优点**：

1. **读写分离**：

* 指定一台数据库为主数据库，负责业务逻辑的读写；其余为从数据库，负责读操作。
* 所有的写操作仅在主数据库上运行，再异步复制给所有的从数据库，实现数据同步
* 所有的从数据库均可进行读操作，以分担主数据库的线程压力

1. **双机热备**

* 一旦主数据库崩溃，一台从数据库将立刻取代其位置，以供业务顺利进展

1. **架构扩展**

* 在项目业务量增大，I/O访问频率增加时，通过多库存储，降低磁盘I/O访问的频率，提高单机I/O性能。
* **主从复制缺点**：
* 难以解决分布式主键自增的一致性问题
* 异步式的数据同步，在高压线程下容易导致主从数据库的不一致性
* **主从复制原理**：

MySQL主从复制涉及到三个线程，一个运行在主节点（log dump thread），其余两个(I/O thread, SQL thread)运行在从节点：

* 主节点 **binary log dump** 线程

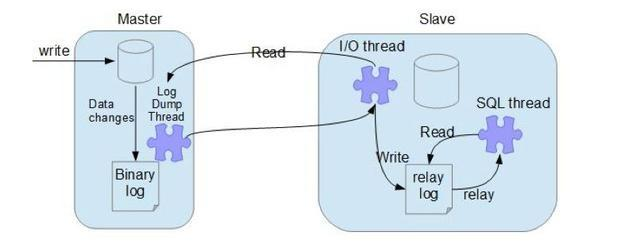
当从节点连接主节点时，主节点会创建一个log dump 线程，用于发送bin-log的内容。在读取bin-log中的操作时，此线程会对主节点上的bin-log加锁，当读取完成，甚至在发动给从节点之前，锁会被释放。

* 从节点 **I/O** 线程

当从节点上执行start slave命令之后，从节点会创建一个I/O线程用来连接主节点，请求主库中更新的bin-log。I/O线程接收到主节点binlog dump 进程发来的更新之后，保存在本地relay-log中。

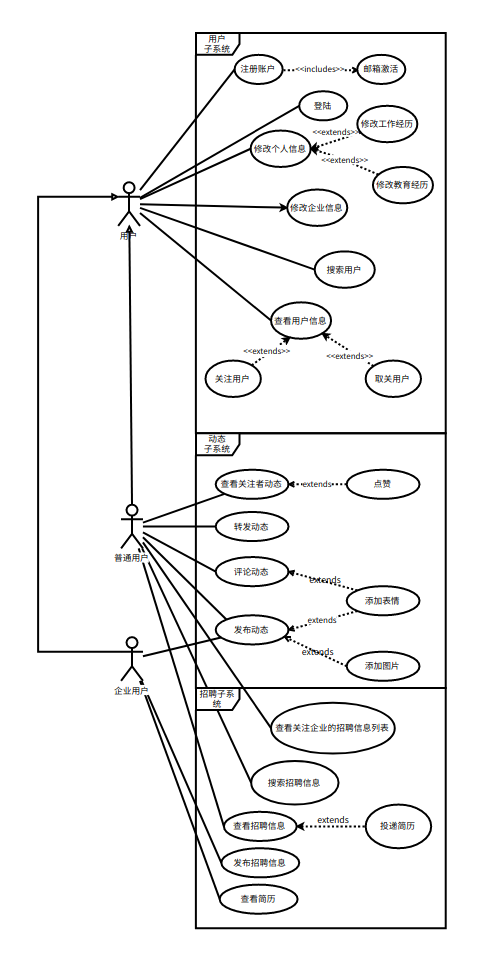
* 从节点 **SQL** 线程

SQL线程负责读取relay log中的内容，解析成具体的操作并执行，最终保证主从数据的一致性。



**2.3 系统总体流程**

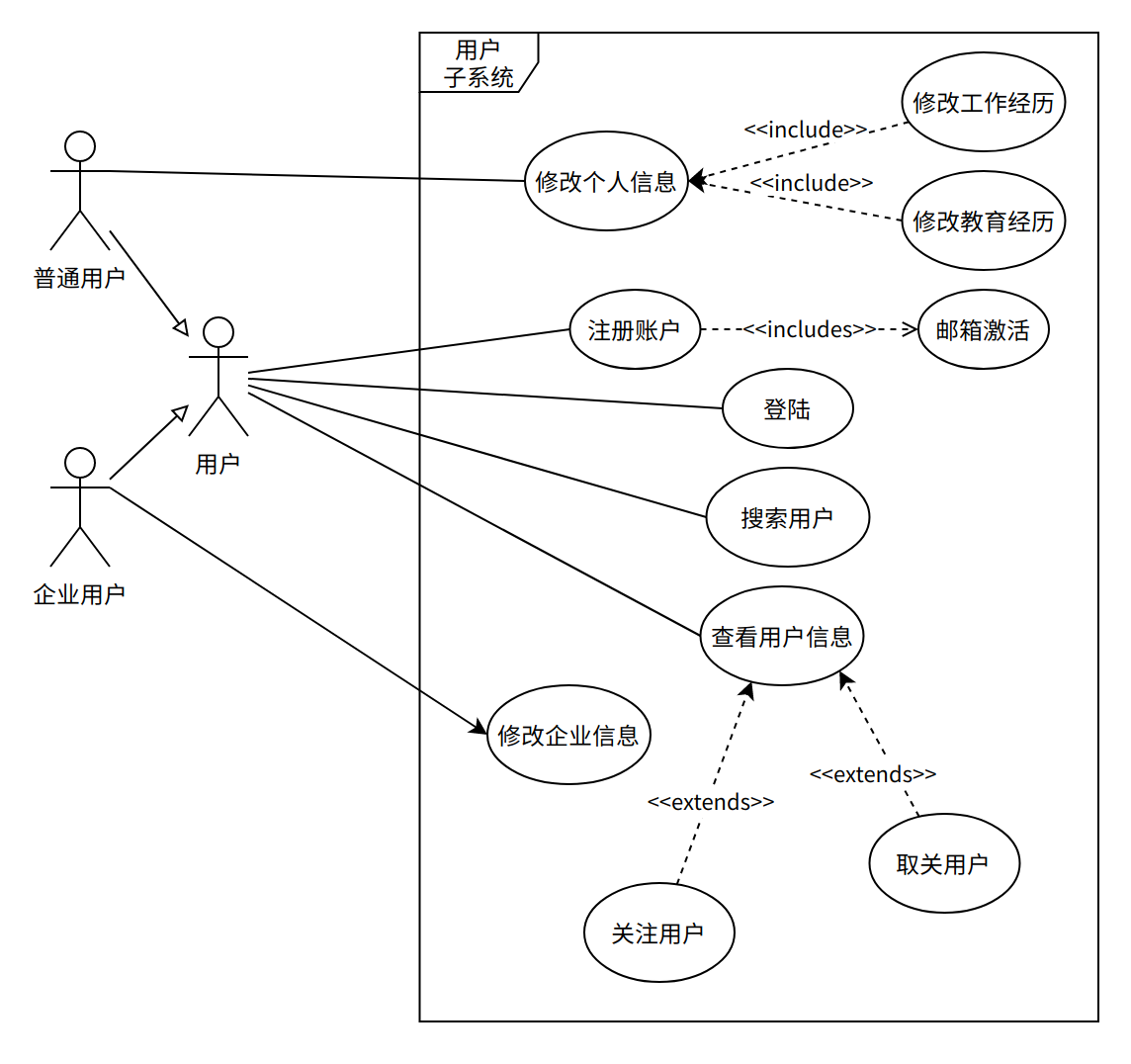
系统总体上分为三个子系统，用户进入系统后进入个人中心可以与用户子系统进行交互，修改和维护和自己相关的信息。用户可以浏览其他用户发布的动态、也可以自己发布新的动态。招聘子系统是项目的核心流程，用户可以浏览企业发布的招聘岗位信息，并向企业的岗位投递简历，企业可以查看所有用户投递过来的简历。



**2.4 需求分析建模**

**2.4.1 用户子系统**

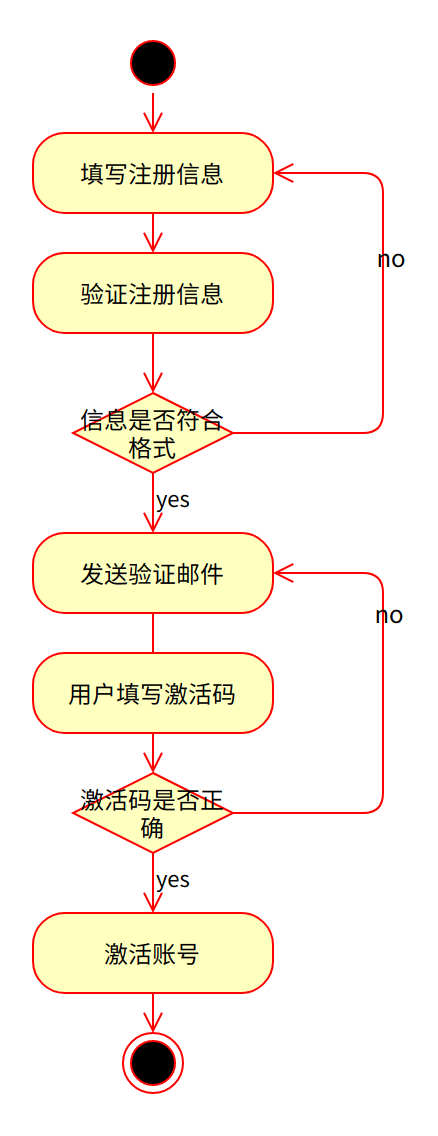
用户子系统的使用者包括普通用户和企业用户（他们共同继承于同一基类）。两者都拥有相同的注册登录功能，但就个人信息的编辑方面略有差异；两者都可以搜索并与其他用户进行交互



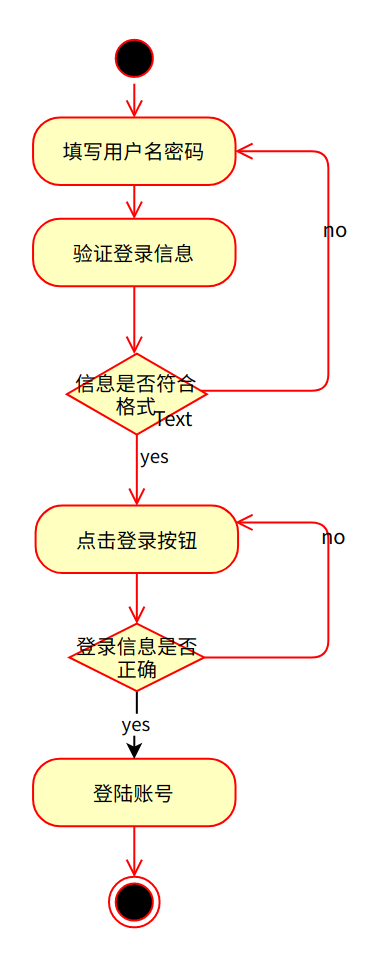
**功能建模**

**注册账户**

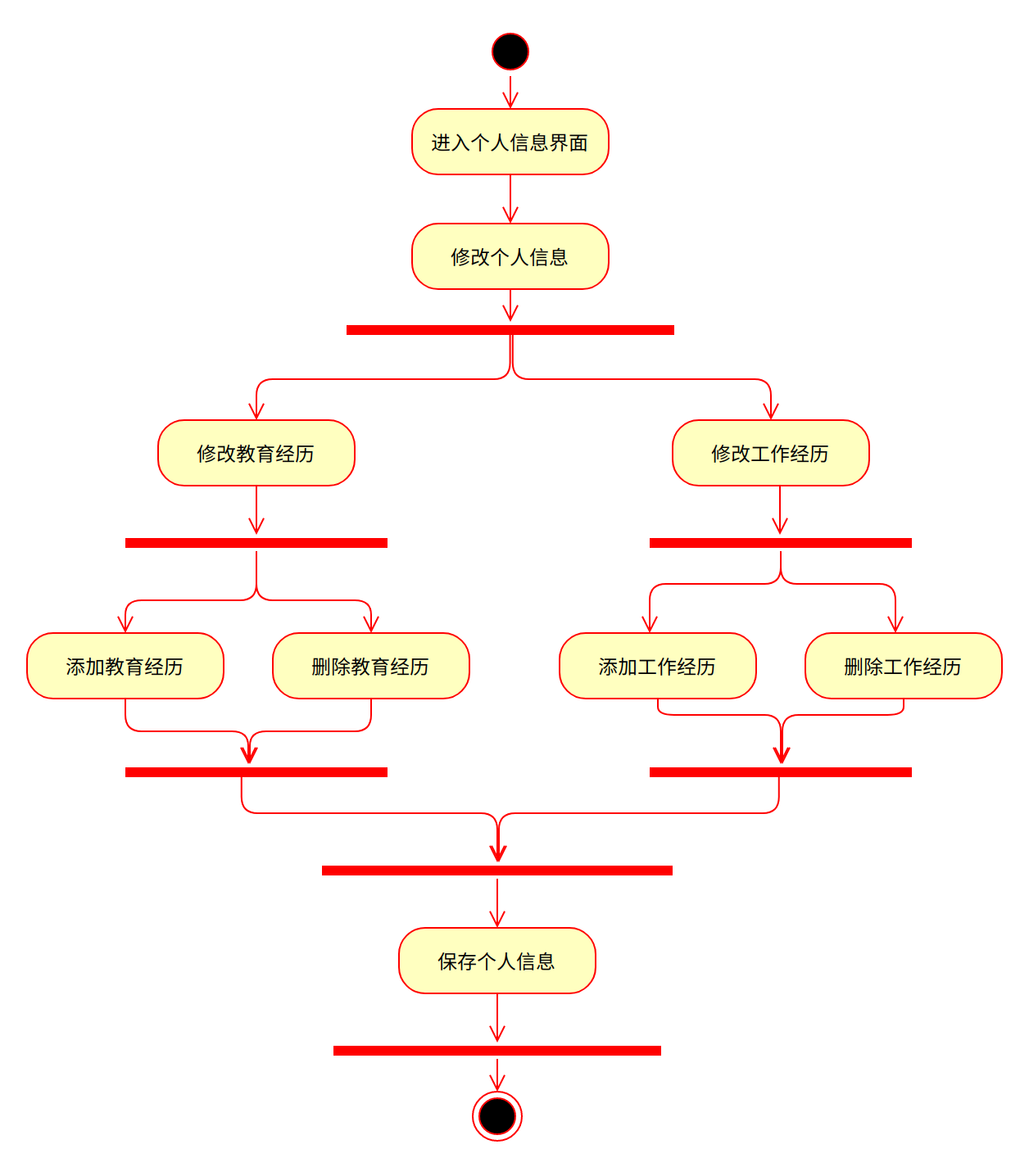
该活动图描述了用户注册账户的整体流程，从开始填写注册信息到成功激活账户结束活动，期间涉及了用户User和邮箱Email实体类，注册表单接口边界类和用户模块控制类



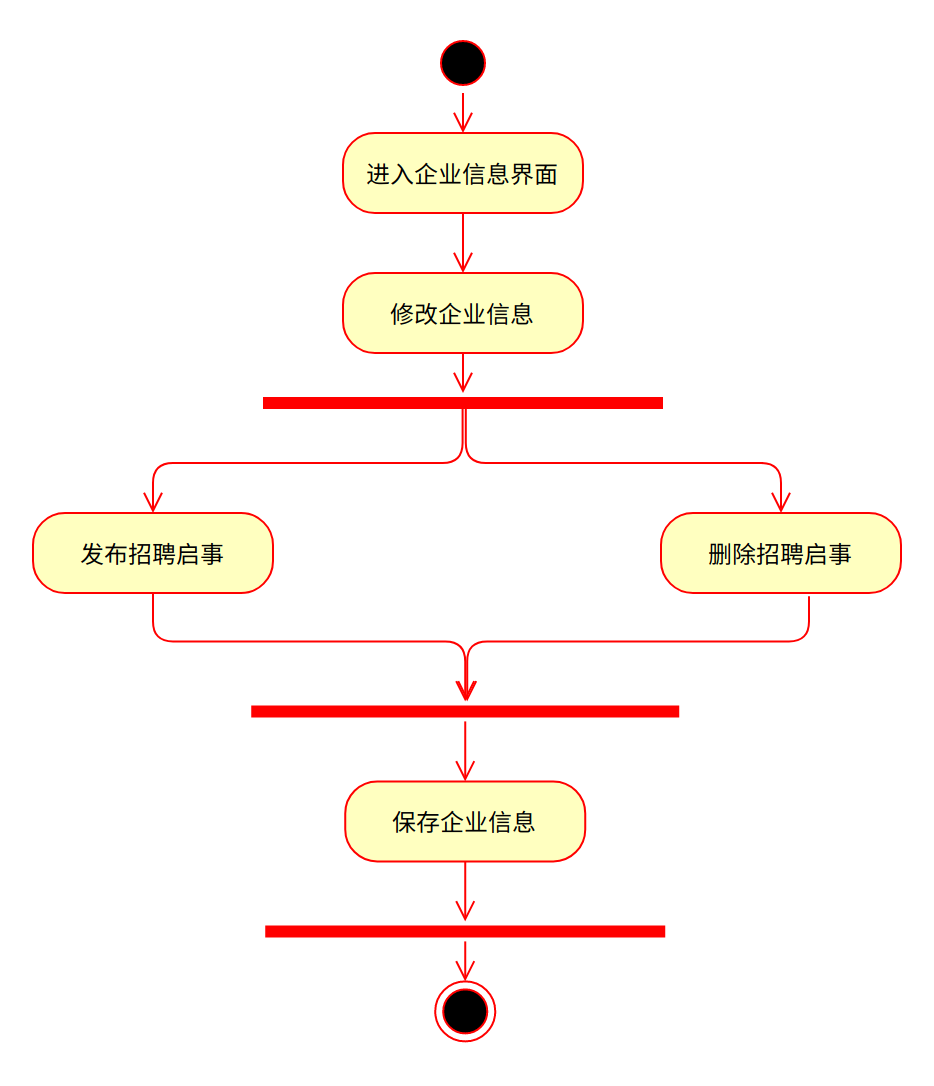
**登录账户**



**修改个人信息**

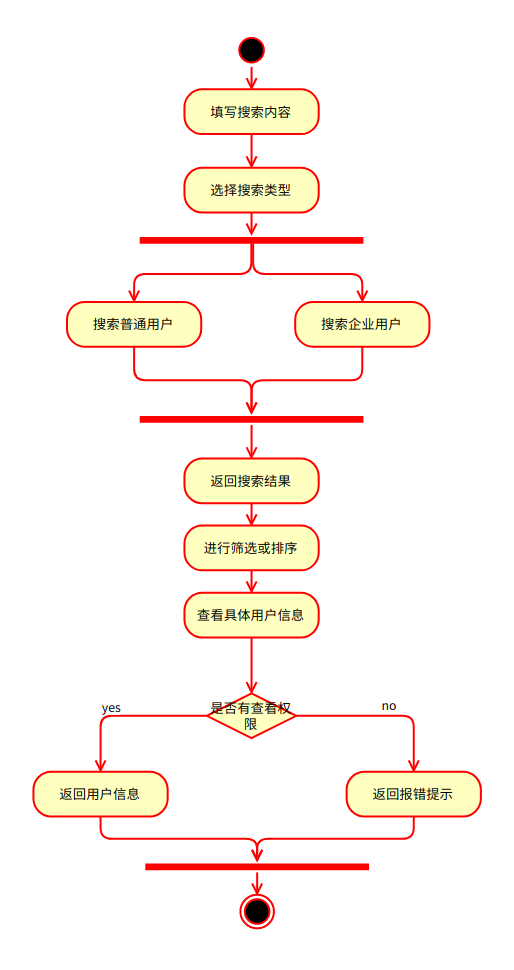


**修改企业信息**

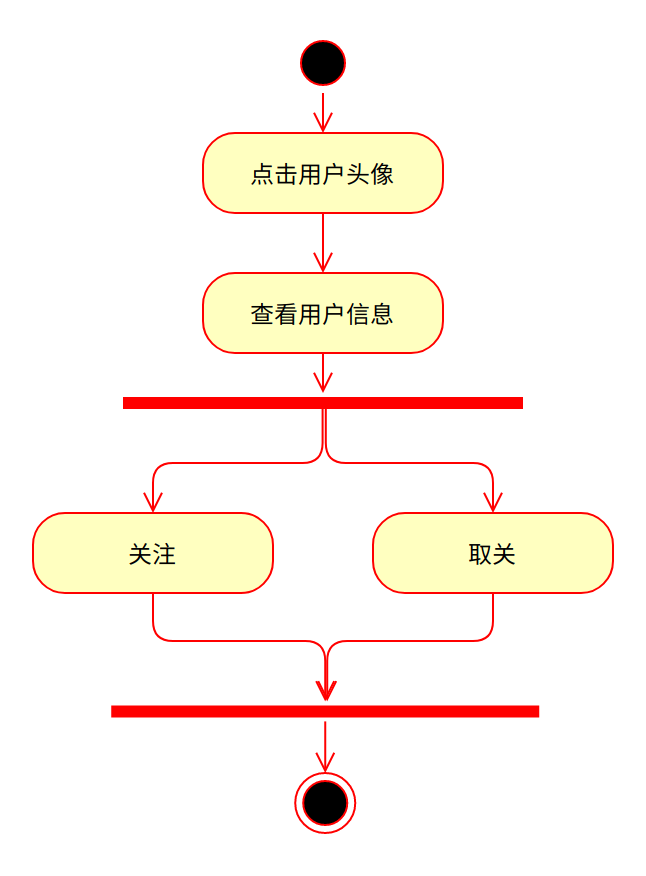


**搜索用户**

该活动图描述了用户搜索的整体流程，用户可以选择搜索的用户类型（普通用户或企业用户）、筛选条件及排序方式，在返回搜索列表，用户还可以点击用户头像，进入“查看用户信息”的扩展事件；此活动涉及用户信息实体类和用户交互的边界类

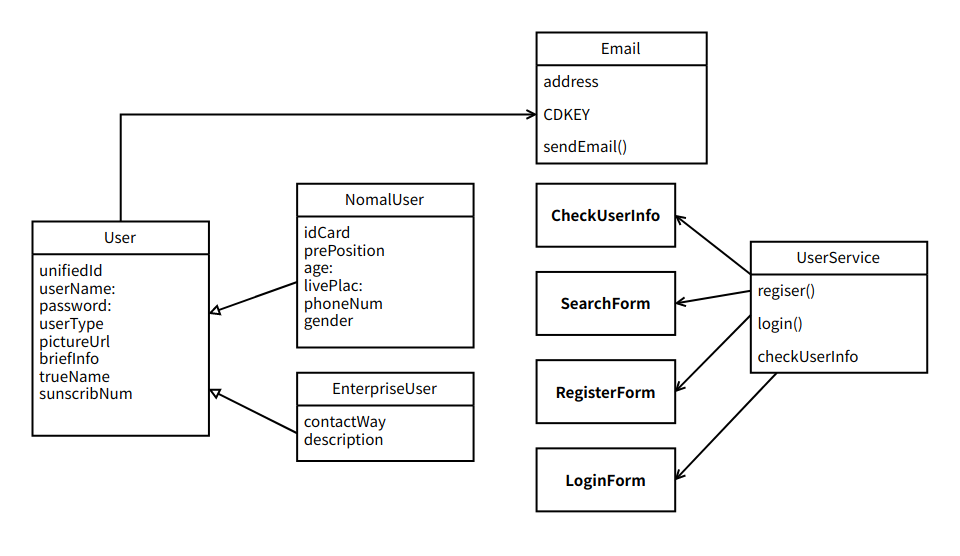


**查看用户信息**

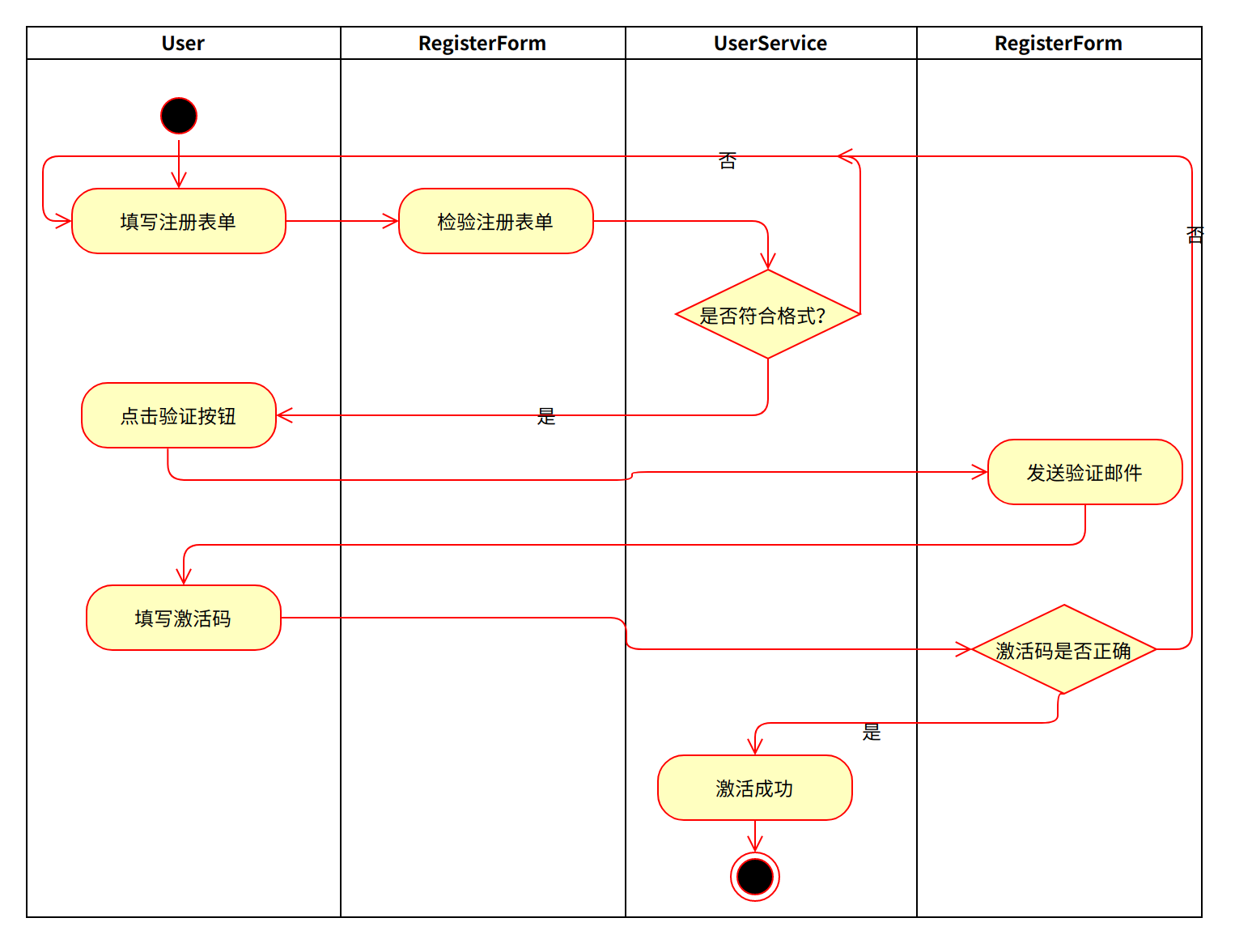


**数据建模**

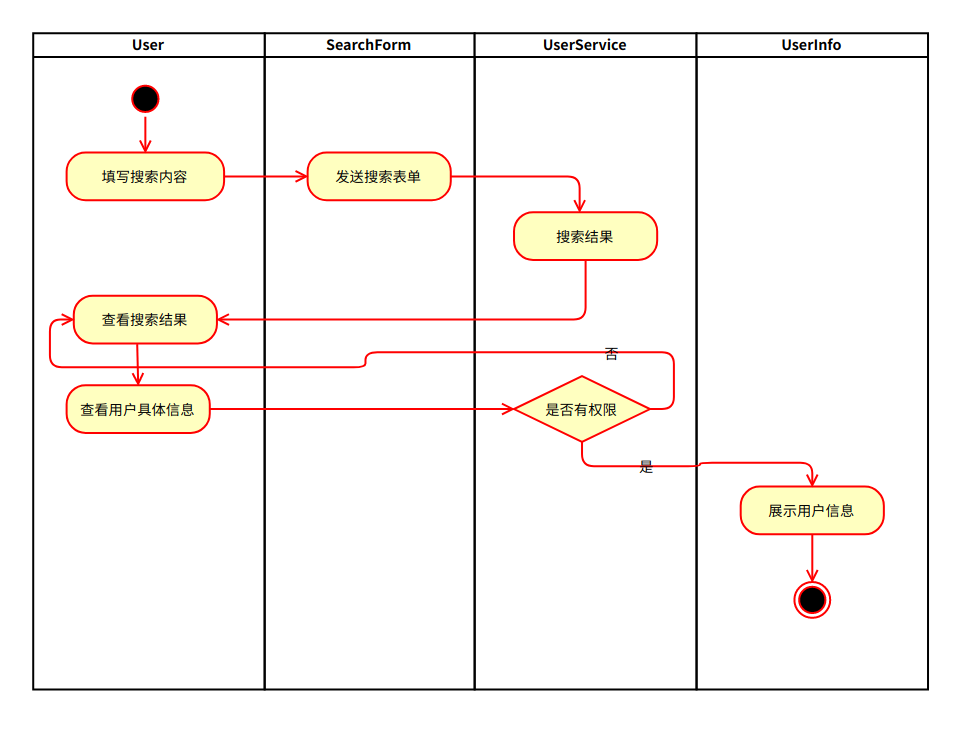
在该类图中共有三种分析类：实体类、边界类和控制类。其中实体类存储具体实例的信息，包括邮箱，基础用户，普通用户和企业用户（由基础用户泛化而来）；边界类负责实体的交互并承担逻辑功能的执行任务，包括注册、登录的表单以及负责实现搜索功能的接口；控制类负责用户模块的统筹协调，编写业务逻辑并为其余分析类提供服务。



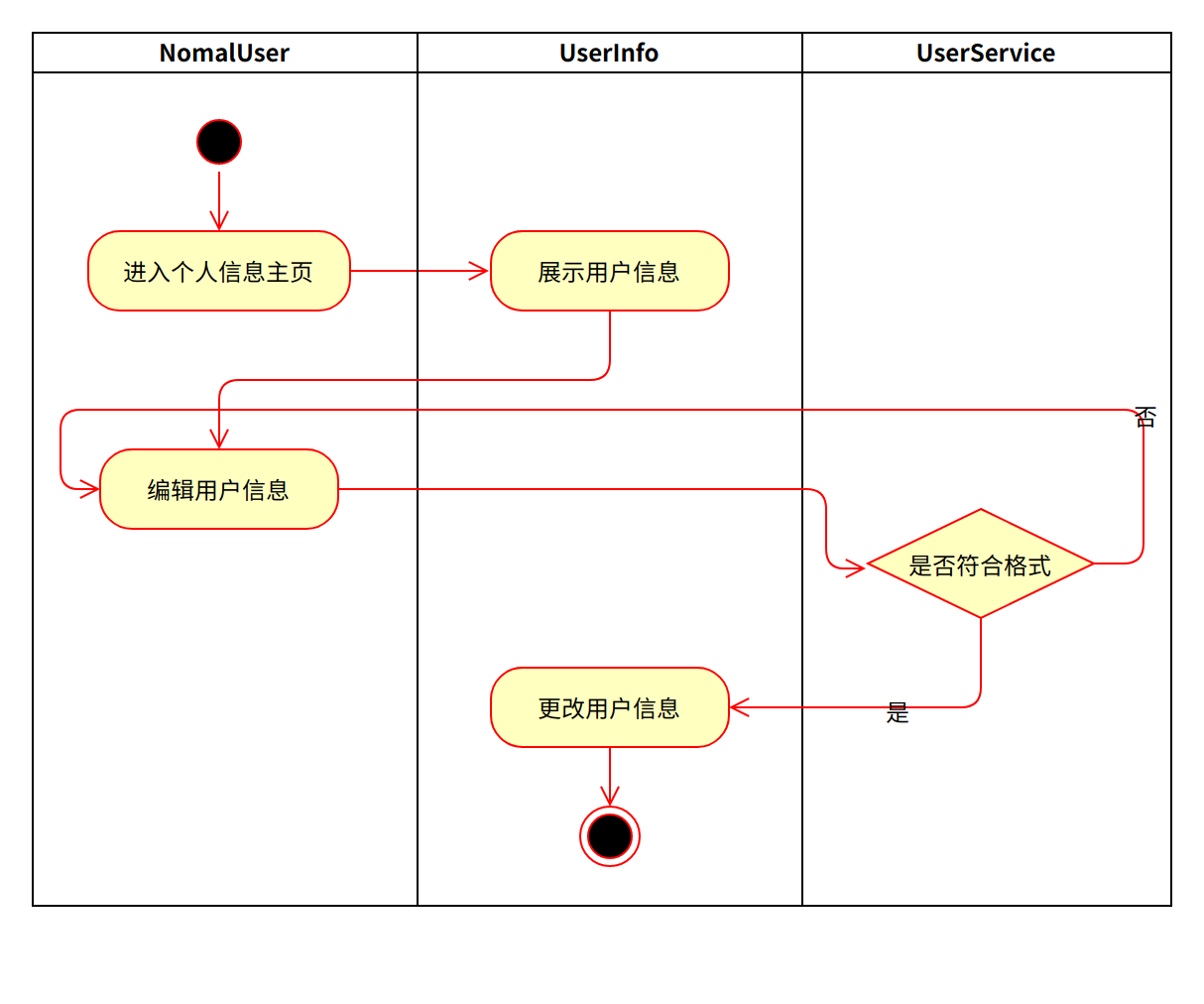
**注册账户**



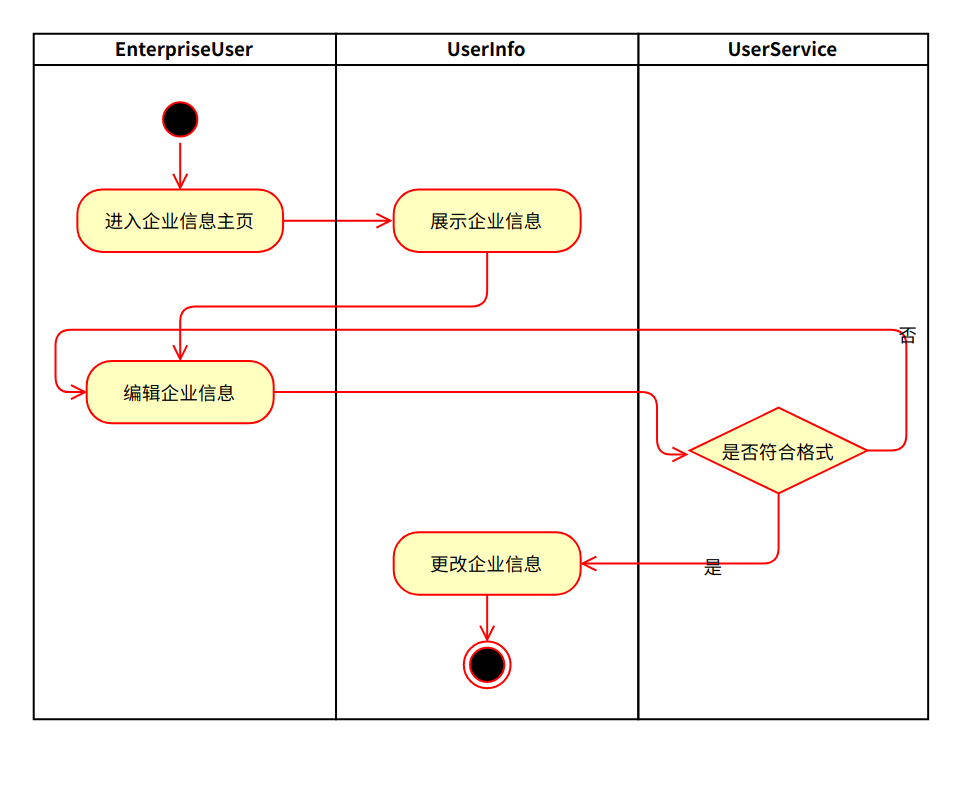
**搜索用户**



**修改个人信息**



**修改企业信息**

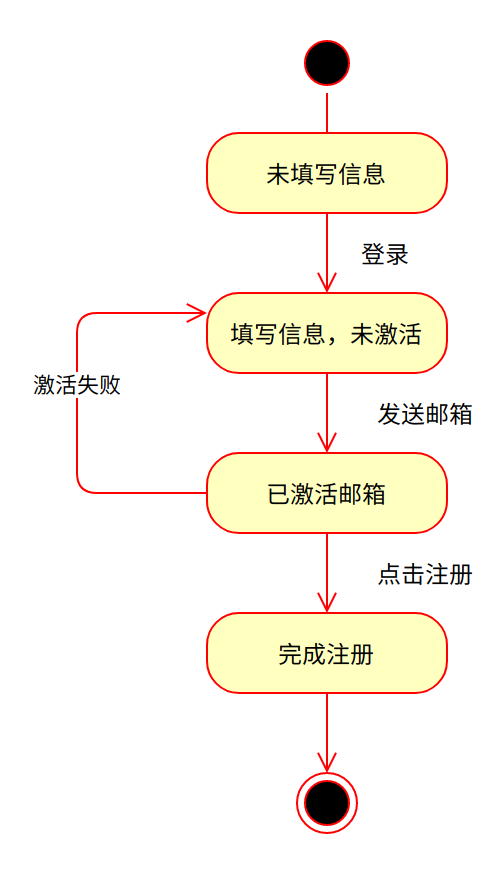


**行为建模**

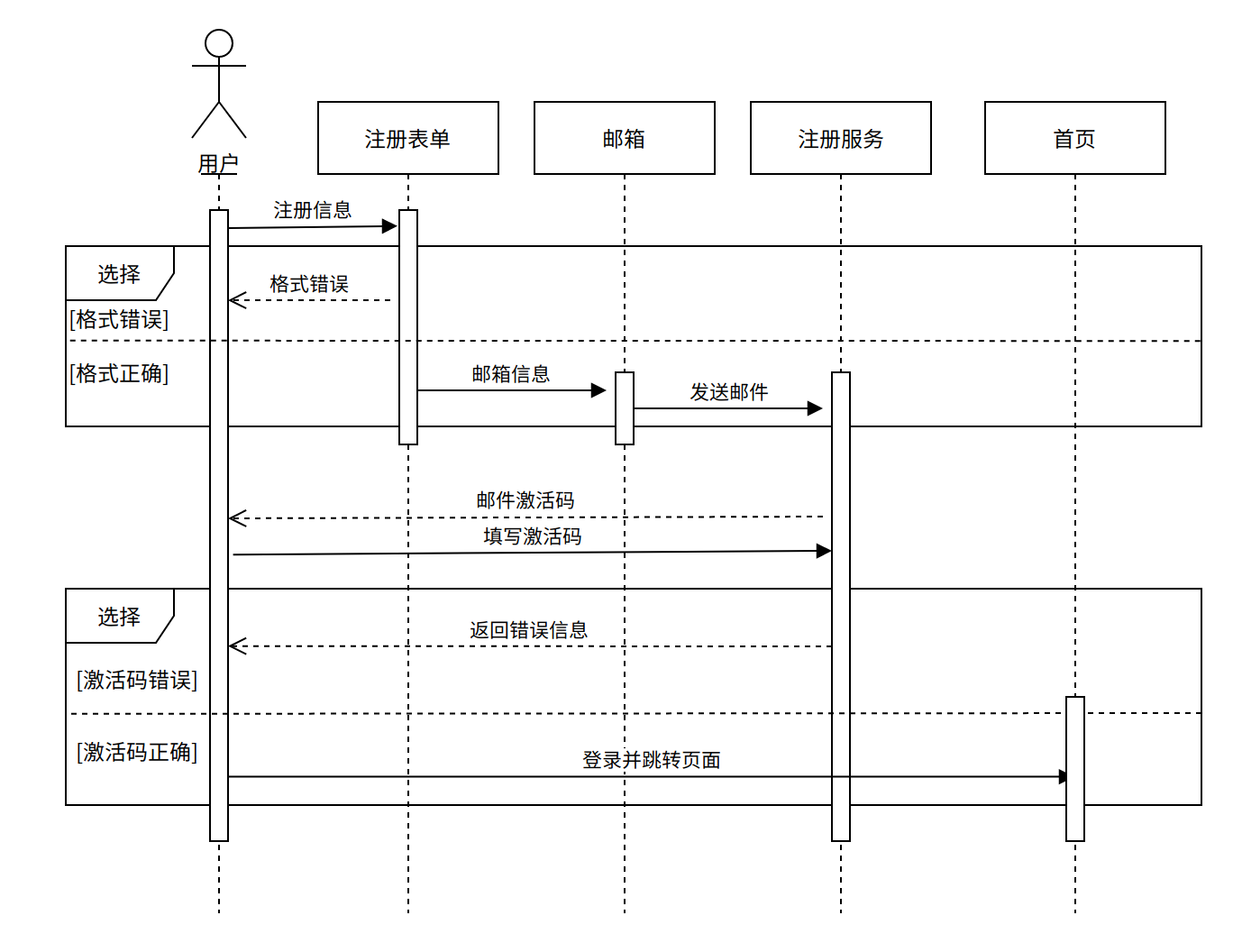
**注册账户**

**用户类（User）**

该状态图描述了在“注册”用例中，用户类的状态变化流程

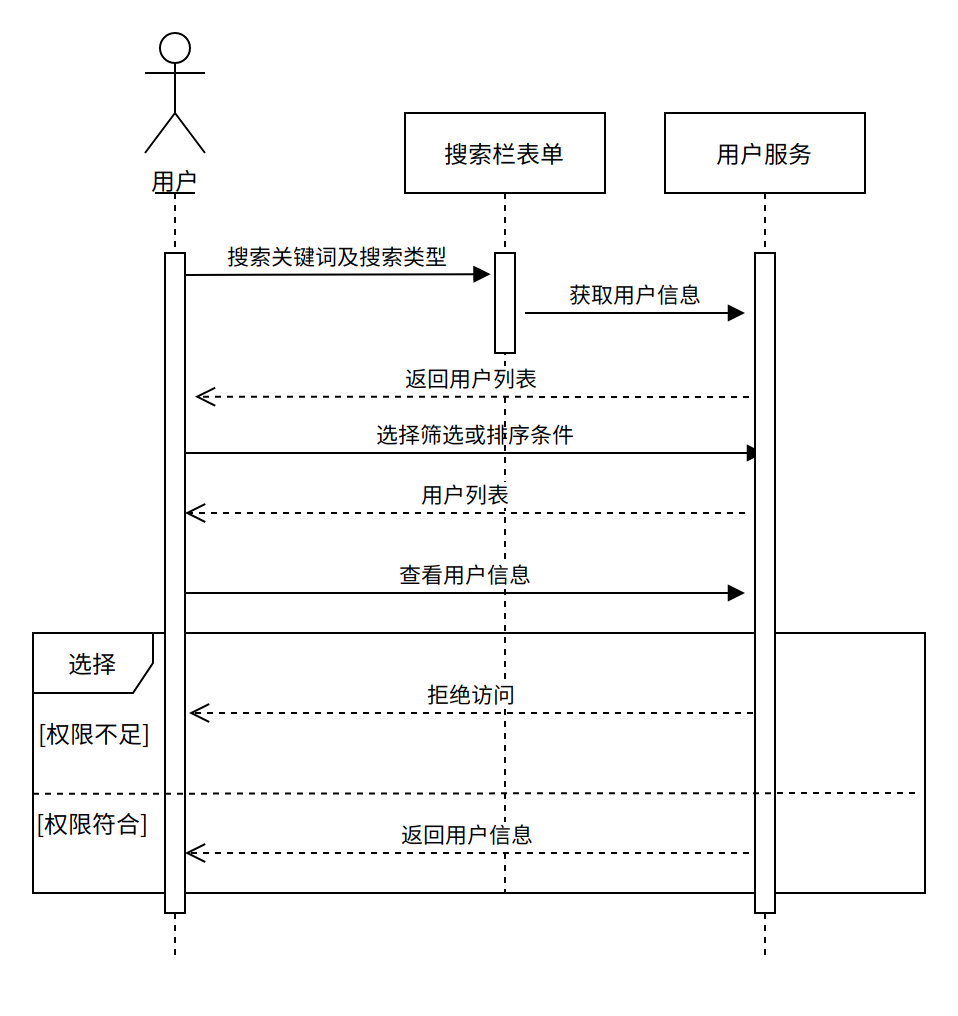


该时序图涉及用例为“注册账户”，涉及对象包含两个实体类（用户、邮箱），一个边界类（注册表单）、一个控制类（注册服务）和一个模块外部系统（首页）。该图规定了“注册账户”这一用例中各个活动的执行顺序和各对象间的信息交互。

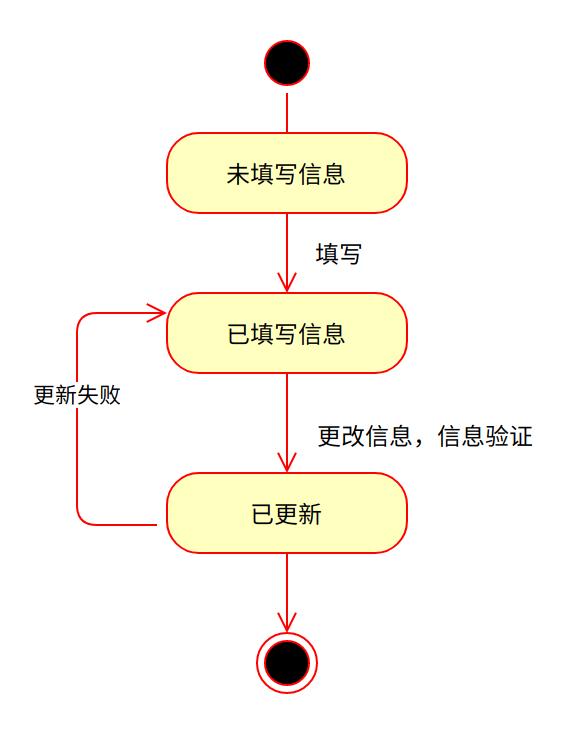


**搜索用户**

该时序图描述了“搜索用户”这一用例中的活动时间顺序及对象信息交互，涉及到了用户实体类，搜索栏表单边界类和用户服务控制类。

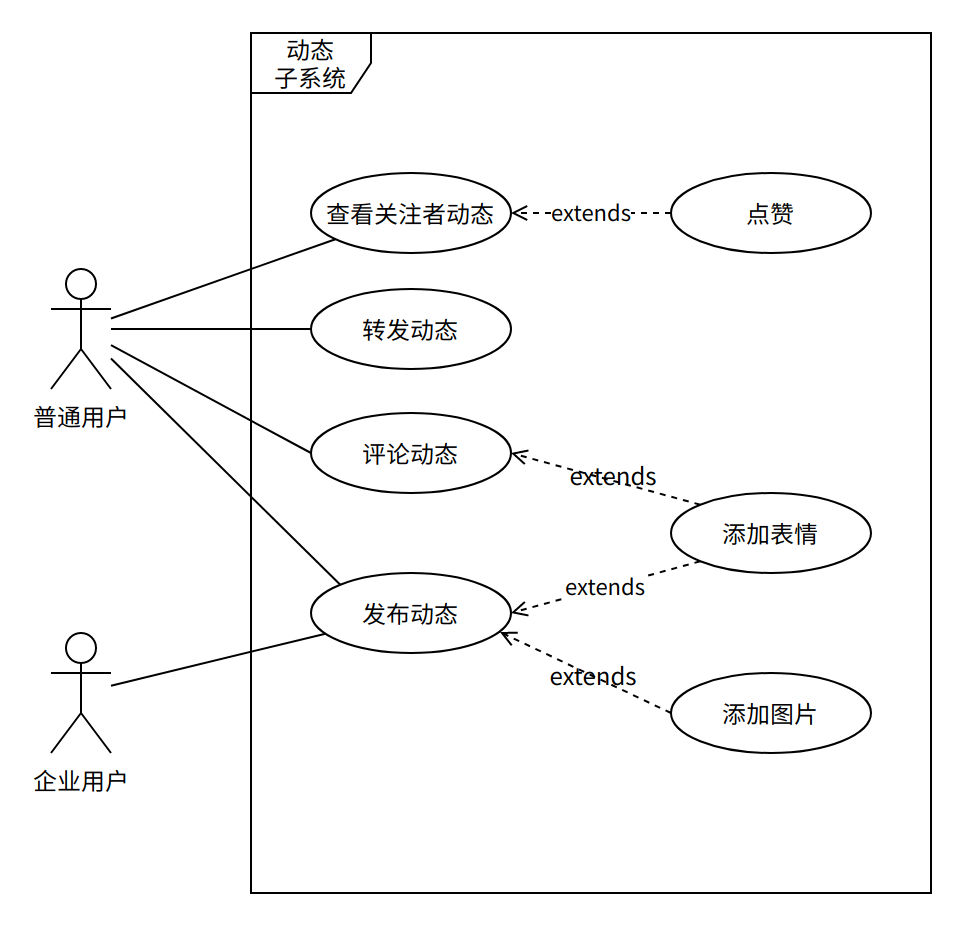


**UserInfo（用户信息类）**



**2.4.2 动态子系统**

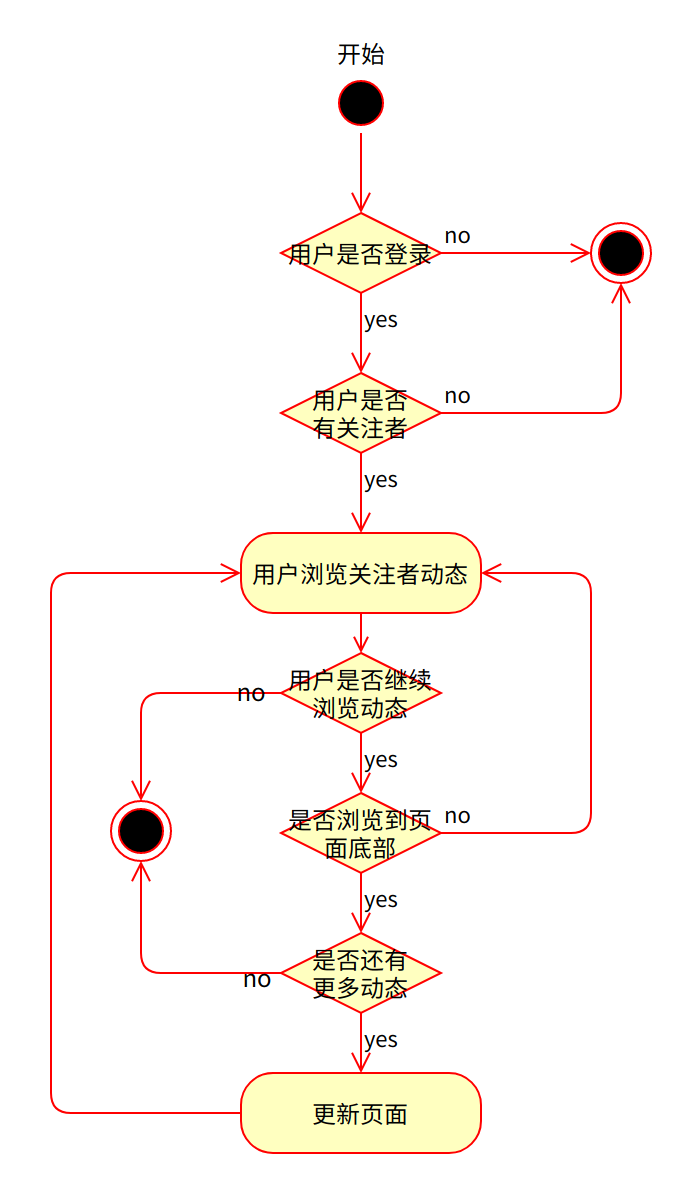
动态子系统主要使用者为普通用户与企业用户。两种用户均可发布动态，在发布时可选择添加图片或者添加表情。普通用户还可以评论动态，转发动态以及查看关注者动态。

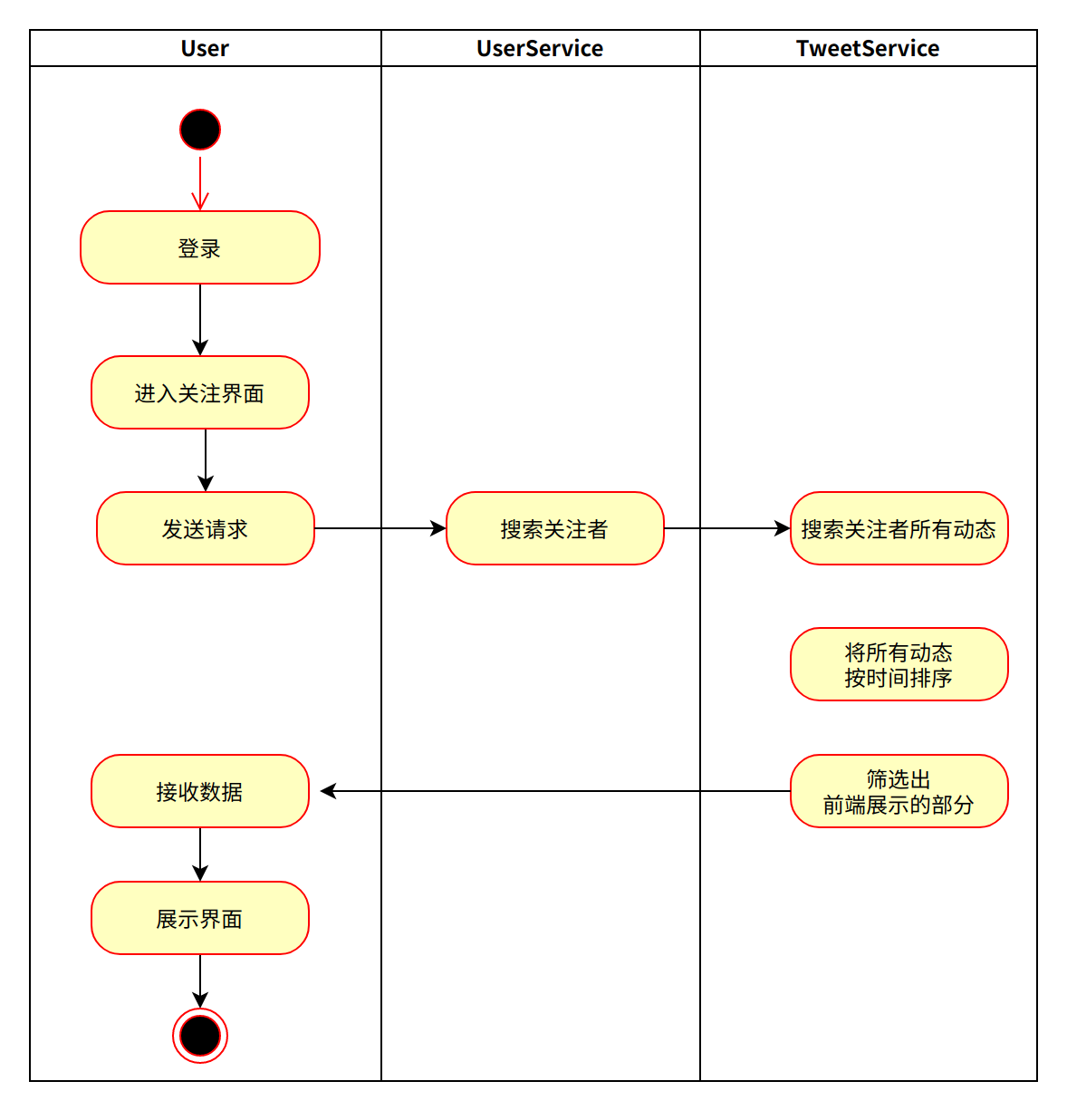


**功能建模**

**查看关注者动态**

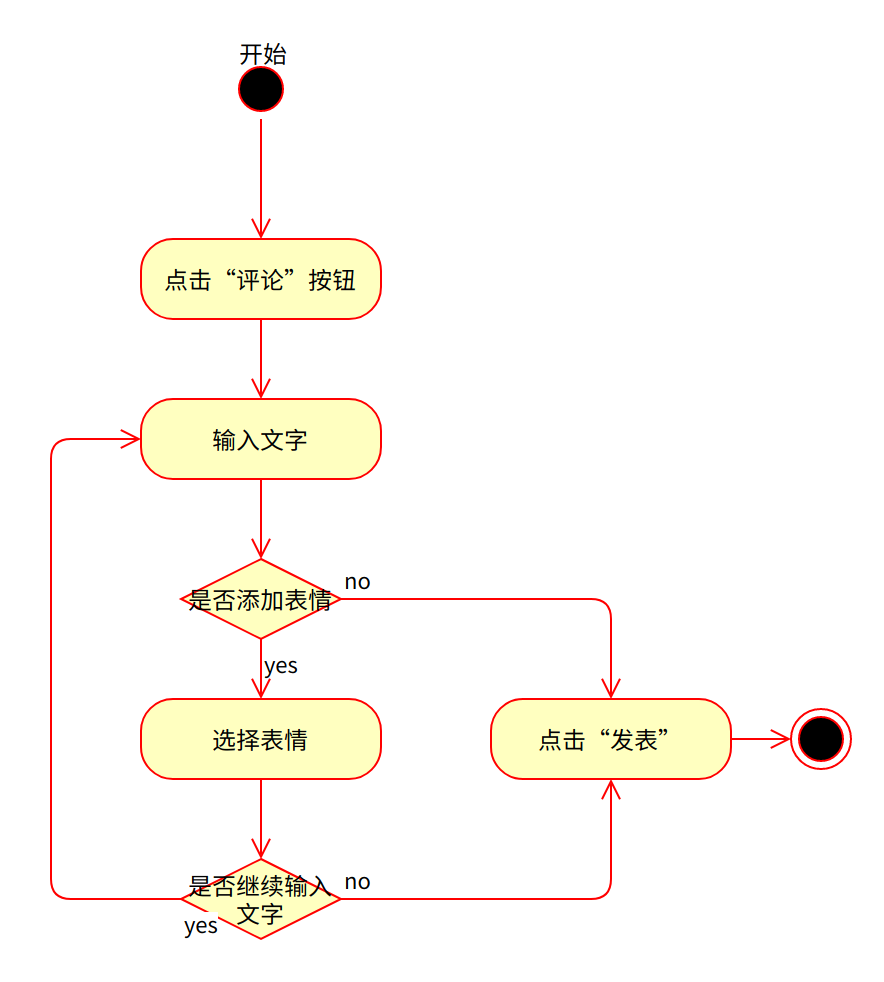
用户登陆后，如果此用户有关注者，系统就会在动态浏览页面推送关注者发布的动态。用户可以浏览这些动态，当浏览到页面底部时，如果还有更多动态，系统就会加载一定数量的动态，用户可以继续浏览。

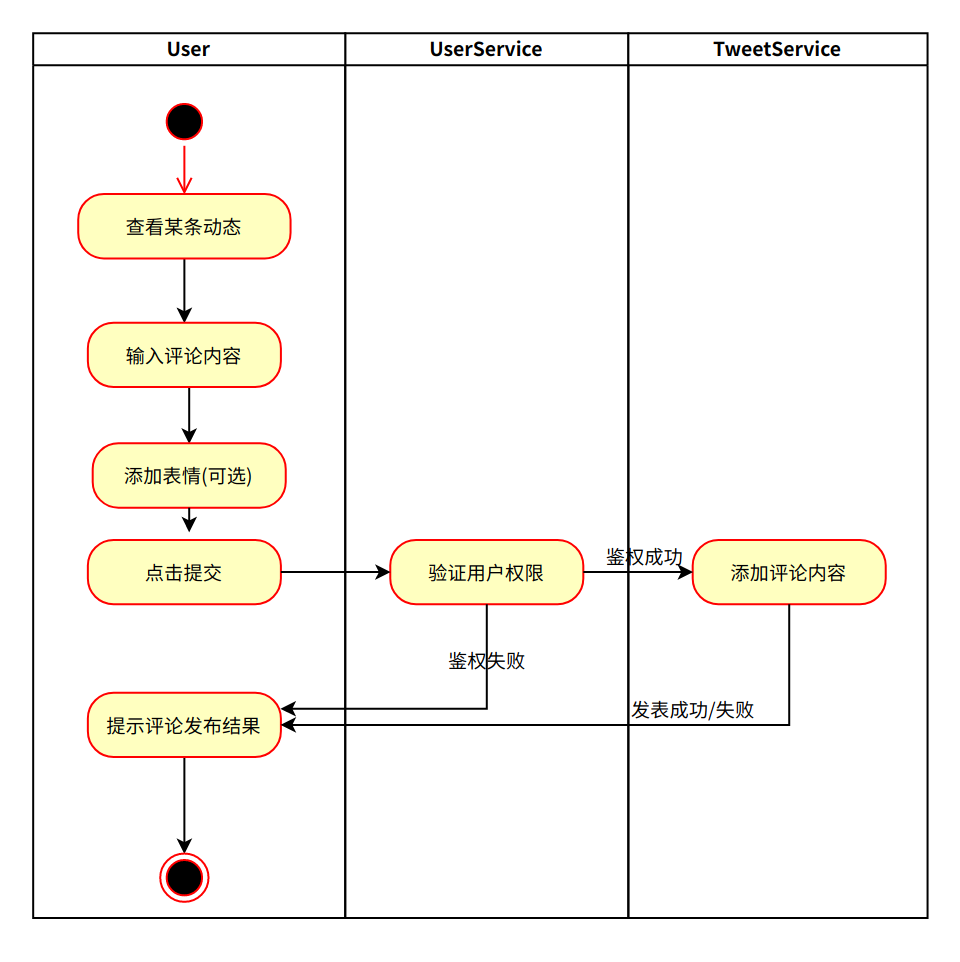




**发表评论**

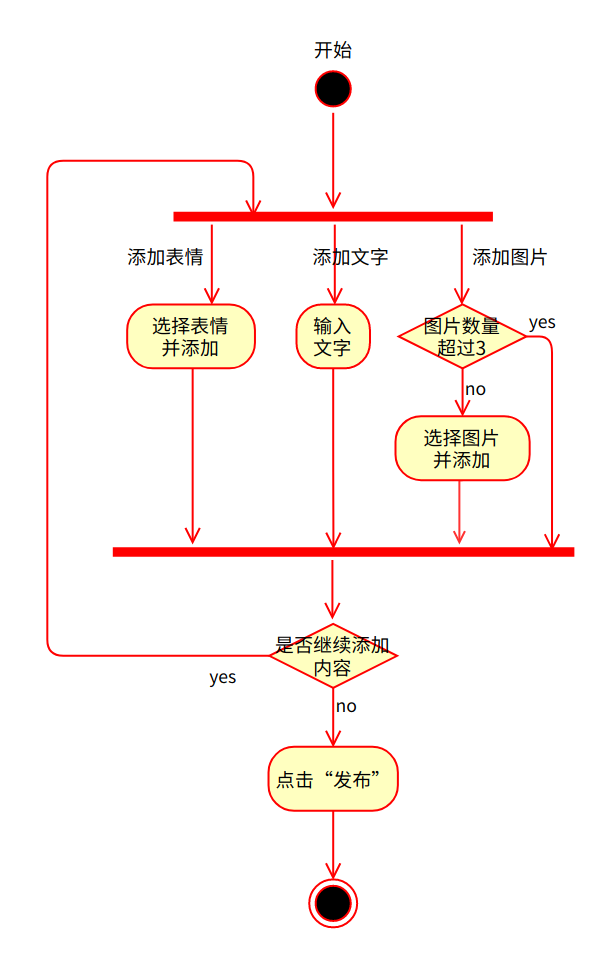
用户浏览到某一条动态时，可以点击“评论”按钮展开评论。用户可以在输入框输入文字，同时也可以点击输入框下方按钮添加表情。评论输入完成后可以点击发表。

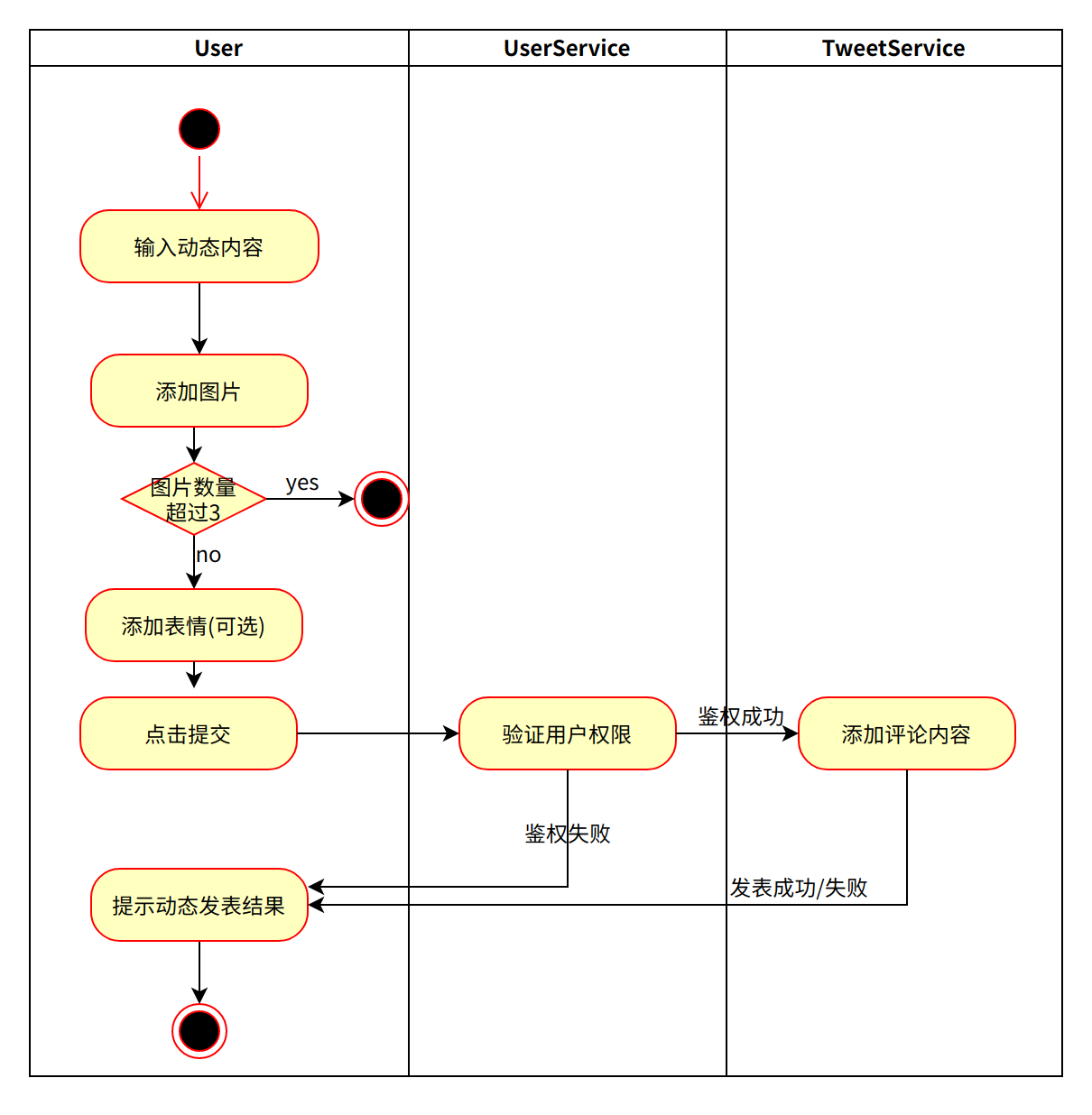




**发布动态**

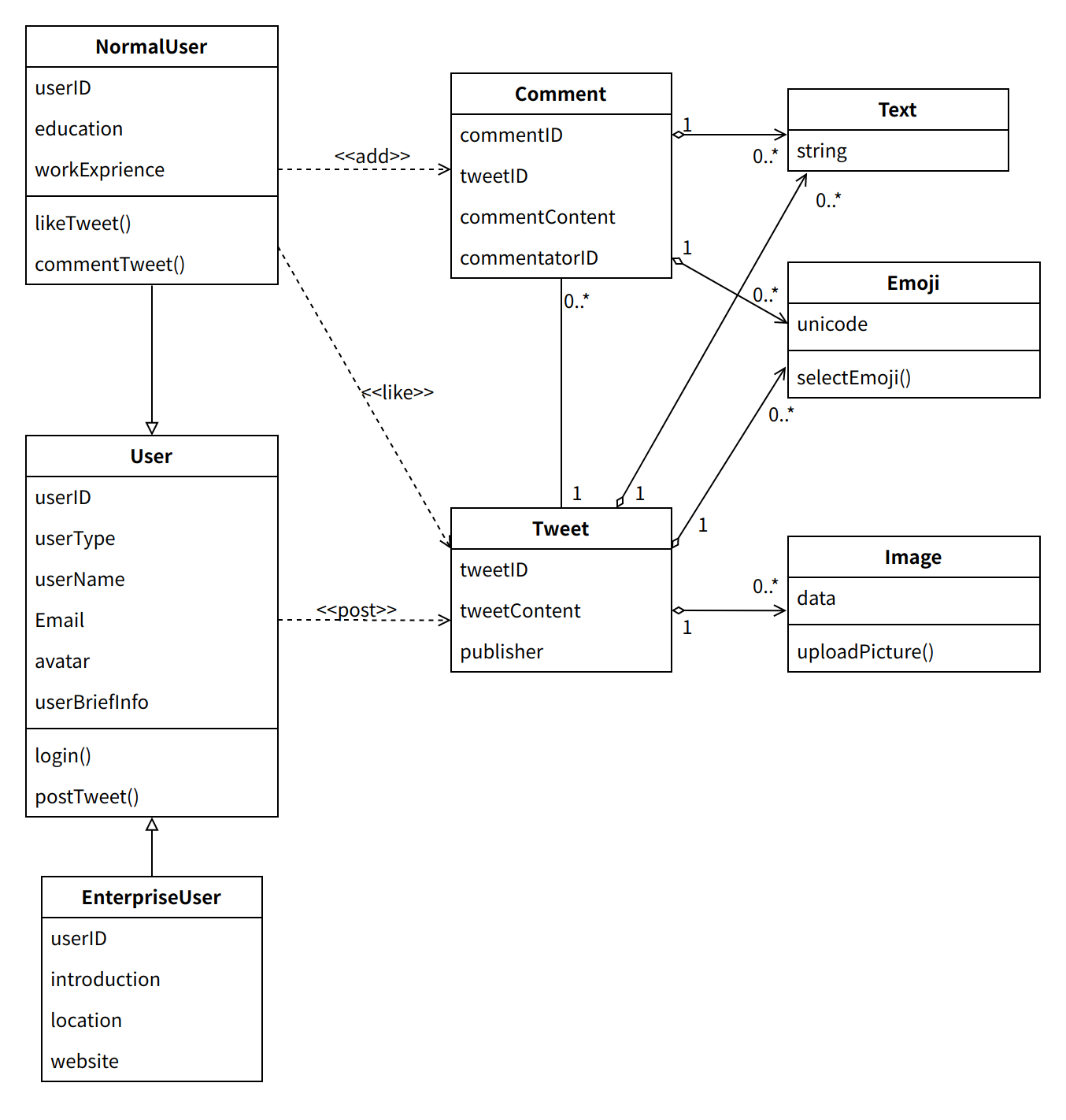
用户登录后可以发布动态。用户可以分别在输入框输入文字，用表情选择器选择表情或者使用文件选择器添加图片。内容输入完成后可以点击发布。



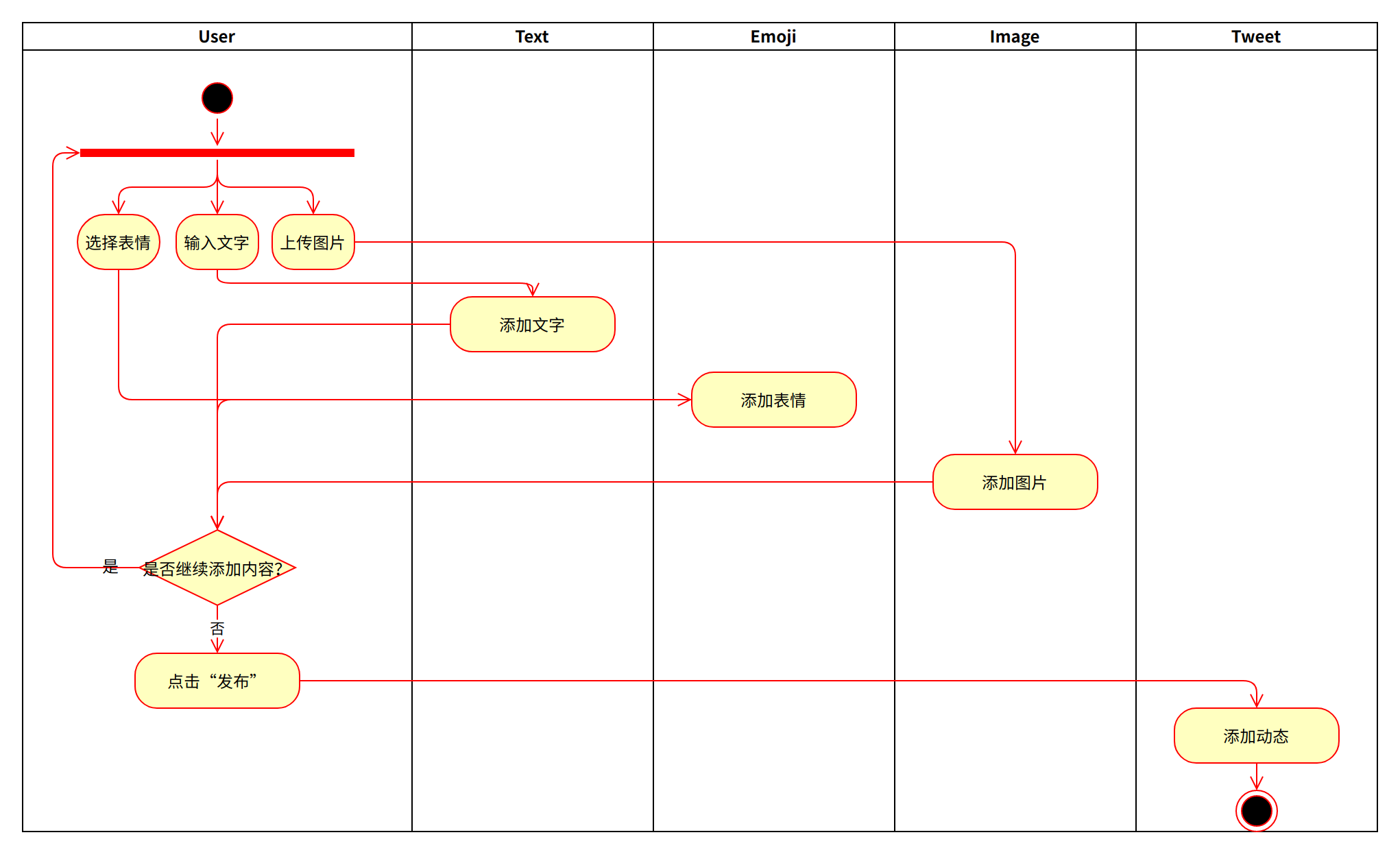


**数据建模**

普通用户（NormalUser）与企业用户（EnterpriseUser）均继承自基类用户（User）。User可以发布动态（Tweet），普通用户还可以点赞Tweet或添加评论（Comment）。Comment与Tweet相关联，一个Tweet可以对应0或多个Comment。同时，一个Comment可以有0或多个文字（Text）与表情（Emoji），一个Tweet也可以有0或多个Text，Emoji与图片（Image）。



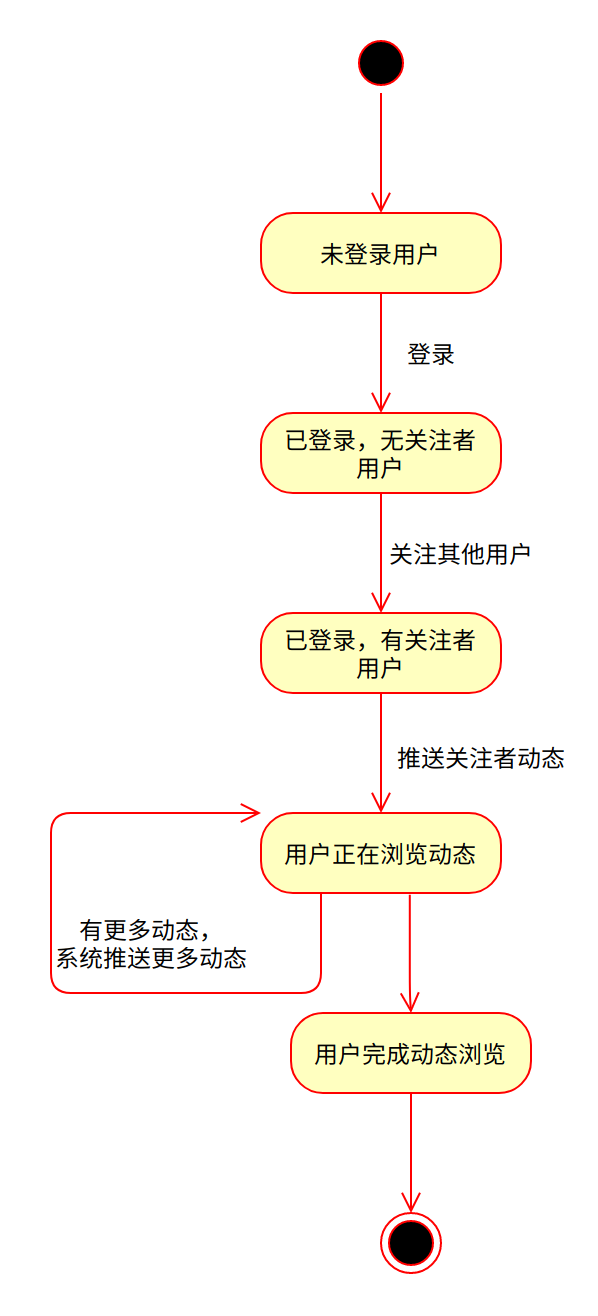
**发布动态**



**行为建模**

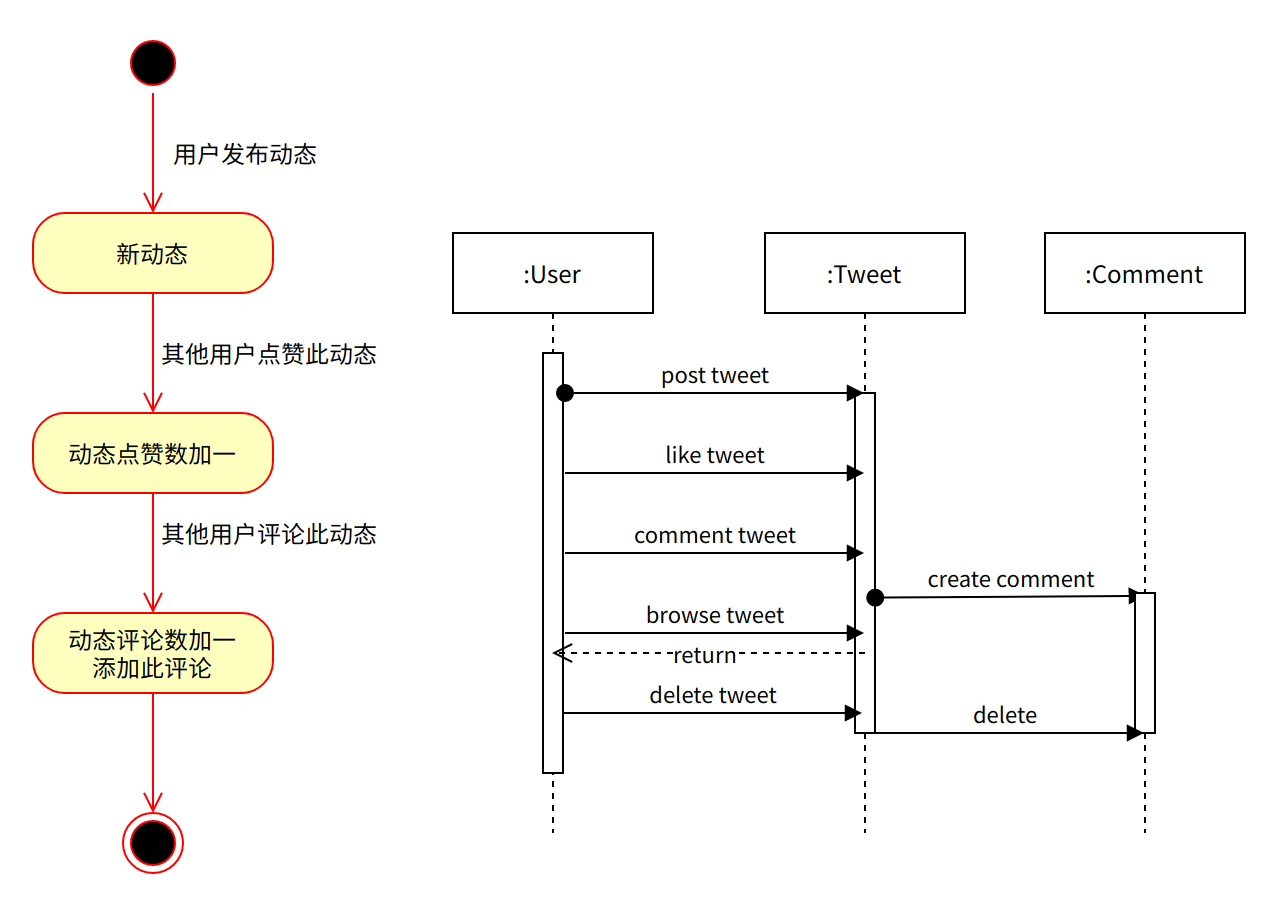
**用户类（User）**

在动态子系统中，如果一个用户登录后，没有关注者，则系统会为此状态用户推送用户关注。如果用户有关注者，则系统会为这种状态用户推送关注者动态。用户将推送动态浏览完毕时，如其关注者有更多动态，则系统会推送更多动态，用户继续浏览。否则用户完成动态浏览状态。



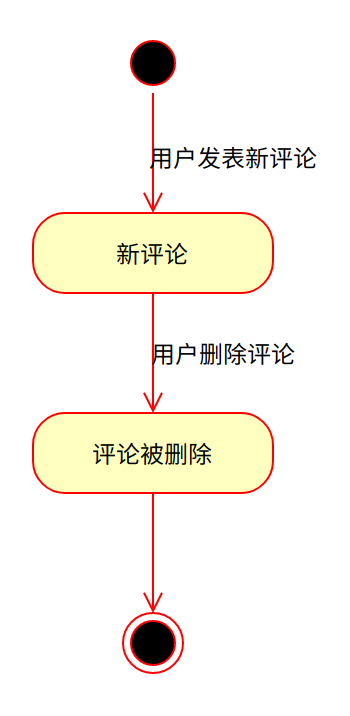
**动态类（Tweet）**

在动态子系统中，用户可以发布动态。其他用户可以点赞此动态，则动态点赞数加一。其他用户也可以添加评论，则动态评论数加一，同时生产新的评论。



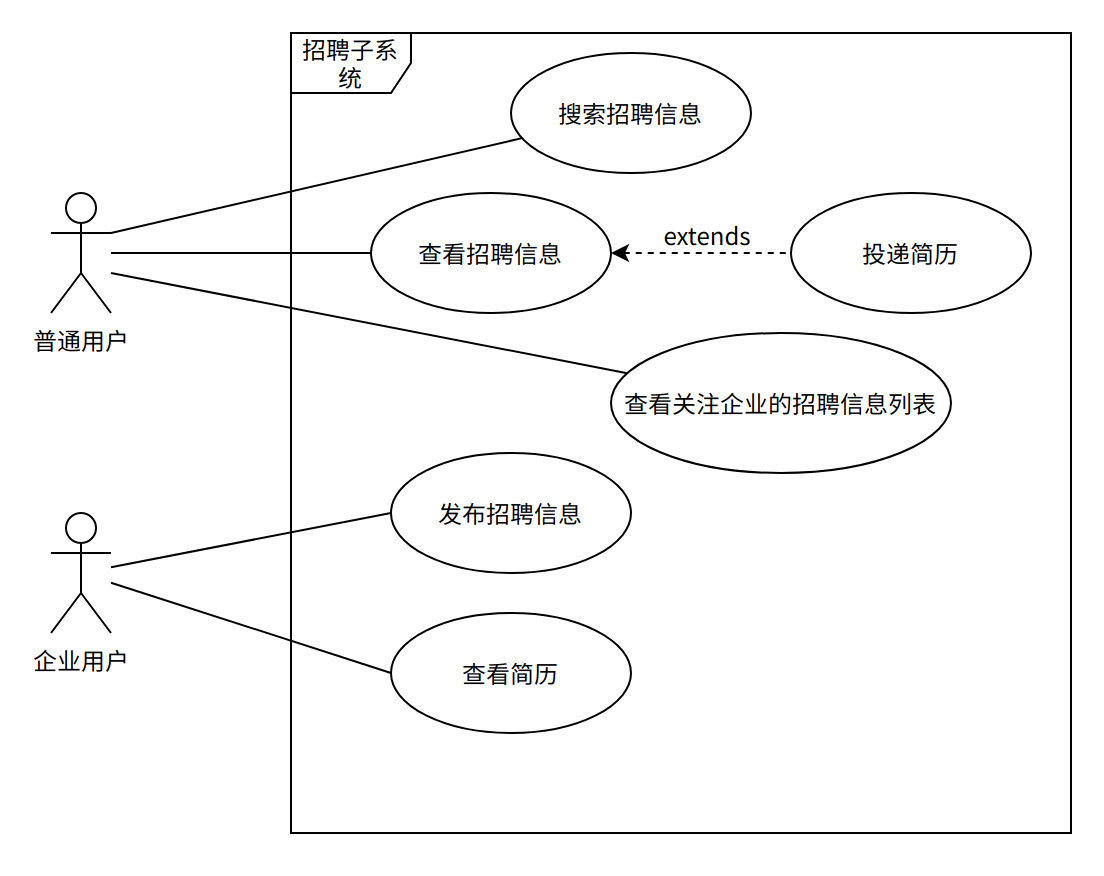
**评论类（Comment）**

用户在浏览动态时可以针对某条动态发表评论。之后也可以随时删除自己发表的评论。



**2.4.3 招聘子系统**

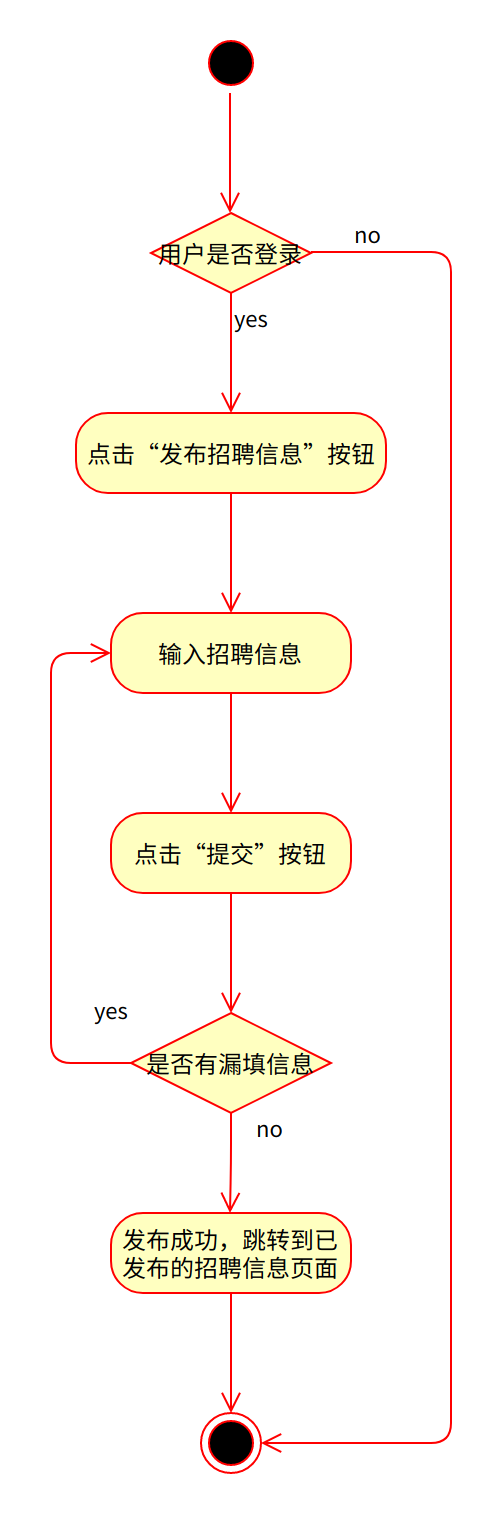
招聘子系统的使用者有普通用户和企业用户。普通用户可以查看企业用户发布的招聘信息，查看招聘信息时可以对招聘信息投递简历。企业用户可以发布招聘信息，也可以查看自己发布的招聘信息所收到的简历。



**功能建模**

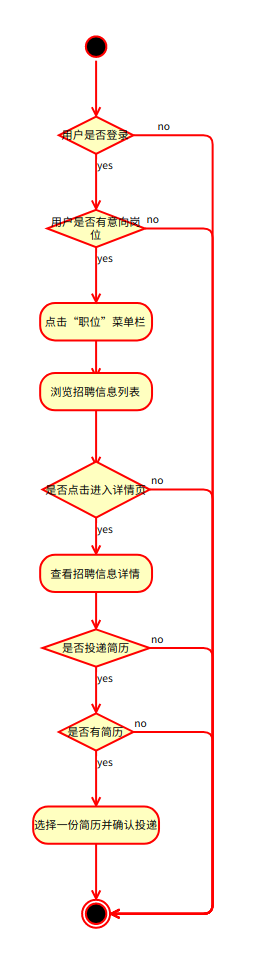
**发布招聘信息**

用户登录后，在自己的个人主页点击“发布招聘信息”按钮，并输入要发布的招聘信息，然后提交。系统会检查输入的信息是否有漏填的项目，如果有则提示用户继续输入完整信息，如果没有则招聘信息发布成功，跳转到用户已发布的招聘信息页面。



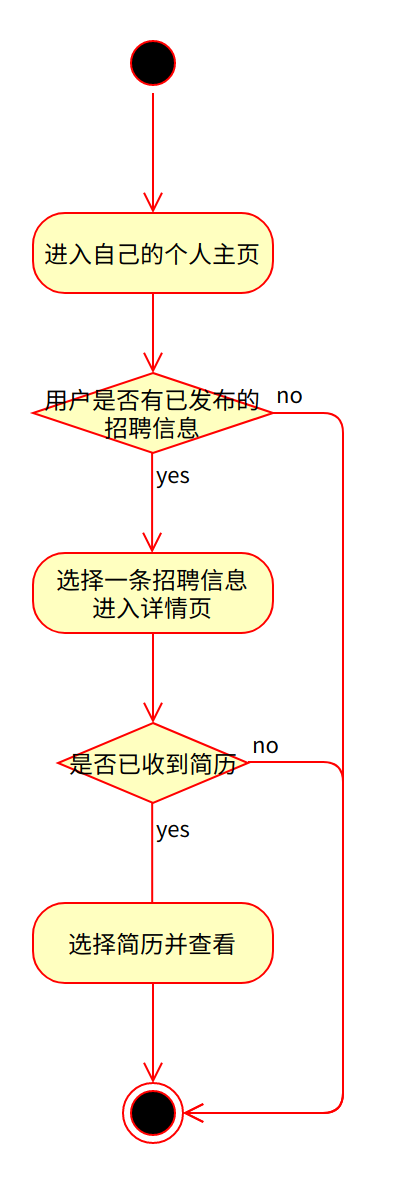
**查看招聘信息**

用户登录后点击“职位”菜单栏，浏览企业发布的招聘信息列表。浏览过程中可以点击某条招聘信息进入其详情页，在详情页中可以选择投递简历。投递简历时若没有简历则显示提示并提供跳转到个人信息页面的选项；若有简历则选择一份简历并确认投递。



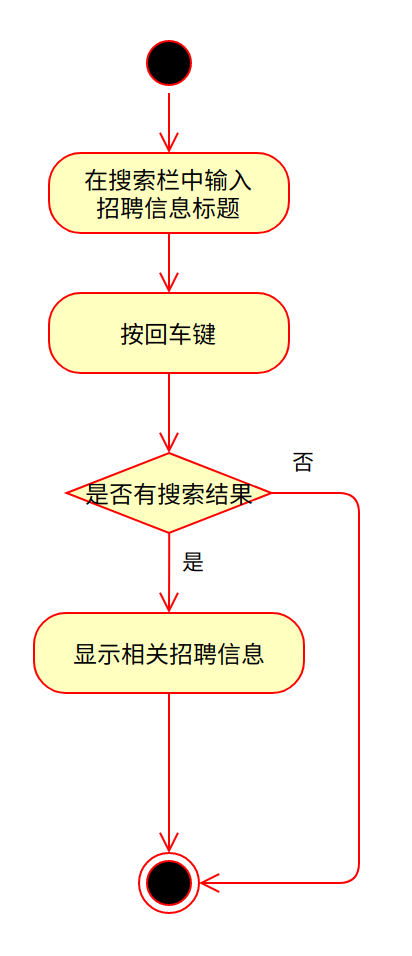
**查看简历**

企业用户进入自己的个人主页，若没有已发布的招聘信息则无法查看简历。用户点击一条招聘信息进入其详情页，然后点击“查看简历”按钮，进入该招聘信息的简历列表。如果列表不为空，则可以选择一份简历并查看。



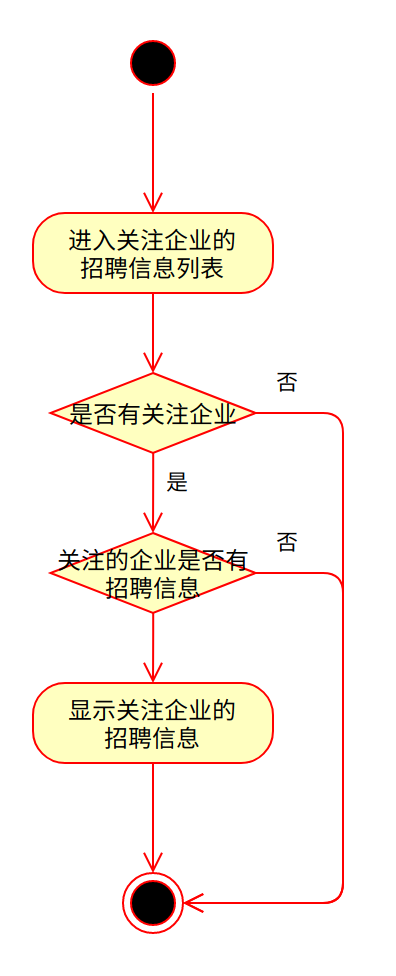
**搜索招聘信息**

用户在搜索栏中输入要搜索的招聘信息标题，并按下回车进行搜索。如果搜索到了相关招聘信息，则会显示相关招聘信息的列表。



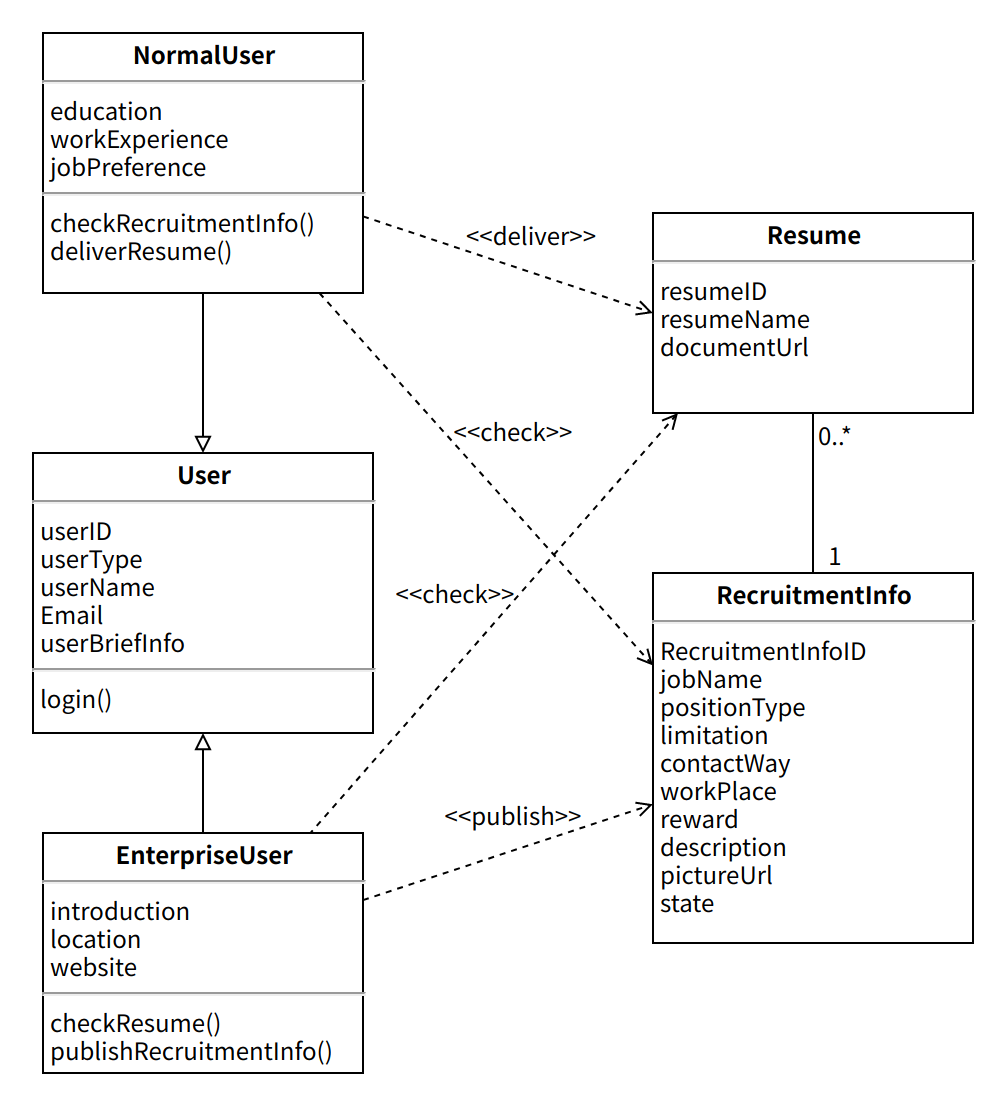
**查看关注企业的招聘信息列表**

用户关注企业后，可以查看所有关注的企业已发布的招聘信息列表。



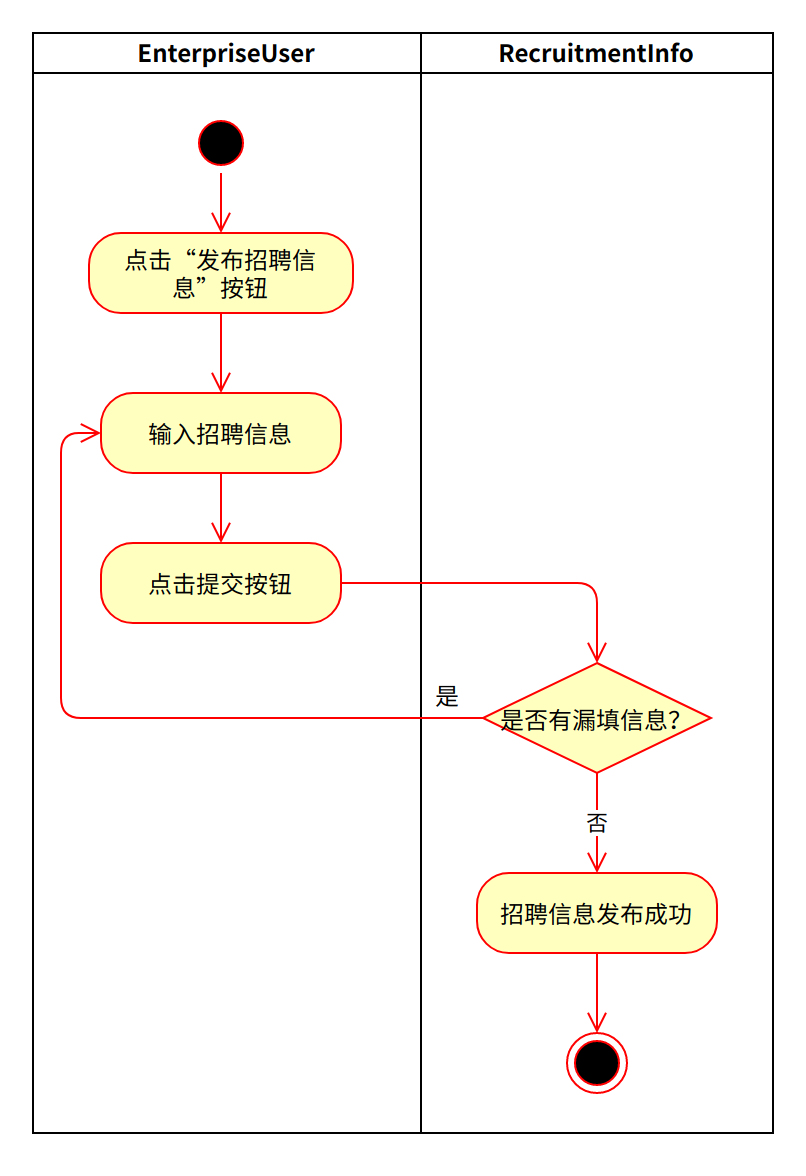
**数据建模**

招聘子系统共有五个实体类，其中普通用户类（NormalUser）和企业用户类（EnterpriseUser）继承自用户类（User）。招聘信息（RecruitmentInfo）与简历（Resume）关联，一个招聘信息（RecruitmentInfo）对应0到多个简历。普通用户可以进行查看招聘信息（checkRecruitmentIfo）和投递简历（deliverResume）两种操作。企业用户可以进行发布招聘信息（publishRecruitmentInfo）和查看简历（checkResume）两种操作。

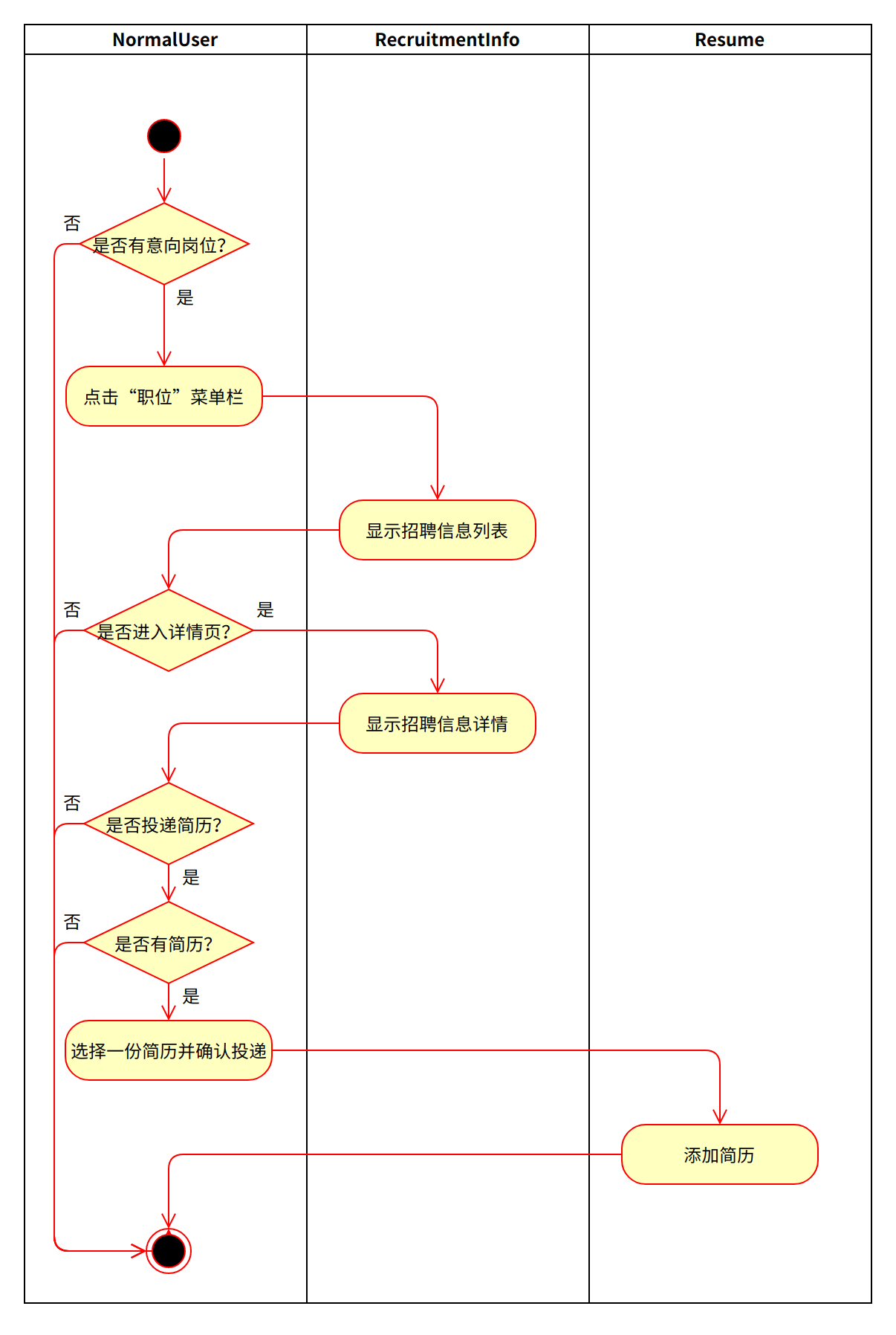


各个用例的泳道图：

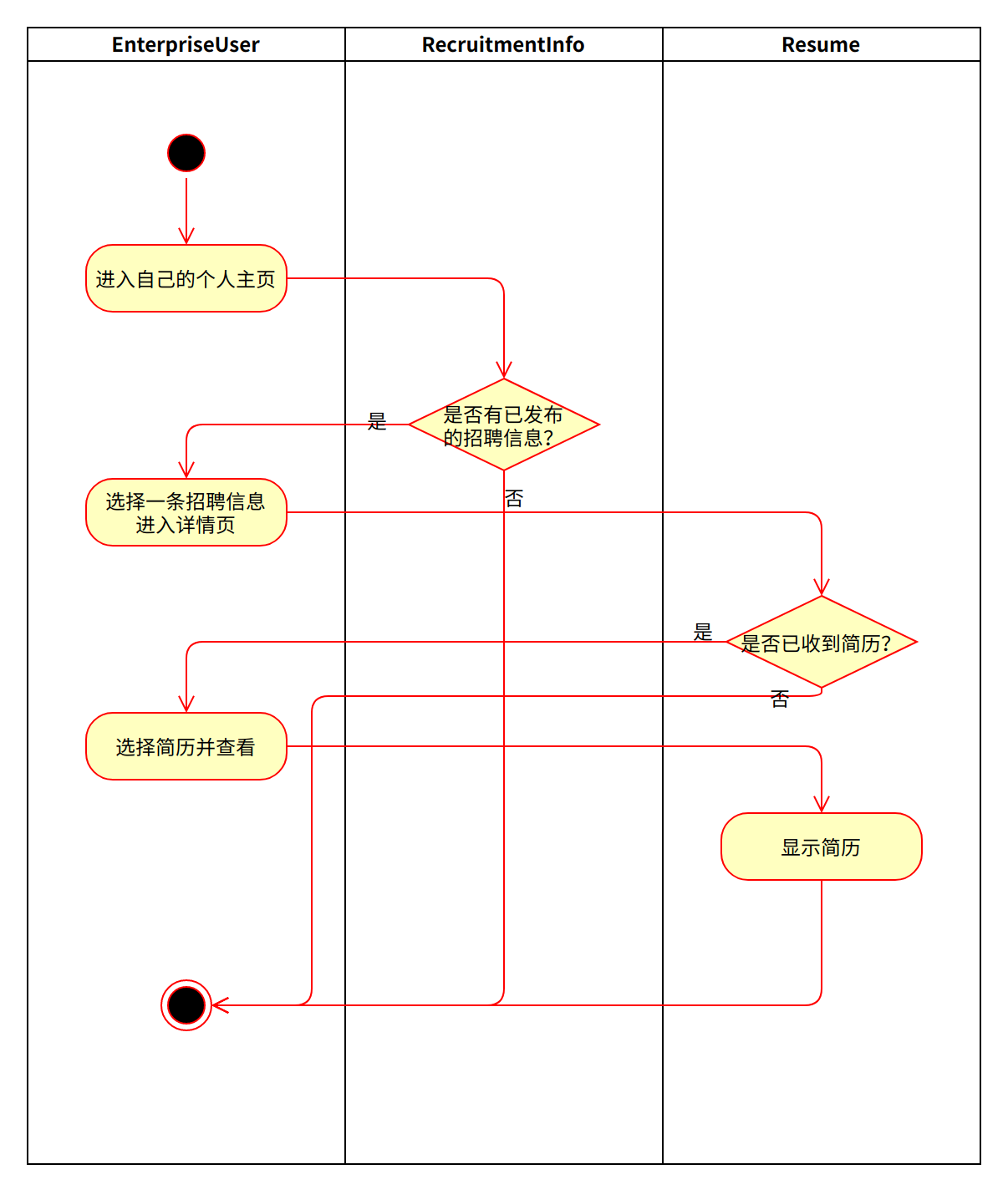
**发布招聘信息**



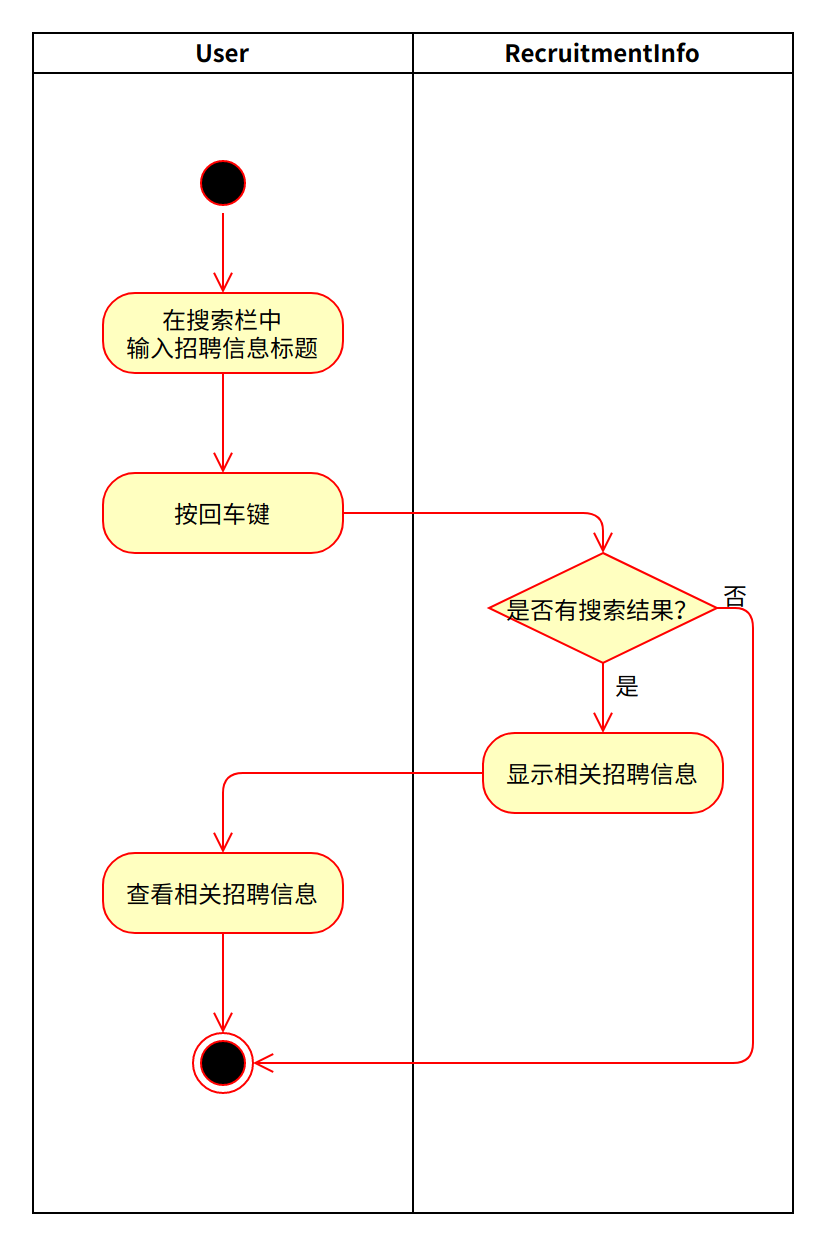
**查看招聘信息**



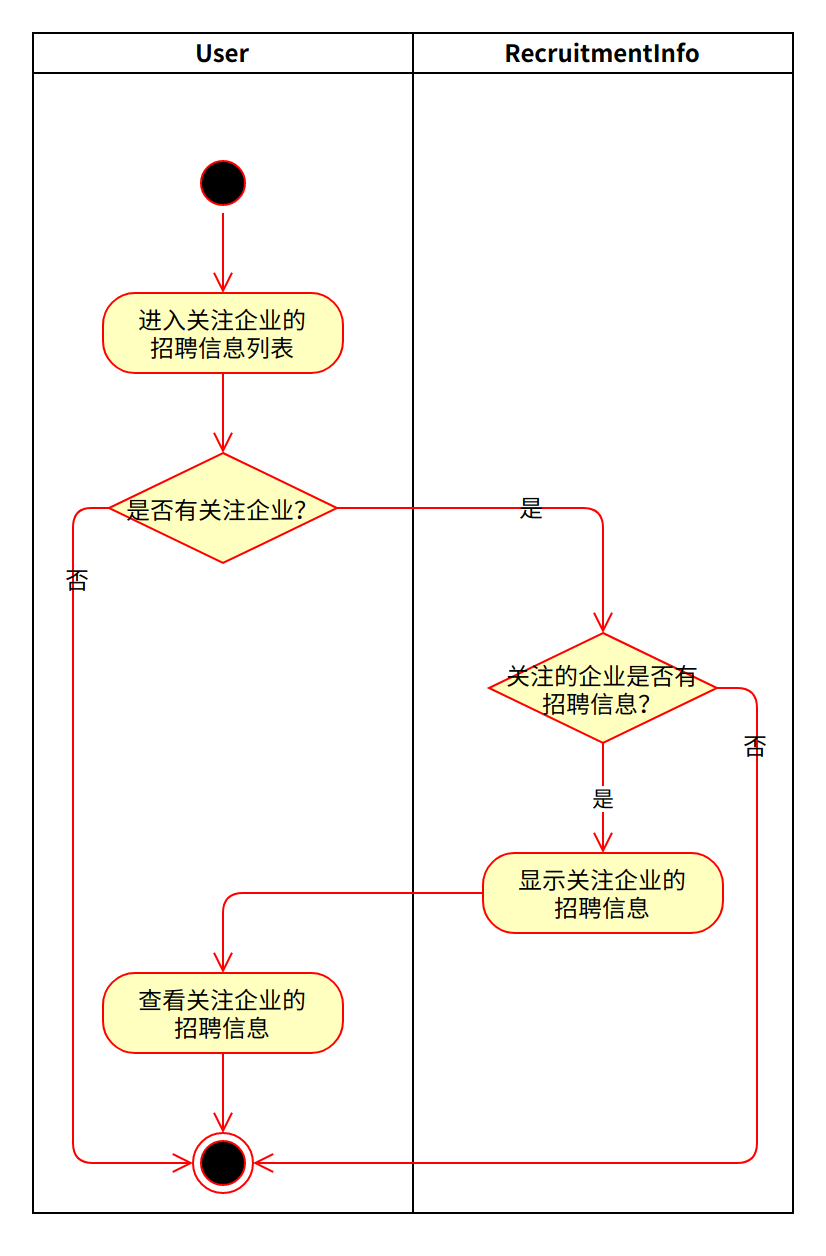
**查看简历**



**搜索招聘信息**



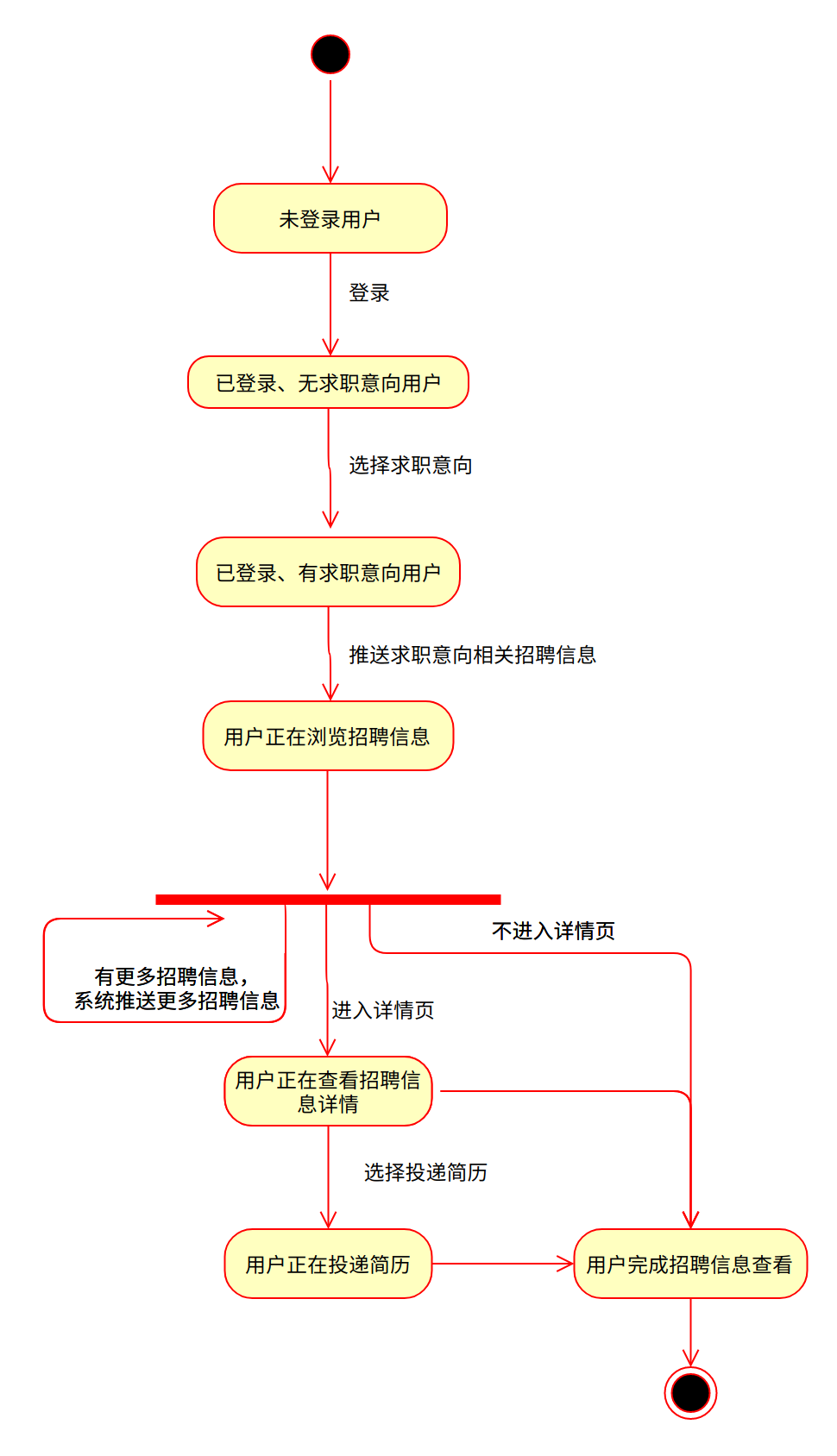
**查看关注企业的招聘信息列表**



**行为建模**

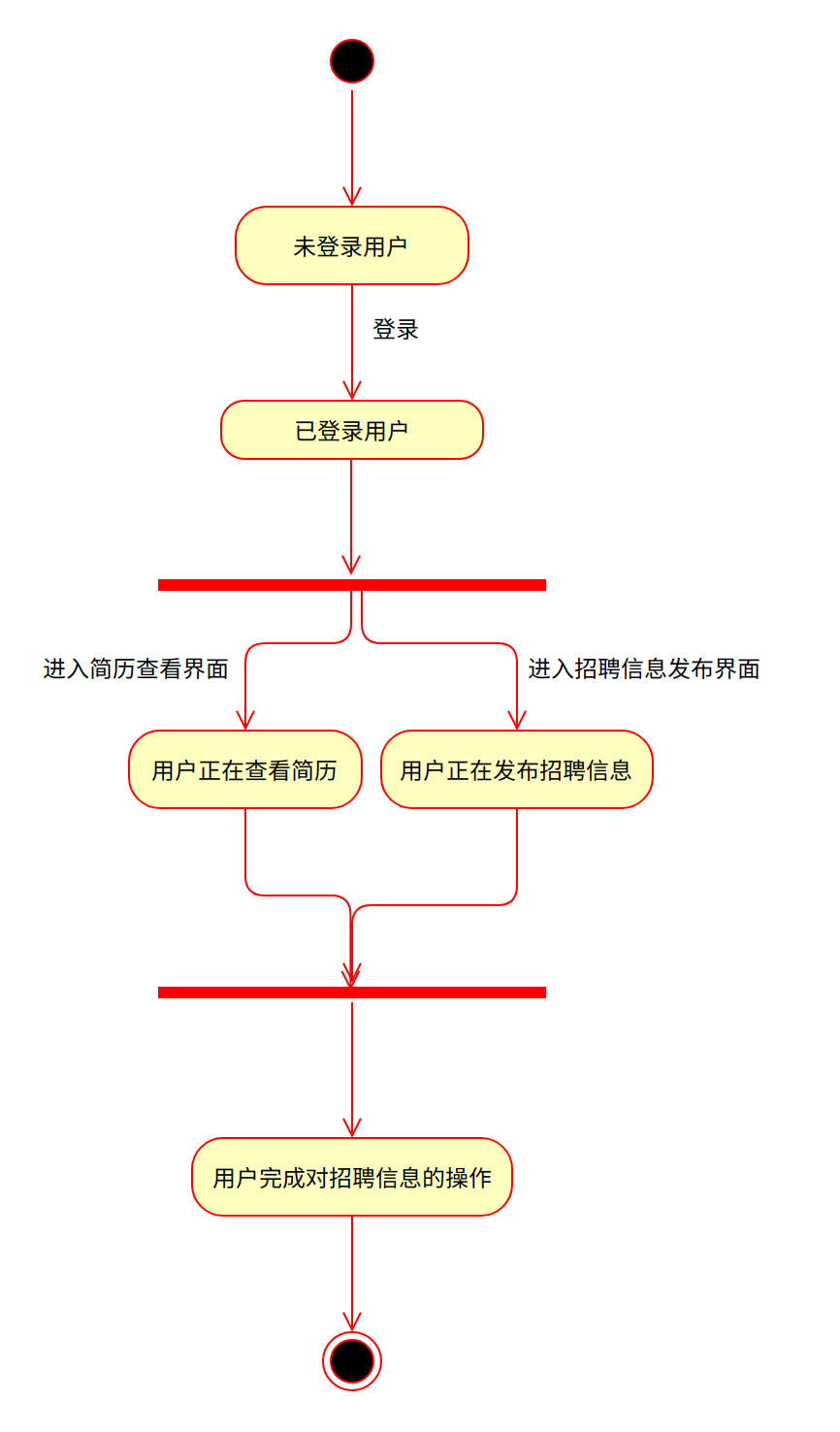
**普通用户类（NormalUser）**

用户登录之后，如果没有选择过求职意向，则无法接收到招聘信息推送。选择求职意向之后，系统会根据求职意向推送相关的招聘信息。用户将推送的招聘信息浏览完毕之后，如果有更多与其求职意向相符的招聘信息，则系统会继续推送。用户可以选择进入招聘信息详情页，进入之后还可以选择投递简历。



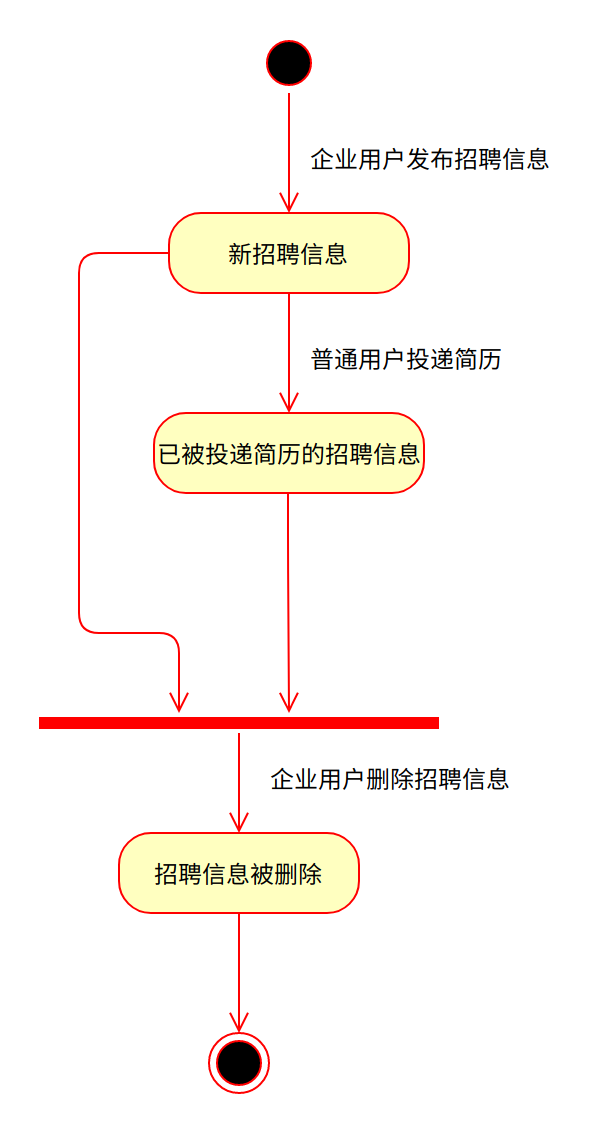
**企业用户类（EnterpriseUser）**

企业用户登录后，可以进入招聘信息发布界面进行招聘信息的发布，也可以进入简历查看收到的简历。



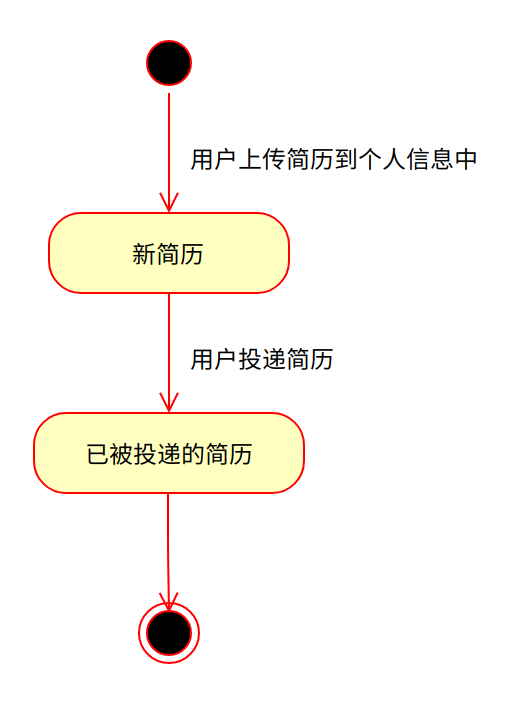
**招聘信息类（RecruitmentInfo）**

企业用户发布招聘信息后，普通用户可以对招聘信息投递简历；招聘信息发布后企业用户也可以删除已发布的招聘信息。



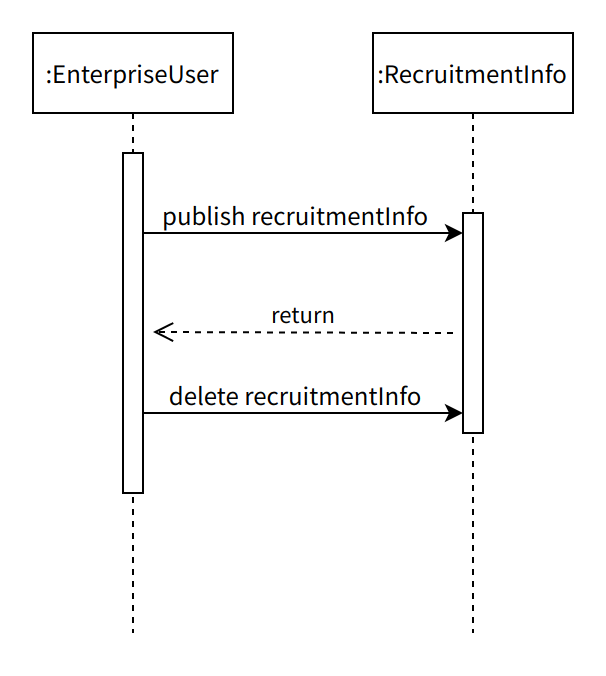
**简历类（Resume）**

用户上传简历后，简历会保存在用户的个人信息页面中。用户可以对已发布的招聘信息投递自己的简历。



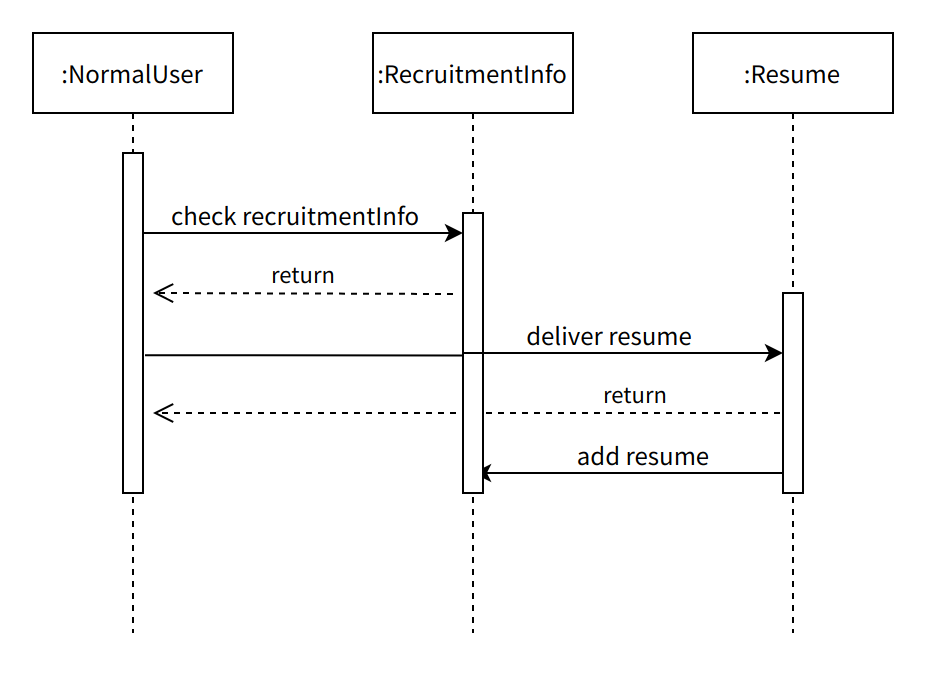
**发布招聘信息**

企业用户发布招聘信息之后，系统返回发布是否成功的信息。招聘信息被发布之后，企业可以将其删除。



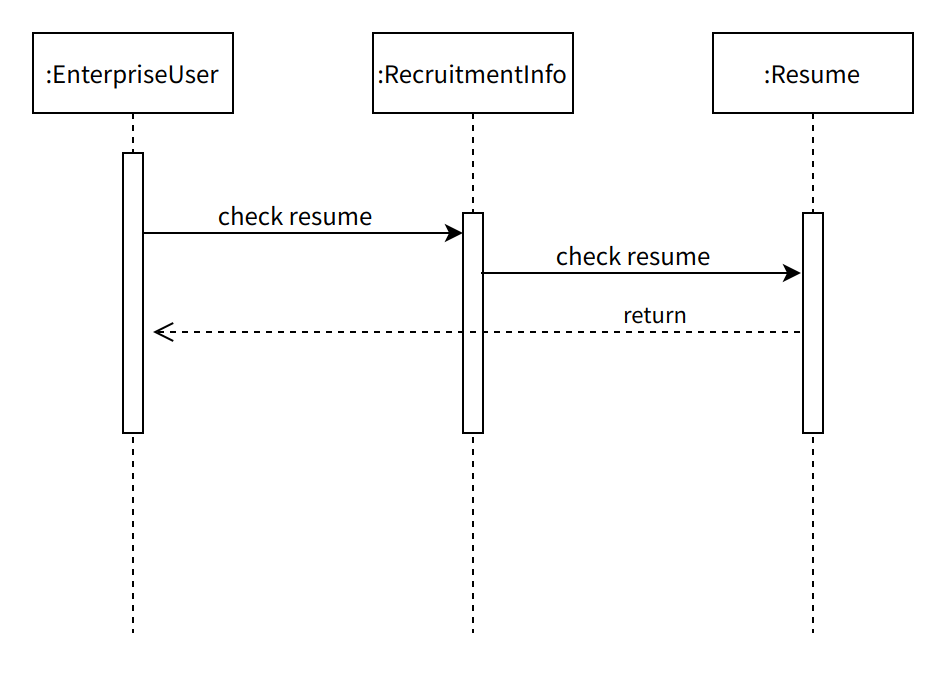
**查看招聘信息**

普通用户查看招聘信息时，系统会将招聘信息返回给用户查看。用户还可以选择向招聘信息投递简历，系统返回是否投递成功的信息。投递成功之后，简历会被添加到招聘信息的简历列表中。



**查看简历**

企业用户选择查看自己发布的某一条招聘信息的简历，系统会将对应的简历返回给用户查看。



1. **非功能性需求**

**3.1 性能要求**

**3.1.1精度**

软件的输入精度:

数字要求：小数点后保留2位有效数字

字符串：长度限定为256位。

输出数据精度的要求:

数字：小数点后保留2位有效数字;

字符串：其长度限定为256位。

传输过程中的精度:

数字：小数点后保留2位有效数字

字符串：取其有效位数。

**3.1.2 时间特性要求**

同步时钟：后端服务器为准

图片传输显示时间<1s

文件预览显示时间<1s

响应时间<0.5s

更新处理时间<0.5s

数据的转换和传送时间<1s

页面整体刷新时间2s左右，不超过4s

**3.1.3 输入输出要求**

1. 登录注册信息：

用户的用户名以字符串格式存储，不可重复，长度在3~15位之间

用户的密码以字符串格式存储，长度6~20为，需保证至少有大写字母、小写字母，阿拉伯数字三种类型，以提升安全性

1. 个人信息：

头像，背景图片：分别为一个文件，格式为png，jepg或jpg，文件最大为2M

姓名：2~5个字符，可以重复

年龄：数字

性别：从“男，女，保密”中选择

身份描述：2~20个字符

电话号码：符合电话号码规范

意向职位：从选择菜单中选择，最多选3条

位置：精确到省，市

简历文件：最多上传3文件，文件格式为pdf，文件最大为3M

1. 教育经历：

学校：从数据库中搜索

专业：2~10个字符

学位：从“专科，本科，硕士，博士”中选择

起止时间：选择时间，精确到日期，起始时间必须大于截至时间

1. 工作经历：

职位：2~10个字符

公司：从数据库中搜索

起止时间：同“教育经历”

描述：2~50个字符

1. 职位信息：

职位名称：2~20个字符

职位分类：从选择菜单中选择，只能选择一项

薪资描述，招聘职位限制：2~10个字符

工作地址：真实地理信息

职位详情：5~2500个字符

**3.2 数据管理能力要求**

数据库共需14张表。

预计系统投入使用后，数据库大小每年增长3G，每3个月需要做数据备份。

图片等文件资源全部存储在对象存储OSS服务器上。

数据库性能：使用主从(Replica)数据库，支持同时从两个数据库读取，提升并发性能。

数据库安全：用户密码存储时使用MD5加密，避免存储明文。主从数据库部署在不同的服务器上，提升容灾能力。

**3.3 安全及保密性要求**

成功申请SSL证书后，需将开发时所用的http服务，更换为https服务，保证加密传输

用户登录后，传入后端的密码需加密处理，数据库中不应明文存储，采用SHA256方式进行加密

用户存在前端浏览器的唯一标识采用token

数据库数据应应定期进行备份，并在多台服务器上留有备份，以防止可能出现的外部攻击、介质损坏故障等造成的数据丢失，引起法律纠纷和经济损失。

数据库系统应严格控制权限，防止非法入侵、恶意篡改等

**3.4 灵活性要求**

a. 操作方式上的变化：电脑端可通过鼠标键盘操作，手机端可通过触控方式操作

b. 运行环境的变化：通过采用B/S架构，使得该软件可在不同的操作系统环境下运行，只需安装浏览器即可浏览运行。

c. 同其他软件的接口的变化：登录和支付模块随接口提供者进行变化

d. 精度和有效时限的变化:可以根据实际情况设置。

e. 计划的变化或改进:在生产过程中,可有效的改变计划,并对其进行改进

**3.5 其他要求**

故障处理要求：可能出现用户用量过大导致系统崩溃，须在1h内解决。

易读性要求：各函数的实现必须加以注释

异常处理要求：所有异常须在抛给JVM前捕获。

可维护性要求：降低维护人员的工作量，发生错误时尽可能为维护人员提供清晰可靠准确的提示信息。

1. **运行环境规定**

**4.1 硬件设备**

|  |  |
| --- | --- |
| CPU | 1Ghz及以上 |
| 内存 | 512MB及以上 |
| 存储 | 20GB及以上 |
| 是否联机 | 是 |
| 带宽 | 1M及以上 |

**4.2 支持软件**

|  |  |
| --- | --- |
| 操作系统 | windows、linux、macos等主流操作系统 |
| 浏览器 | Microsoft Edge、Google Chrome、Firefox、Safari、Opera |
| 数据库 | Oracle、MySql、Access |

**4.3 接口**

Iteration1：

外部地图接口：在多家地图服务提供商中，例如：百度地图、高德地图、腾讯地图，选择一家数据准确、性价比高的提供商，预留地图对接接口，方便用户查看岗位工作所在地。

外部数据接口：对接全国高校数据接口，在用户添加教育经历时可以方便获取到全国高校数据信息，而无需平台自行维护相关数据库。

Iteration2：

支持外部登录接口：例如微博、微信、QQ登录，方便用户使用而无需新注册账号，提高平台的注册率。

支持外部支付接口：对接包括微信支付、支付宝支付、云闪付等，在后续迭代开发中将为不同用户提供高级付费功能，以提升用户体验，维持平台盈利。为此需要提供常见的支付手段。