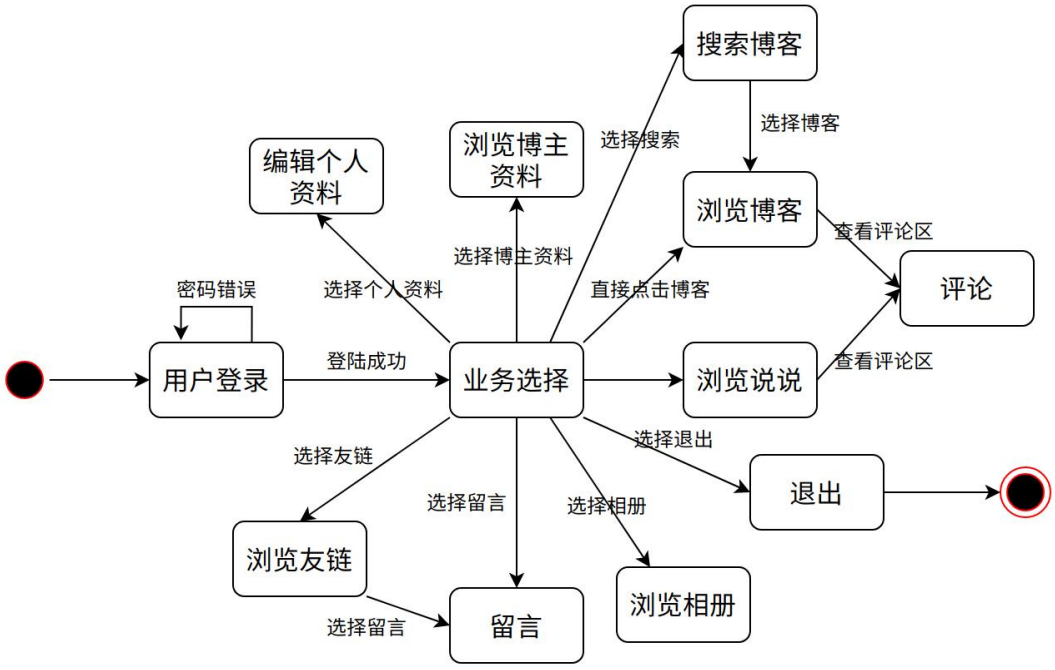
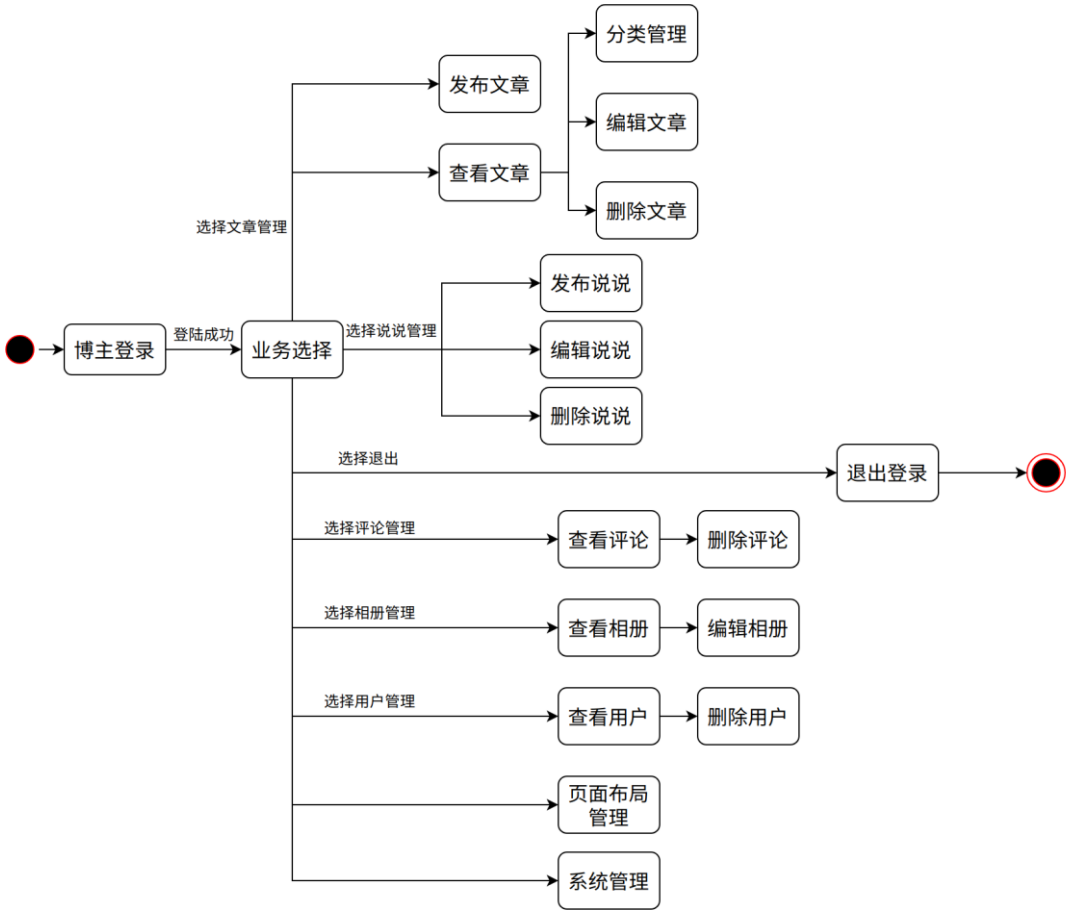
动态建模工具

1.状态图

前台用户状态图:

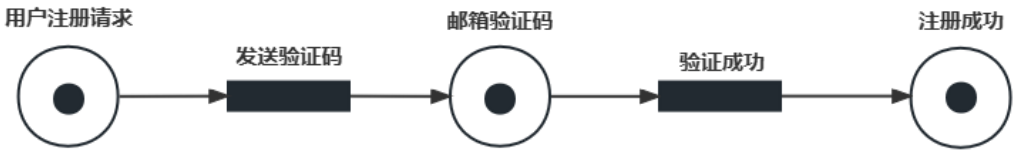


后台博主状态图:

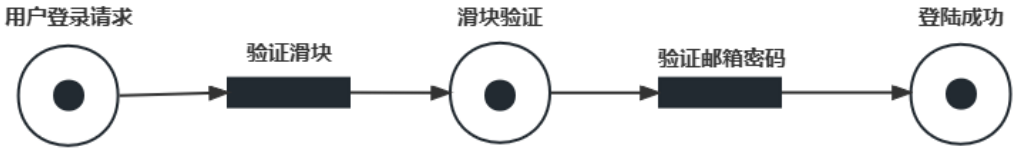


2.Petri 网

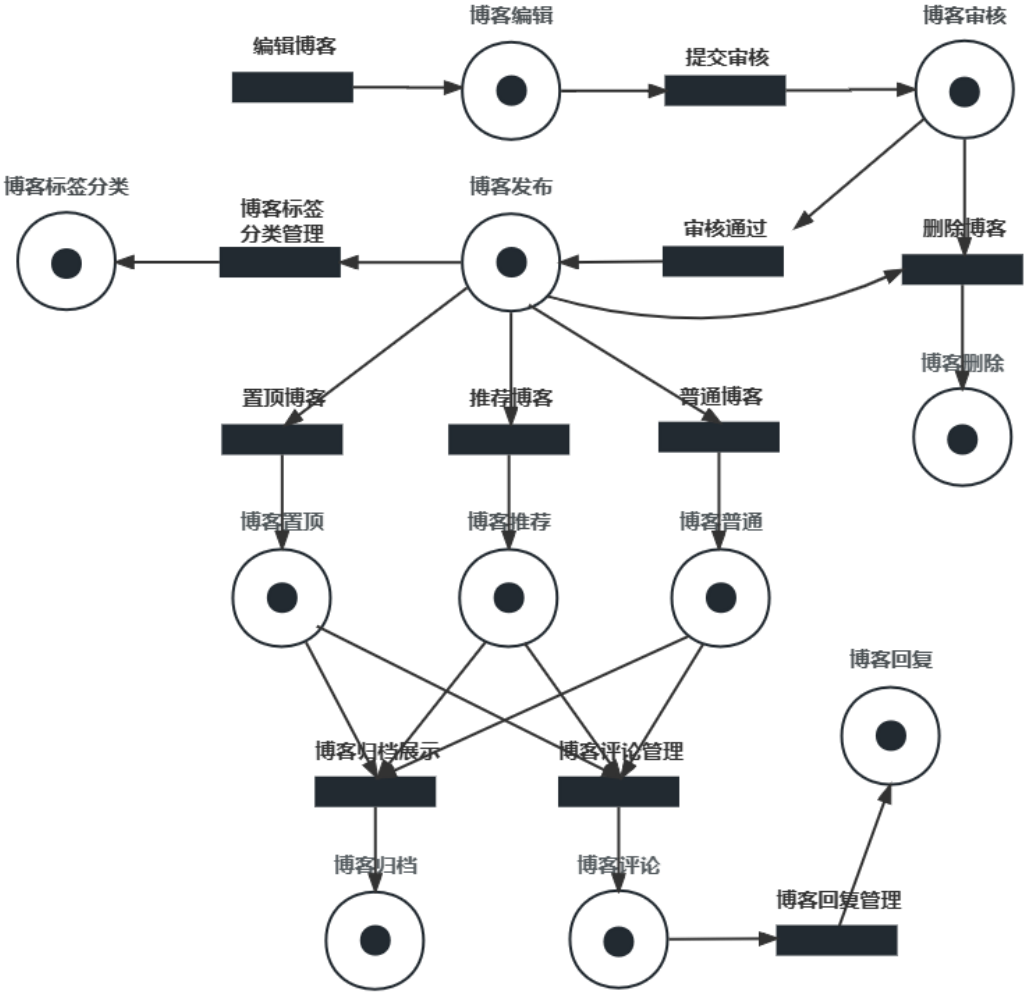
用户注册 Petri 网



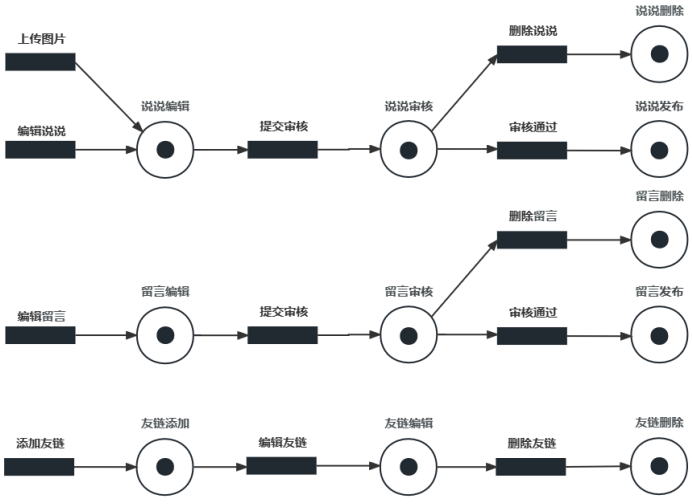
用户登录 Petri 网

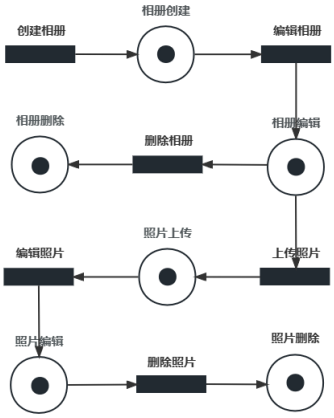


博客 Petri 网

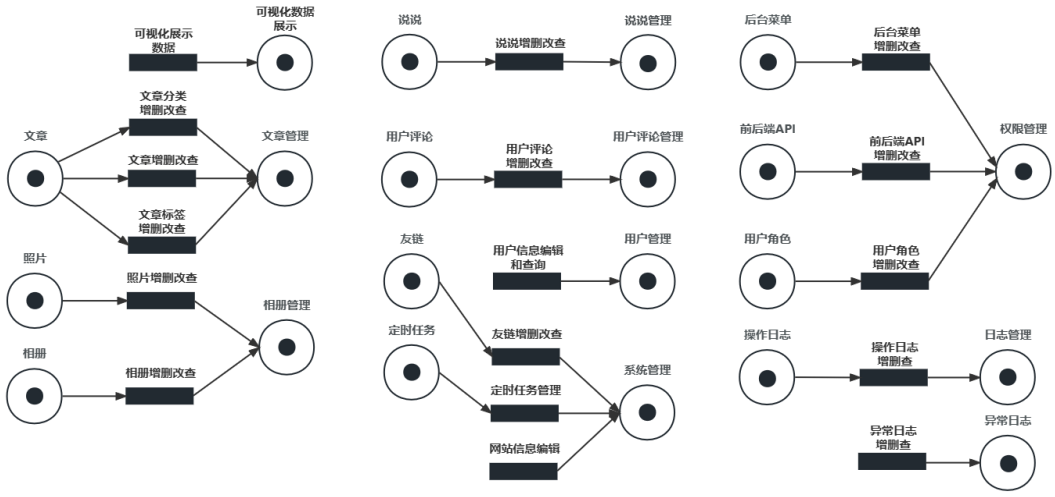

博客编辑
博客审核
编辑博客
提交审核
博客标签分类
博客发布
博客标签
分类管理
审核通过
删除博客
博客删除
置顶博客
推荐博客
普通博客
博客置顶
博客推荐
博客普通
博客回复
博客归档展示
博客评论管理
博客归档
博客评论
博客回复管理

其他模块 Petri 网

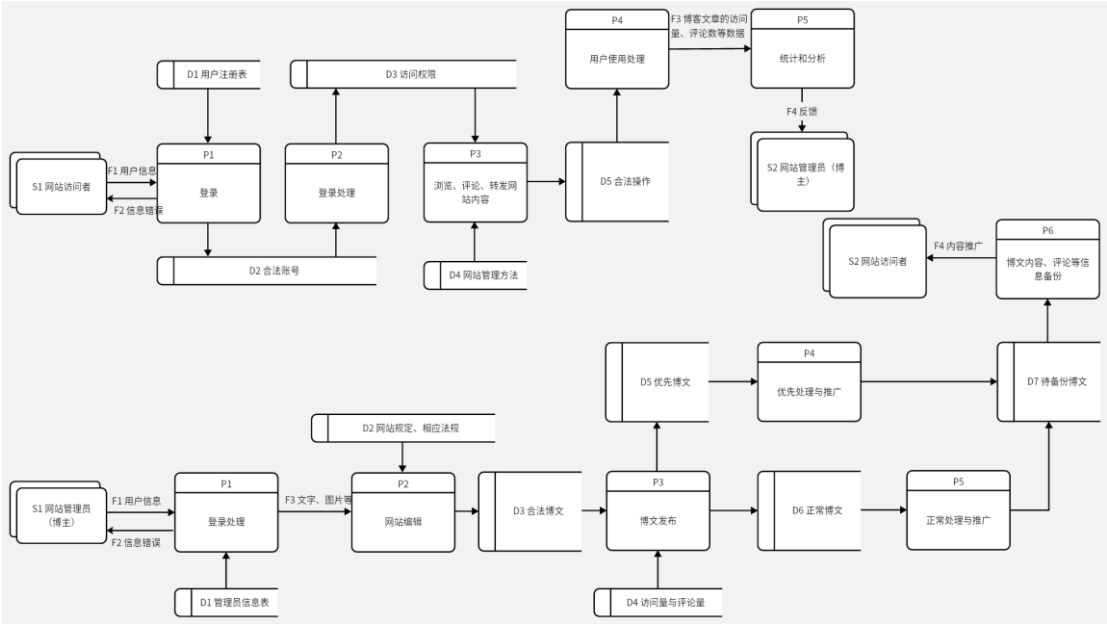

上传图片
删除说说
说说删除
说说编辑
说说审核
编辑说说
提交审核
审核通过
说说发布
删除留言
留言删除
留言编辑
留言审核
编辑留言
提交审核
审核通过
留言发布
友链添加
友链编辑
添加友链
编辑友链
删除友链
友链删除


相册创建
创建相册
编辑相册
相册删除
删除相册
相册编辑
照片上传
编辑照片
上传照片
照片编辑
删除照片
照片删除

后台管理 Petri 网



3.数据流图



4.OCL 逻辑

1.登陆处理，检查用户是否存在以及用户名和密码是否匹配

```
Dockerfile
context User
  def: authenticate(username: String, password: String) : Boolean
=
    self.username = username and self.password = password
  enddef
endcontext
```

```
context LoginController
  pre: User.allInstances()->exists(user | user.username =
username)
  pre: User.allInstances()->select(user | user.username =
username)->one().authenticate(username, password) = true
  post: currentUser = User.allInstances()->select(user |
user.username = username)->one()
endcontext
```

2.操作合法性验证，以验证输入内容字数为例

```
Dockerfile
context BlogPost
  inv: self.title.size() <= 100
endcontext
```

3.用户权限检测，检查当前用户为普通用户或管理员

```
Dockerfile
context BlogPost
  inv: self.owner = currentUser or currentUser.role = 'admin'
endcontext

context Comment
  inv: self.author = currentUser or currentUser.role = 'admin'
endcontext
```

4.文章标签约束

```
Dockerfile
context BlogPost
  inv: self.tags->forAll(tag | allTags->includes(tag))
endcontext

context Tag
  inv: self.posts->forAll(post | allPosts->includes(post))
endcontext
```

5.主楼和评论的状态

```
Dockerfile
context BlogPost
  pre: self.status = 'draft'
```

```
  post: self.status = 'published'
endcontext

context Comment
  pre: self.status = 'published'
  post: self.status = 'edited'
endcontext
```

6.检查文章和用户的关系

```
Dockerfile
context BlogPost
  inv: self.owner = currentUser or self.status = 'published'
endcontext

context User
  inv: self.posts->forAll(post | post.owner = self)
endcontext
```

7.非注册用户（游客）的访问权限

```
Dockerfile
context BlogPost
  inv: currentUser.role <> 'guest' or self.status = 'published'
endcontext

context Comment
  inv: currentUser.role <> 'guest' or self.status = 'published'
endcontext
```