

文档

系统需求文档

1. 项目可行性分析与描述

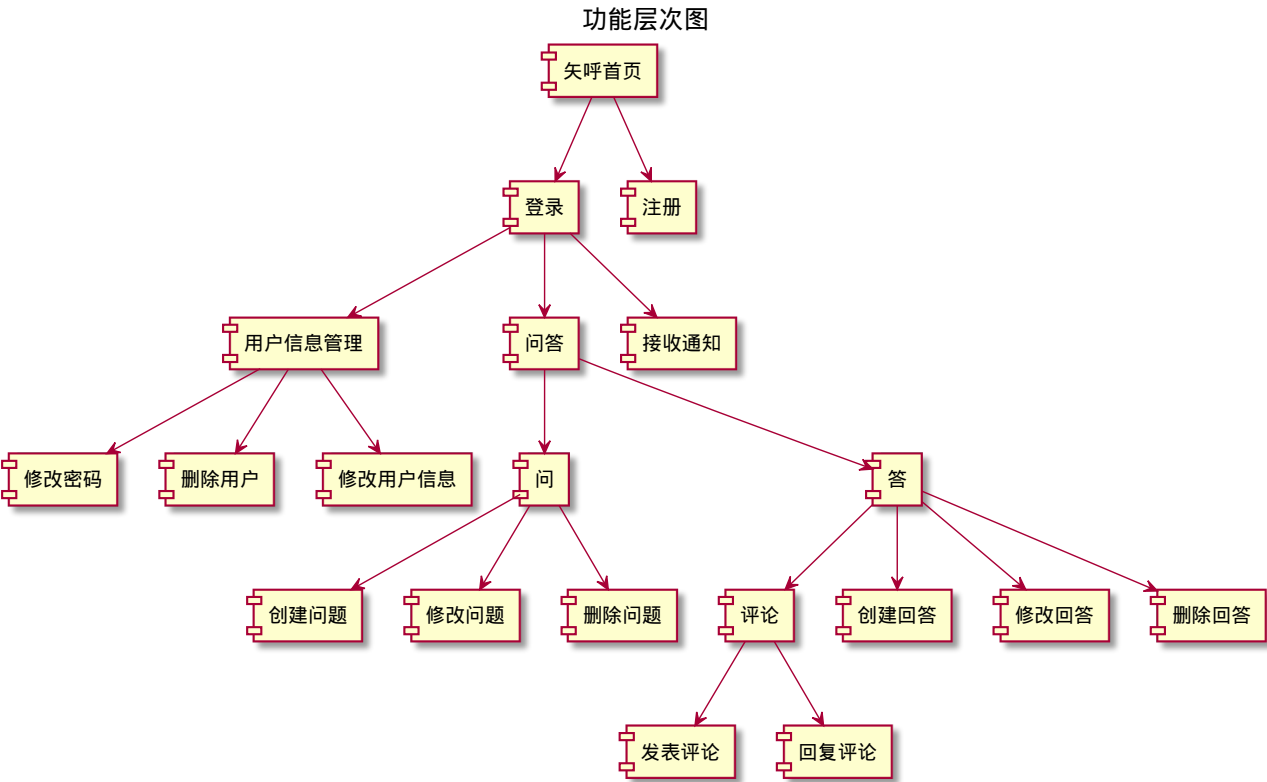
该项目要解决的问题归属于“网络知识社区”领域

该软件系统服务的用户是好奇的网民与乐于分享知识的朋友。它满足了用户提问、回答的需求，解决了用户想要一个开源的知乎的问题。

我们要解决的问题知乎已经解决了。与知乎相比，我们的系统优势在于简单（“KISS”）以及不用充钱；劣势在于出现的比知乎晚，用户已经被知乎抢光了、功能不如知乎多（没有电子书、音频课等，但这也算优点）

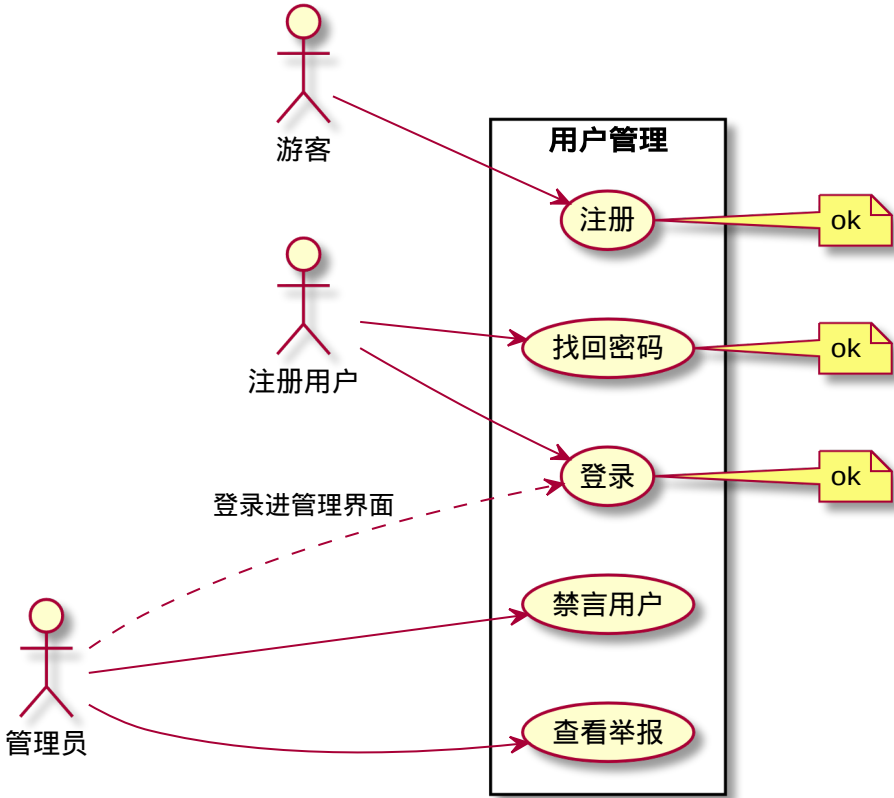
由于知乎本体未开源所以没法在代码级别比较优劣，但我看他们讲知乎有[千万级高性能长连接网关](#)，性能方面一定不如知乎本体。

2. 系统功能

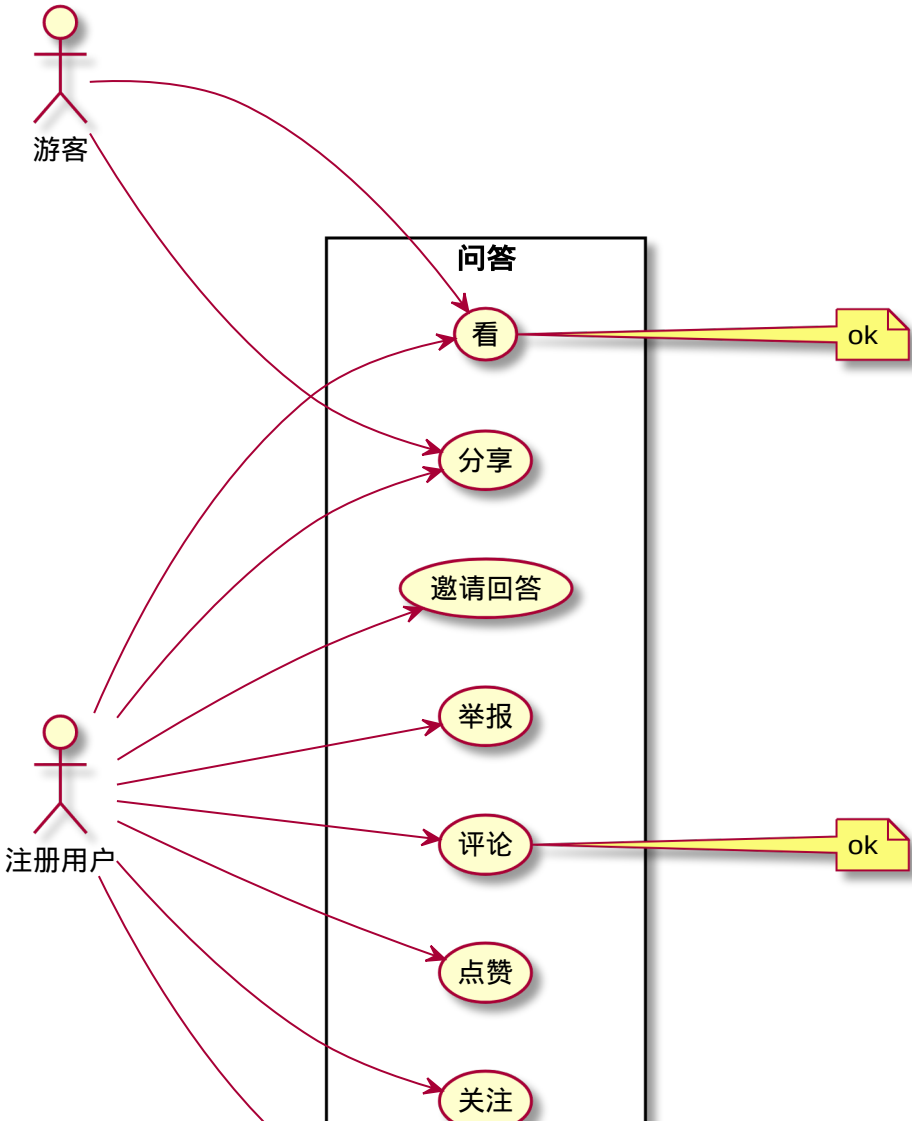


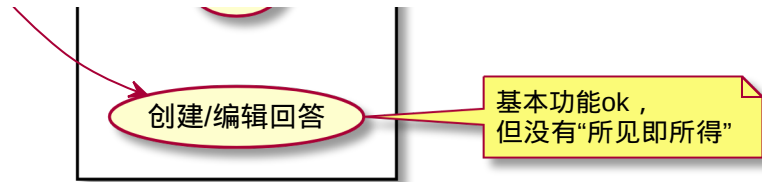
用例图：

不同用户在用户管理上的权限



不同用户在问答上的权限（管理员禁止参与问答）





客户端应用与服务端应用各自实现哪些功能:

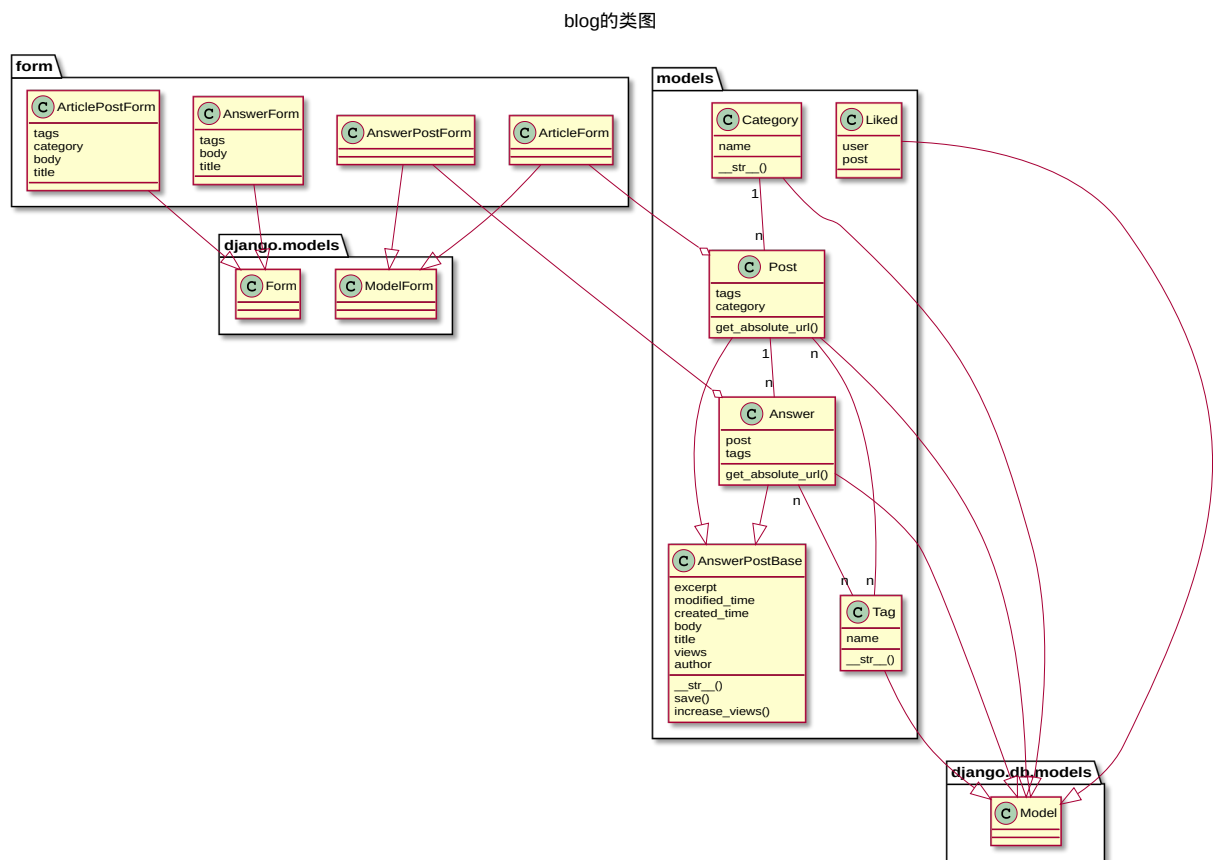
- 客户端实现渲染界面、把请求发给服务端的功能（比如把用户发文章的Post发给服务端调用真正发文章的函数）
- 服务端负责处理客户端的请求，然后对数据库进行存储、修改。还要告诉客户端下一步渲染哪个网页。

TODO 截图

3. 领域模型

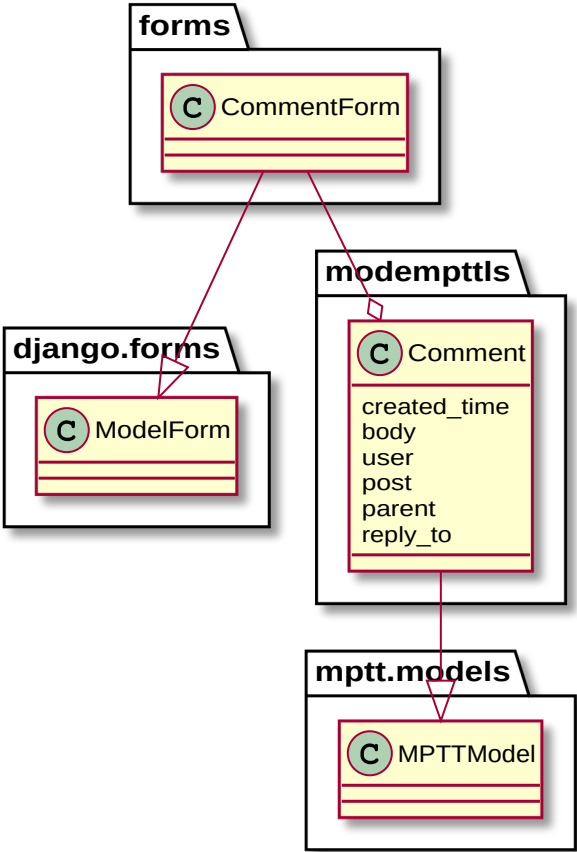
类图：

blog的类图



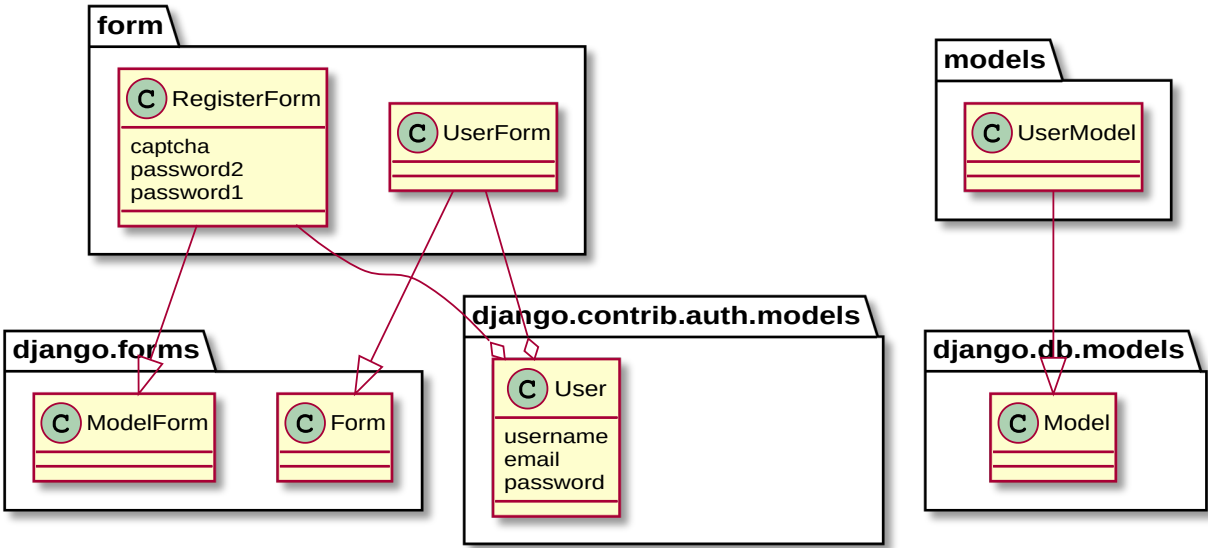
comment的类图

comment的类图



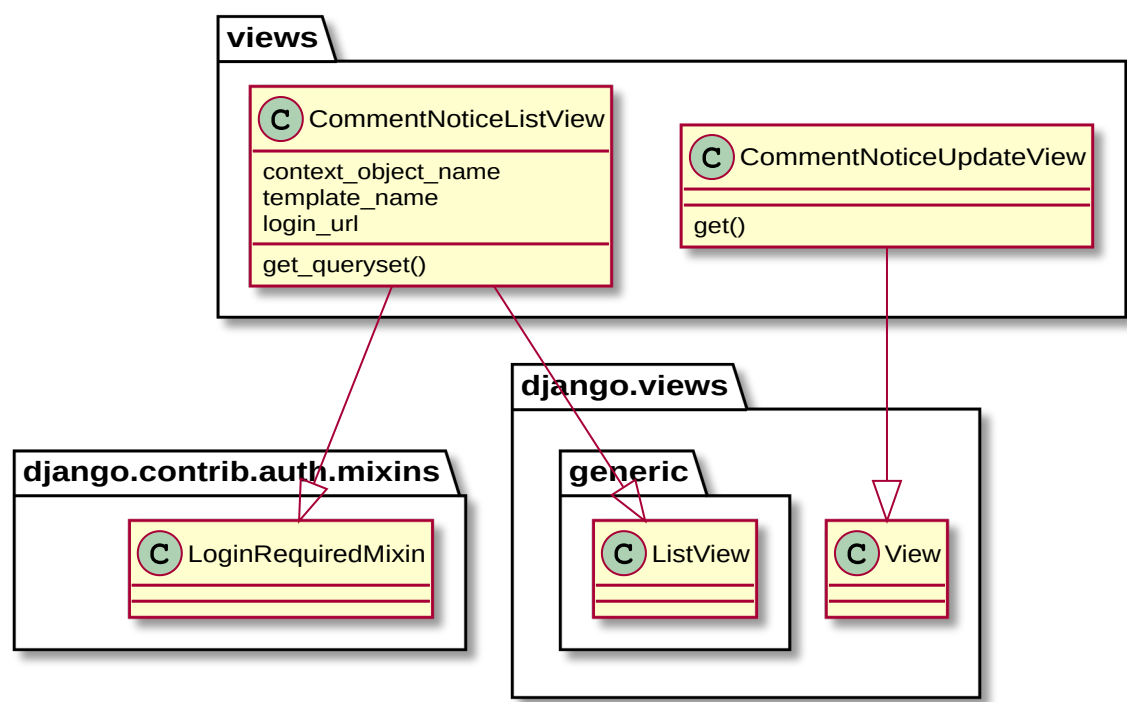
login的类图

login的类图



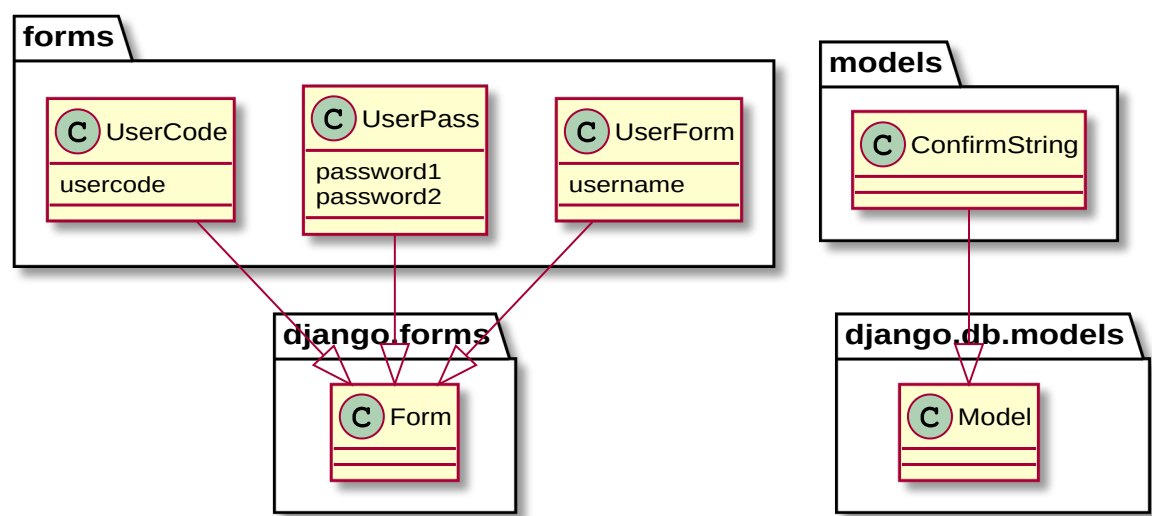
notice的类图

notice的类图



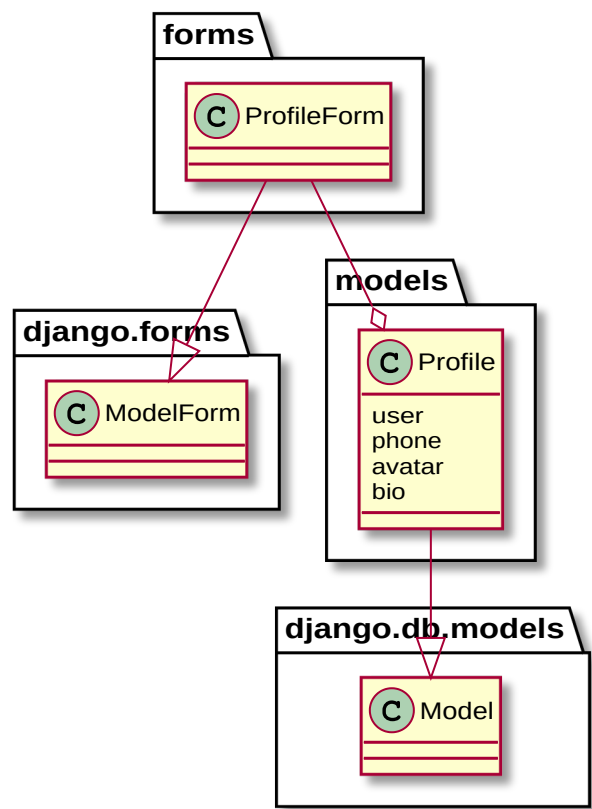
reset_password的类图

reset_password的类图

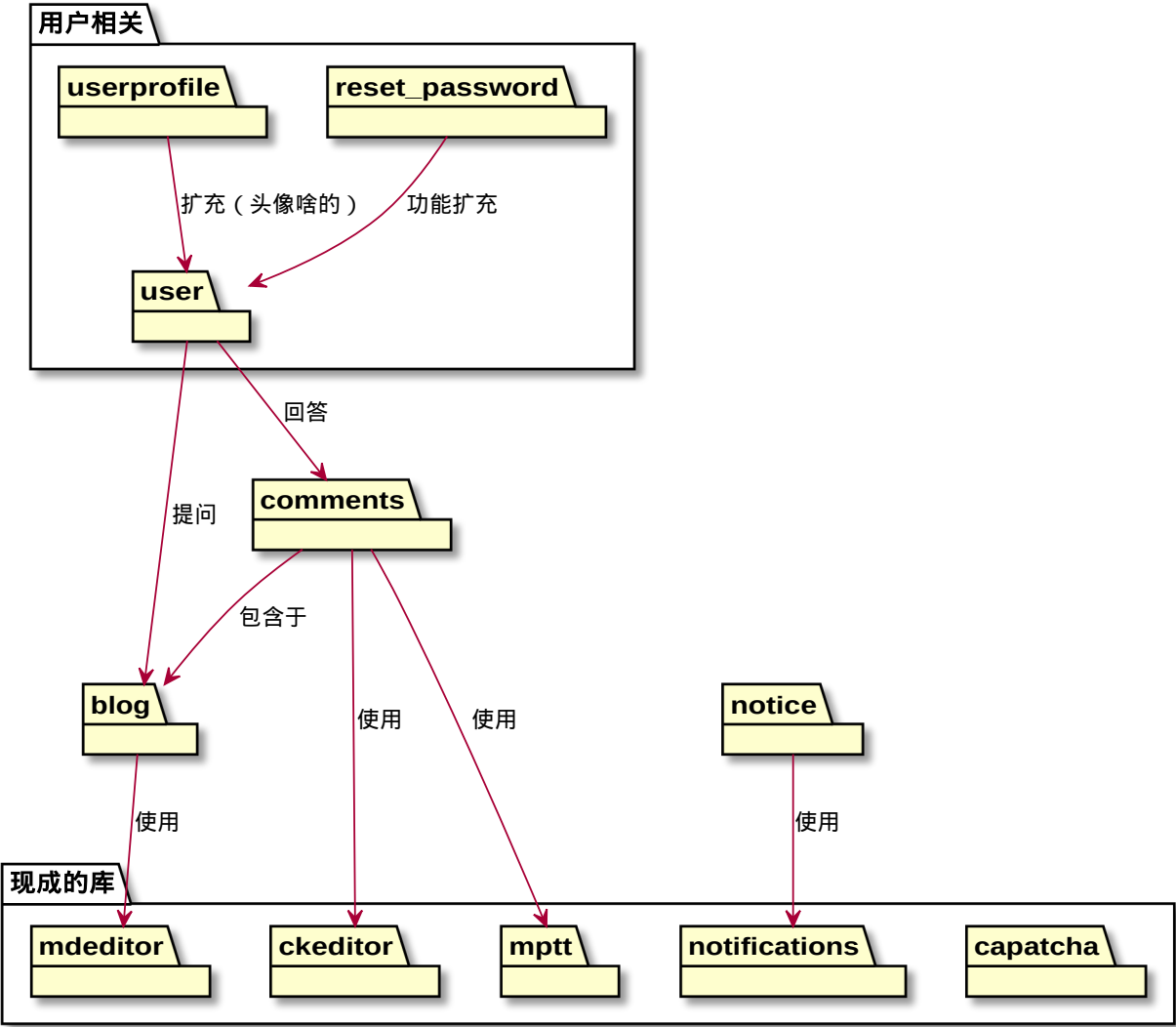


userprofile的类图

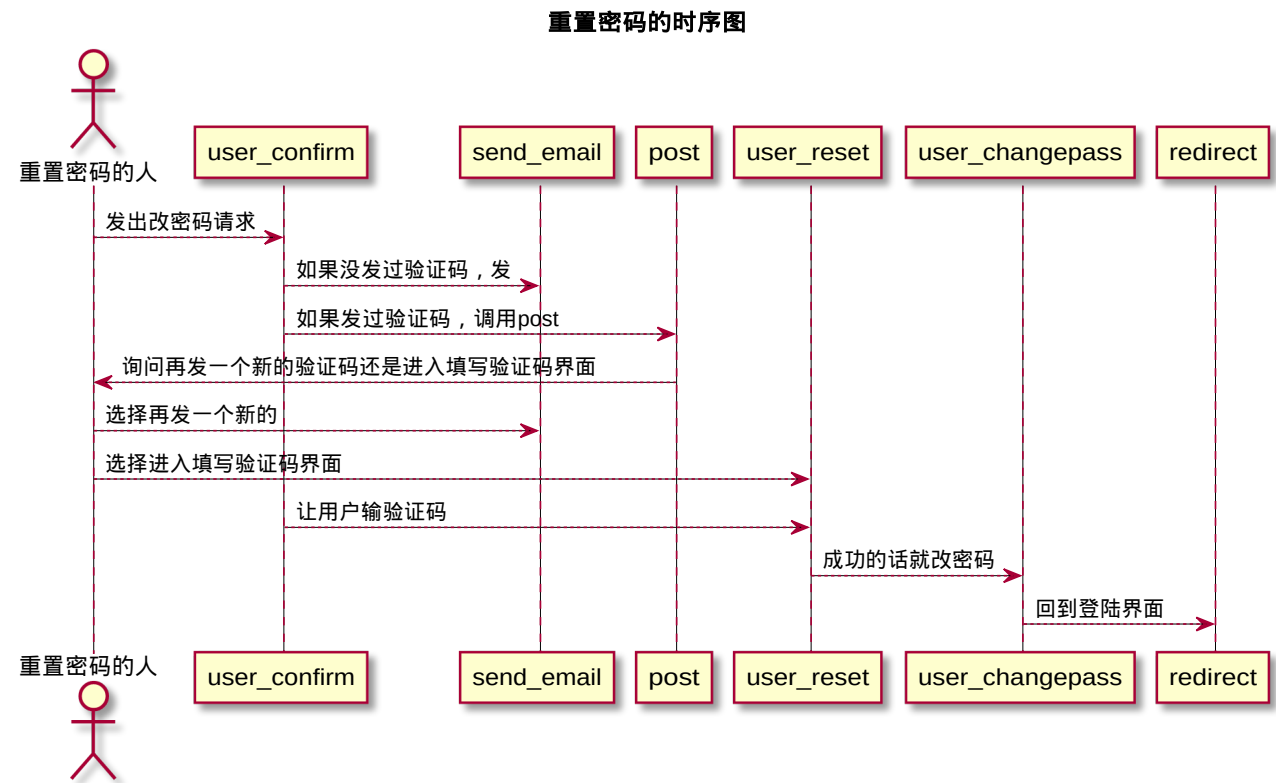
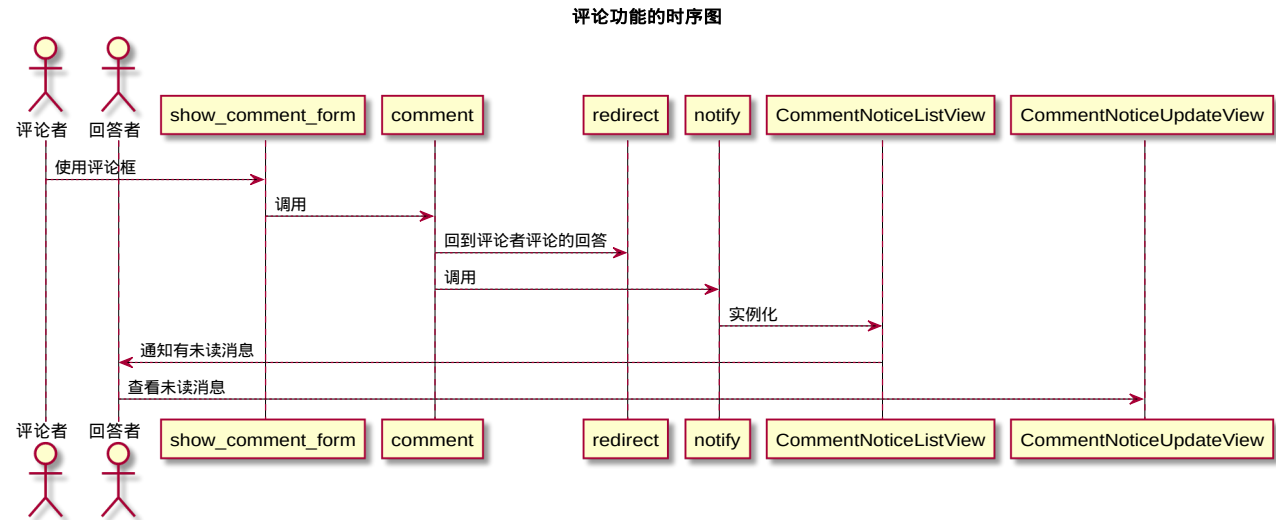
userprofile的类图



包图：

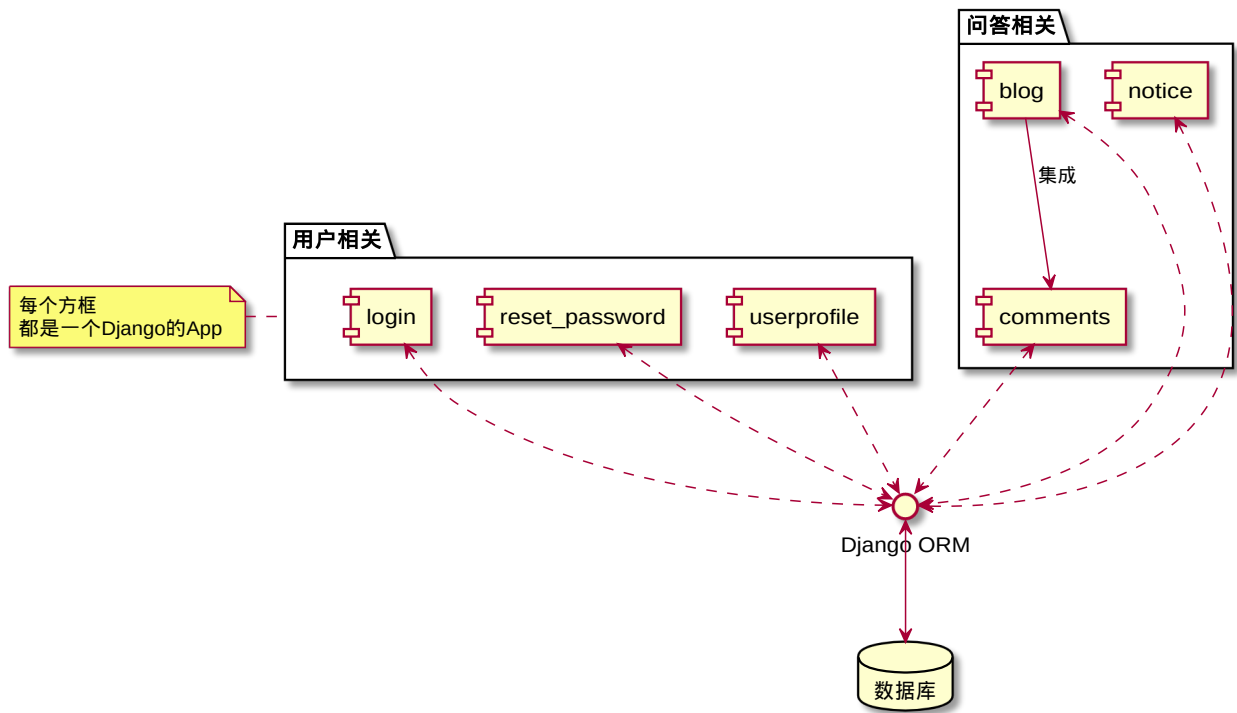


几个典型例子的时序图：



系统设计文档

1. 系统总体架构



我们的程序中每个Django App相当于一个组件，另外引入了一些第三方包，这在之前的图中已经写过了，不再赘述。

各个组件（Django App）都至少有forms, models, views三个文件。其中views主要是函数（没必要封装成一个类）；models里的各个类主要对应于ORM，描述了相关对象应该怎么被存进数据库；forms是models和views之间的桥梁，基本每个models中的类会对应一个models里的类。

由于我们的client端只是运行于浏览器里的web页面，所以不需要单独部署。和server端一起放在租来的服务器上就可以了。

TODO 多服务器？？

2. 技术关键点、解决方案

TODO ???

系统附加说明

1. 软件的运行与开发环境

Django是跨平台的，理论上Windows、Linux和OS X都可以运行。但实际使用时，因为中文版的Windows默认文件为gbk编码，但有一个依赖的安装脚本用utf-8读取文件，导致没法装那个包。因此本软件不能在Windows上使用。Linux与OS X都可以用，这是经过测试的。

2. 软件的安装与卸载方法

首先推荐用conda创建一个新环境，把用来运行本软件的pyhon和系统用的python分别开来

```
conda create -n shihu python
conda activate shihu
```

然后安装所需依赖

```
pip install -r requirements.txt
```

创建数据库

```
python manage.py makemigrations  
python manage.py migrate --run-syncdb
```

启动

```
python manage.py runserver
```

卸载

```
conda remove -n shihu --all  
cd ..  
rm -rf shihu/
```

本系统的实际开发记录

1. 具体开发活动

- TODO 之前，尝试ror...
- 12.8 上传到github上，给评论加上markdown渲染功能，增加创建问题功能
- 12.9 增加删除问题功能，添加jQuery插件，增加编辑问题功能（指创建之后的修改），更正页面间链接
- 12.10 增加密码找回功能（利用腾讯邮箱的smtp服务）
- 12.11 创建修改用户信息的Django app
- 12.13 给创建、修改文章加上实时预览的markdown输入框
- 12.18 画好用例图，用来告知大家还需要完成什么功能
- 12.25 增加个人信息主页功能，写清楚依赖文件以便其他同学安装，增加删除用户功能（模仿欧盟的GDPR）
- 12.26 增加评论嵌套的支持（TODO 是么？），完成评论功能
- 12.27 增加通知功能
- 1.2 重构两个极端相似的类，并且画出报告要用的类图、层次图、包图、顺序图
- 1.3 增加简单的测试（其实应该最开始写的），使用github action + 配置文件来完成push到主分支、pr的自动测试
- 1.8 修改创建文章的标签功能为用户输入（之前只能从已有标签选择）

2. 项目组员分工表

- 崔程远 Django主力开发

- 苑浩然 组长、Django辅助
- 宁昭铤 Web端美化（Vue、jQuery之类的）
- 张秀程 前端

成员表现：崔程远 = 苑浩然 >> 宁昭铤 = 张秀程