## 遗留问题：

1. indexController中，如果没有cookie的已注册用户登录，将会重新再数据库中插入用户，但实际是同一用户（已解决）
2. 使用spring拦截器样式失效？？（WebConfig注解问题）
3. Mybatis Generator：

小匠改为mapper，而我是model却没有问题用了

mybatis generator之后question.description消失了

（改用selectByExampleWithBLOBsWithRowbounds）

## 类关系：

Question+User=QuestionDTO

n\*QuestionDTO=QuestionDTOs(list)

QuestionDTOs(list)+page’s information=PaginationDTO

## 安装运行

1. git clone ${address}
2. mvn flyway:migrate -Pdev 运行数据库脚本，创建数据库文件
3. mvn clean compile package 打出可运行的jar包
4. java -jar target/community-0.0.1-SNAPSHOT.jar 运行该jar包
5. 浏览器访问localhost:8887

## 项目结构介绍

1. .gitignore git忽略文件，commit时不提交的文件
2. 目录结构大致如下（可能有细微区别）



1. springboot项目，在Application.class启动类同级或下级带注解的类，会自动交给spring容器进行管理

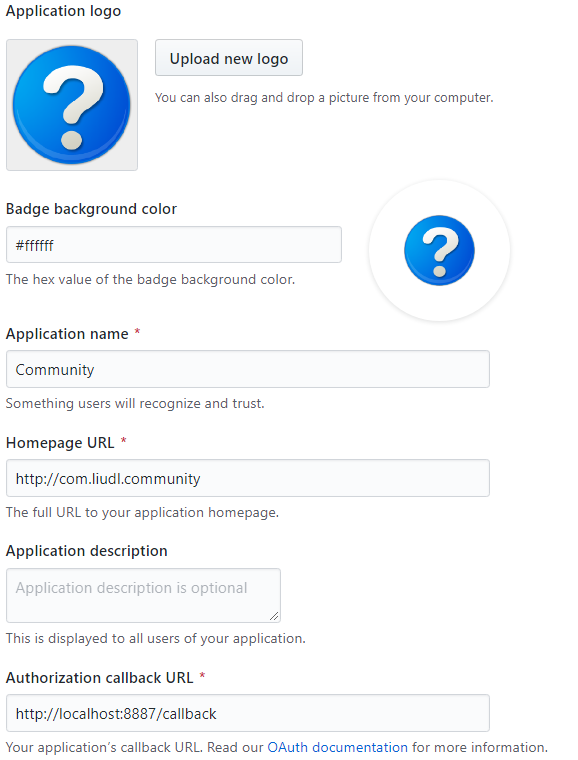
疑问：idea创建spring项目时默认使用了maven，springboot和git？

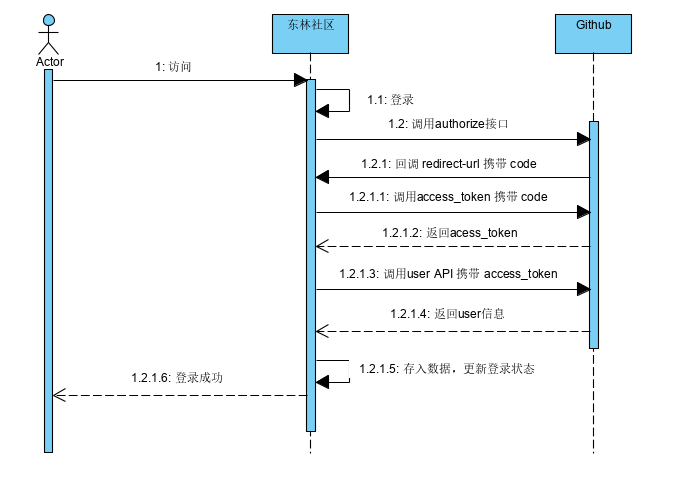
## Bootstrap快速布局导航栏

1. <https://v3.bootcss.com/getting-started/>下载前端资源包并copy到resources/static目录
2. index.html的head中引入css和js
3. copy<https://v3.bootcss.com/components/>中导航条的html代码到index.html并修改

## 注册Github app

1. github网页底API找OAuth Apps，按照说明注册一个OAuth Apps



1. 登录时序图

access\_token:4c25c314dc0e19d58acb383e141a5a1dd8787fbb

index.html 登录向github发起请求，得到code,state并请求callback

AuthorizeController.callback 将得到的code,state以及注册的github OAuthx信息封装到AcessTokenDTO，将该对象传递给GithubProvider. getAccessToken。得到access\_token。

GithubProvider.GithubUser(access\_token)得到user信息

## 不同环境读取不同配置文件

1. properties中给定值
2. 在相应类中用@Value注入properties的值

spring启动时会把application.properties中的键值对一起创建出一个map

## session cookie

session 会话（存有用户以及客户端的信息）

cookie 钥匙（连接特定对话）

P14内容：将user存到session中，index页面可取值显示，使用thymeleaf模板

## H2数据库

pom.xml导入H2包

idea datebase菜单栏创建H2数据库（版本和导入的包不一样，不知为何）

创建表

application.properties中写好连接url user password

spring默认有HikariPool连接池

## Mybatis连接Springboot

驱动及方法:

<http://mybatis.org/spring-boot-starter/mybatis-spring-boot-autoconfigure/>

spring默认有连接池可以使用

<https://docs.spring.io/spring-boot/docs/2.2.0.RC1/reference/htmlsingle/>

## 登录成功存入用户数据

github返回user信息存入githubUser，再将其信息存入model层User，并插入数据库

## 持久化登录状态获取

cookie中存的token（session？）是否和数据库中user的token一致来决定是否自动登录

## Flyway Migration

多人协作时，对数据库的操作不能修改任何已产生的sql

只能新增语句文件来执行

**https://flywaydb.org/getstarted/why**

## 完成文章发布

如果是get就渲染页面，post就执行请求？？

发布按钮以post方式路由到PostMapping(/publish)

页面回显，保证用户已填写信息不丢失

验证填写信息是否空，验证是否登录，插入数据库

## Lombok

@Data 自动生成setter，getter，to string()等方法

## 首页列表

mybatis解决驼峰命名与数据库命名不映射的问题’

mybatis.configuration.map-underscore-to-camel-case=true

questionDTO(question,user)对象创建，分别question和对应的user set进questionDTO

## 热部署

JReble / Spring Dev tool

1. 引入包
2. settings,compiler,build project automatically
3. ctrl shift alt ?, registry, compiler.automake.allow.when.app.running

## 分页功能

每一页起点：每页行数\*（页码-1）

总页数：总行数/每页行数 or 总行数/每页行数+1

PaginationDTO囊括原有的questionDTO类，并附加页码信息。

将传入方法的参数赋值给类的同名变量后（this.page = page），在方法中使用的什么参数？

（当参数与类变量同名，必须用this可调用类变量）

PaginationDTO.setPagination()方法中page大于totalPage的处理晚了，应在用page计算offset之前确定

（已解决，详见代码）

## 通过源码分析不能加载资源问题

WebMvcConfigurer. addResourceHandlers

1.

WebMvcAutoConfiguration中ResourceProperties.staticLocations：设置资源默认路径

@ConditionalOnMissingBean(WebMvcConfigureationSupport.class)

疑问：虽然WebMvcAutoConfiguration实现了addResourceHandlers方法，但是却未实现WebMvcConfigurer接口

2.

WebMvcConfigurerComposite.delegates通过addWebConfigurers添加

而该方法被DelegatingWebMvcConfiguration调用

DelegatingWebMvcConfiguration.setConfigurers(List<WebMvcConfigurer> configurers)其中参数是WebConfig（自写）的父类。当前类继承自WebMvcConfigureationSupport

@EnableWebMvc(@Import(DelegatingWebMvcConfiguration.class))

## 修复登录问题

创建UserService.createOrUpdate来代替简单的UserMapper.insert功能

## Mybatis Generator

小匠改为mapper，而我是model却没有问题

用了mybatis generator之后question.description消失了

## 异常处理

为什么地址访问questions/n n比较小和比较大时得到的异常不一样

[https://docs.spring.io/spring-boot/docs/2.2.0.RC1/reference/htmlsingle/#boot-features-error-handling](https://docs.spring.io/spring-boot/docs/2.2.0.RC1/reference/htmlsingle/" \l "boot-features-error-handling)

白页问题

业务异常

4xx和5xx异常

## 增加阅读数

sql语句只会一条一条的执行

所以在数据库中set viewCount = viewCount + 1

可以避免高并发导致的漏加问题

## 后台实现回复功能

异常：

CustomizeErrorCode：定义所有能想到的业务逻辑异常的code和message

CustomizeException：接收CustomizeErrorCode的message和code，继承RuntimeException

CustomizeExceptionHandler:处理能捕获到的异常，分为json和html两个部分。

CustomizeErrorController：该类处理CustomizeExceptionHandler不能处理的异常或错误

回复：

ResultDTO：返回前端的对象类，有接收各种参数的静态构造方法。

思考：

既然本节的重点之一在于将异常或错误一json的方式返回，那何必throw Exception呢，这样还要在CustomizeErrorController中将接收到的异常转化为ResultDTO,为何不将CommentService.insert方法返回值设为ResultDTO,而在碰到问题时直接return ResultDTO.(CustomizeErrorCode)

## 回复功能-事务

@Transactional