springboot\_h5开发框架说明

## 一：导出项目

1. 在svn中导出springboot\_h5项目到本地
2. 断开svn链接（同时**删除svn链接**信息）
3. **重命名**工程项目名称

## 二：微信授权

1. Springboot\_h5框架已经默认自带了微信授权认证功能，只要在application-prod.properties或application-dev.properties文件中修改**wechat.appId和wechat.appSecret**的值即可。**wechat.auth.callback**如果正式上线，其值可以为空。也可以自己设置。
2. 微信授权的实现原理是：添加一个微信拦截器，拦截用户的活动访问，如果拦截到cookie中没有授权信息，则重定向到组织微信授权响应url的action那里，组织好微信回调url及重定向url后，调用微信官方的授权认证接口。微信授权完成后，会回调之前设定的回调url并回传授权结果及重定向url。把授权结果写入cookie，然后重定向到一开始访问的页面。
3. 如果**不**需要进行**微信授权**，则只要在**config**包下的**AppWebConfig**类的public void addInterceptors(InterceptorRegistry registry)方法中**注释掉addWxAuthInterceptor**(registry);这行代码即可。

## 三：不使用session而cookie的使用

1. 由于h5的活动经常是访问量比较大，需要做负载均衡。为了防止负载均衡后session不同步的问题，统一采用cookie。
2. 注意cookie中保存的信息**不能超过2KB**，如果需要保存更大的数据，请使用redis缓存。

## 四：数据库Dao操作使用apache的DBUtils

1. 为了便于后续的维护，Dao内的数据库操作统一使用我们自己对apache DBUTils包装好的**DbOperator**进行。具体使用可以参考其使用说明文档。
2. DbOperator的配置文件默认是**config.properties**，如果有特殊情况，可以自己修改为其他配置文件名称。

## 五：页面模版使用freemarker

1. 为了减少开发和学习成本，页面模版我们默认使用**freemarker**，不推荐使用springboot自带的**thymeleaf**
2. 如果确实有特别原因，也可以采用其他页面模版，但是尽量使用freemarker。

## 六：静态资源的存放

1. 图片，音乐，视频，css，js等静态资源，一般情况下放到**oss**中。如果确定访问量不大的情况，可以放在本地服务器（静态资源和程序在同一台服务器）。
2. 图片，音乐，视频的存放oss的路径在页面模版的common目录的base.html中设置和修改，即**itx**变量。
3. Css和js的存放oss的路径页面模版的common目录的base.html中设置和修改，即**rtx**变量。
4. 页面模版的项目路径在common目录的base.html中设置和修改，即**ctx**变量。

## 七：开发注意点

1. 所有的**Controller**都需要继承**BaseController**，同理，所有的Service要继承**BaseService**，Dao要继承**BaseDao**。
2. 没有特殊情况，注入的使用最多到Service层。一般情况下，**Service**类内部**不要**使用**自动注入**。

## 八：OSS接口上传图片及视频

1. ……….