

Spring Boot 集成 PageOffice

开发环境：JDK1.8、Eclipse、Spring Boot + Thymeleaf 框架。

一. 构建 Spring Boot + Thymeleaf 框架的项目(不再详述):

1. 新建一个 maven project 项目：demo。
2. 修改 pom.xml 配置，把项目配置为 Spring Boot 项目；
3. 配置 Thymeleaf:
添加 Thymeleaf 依赖，并在 application.properties 文件中添加 Thymeleaf 的配置；、
4. 新建 DemoController，添加 showWord、showIndex 方法：

```
@RequestMapping(value="/word", method=RequestMethod.GET)
public ModelAndView showWord(HttpServletRequest request,
Map<String, Object>map){

    ModelAndView mv = new ModelAndView("Word");
    return mv;
}

@RequestMapping(value="/index", method=RequestMethod.GET)
public ModelAndView showIndex(){
    ModelAndView mv = new ModelAndView("Index");
    return mv;
}
```
5. 新建 Thymeleaf 模板页：Word.html、Index.html；
6. 运行 demo 项目，并成功访问：http://localhost:8080/index

二. 集成 PageOffice

1. 在 pom.xml 中添加 PageOffice 的依赖：

```
<!-- 添加Sqlite依赖（可选：如果不需要使用印章功能的话，不需要添加此依赖）-->
<dependency>
<groupId>org.xerial</groupId>
<artifactId>sqlite-jdbc</artifactId>
<version>3.7.2</version>
</dependency>
<!-- 添加PageOffice依赖（必须） -->
<dependency>
```

```

<groupId>com.zhuozhengsoft</groupId>
<artifactId>pageoffice</artifactId>
<version>4.4.0.4</version>
</dependency>

```

2. 在 application.properties 文件中添加两个自定义参数配置，**posyspath**：指定一个磁盘目录用来存放 PageOffice 注册成功之后生成的 license.lic 文件；**popassword**：设置 PageOffice 自带印章管理程序的登录密码；以备给 PageOffice 的服务器端 Servlet 程序使用：

```

#####
###PageOffice
#####
posyspath=d:/lic/
popassword=111111

```

3. 在 DemoController 中添加代码获取上一步在 application.properties 中定义的两个参数：

```

@Value("${posyspath}")
private String poSysPath;
@Value("${popassword}")
private String poPassWord;

```

4. 在 DemoController 中添加 PageOffice 的 Servlet 的注册代码：

```

/**
 * 添加PageOffice的服务器端授权程序Servlet（必须）
 * @return
 */
@Bean
public ServletRegistrationBean servletRegistrationBean() {
    com.zhuozhengsoft.pageoffice.poserver.Server poserver = new
com.zhuozhengsoft.pageoffice.poserver.Server();
    //设置PageOffice注册成功后,license.lic文件存放的目录
    poserver.setSysPath(poSysPath);
    ServletRegistrationBean srb = new ServletRegistrationBean(poserver);
    srb.addUrlMappings("/poserver.zz");
    srb.addUrlMappings("/posetup.exe");
    srb.addUrlMappings("/pageoffice.js");
    srb.addUrlMappings("/jquery.min.js");
    srb.addUrlMappings("/pobstyle.css");
    srb.addUrlMappings("/sealsetup.exe");
    return srb;
}

/**
 * 添加印章管理程序Servlet（可选）

```

```

    * @return
    */
@Bean
public ServletRegistrationBean servletRegistrationBean2() {
    com.zhuozhengsoft.pageoffice.poserver.AdminSeal adminSeal = new
com.zhuozhengsoft.pageoffice.poserver.AdminSeal();
    adminSeal.setAdminPassword(poPassWord);//设置印章管理员admin的登录密码
    adminSeal.setSysPath(poSysPath);//设置印章数据库文件poseal.db存放目录
    ServletRegistrationBean srb = new ServletRegistrationBean(adminSeal);
    srb.addUrlMappings("/adminseal.zz");
    srb.addUrlMappings("/sealimage.zz");
    srb.addUrlMappings("/loginseal.zz");
    return srb;
}

```

5. 在 DemoController 的 showWord 方法中添加创建 PageOfficeCtrl 对象的代码，其中 WebOpen 方法的第一个参数是 office 文件在服务器端的磁盘路径，在此 demo 中暂时使用常量：d:\\test.doc

```

@RequestMapping(value="/word", method=RequestMethod.GET)
public ModelAndView showWord(HttpServletRequest request,
Map<String, Object> map){
    //--- PageOffice的调用代码开始 -----
    PageOfficeCtrl poCtrl=new PageOfficeCtrl(request);
    poCtrl.setServerPage("/poserver.zz");//设置授权程序servlet
    poCtrl.addCustomToolButton("保存","Save",1);//添加自定义按钮
    poCtrl.setSaveFilePage("/save");//设置保存的action
    poCtrl.webOpen("d:\\test.doc",OpenModeType.docAdmin,"张三");
    map.put("pageoffice",poCtrl.getHtmlCode("PageOfficeCtrl1"));
    //--- PageOffice的调用代码结束 -----
    ModelAndView mv = new ModelAndView("Word");
    return mv;
}

```

6. 在 Word.html 中添加 PageOffice 客户端控件所在的 div 和 js 代码：

```

<div style="width:1000px;height:700px;"
th:utext="${pageoffice}"></div>
<script type="text/javascript">
function Save() {
document.getElementById("PageOfficeCtrl1").WebSave();
}
</script>

```

7. 在 Word.html 中添加 pageoffice.js 和 jquery.min.js 的引用，并添加打开文件的超链接：

```

<script type="text/javascript" src="jquery.min.js"></script>
<script type="text/javascript" src="pageoffice.js" id="po_js_main">
</script>
<a
href="javascript:POBrowser.openWindowModeLess('/word','width=1200px;height=800px;');">打开文件</a>

```

- 在 DemoController 添加 saveFile 方法，用来接收 PageOffice 客户端上传的文件流并保存到服务器指定磁盘目录，在此 demo 中暂时使用常量：d:\

```

@RequestMapping("/save")
public void saveFile(HttpServletRequest request, HttpServletResponse
response){
    FileSaver fs = new FileSaver(request, response);
    fs.saveToFile("d:\\\" + fs.getFileName());
    fs.close();
}

```

- 在 d 盘根目录下准备一个 test.doc 文件（不要用 0 字节的文件）以备测试；
- 运行 demo 项目，访问：<http://localhost:8080/index> 点击“打开文件”的超链接即可在线打开、编辑和保存文件。

提示注册时，使用 PageOffice V4.0 专业版试用序列号：**CA1XB-MF7Y-12ST-PSBP2** 注册即可。

- 继续添加印章功能。
拷贝印章数据库文件 poseal.db，到 `adminSeal.setSysPath(poSysPath)`；指向的目录，也就是 `posyspath=d:/lic/` 目录下；

- 添加自定义盖章按钮：

```
poCtrl.addCustomToolButton("盖章","AddSeal",2); //添加自定义盖章按钮
```

- 添加 js 的 AddSeal 方法：

```

<script type="text/javascript">
function AddSeal() {
    try{
        document.getElementById("PageOfficeCtrl1").ZoomSeal.AddSeal();
    }catch (e){ }
}
</script>

```

- 运行 demo 项目，访问：<http://localhost:8080/index>，在线打开文件之后，点击盖章按钮即可盖章。用户名：李志，密码：111111 或 123456

15. 访问 <http://localhost:8080/loginseal.zz>，用户名：admin，密码：111111（也就是 `popassword=111111` 配置的密码），登录后可以添加删除印章。

简易印章管理平台中没有提供用户管理，如果新建印章的时“签章人”中输入的用户名在印章数据库文件 `poseal.db` 的用户表中不存在的话，印章管理平台会自动新建用户，并设置初始密码为：111111；如果新建印章的时“签章人”中输入的用户名已经存在于用户表了，新建的印章就归属于这个用户。