

# production\_ssm

---

production\_ssm

项目简介

项目技术架构(Spring+SpringMVC+Mybatis )

系统架构

数据库设计 ( 详见sql文件 )

软件运行说明

登录界面

运行界面

图片上传

文件上传

富文本编辑

关联信息

search-box查找

session过期跳转登录

动态权限控制

备注

导入项目到STS

文件上传配置

idea classpath配置

Mybatis逆向工程

## 1.项目简介

aproduction\_ssm 是一个生产管理 ERP 系统。依托科技计划重点项目“制造装备物联及生产管理系统研发”，主要包括：计划进度、设备管理、工艺监控、物料监控、人员监控、质量监控、系统管理 7 大模块。

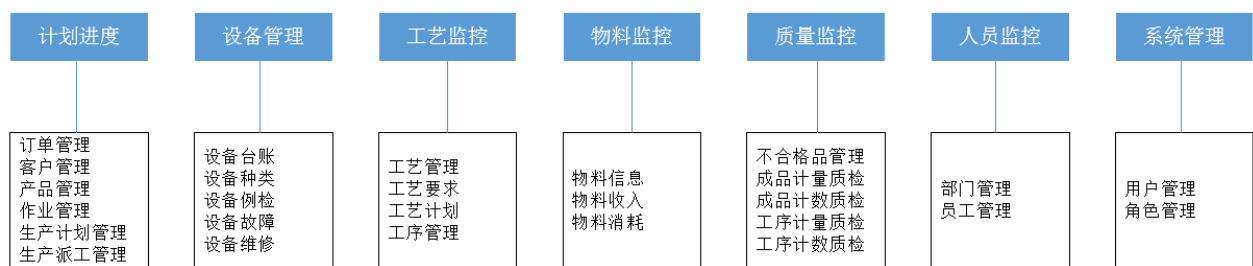
## 2. 项目技术架构

(Spring+SpringMVC+Mybatis )

- Maven

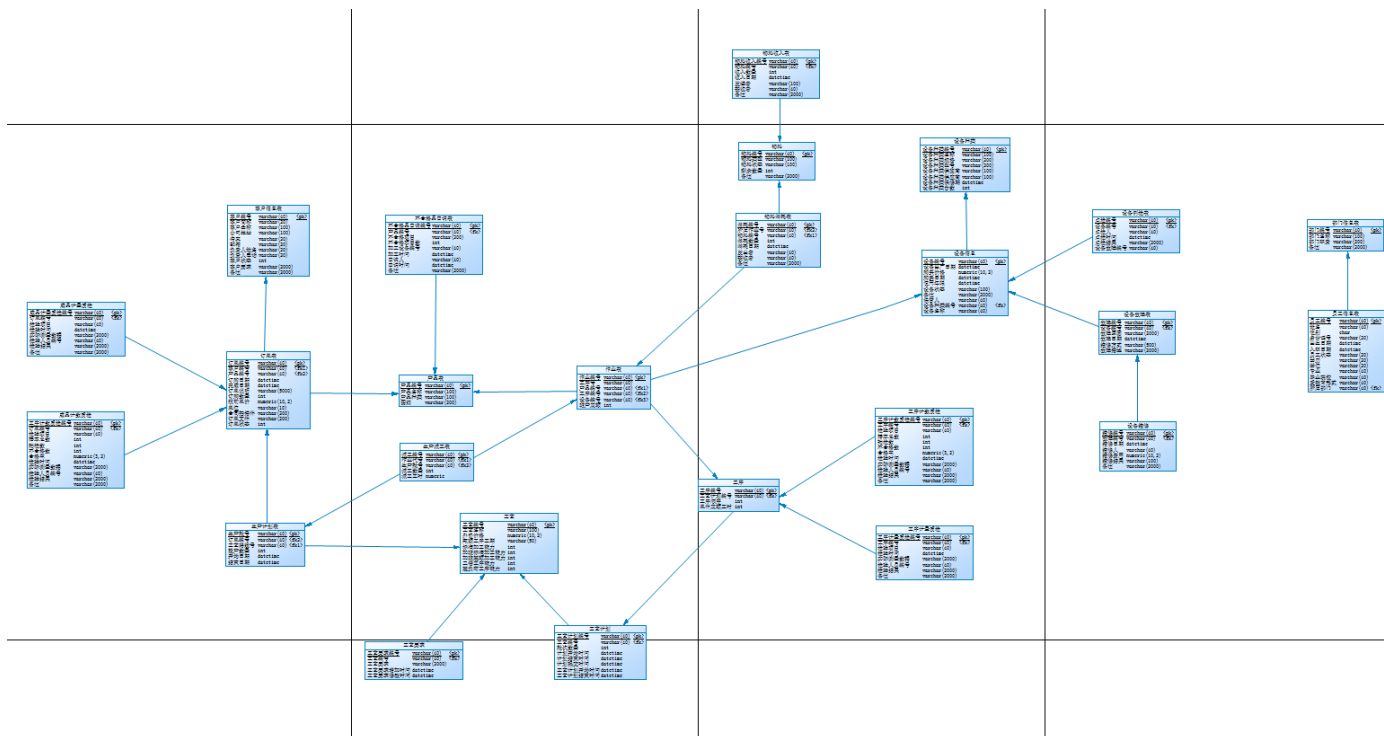
- Spring ( IOC DI AOP 声明式事务处理 )
- SpringMVC ( 支持 Restful 风格 )
- Hibernate Validator ( 参数校验 )
- Mybatis ( 最少配置方案 )
- shiro 权限控制,结合 ajax 实现了异步认证与异步授权,同时实现了细粒度的权限动态分配 ( 到按钮级别 ); 添加了 shiro session 过期的登录跳转
- jQuery EasyUI 开发前端页面,利用 jQuery 文件上传插件实现拖拽上传的效果并对文件类型、大小、数量进行控制;利用 search-box 实现查找功能
- Druid ( 数据源配置 sql 防注入 sql 性能监控)
- 统一的异常处理
- JSP JSTL JavaScript
- kindeditor 富文本编辑器,处理图片上传和富文本编辑

### 3. 系统架构



### 4. 数据库设计

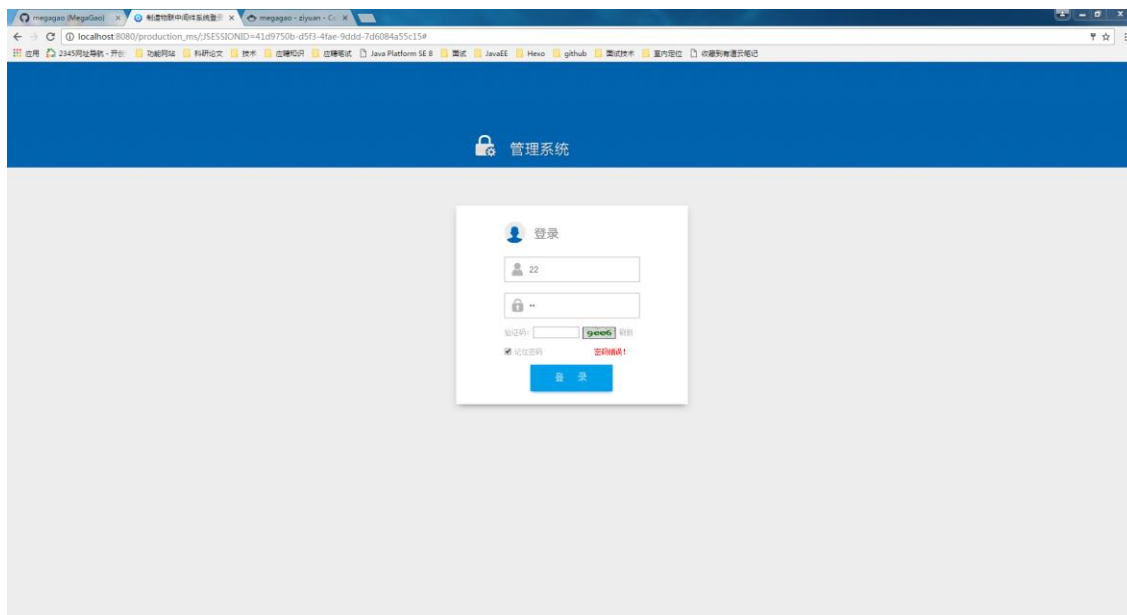
( 详见 sql 文件 )



## 5. 软件部分功能展示

### 登录界面

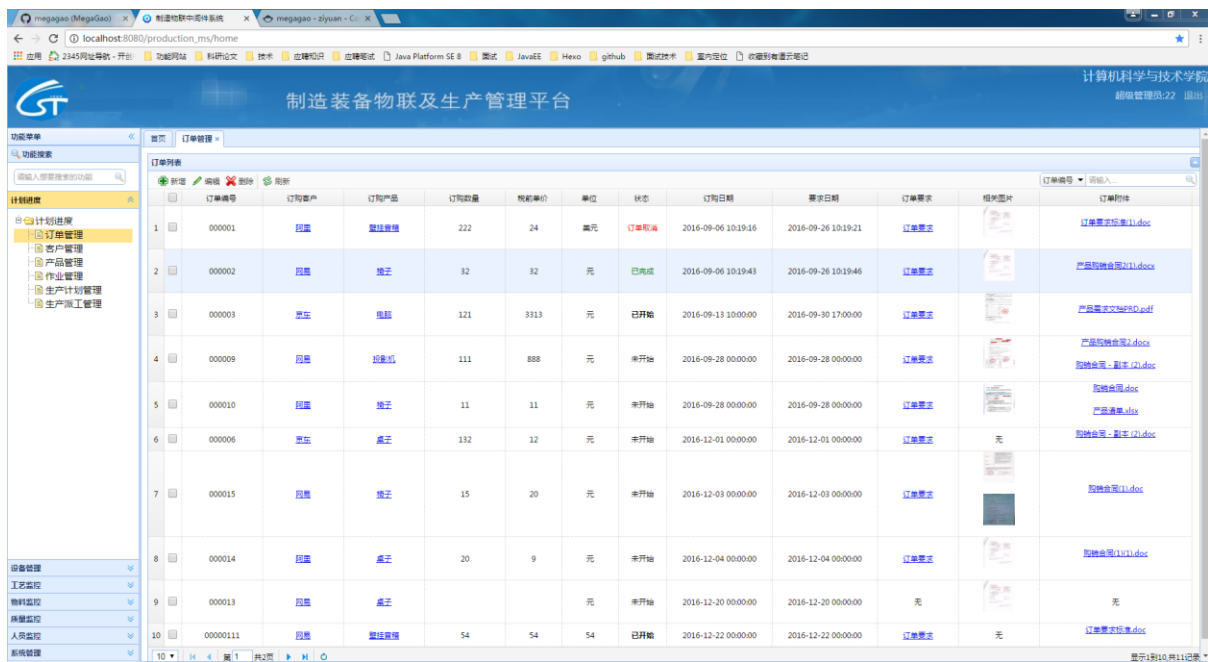
登录可使用账号：22，密码：22 的超级管理员登录，若密码输错，下次登录需输入验证码。



# 运行界面

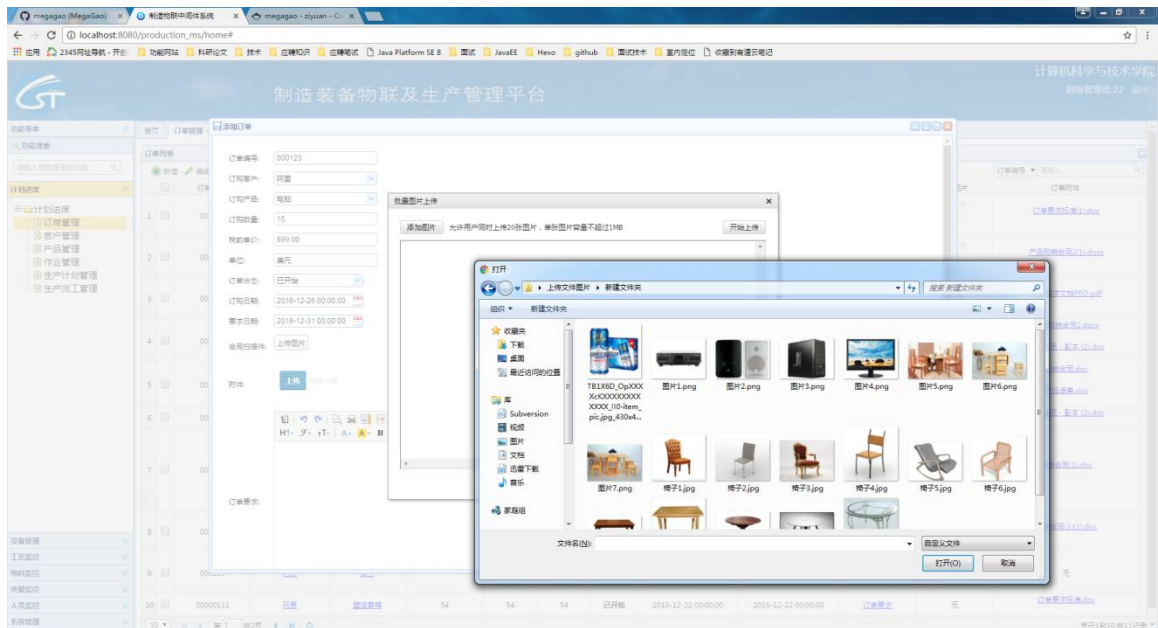
超级管理员可显示系统管理模块进行系统权限分配与管理，其他角色可查看除系统管理外的剩余模块的信息（包括下载附件、查看图片等），但是只能维护该角色对应权限内的信息。

左边功能搜索栏可进行功能模糊查找。



# 图片上传

图片上传的配置请查看文档尾部的注释，图片大小要求不能超过 1M，支持 jpg、png 等多种格式的图片，上传成功后可在相应的展示栏进行回显。

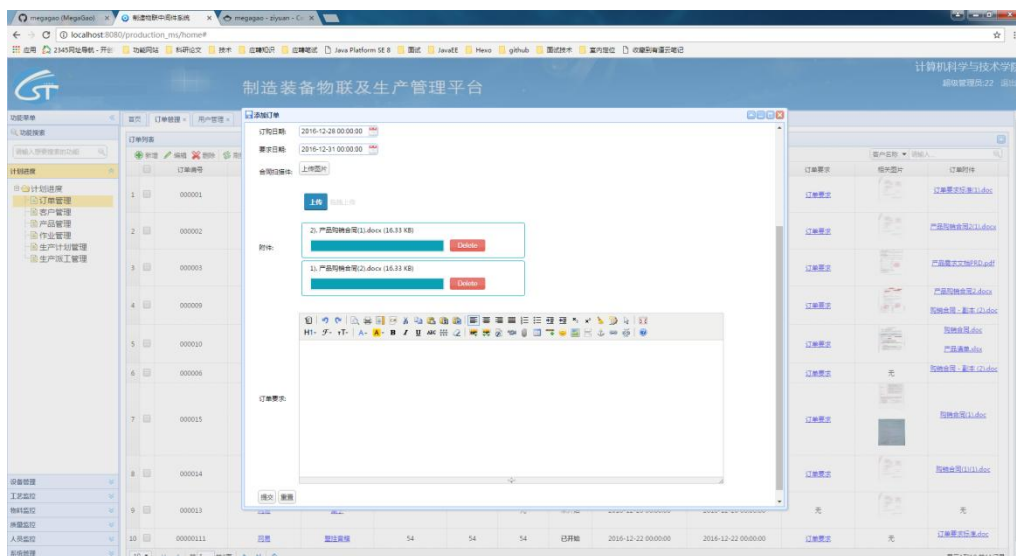


## 文件上传

文件上传使用了一个开源的 jQuery 文件上传插件，可以在 common.js 里面修改上传文件的参数，包括上传个数，支持的文件类型等，配置信息如下：

```
1 url: "file/upload",
2 maxFileCount: 5,           //上传文件个数（多个时修改此处
3   returnType: 'json',      //服务返回数据
4   allowedTypes: 'doc,docx,excel,sql,txt,ppt,pdf', //允许上传的文件式
5   showDone: false,        //是否显示"Done"(完成)按钮
6   showDelete: true,       //是否显示"Delete"(删除)按钮
```

文件上传的界面如下：

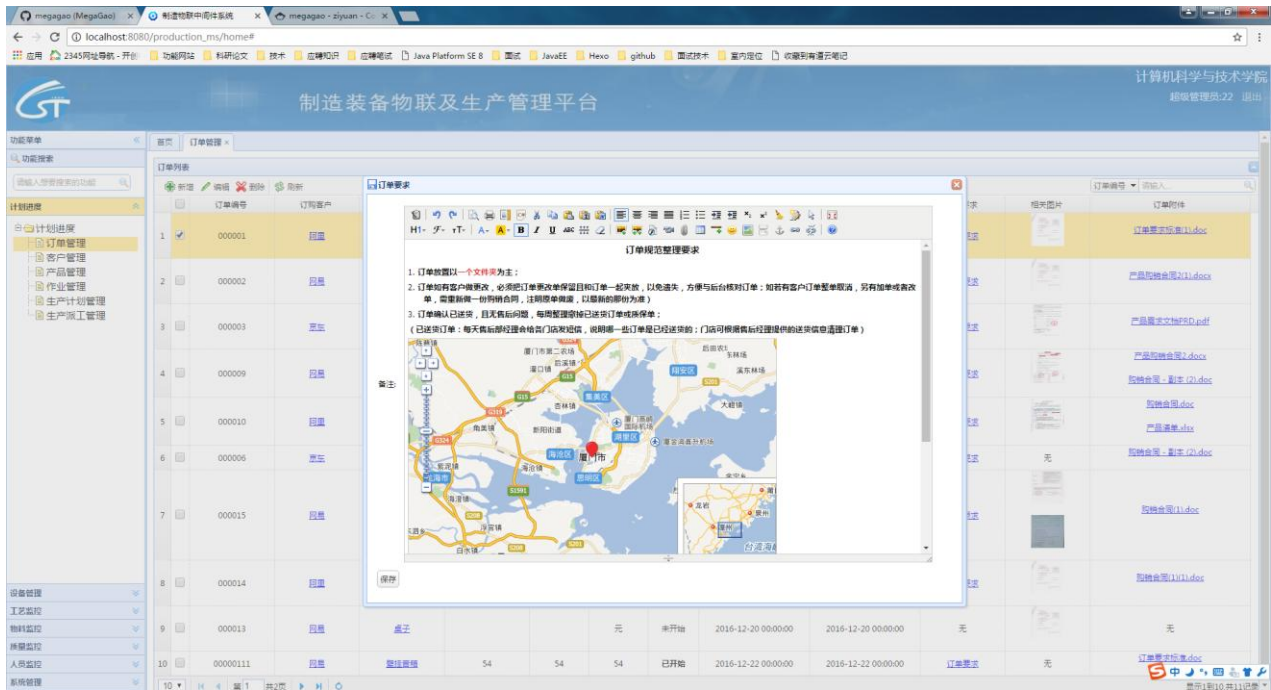


# 富文本编辑

本系统采用了开源的 KindEditor 富文本编辑器，它是一套在线 HTML 编辑器，主要用于让用户在网站上获得所见即所得编辑效果。KindEditor 把传统的多行文本输入框(textarea)替换为可视化的富文本输入框。

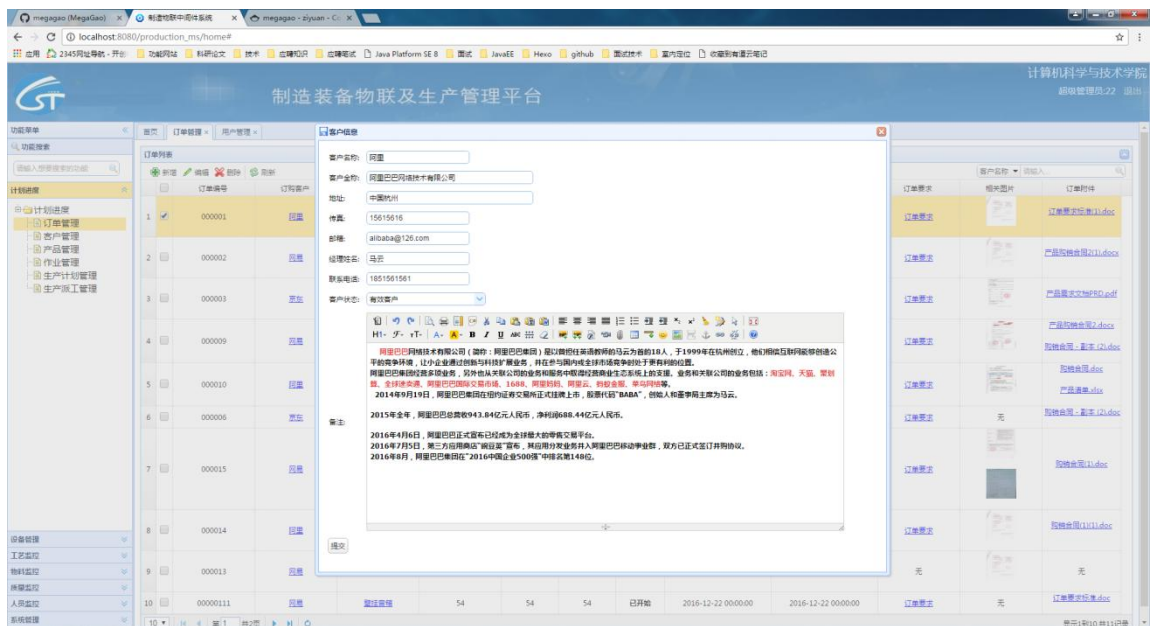
## KindEditor 主要特点

- 快速：体积小，加载速度快
- 开源：开放源代码，高水平，高品质
- 底层：内置自定义 DOM 类库，精确操作 DOM
- 扩展：基于插件的设计，所有功能都是插件，可根据需求增减功能
- 风格：修改编辑器风格非常容易，只需修改一个 CSS 文件
- 兼容：支持大部分主流浏览器，比如 IE、Firefox、Safari、Chrome、Opera



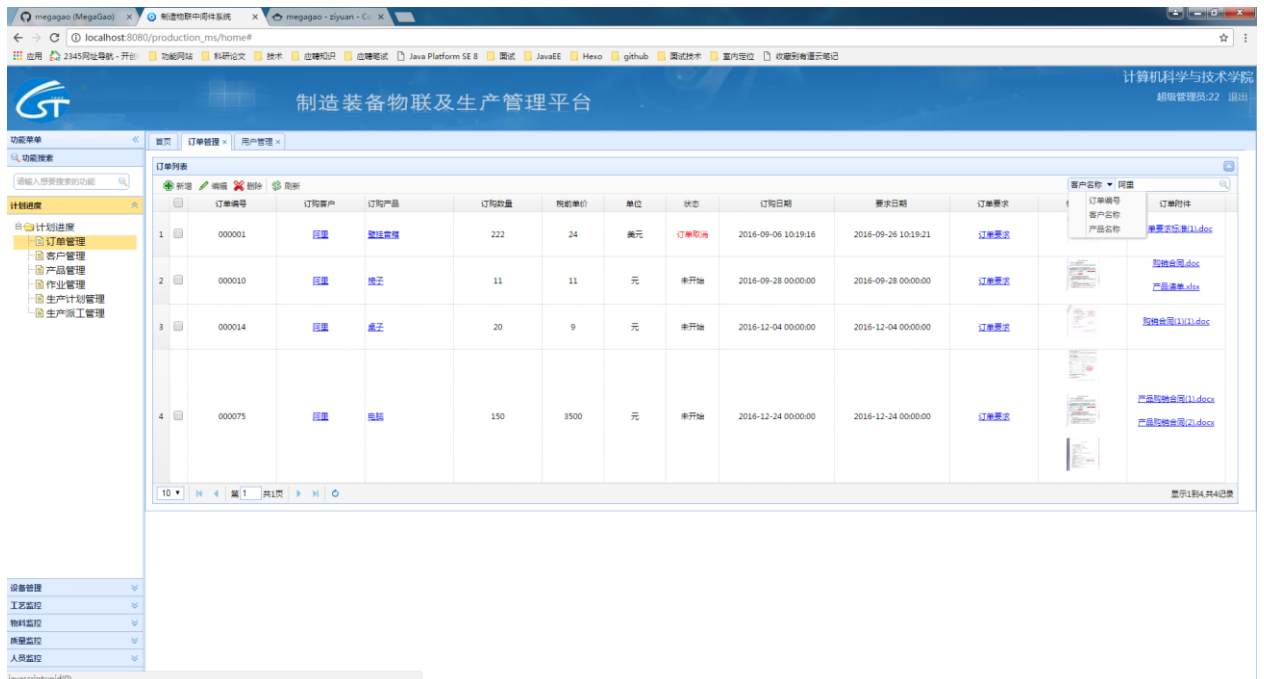
- ## 关联信息

关联对象的信息，点击以弹窗的形式显示，若具有该模块对应的修改权限，则也可在此进行信息维护。



## search-box 查找

可以在右上角的 search-box 选择查询条件，输入关键字进行对应信息的模糊查找。



## session 过期跳转登录

用户登录后，会建立相应的 session，系统默认过期时间为 10 分钟，若需更改，可在 applicationContext-shiro.xml 配置文件中，更改如下配置。



```

1 <!-- 会话管理器 -->
2 <bean id="sessionManager"
3 class="org.apache.shiro.web.session.mgt.DefaultWebSessionManager">
4 <!-- session 的失效时长，单位毫秒，这里设置为 10 分钟-->
5 <property name="globalSessionTimeout" value="600000"/>
6 <!-- 删除失效的 session -->
7 <property name="deleteInvalidSessions" value="true"/>
8 <!-- 指定本系统 sessionId，默认为: JSESSIONID 问题: 与 Servlet 容器名冲突,
9 如 Jetty, Tomcat 等默认 JSESSIONID,
10 当跳出 shiro Servlet 时如 Error-page 容器会为 JSESSIONID 重新分配值
    导致登录会话丢失! -->
    <property name="sessionIdCookie" ref="sessionIdCookie"/>
  </bean>

```

若 session 过期，则应跳转登录界面重新登录。系统采用的方式是设置一个 sessionfilter，如下：

```

1 public class SessionFilter implements Filter {
2
3   public void doFilter(ServletRequest servletRequest, ServletResponse
4 servletResponse, FilterChain filterChain) throws IOException,
5     ServletException {
6
7     HttpServletRequest request = (HttpServletRequest) servletRequest;
8     HttpServletResponse response = (HttpServletResponse) servletResponse;
9     if (!SecurityUtils.getSubject().isAuthenticated()) {
10        //判断 session 里是否有用户信息,且是否为 ajax 请求，如果是 ajax 请求响
11        应头会有，x-requested-with
12        if (request.getHeader("x-requested-with") != null
13            && request.getHeader("x-requested-
14            with").equalsIgnoreCase("XMLHttpRequest")) {
15            response.setHeader("session-status", "timeout");//在响应头设置 session 状
16            态
17        }
18    }
19    filterChain.doFilter(request, servletResponse);
20 }
21
22 @Override
23 public void destroy() {
24    // TODO Auto-generated method stub
25 }

```



```

26
27 @Override
    public void init(FilterConfig arg0) throws ServletException {
        // TODO Auto-generated method stub
    }
}

```

并在 web.xml 中配置该 filter。当 session 过期时，在响应头设置 session 状态为 timeout，然后采用全局的 ajax 事件对 session 状态进行判断，跳转到登录页面。该事件定义在 common.js 中，如下：

```

//全局 ajax 事件，处理 session 过期跳转登录
1$.ajaxSetup({
2    complete:function(XMLHttpRequest,sessionStatus){
3        var sessionstatus = XMLHttpRequest.getResponseHeader("session-
4status");
5        if(sessionstatus=="timeout"){
6            $.messager.alert('提示信息','登录超时！请重新登录！',
7'info',function(){
8                window.location.href = 'login';
9            });
10       }
11   }
12   });

```

GT 制造装备物联及生产管理平台

客户列表

序号	客户编号	客户名称	客户全称	地址	传真	邮箱	经理姓名	联系电话	客户状态	操作
1	001	阿里	阿里巴巴网络技术有限公司	中国杭州	15615616	alibaba@126.com	马云	1851561561	有效客户	<a href="#">查看详情</a>
2	002	网易	杭州网易传媒有限公司	杭州东工区	4214324	2432432	丁磊	32143214	有效客户	<a href="#">查看详情</a>
3	003	京东	北京京东世纪贸易有限公司	中国北京市朝阳区北京西路8号北京世纪中心A座	5001363	jd@126.com	刘强东	1563156456	有效客户	<a href="#">查看详情</a>
4	004	新浪	新浪网络技术股份有限公司	北京市北四环西路58号建外SOHO大厦20层	15612561	15611561@163.com	曹国伟	15614145656	有效客户	<a href="#">查看详情</a>
5	005	美团	北京三快在线科技有限公司	北京市朝阳区望京东路6号 望京国际开发园三期	15156156156	156115115@126.com	王兴	1561511151	无效客户	<a href="#">查看详情</a>
6	006	蒙牛	内蒙古蒙牛乳业集团	呼和浩特	321321321	3213213@111	3132	3213	有效客户	<a href="#">查看详情</a>
7	007	百度	北京百度网讯有限公司	北京市海淀区中关村科技园二期百度大厦	156151	561455@baidu.com	李彦宏	1561561511	有效客户	<a href="#">查看详情</a>
8	1253	aaa							有效客户	<a href="#">查看详情</a>

提示信息: 登录超时！请重新登录！

确定

## 文件上传配置

在本地建立上传图片和文件的文件夹，如我的存放路径是在 D:\upload\temp\img，D:\upload\temp\file 文件夹下，然后修改 tomcat 的配置文件 server.xml，添加虚拟路径，将对图片和文件的请求 url 映射到本地硬盘的相应路径，如下：

```
<Host name="localhost" appBase="webapps"
  unpackWARs="true" autoDeploy="true">
1   <!-- SingleSignOn valve, share authentication between web applications
2       Documentation at: /docs/config/valve.html -->
3   <!--
4   <Valve className="org.apache.catalina.authenticator.SingleSignOn" />
5   -->
6
7   <!-- Access log processes all example.
8       Documentation at: /docs/config/valve.html
9       Note: The pattern used is equivalent to using pattern="common" -->
10  <Valve className="org.apache.catalina.valves.AccessLogValve"
11      directory="logs"
12      prefix="localhost_access_log" suffix=".txt"
13      pattern="%h %l %u %t &quot;%r&quot; %s %b" />
14
15  <!-- 在 Host 标签下添加下面两行，配置虚拟路径到你本机的文件夹 -->
16  <Context path="/pic" docBase="D:\upload\temp\img" crossContext="true"
17      trusted="true" reloadable="true"/>
18  <Context path="/file" docBase="D:\upload\temp\file" crossContext="true"
19      trusted="true" reloadable="true"/>
  </Host>
```

## Mybatis 逆向工程

系统使用了 Mybatis 的逆向工程，依据数据库表自动生成 domain 和 mapper（注：自动生成的代码都是针对单表，若需多表整合，则要手动修改实现），其中，针对每个数据库表，都会生成两个封装对象，可以认为 example 对象是对其相应的 ORM 映射对象查询条件的封装。逆向工程的实现-Mybatis Generator 的代码托管在 <https://github.com/mybatis/generator>，代码 down 下来后，配置 generatorConfig.xml 文件，如下：

```
1<?xml version="1.0" encoding="UTF-8"?>
2<!DOCTYPE generatorConfiguration
```

```

3 PUBLIC "-//mybatis.org//DTD MyBatis Generator Configuration 1.0//EN"
4 "http://mybatis.org/dtd/mybatis-generator-config_1_0.dtd">
5
6<generatorConfiguration>
7    <context id="testTables" targetRuntime="MyBatis3">
8
9        <commentGenerator>
10            <!-- 是否去除自动生成的注释 true: 是 : false:否 -->
11            <property name="suppressAllComments" value="true"
12/>
13
14        </commentGenerator>
15
16        <!-- 数据库连接的信息：驱动类、连接地址、用户名、密码 -->
17        <jdbcConnection driverClass="com.mysql.jdbc.Driver"
18            connectionURL="jdbc:mysql://localhost:3306/production_ms" userId="root"
19            password="root">
20        </jdbcConnection>
21
22        <!-- 默认 false，把 JDBC DECIMAL 和 NUMERIC 类型解析为
23Integer，为 true 时把 JDBC DECIMAL 和
24            NUMERIC 类型解析为 java.math.BigDecimal -->
25        <javaTypeResolver>
26            <property name="forceBigDecimals" value="false" />
27        </javaTypeResolver>
28
29        <!-- targetProject:生成 PO 类的位置 -->
30        <javaModelGenerator
31targetPackage="org.hqu.production_ms.domain"
32            targetProject=".\\src">
33            <!-- enableSubPackages:是否让 schema 作为包的后缀 -
34->
35            <property name="enableSubPackages" value="false" />
36            <!-- 从数据库返回的值被清理前后的空格 -->
37            <property name="trimStrings" value="true" />
38        </javaModelGenerator>
39
40        <!-- targetProject:mapper 映射文件生成的位置 -->
41        <sqlMapGenerator
42targetPackage="org.hqu.production_ms.mapper"
43            targetProject=".\\src">

```

```

44         <!-- enableSubPackages:是否让 schema 作为包的后缀 -
45->
46         <property name="enableSubPackages" value="false" />
47     </sqlMapGenerator>
48
49     <!-- targetPackage: mapper 接口生成的位置 -->
50     <javaClientGenerator type="XMLMAPPER"
51         targetPackage="org.hqu.production_ms.mapper"
52         targetProject=".\\src">
53         <!-- enableSubPackages:是否让 schema 作为包的后缀 -
54->
55         <property name="enableSubPackages" value="false" />
56     </javaClientGenerator>
57
58     <!-- 指定数据库表 -->
59     <table schema="" tableName="unqualify_apply"></table>
60
61 </context>
62 </generatorConfiguration>

```

修改完成后，执行 main 方法即可。需要注意的是，若要重新生成，则需把已经生成的文件删除，因为它不会自己覆盖，导致文件混乱。

关于 Mybatis 逆向工程的更多详细信息，读者可以自行上网查阅。