

2022 年福建省职业教育教学成果奖申报佐证资料

软件著作权白皮书

成果名称：“教、训、赛、创”一体化助推信息技术类专业人才培养模式研究与实践

申报时间：2022 年 5 月 17 日

成果所属类别：中等职业教育

目 录

- 一、 知识互补结构下的中职教师研修平台 V1.0……………(1)
- 二、 知识互补结构下的中职教师研修平台 V1.0……………(85)
- 三、 知识互补结构下的中职教师研修平台 V1.0……………(134)

友好 十一

中华人民共和国国家版权局 计算机软件著作权登记证书

证书号： 软著登字第7308277号

软件名称： 知识互补结构下的中职教师研修平台
V1.0

著作权人： 集美工业学校;北京世纪超星信息技术发展有限责任公司

开发完成日期： 2021年03月05日

首次发表日期： 2021年03月09日

权利取得方式： 原始取得

权利范围： 全部权利

登记号： 2021SR0585651

根据《计算机软件保护条例》和《计算机软件著作权登记办法》的规定，经中国版权保护中心审核，对以上事项予以登记。



No. 07830960



2021年04月23日

中华人民共和国国家版权局
计算机软件著作权登记证书

(副本)

证书号： 软著登字第7308277号

软件名称： 知识互补结构下的中职教师研修平台
V1.0

著作权人： 集美工业学校;北京世纪超星信息技术发展有限责任公司

开发完成日期： 2021年03月05日

首次发表日期： 2021年03月09日

权利取得方式： 原始取得

权利范围： 全部权利

登记号： 2021SR0585651

根据《计算机软件保护条例》和《计算机软件著作权登记办法》的规定，经中国版权保护中心审核，对以上事项予以登记。



No. 07330961



2021年04月23日

流水号

2021R11L1337118



受理签字: _____

审查签字: _____

计算机软件著作权登记申请表

软件基本信息	软件全称	知识互补结构下的中职教师研修平台				版本号	V1.0
	软件简称					软件分类	应用软件
	软件作品说明	<input checked="" type="radio"/> 原创 <input type="radio"/> 修改(含翻译软件、合成软件) <input type="checkbox"/> 修改软件须经原权利人授权 <input type="checkbox"/> 原有软件已经登记 原登记号: 修改(翻译或合成)软件作品说明:					
开发完成日期	2021年03月05日						
发表状态	<input checked="" type="radio"/> 已发表 首次发表日期: 2021年03月09日 首次发表地点: 中国 福建 厦门 <input type="radio"/> 未发表						
开发方式	<input type="radio"/> 单独开发 <input checked="" type="radio"/> 合作开发 <input type="radio"/> 委托开发 <input type="radio"/> 下达任务开发						
著作权人	姓名或名称	类别	证件类型	证件号码	国籍	省份/城市	成立/出生日期
	集美工业学校	事业单位法人	统一社会信用代码证书	12350200B3695288X0	中国	福建 厦门	2016年08月30日
	北京世纪超星信息技术发展有限责任公司	企业法人	统一社会信用代码证书	91110108700242692T	中国	北京 海淀	2000年01月27日

流水号

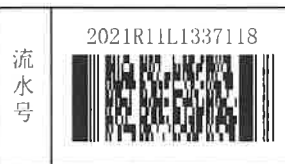
2021R11L1337118



权利说明	权利取得方式	<input checked="" type="radio"/> 原始取得 <input type="radio"/> 继受取得 (<input type="radio"/> 受让 <input type="radio"/> 承受 <input type="radio"/> 继承) <input type="checkbox"/> 该软件已登记 (原登记号:) <input type="checkbox"/> 原登记做过变更或补充 (变更或补充证明编号:)	
	权利范围	<input checked="" type="radio"/> 全部 <input type="checkbox"/> 部分 (<input type="checkbox"/> 发表权 <input type="checkbox"/> 署名权 <input type="checkbox"/> 修改权 <input type="checkbox"/> 复制权 <input type="checkbox"/> 发行权 <input type="checkbox"/> 出租权 <input type="checkbox"/> 信息网络传播权 <input type="checkbox"/> 翻译权 <input type="checkbox"/> 应当由著作权人享有的其他权利)	
软件鉴别材料	程序鉴别材料	<input checked="" type="radio"/> 一般交存: 提交源程序前连续的30页和后连续的30页 <input type="checkbox"/> 例外交存: 使用黑色宽斜线覆盖, 页码为: <input type="checkbox"/> 例外交存: 前10页和任选连续的50页 <input type="checkbox"/> 例外交存: 目标程序的连续的前、后各30页和源程序任选连续的20页	
	文档鉴别材料	<input checked="" type="radio"/> 一般交存: 提交任何一种文档的前连续的30页和后连续的30页 <input type="checkbox"/> 例外交存: 使用黑色宽斜线覆盖, 页码为: <input type="checkbox"/> 例外交存: 前10页和任选连续的50页	
软件功能和技术特点	硬件环境	开发: CPU: Core i9 2.8GHz, 八核十六线程, 内存容量: 64GB, 硬盘容量: 2TB 运行: CPU主频: 2.1GHz, CPU核心数: 四核心, 显存类型: GDDR5, 内存容量: 4GB	
	软件环境	开发: Windows 10; MS VC++, MyEclipse 运行: Windows 10; Windows 10, 阿里服务器	
	编程语言	C# ; C++	源程序量 10703行
	主要功能和技术特点	开发目的: 解决知识互补结构下的中职教师研修问题 面向领域/行业: 面向知识互补结构下的中职教师研修领域 主要功能: 知识互补结构下的中职教师研修平台能够快速采集软件运行过程中间文件以及对数据进行实时优化和记录, 并有效应对大流量、高并发用户访问, 有效缓解系统压力。 技术特点: 大数据软件: 系统具有信息处理、后台驱动管理等软件参数, 并对软件进行在线控制管理。该系统还能够进行背景图片设置, 设置信息的过滤以及设置自动登录等功能, 综合提升用户的体验效果。	

流水号	2021R11L1337118 
-----	--

申请办理方式	<input checked="" type="radio"/> 由著作权人申请 <input type="radio"/> 由代理人申请
申请人姓名或名称	集美工业学校
<p>申请人委托下述代理人办理登记事宜，具体委托事项如下： 全权负责本软件著作权登记相关事宜，直至取得登记证书</p>	
代理人姓名或名称	北京友尚文化发展有限公司
<p>申请人认真阅读了填表说明，准确理解了所需填写的内容，保证所填写的内容真实。</p> <div style="text-align: right;">  <p>申请人签章 2021年03月11日</p> </div>	



证书份数	1 份正本 1 份副本	
请确认所需要的计算机软件著作权登记证书副本数量。登记证书和副本数量之和不能超过软件著作权人的数量		
提交申请材料清单		
材料类型	材料名称	
申请表	打印签字或盖章的登记申请表	1页
身份证明文件	著作权人(1) - 集美工业学校的统一社会信用代码证书复印件 著作权人(2) - 北京世纪超星信息技术发展有限责任公司的统一社会信用代码证书复印件 代理人北京友尚文化发展有限公司的统一社会信用代码证书复印件	1页 1页 1页
权利归属证明文件	合作开发合同或协议	2页
软件鉴别材料	程序鉴别材料 - 一般交存 文档鉴别材料 - 一般交存	60页 15页
其他材料		

合作开发协议书

甲方：集美工业学校

证件号码：12350200B3695288X0

乙方：北京世纪超星信息技术发展有限责任公司

证件号码：91110108700242692T

鉴于，协议各方均为计算机软件专业开发人员，能够进行创造性的软件开发活动。并且，协议各方有意愿共同从事：知识互补结构下的中职教师研修平台 版本号为 V1.0 软件的开发工作。为了规范各方的权利义务，在《中华人民共和国合同法》及其他相关法规政策的原则指导下，订立本协议书，各方共同遵守：

第一条、合作宗旨

为完成 知识互补结构下的中职教师研修平台 版本号为 V1.0 软件的开发工作，并共同享有开发成果而合作。

第二条 合作项目和范围

协议各方共同开发 知识互补结构下的中职教师研修平台 版本号为 V1.0 软件，合作范围包括软件的代码编写、调试、测试等开发工作。

第三条 合作期限

合作期限为一年。

第四条 合作方式

1. 协议各方按照软件编程工作的正常分工进行编写，任何一方不得随意更改软件的重大功能和事项，以免对其余各方造成履约困难。

2. 合作各方应坚持勤勉努力诚实信用的原则，进行各方分别负责的软件的编程工作，并考虑到各方软件的兼容和接合。如部分合作人发生特殊技术困难，其余合作方有义务为其提供合理适当的技术帮助。

第五条 知识产权

1. 各方编写的软件源代码、技术文档及汇编而成的程序本身，其著作权均由合作方共同享有。

2. 合作各方在编写软件的过程中，不得有侵犯他人知识产权的行为，否则，应对外承担全部侵权责任。

第六条 协议变更

1. 经合作各方协商同意，本协议可以作相应变更；

2. 任何合作方未经与其他各方协商，擅自变更本协议条款或者将本协议权利义务转让给他人，均为无效。

第七条 禁止行为

1. 未经全体合作人同意，禁止任何合作人私自以团体名义进行业务活动；如其业务获得利益归合作各方共有，造成损失按实际损失赔偿。
2. 禁止合作人经营与团队相竞争的业务。
3. 禁止合作方泄露本协议所涉及的相关商业秘密。
4. 如合作人违反上述各条，应按实际损失赔偿。

第八条 合作的终止

合作开发活动因以下事由之一得终止：

- ①全体合作人同意终止合作关系；
- ②合作项目因技术原因，根本不能完成；
- ③合作项目违反法律被撤销。

第九条 纠纷的解决

合作各方之间如发生纠纷，应共同协商，本着有利于事业发展的原则予以解决。如协商不成，可以诉诸法院。

第十条 本协议如有未尽事宜，应由合作人集体讨论补充或修改。补充和修改的内容与本协议具有同等效力。

各方签署：

甲方：

乙方：



2020年8月29日

中华人民共和国 事业单位法人证书

(副本)

统一社会信用代码 12350200B3695288X0



仅用于数据申请



gjsy.gov.cn

有效期自2019年03月29日至2024年03月29日

名称 集美工业学校
宗旨和业务范围 实施中等职业教育；开展相关领域现代职业教育研究；开展相关职业培训
住所 福建省厦门市集美区杏前路22号
法定代表人 陶顺生
经费来源 财政拨款
开办资金 ¥6548万元
举办单位 厦门市教育局
登记管理机关



1.1 系统简介

知识互补结构下的中职教师研修平台能够快速采集软件运行过程中间文件以及对数据进行实时优化和记录，并有效应对大流量、高并发用户访问，有效缓解系统压力。系统具有信息处理、后台驱动管理等软件参数，并对软件进行在线控制管理。该系统还能够进行背景图片设置，设置信息的过滤以及设置自动登录等功能，综合提升用户的体验效果。主要有院校参数、数据备份、教师评价等功能。

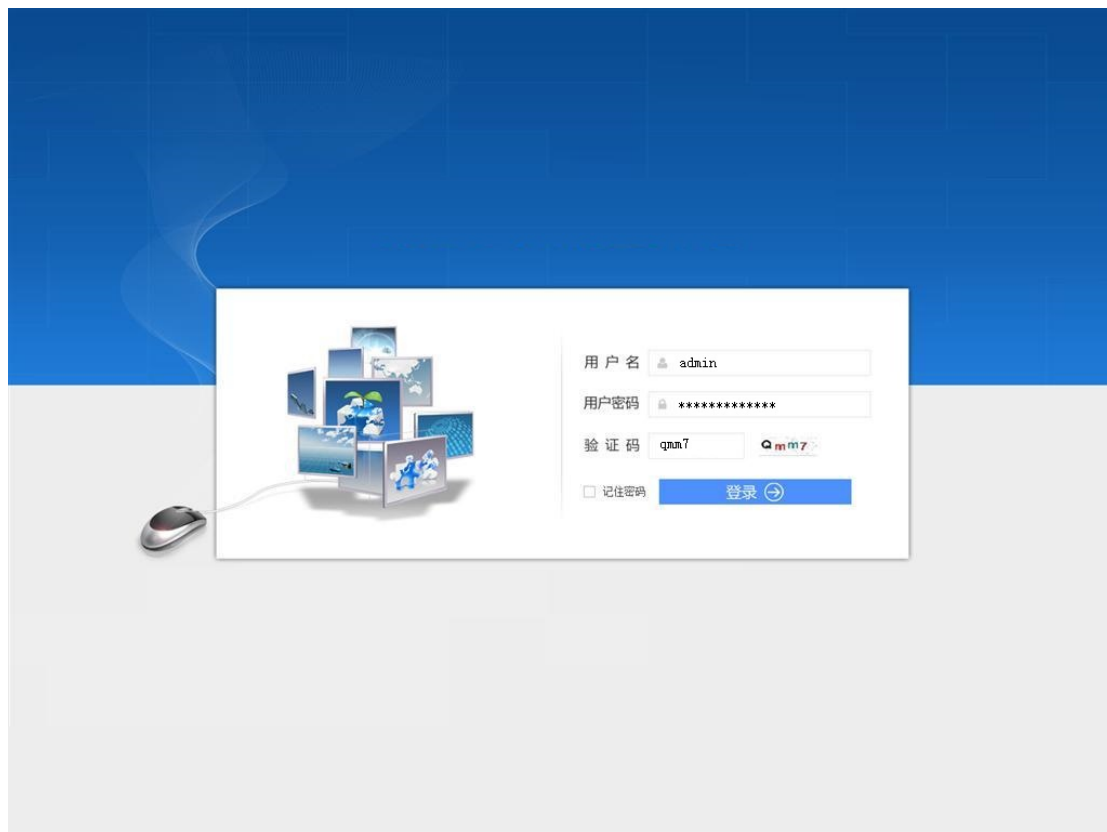
1.2 操作和界面解说

主要是对软件的基本的操作的内容和信息进行介绍的操作，帮助用户更好的使用软件系统。

1.3 登录页面介绍

在桌面上双击系统的图标就会出现以下的界面，这时候就会弹出了用户登录的界面，从这个界面中可以到有用户名和密码需要输入，输入完成之后点击登入按钮。而且必须在用户名与密码同时输入正确的情况下才能通过认证登录成功，账号和密码必须有正确才可以登录该软件进行使用操作，如果不能正确登录该软件，将无法正常使用该软件，详情如下图所示：

- 登陆系统。
- 修改密码。



点击登录按钮，即可进行登录的操作了。

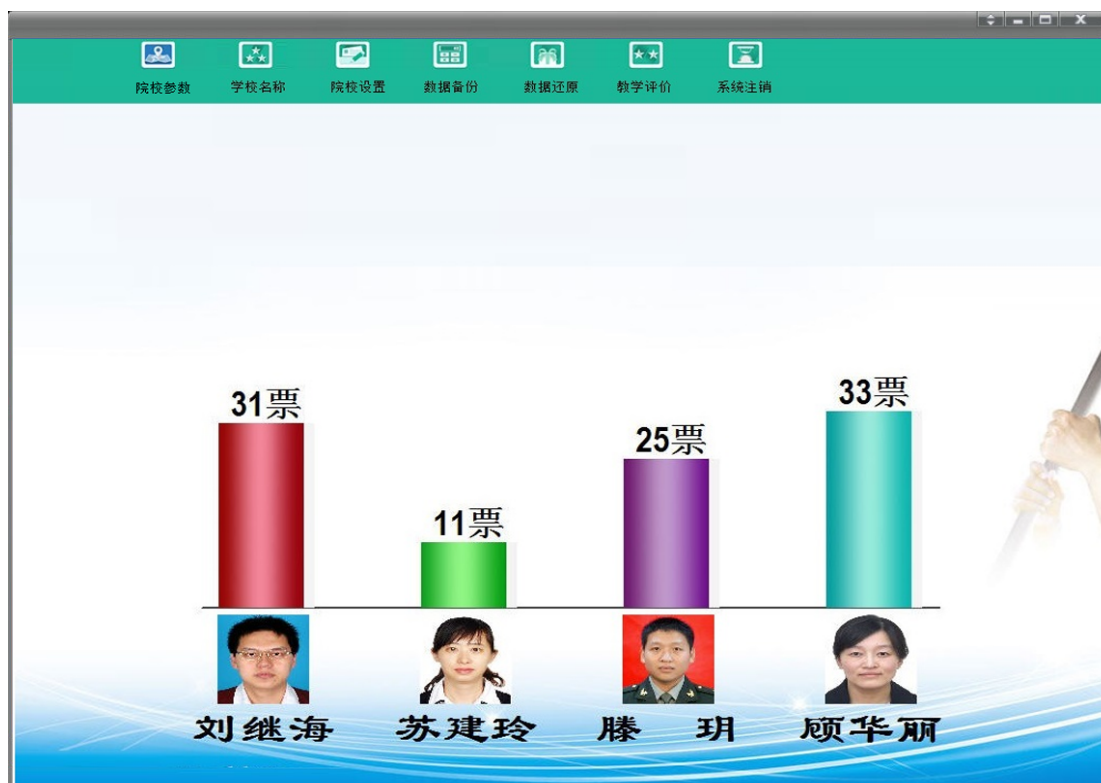
1.3.1 登录系统

操作方法

输入用户名和密码。在“用户名”中输入用户名，“密码”中输入密码，用户名和密码是系统管理员告诉你的。点击“登录”按钮。系统验证通过后，进入系统主页。

1.4 主页面介绍

您在进入系统后，会看到系统的主页面，如图：



该界面是软件系统的主要的操作界面，根据界面的展示，使用者可以更好的操作软件系统了。

- 功能按钮区：使用者可以在该区域进行软件的操作与使用。

2 软件功能

2.1 软件结构

主要是帮助用户进行软件的一个结构的介绍。

2.1.1 系统功能

系统功能就是帮助用户介绍一下系统的功能信息了，如下图所示



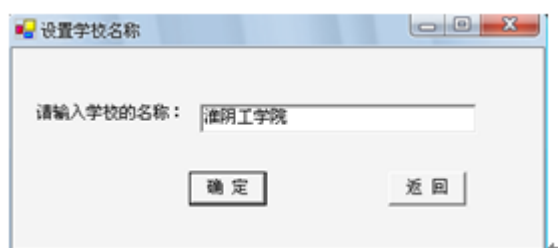
2.2 院校参数

院校的参数信息就是帮助用户操作和处理相关的数据信息，完成最近的院校的参数信息的处理人，如下图所示：



2.3 学校名称

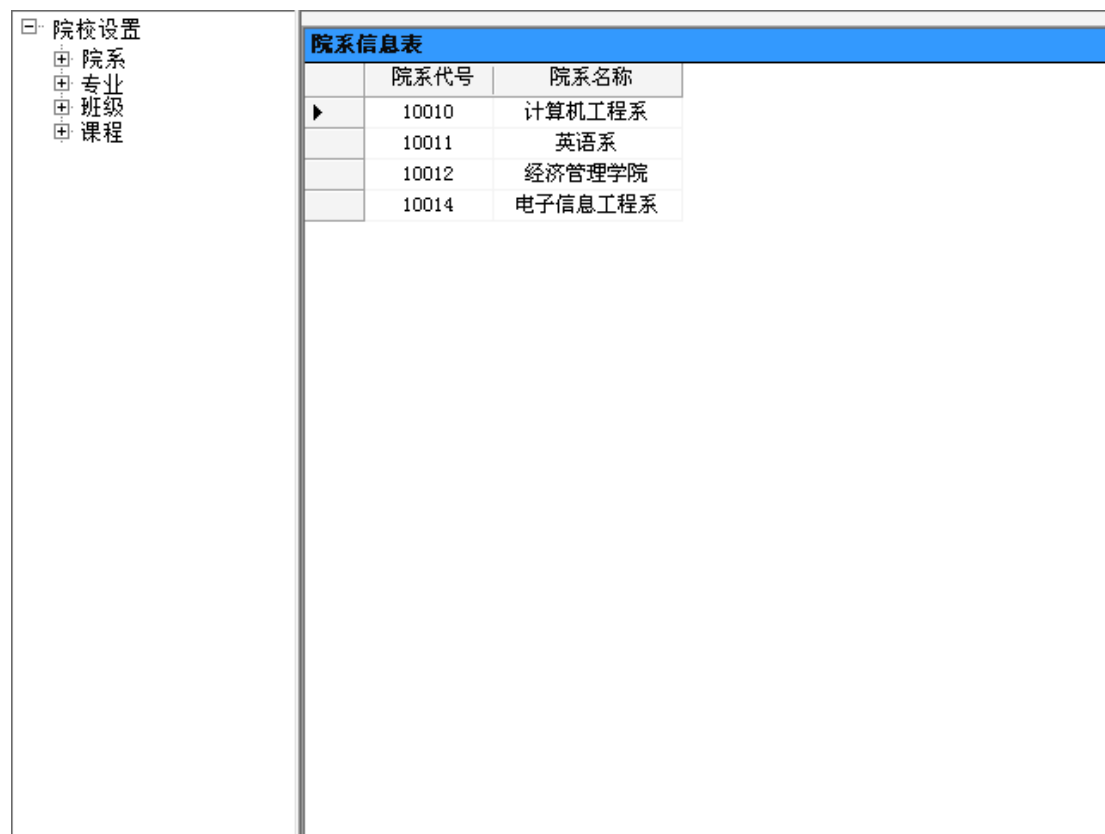
在主页面点击学校名称导航，在设置页面输入学校名称，点击确定按钮完成操作，如下图所示：



以上界面就是学校名称的信息页面了。

2.4 院校设置

在院校设置页面用户可以对院系、专业、班级课程等信息进行设置操作，帮助用户进行教学评估，如下图所示：

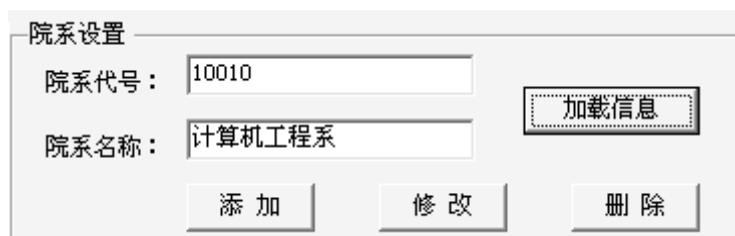


院系代号	院系名称
10010	计算机工程系
10011	英语系
10012	经济管理学院
10014	电子信息工程系

以上界面就是院校设置的信息页面了。

2.4.1 院系设置

点击院系设置，输入院系代号、院系名称，点击添加按钮完成操作，如下图所示：



院系设置

院系代号： 10010

院系名称： 计算机工程系

加载信息

添加 修改 删除

以上界面就是院系设置的信息页面了。

2.4.2 专业设置

点击转设置，选择院系名称，输入专业代号、专业名称，点击添加按钮完成操作，如下图所示：



The screenshot shows a form titled "专业设置" (Professional Settings). It contains the following elements:

- A dropdown menu labeled "院系名称" (Department Name).
- A text input field labeled "专业代号" (Professional Code).
- A text input field labeled "专业名称" (Professional Name).
- A button labeled "加载信息" (Load Information) positioned to the right of the "专业代号" and "专业名称" fields.
- Three buttons at the bottom: "添加" (Add), "修改" (Modify), and "删除" (Delete).

以上界面就是专业设置的信息页面了。

2.4.3 班级设置

点击班级设置，选择院系名称、专业名称，输入班级代码、班级名称，点击添加按钮完成操作，如下图所示：



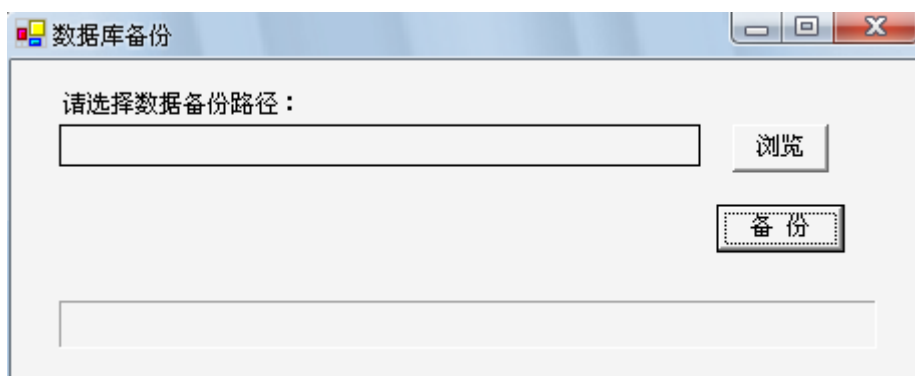
The screenshot shows a form titled "班级设置" (Class Settings). It contains the following elements:

- A dropdown menu labeled "院系名称" (Department Name).
- A dropdown menu labeled "专业名称" (Professional Name).
- A text input field labeled "班级代号" (Class Code).
- A text input field labeled "班级名称" (Class Name).
- A button labeled "加载信息" (Load Information) positioned to the right of the "专业名称" and "班级代号" fields.
- Three buttons at the bottom: "添加" (Add), "修改" (Modify), and "删除" (Delete).

以上界面就是班级设置的信息页面了。

2.5 数据备份

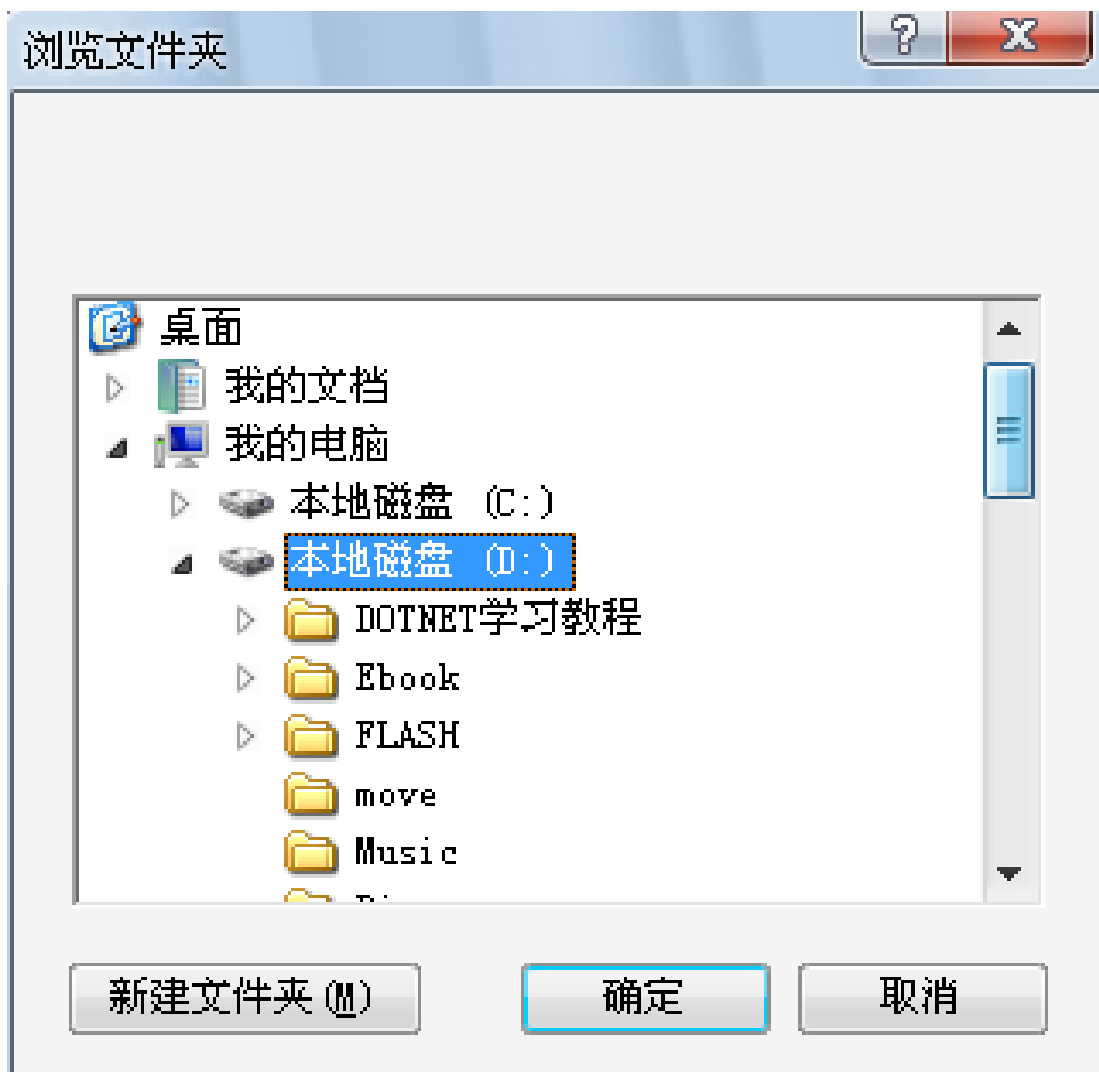
为了保证数据的安全性，用户可以对数据进行备份操作，点击浏览按钮选择备份位置，点击备份按钮进行备份操作，如下图所示：



以上界面就是数据备份的信息页面了。

2.5.1 浏览文件夹

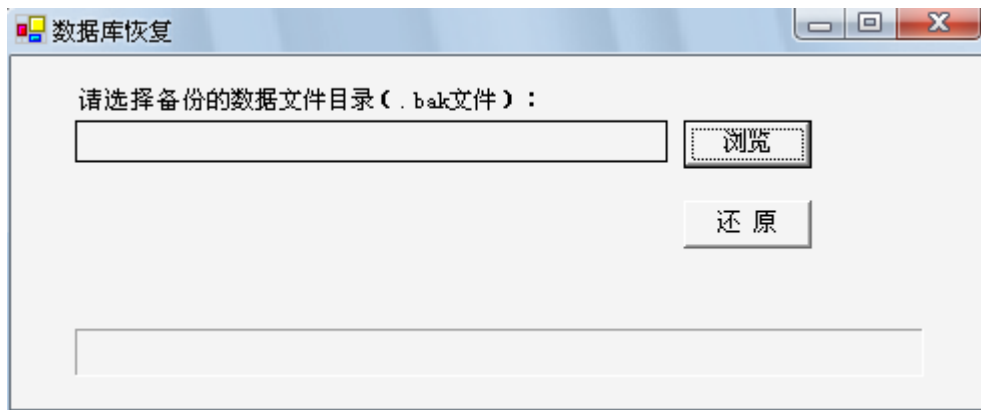
在数据备份页面点击浏览按钮，选择要备份文件的存储位置，点击确定按钮完成操作，如下图所示：



以上界面就是浏览文件夹的信息页面了。

2.6 数据还原

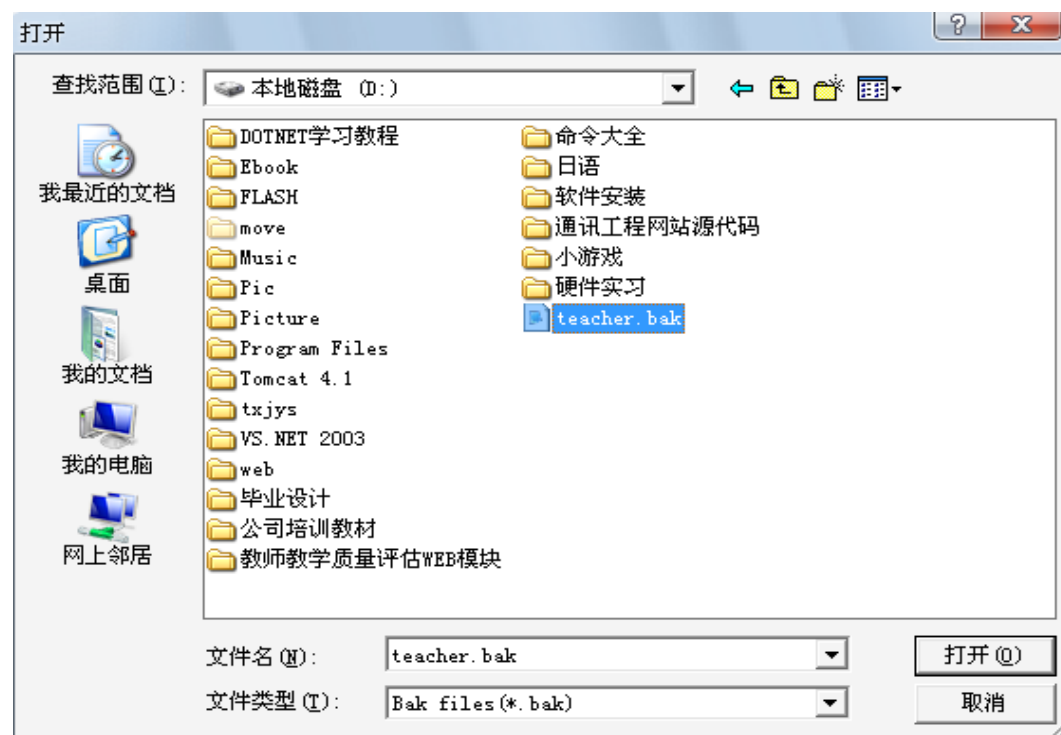
当数据出现丢失的情况时，用户可以根据系统备份的文件进行数据还原操作，选择备份文件，点击还原按钮进行还原，如下图所示：



以上界面就是数据还原的信息页面了。

2.6.1 打开文件

在数据还原页面点击浏览按钮，选择备份文件的位置，选中备份文件，点击打开按钮进行数据还原操作，如下图所示：



以上界面就是打开文件的信息页面了。

2.7 教师评价

帮助用户处理基本的评价的操作，评价的数据和操作的信
息，根据基本的数据操作管理，根据每个课程的信息管理，进行教学信息的评价反馈，如下图所示：



以上界面就是教学评价的信息页面了。

2.7.1 自我评价

点击自我评价，教师可以对自己的教学行为进行评价，如下图所示：

编号	指标名称	得分
25	精神饱满，治学严谨，为人师表，严格管理，备课充分	请选择..... ▾
26	内容正确，信息量大，重点突出，能吸收、介绍该学科发展新成果，严格按照教学进度开展教学	请选择..... ▾
27	方法灵活多样，开拓思路，联系实际，因材施教，积极采用现代化教学手段	请选择..... ▾
28	有效利用上课时间，思路清晰，条理分明，师生互动，气氛活跃	请选择..... ▾
29	作业适中，批改认真、及时，讲解及时，常进行辅导、答疑	请选择..... ▾

以上界面就是自我评价的信息页面了。

2.7.2 查看留言

点击查看留言，用户可以对留言信息进行查看，选择学期、评估类型等信息，点击按学期查看或按学年查看，如下图所示：

选择学期: 至 选择评估者类型:

按课程查看: 按评估者类型查看:

以上界面就是查看留言的信息页面了。

2.7.3 评估得分

点击评估得分，用户可以对教学评估得分信息进行查看，选择学期，点击查询按钮进行查看，如下图所示：

选择学期: 至 上

序号	学期	工号	姓名	所属院系	课程名	专家评分	领导评分	同行评分	学生评分	自评评分	总分
		101001	步山岳	计算机工程系	计算机网络	18.00000	18.80000	13.46250	36.60000	4.75000	91.61250

以上界面就是评估得分的信息页面了。

2.7.4 修改密码

在系统界面中用户选择“修改密码”功能之后，系统就会自动切换到修改密码功能界面，用户可以在这里可以对修改密码进行修改。功能界面详情如下图所示：

The screenshot displays a web application interface with a blue header and a light blue sidebar. The main content area is white. At the top, there is a navigation bar with buttons for '首页', '教案中心', '我要备课', '我的教案', '个人信息', and '修改密码'. Below this, a status bar shows '单位: 活在当下 老师, 欢迎您!' and '身份: 教师'. The '修改密码' button is highlighted. The main content area contains a form titled '修改用户密码' with the following fields: '用户名: 活在当下', '旧密码:', '新密码:', and '确认密码:'. A '保存' button is located below the form. The bottom right corner of the page has a '退出' button.

2.7.5 日志查询

操作人员点击日志查询，系统弹出相应的界面，操作人员在此界面可以查询相应的东西详情如下图所示：

类型	时间	代码或名称	内容
登录	2015-09-28 16:46	0001	用户[管理员]登录本系统.
登录	2015-09-28 16:46	0001	用户[管理员]登录本系统.
登录	2015-09-28 14:07	0001	用户[管理员]登录本系统.
登录	2015-09-28 14:01	0001	用户[管理员]登录本系统.
登录	2015-09-28 13:58	0001	用户[管理员]登录本系统.
登录	2015-09-28 13:43	0001	用户[管理员]登录本系统.
登录	2015-09-28 13:42	0001	用户[管理员]登录本系统.
登录	2015-09-28 13:39	0001	用户[管理员]登录本系统.
充值	2015-09-28 09:20	288802000068	充值金额:1000, 增加次数: 7, 余额:2280.00, 保存成功
登录	2015-09-28 09:10	0001	用户[管理员]登录本系统.
		10	

3 系统注销

点击退出系统按钮，就给出相应提示，是否确定要退出系统，确定退出，直接点击退出系统按钮，则系统就会关闭，如需再次使用系统，则会回到我们的第一步操作，就是点击桌面的图标即可。具体退出系统界面如下图所示：



点击确定按钮，即可进行退出的操作。

```

using System;
using System.Collections.Generic;
using System.ComponentModel; //登录页面介绍
using System.Drawing;
using System.Data;
using System.Text;
using System.Windows.Forms;
namespace ComponentSysPkasa
{
    public partial class ChartPanel : UserControl
    {
        #region add shuxing
        private DataSet PDataSource = null;
        [Browsable(true), Category("system tape"), Description("connect sql")] //在“shuxing”
shituDataSourceshuxing
        public DataSet DataSource
        {
            get { return PDataSource; }
            set
            {
                PDataSource = value;
                ChartFirst();
                Invalidate();
            }
        }
        private AxesStyle PSumAxes = AxesStyle.Null;
        [Browsable(true), Category("system tape"), Description("attention")]
        public AxesStyle SumAxes
        {
            get { return PSumAxes; }
            set
            {
                PSumAxes = value;
                if (PSumAxes != AxesStyle.Null)
                {
                    this.SumYAxis = "";
                }
            }
        }
        private string PSumXAxis = "";
        public string SumXAxis
        {
            get { return PSumXAxis; }
            set
            {
                PSumXAxis = value;
            }
        }
        private string PSumYAxis = "";
        [Browsable(true), Category(""), Description("")]

```

```

public string SumYAxis
{
    get { return PSumYAxis; } // 院校参数
    set
    {
        PSumYAxis = value;
        if (PSumYAxis != "")
            this.SumAxes = AxesStyle.Null;
    }
}
public enum CharMode
{
    Bar = 0, //
    Mark = 1,
    Line = 2, //
    Area = 3, //
    none = 4, //
}

public enum ChartTitleStyle
{
    TopLeft = 0, //
    TopCenter = 1, //
    TopRight = 2, //
    BottomLeft = 3, //
    BottomCenter = 4, //
    BottomRight = 5, //
}

private CharMode PChartStyle = CharMode.Bar;
[Browsable(true), Category(""), Description("")]
public CharMode
ChartStyle
{
    get
    {
        return PChartStyle;
    }
    set
    {
        PChartStyle = value;
        if (this.ChartStyle == CharMode.Bar || this.ChartStyle == CharMode.Mark)
        {
            if (this.RowWeave == CharRowWeaveStyle.Stavked)
            {
                if (this.RowList == 1)
                    RowSideMax();
                else
                    RowStackedSum();
            }
            if (this.RowWeave == CharRowWeaveStyle.Side)
                RowSideMax();
        }
    }
}

```

```
    }
    if (this.ChartStyle== CharMode.Line)
        RowSideMax();
    if (this.ChartStyle == CharMode.Area)// 院校设置
    {
        AreaPercent();
    }
    Invalidate();
}
}
private ChartTitleStyle PTitleStyle = ChartTitleStyle.TopCenter;
public ChartTitleStyle TitleStyle
{
    get { return PTitleStyle; }
    set
    {
        PTitleStyle = value;
        Invalidate();
    }
}
private Color PTitleColor = Color.Black;
public Color TitleColor
{
    get { return PTitleColor; }
    set
    {
        PTitleColor = value;
        Invalidate();
    }
}
private Color PChartColor = Color.Red;
public Color ChartColor
{
    get { return PChartColor; }
    set
    {
        PChartColor = value;
        Invalidate();
    }
}
private bool PChartWearColor = false;
public bool ChartWearColor
{
    get { return PChartWearColor; }
    set
    {
        PChartWearColor = value;
        if (this.RowList == 1)
            if (this.ChartStyle == CharMode.Bar)
                Invalidate();
    }
}
```

```
    }
}
private bool PChartmark = true;
[Browsable(true), Category("system tape"), Description("whether appear data table")]
// public bool Chartmark
{
    get { return PChartmark; }
    set
    {
        PChartmark = value;
        Invalidate();
    }
}
private string PTitleName = "Title";
[Browsable(true), Category("system tape"), Description("system name")]
public string TitleName
{
    get { return PTitleName; }
    set
    {
        PTitleName = value;
        Invalidate();
    }
}
private Font PTitleFont = new Font("songti", 9);
[Browsable(true), Category("system tape"), Description("system namewenzi yangshi")]
//在 “shuxing” shituTitleFontshuxing
public Font TitleFont
{
    get { return PTitleFont; }
    set
    {
        PTitleFont = value;
        Invalidate();
    }
}
private int PFoulCalcar = 5;
public int FoulCalcar
{
    get { return PFoulCalcar; }
    set
    {
        PFoulCalcar = value;
        if (PFoulCalcar < 0)
            PFoulCalcar = 0;
        Invalidate();
    }
}

private Color PFoulLineColor = Color.Black;
```

```
public Color FoulLineColor
{
    get { return PFoulLineColor; }
    set
    {
        PFoulLineColor = value;
        Invalidate();
    }
}
public Font XYFont
{
    get { return PXYFont; }
    set
    {
        PXYFont = value;
        Invalidate();
    }
}
public Color XYColor
{
    get { return PXYColor; }
    set
    {
        PXYColor = value;
        Invalidate();
    }
}
public int PageList
{
    get { return PPageList; }
    set
    {
        PPageList = value;
        if (PPageList < 1)
            PPageList = 1;
        if (this.DataSource == null)
            ChartFirst();
        if (this.ChartStyle == CharMode.Bar || this.ChartStyle == CharMode.Mark)
        {
            if (this.RowWeave == CharRowWeaveStyle.Stavked)
            {
                if (this.RowList == 1)
                    RowSideMax();
                else
                    RowStackedSum();
            }
            if (this.RowWeave == CharRowWeaveStyle.Side)
                RowSideMax();
        }
        if (this.ChartStyle == CharMode.Area)
```

```

        {
            AreaPercent();
        }
        Invalidate();
    }
}
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO;
using System.Text.RegularExpressions;
namespace MainTdeal
{
    public partial class FormSystemMain : FormSystem
    {
        private Point FormSystemPosition = new Point();
        private Timer timer = new Timer();
        private int progress = 0;
        private const int FIFTY = 30;
        public FormSystemMain()
        {
            InitializeComponent();
            timer.Interval = 5;
            this.timer.Tick += new System.EventHandler(this.timer_func);
        }
        private void FormSystemMain_Load(object sender, EventArgs e)
        {
            pictureBox1.Visible = false;
            pictureBox2.Visible = false;
            pictureBox3.Visible = false;
            pictureBox4.Visible = false;
            pictureBox5.Visible = false;
            panelwork.Hide();
            panelmen.Hide();
            panelku.Hide();
            webBrowser2.Navigate(Application.StartupPath + @"\map.html");
        }
        private void FormSystemMain_Shown(object sender, EventArgs e)
        {
            try
            {
                uint numDevices;          /* Number of available devices. */
                PYLON_DEVICECALLBACK_HANDLE hCb;          /* Required for
deregistering the callback. */

```



```

int loopCount;          /* Counter. */
#if DEBUG
bool isGigECamera;     /* 1 if the device is a GigE device. */
#endif

DeviceCallbackHandler callbackHandler = new DeviceCallbackHandler(); /*
Handles callbacks from a device. */
/* Before using any pylon methods, the pylon runtime must be initialized. */
Pylon.Initialize();
/* Enumerate all camera devices. You must call
PylonEnumerateDevices() before creating a device. */
numDevices = Pylon.EnumerateDevices();
if (0 == numDevices)
{
    throw new Exception("No devices found.");
}
/* Get a handle for the first device found. */
hDev = Pylon.CreateDeviceByIndex(0);
/* Before using the device, it must be opened. Open it for configuring
parameters, for grabbing images, and for grabbing events. */
Pylon.DeviceOpen(hDev, Pylon.cPylonAccessModeControl |
Pylon.cPylonAccessModeStream);
/* Print out the name of the camera we are using. */
{
    bool isReadable = Pylon.DeviceFeatureIsReadable(hDev,
"DeviceModelName");
    if (isReadable)
    {
        string name = Pylon.DeviceFeatureToString(hDev,
"DeviceModelName");
        Console.WriteLine("Using camera {0}", name);
    }
}
}
//
private void timer_func(object sender, EventArgs e)
{
    //panelProgress.Show();
    this.panelProgress.Left = this.Width / FIFTY * progress;
    if (progress == FIFTY)
    {
        this.timer.Stop();
        progress = 0;
        // panelProgress.Hide();
    }
    else
    {
        progress++;
    }
}
private void btnX_Click(object sender, EventArgs e)

```

```

        {
            this.Close();
        }
        //
        private void panell1_MouseDown(object sender, MouseEventArgs e)
        {
FormSystemPosition = e.Location;
        }
        private void panell1_MouseUp(object sender, MouseEventArgs e)
        {
            this.Location = new Point(this.Location.X + e.Location.X -
FormSystemPosition.X,
                                     this.Location.Y +
e.Location.Y - FormSystemPosition.Y);
        }
        private void showFlowEffect(string name, string number, string weight, string air, Point
po)
        {
            panelFlow.Show();
            this.labelName.Text = name;
            this.labelNUmber .Text= number;
            this.labelWeight.Text = weight;
            this.labelair.Text = air;
            panelFlow.Location = po;
        }
    }
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace InernetDataDistribution
{
    public partial class FormSystemProgress : FormSystem
    {
        private Timer timer = new Timer();
        private int progress = 0;
        private const int FIFTY = 20;
        public FormSystemProgress()
        {
            InitializeComponent();
            timer.Interval = 100; // millisecond
            this.timer.Tick += new System.EventHandler(this.timer_func);
        }
        private void FormSystemProgress_Load(object sender, EventArgs e)
        {

```

```

        timer.Start();
    }
    private void timer_func(object sender, EventArgs e)
    {
        //panelProgress.Show();
        this.panelProgress.Width = this.Width * progress / FIFTY;
        this.labelPerc.Text = (progress * 100 / FIFTY).ToString()+"%";
        if (progress == FIFTY)
        {
            this.timer.Stop();
            progress = 0;
            this.Close();
            // panelProgress.Hide();
        }
        else
        {
            progress++;
        }
    }
    private void FormSystemProgress_Shown(object sender, EventArgs e)
    {
    }
}
}
int main(int argc, char **argv)
{
    Mat gau, gau1;
    int start = GetTickCount();
    GaussianBlur(src, gau, Size(0,0), 16);
    cout << "GaussianBlur: " << GetTickCount()-start << "ms" << endl;
    imshow("gau", gau);
    // cvSmooth(gray, gau1, 2, 0,0, 0.0, 0.0);
    // GaussianBlur(gray, gau1, Size(0,0), 0.0);
    // imshow("gau1", gau1);
    Mat gauss;
    start = GetTickCount();
    GaussianTemplateSmooth(gray, gauss, 0.0);
    cout << "GaussianTemplateSmooth: " << GetTickCount()-start << "ms" << endl;
    imshow("gauss", gauss);
    // imwrite("dst.jpg", dst);
    // imwrite("gauss.jpg", gauss);
    Mat gs;
    start = GetTickCount();
    GaussianSmooth(src, gs, 16);
    cout << "GaussianSmooth: " << GetTickCount()-start << "ms" << endl;
    imshow("gs", gs);
    // imwrite("small.jpg", small);
    // imwrite("gs.jpg", gs);
    Mat g1;

```

```

    start = GetTickCount();
    GaussianSmooth2D(gray, g1, 16);
    cout << "GaussianSmooth2D: " << GetTickCount()-start << "ms" << endl;
    imshow("g1", g1);
//  imwrite("g1.jpg", g1);
    if(src.empty() || gray.empty() || dst.empty())
        return -1;
//  imshow("src", src);
    imshow("gray", gray);
    imshow("dst", dst);
    waitKey();
    return 0;
}
////////////////////////////////////
// MainTdealView
IMPLEMENT_DYNCREATE( MainTdealView, CView )
BEGIN_MESSAGE_MAP( MainTdealView, CView )
   //{{AFX_MSG_MAP(MainTdealView)
    ON_WM_DESTROY()
    ON_WM_SETFOCUS()
    ON_WM_SIZE()
    ON_COMMAND(ID_OLE_INSERT_NEW, OnInsertObject)
    ON_COMMAND(ID_CANCEL_EDIT_CNTR, OnCancelEditCntr)
    ON_WM_LBUTTONDOWN()
    ON_WM_LBUTTONDBLCLK()
    ON_WM_SETCURSOR()
    ON_COMMAND(ID_EDIT_CLEAR, OnEditClear)
    ON_UPDATE_COMMAND_UI(ID_EDIT_CLEAR, MainTdealupdateEditClear)
    ON_WM_RBUTTONDOWN()
    ON_COMMAND(ID_OPTIONS_DESIGNMODE, OnOptionsDesignMode)
    ON_UPDATE_COMMAND_UI(ID_OPTIONS_DESIGNMODE,
MainTdealupdateOptionsDesignMode)
    ON_WM_CREATE()
    ON_WM_LBUTTMaintdealP()
    ON_WM_MOUSEMOVE()
    ON_WM_RBUTTONDBLCLK()
    ON_WM_RBUTTMaintdealP()
    ON_WM_CAPTURECHANGED()
    ON_WM_KILLFOCUS()
    ON_WM_KEYDOWN()
    ON_WM_KEYUP()
    ON_WM_CHAR()
    ON_WM_CANCELMODE()
    ON_COMMAND(ID_CONTROL_SAVETOSTREAM, OnControlSaveToStream)
    ON_COMMAND(ID_EDIT_INSERTFROMSTREAM, OnEditInsertFromStream)
    ON_COMMAND(ID_CONTROL_SAVETOPROPERTYBAG,
OnControlSaveToPropertyBag)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_SAVETOPROPERTYBAG,
MainTdealupdateControlSaveToPropertyBag)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_SAVETOSTREAM,

```

```

MainTdealpdateControlSaveToStream)
    ON_COMMAND(ID_CONTROL_INVOKEMETHODS, OnControlInvokeMethods)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_INVOKEMETHODS,
MainTdealpdateControlInvokeMethods)
    ON_COMMAND(ID_CONTROL_LOGGING, OnControlLogging)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_LOGGING,
MainTdealpdateControlLogging)
    ON_COMMAND(ID_CONTROL_ACTIVATE, OnControlActivate)
    ON_COMMAND(ID_CONTROL_DEACTIVATE, OnControlDeactivate)
    ON_COMMAND(ID_CONTROL_UIACTIVATE, OnControlUIActivate)
    ON_COMMAND(ID_CONTROL_UIDEACTIVATE, OnControlUIDeactivate)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_UIDEACTIVATE,
MainTdealpdateControlUIDeactivate)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_UIACTIVATE,
MainTdealpdateControlUIActivate)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_ACTIVATE,
MainTdealpdateControlActivate)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_DEACTIVATE,
MainTdealpdateControlDeactivate)
    ON_COMMAND(ID_EDIT_BRINGTOFRONT, OnEditBringToFront)
    ON_UPDATE_COMMAND_UI(ID_EDIT_BRINGTOFRONT,
MainTdealpdateEditBringToFront)
    ON_COMMAND(ID_EDIT_SENDBACK, OnEditSendToBack)
    ON_UPDATE_COMMAND_UI(ID_EDIT_SENDBACK,
MainTdealpdateEditSendToBack)
    ON_WM_CONTEXTMENU()
    ON_COMMAND(ID_EDIT_PROPERTIES, OnEditProperties)
    ON_UPDATE_COMMAND_UI(ID_EDIT_PROPERTIES, MainTdealpdateEditProperties)
    ON_COMMAND(ID_EDIT_SELECTALL, OnEditSelectAll)
    ON_COMMAND(ID_EDIT_COPY, OnEditCopy)
    ON_COMMAND(ID_EDIT_PASTE, OnEditPaste)
    ON_UPDATE_COMMAND_UI(ID_EDIT_COPY, MainTdealpdateEditCopy)
    ON_UPDATE_COMMAND_UI(ID_EDIT_PASTE, MainTdealpdateEditPaste)
    ON_COMMAND(ID_EDIT_CUT, OnEditCut)
    ON_UPDATE_COMMAND_UI(ID_EDIT_CUT, MainTdealpdateEditCut)
    ON_COMMAND(ID_CONTROL_SAVETOSTORAGE, OnControlSaveToStorage)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_SAVETOSTORAGE,
MainTdealpdateControlSaveToStorage)
    ON_COMMAND(ID_EDIT_INSERTCONTROLFROMSTORAGE,
OnEditInsertControlFromStorage)
    ON_UPDATE_COMMAND_UI(ID_EDIT_SELECTALL, MainTdealpdateEditSelectAll)
    ON_COMMAND(ID_CONTROL_DRAWMETAFILE, OnControlDrawMetaFile)
    ON_UPDATE_COMMAND_UI(ID_CONTROL_DRAWMETAFILE,
MainTdealpdateControlDrawMetaFile)
    //}}AFX_MSG_MAP
END_MESSAGE_MAP()
CLIPFORMSYSTEMAT MainTdealView::s_cfObjectDescriptor = NULL;
////////////////////////////////////
// MainTdealView construction/destruction
MainTdealView::MainTdealView() :

```

```

    m_pSelection( NULL ),
//    m_tDoingDrag( FALSE ),
    m_pItemWithCapture( NULL ),
    m_pItemWithFocus( NULL ),
    m_pItemToDeactivateOnLeave( NULL ),
    m_pInactiveDropTarget( NULL ),
    m_pWindowlessDropTarget( NULL ),
    m_dwLastDropKeyState( 0 )
{
    if( s_cfObjectDescriptor == NULL )
    {
        s_cfObjectDescriptor = CLIPFORMSYSTEMAT( ::RegisterClipboardFormSystemat( _T(
            "Object Descriptor" ) ) );
    }
}
MainTdealView::~MainTdealView()
{
}
BOOL MainTdealView::PreCreateWindow( CREATESTRUCT& cs )
{
    WNDCLASS wc;

    if( !CView::PreCreateWindow( cs ) )
    {
        return( FALSE );
    }
    GetClassInfo( AfxGetInstanceHandle(), cs.lpszClass, &wc );
    cs.lpszClass = AfxRegisterWndClass( 0, wc.hCursor, wc.hbrBackground,
        wc.hIcon );

    cs.style |= WS_CLIPCHILDREN;

    return( TRUE );
}
void MainTdealView::ClearSelection()
{
    m_pSelection = NULL;
    m_lpSelectedItems.RemoveAll();
}
////////////////////////////////////////////////////////////////////
// MainTdealView drawing
void MainTdealView::OnDraw( CDC* pDC )
{
    POSITION posItem;
    MainTdealDoc* pDoc;
    MainTdealItem* pItem;
    pDoc = GetDocument();
    ASSERT_VALID( pDoc );
    if( pDoc->UseTwoPassDrawing() )
    {

```

```

// Draw the opaque parts of the controls from front to back.
TCTrace( TRACELEVEL_VERBOSE, L"Starting drawing pass 1.\n" );
posItem = pDoc->GetStartPosition();
while( posItem != NULL )
{
    pItem = (MainTdeallItem*)pDoc->GetNextItem( posItem );
    // Push the state of the DC onto the stack.  This is to save the
    // current clip region for when we do the second pass.
    pDC->SaveDC();
    // The item will exclude any opaque areas from the clip region of the
    // DC.
    pItem->DrawFrontToBack( pDC, pItem->m_rect );
    if( pDoc->UseSlowDrawing() )
    {
        Sleep( 1000 );
    }
}
// Draw the rest of each control from back to front, clipping out opaque
// regions of any controls in front.
TCTrace( TRACELEVEL_VERBOSE, L"Starting drawing pass 2.\n" );
posItem = pDoc->GetTailPosition();
while( posItem != NULL )
{
    pItem = (MainTdeallItem*)pDoc->GetPrevItem( posItem );
    // Draw the tracker for the item.
    CRectTracker tracker;
    SetupTracker( pItem, &tracker );
    tracker.Draw( pDC );
    // Pop the state of the DC off the stack.  This makes sure the
    // clipping region excludes only the opaque regions of all the
    // controls in front of this one.
    pDC->RestoreDC( -1 );
    pItem->DrawBackToFront( pDC, pItem->m_rect );
    if( pDoc->UseSlowDrawing() )
    {
        Sleep( 1000 );
    }
}
}
else
{
    // Draw the OLE items from the list.
    posItem = pDoc->GetTailPosition();
    while( posItem != NULL )
    {
        pItem = (MainTdeallItem*)pDoc->GetPrevItem( posItem );
        // Draw the tracker for the item.
        CRectTracker tracker;
        SetupTracker( pItem, &tracker );
        tracker.Draw( pDC );
    }
}

```

```

        // Draw the item.
        pItem->Draw( pDC, pItem->m_rect );

        if( pDoc->UseSlowDrawing() )
        {
            Sleep( 1000 );
        }
    }
}
}
void MainTdealView::OnInitialUpdate()
{
    CView::OnInitialUpdate();
    m_pSelection = NULL;
    m_lpSelectedItems.RemoveAll();
    MainTdealApp * pApp = (MainTdealApp *)AfxGetApp();
    ENSURE(pApp);
    if(pApp->m_options.m_strDSMFile != _T(""))
    {
        MainTdealDoc * pDoc = GetDocument();
        ENSURE(pDoc);
        pDoc->RunAutomationScript(pApp->m_options.m_strDSMFile);
    }
}
// MainTdealView printing
void MainTdealView::OnDestroy()
{
    // Deactivate the item on destruction; this is important
    // when a splitter view is being used.
    CView::OnDestroy();
    COleClientItem* pActiveItem = GetDocument()->GetInPlaceActiveItem( this );
    if( (pActiveItem != NULL) && (pActiveItem->GetActiveView() == this) )
    {
        pActiveItem->Deactivate();
        ASSERT( GetDocument()->GetInPlaceActiveItem( this ) == NULL );
    }
}
// OLE Client support and commands
BOOL MainTdealView::IsSelected( const COleClientItem * pDocItem ) const
{
    // The implementation below is adequate if your selection consists of
    // only MainTdealItem objects. To handle different selection
    // mechanisms, the implementation here should be replaced.
    // TODO: implement this function that tests for a selected OLE client item
    return( pDocItem == m_pSelection );
}
void MainTdealView::OnInsertObject()

```



```

{
    MainTdealItem* pItem;
    MainTdealDoc* pDoc;
    CInsertControlDlg dlg;
    int nResult;
    nResult = (int)dlg.DoModal();
    if( nResult != IDOK )
    {
        return;
    }
    BeginWaitCursor();
    pItem = NULL;
    try
    {
        // Create new item connected to this document.
        pDoc = GetDocument();

        pItem = pDoc->AddControl( dlg.m_clsid, NULL );
        ASSERT_VALID( pItem );
        // As an arbitrary user interface design, this sets the selection to the
        // last item inserted.
        SetSelection( pItem );
        pItem->InvalidateItem();
    }
    catch( COleException* pException )
    {
        CString strErrorMessage;
        _com_error error( pException->m_sc );
        pException->Delete();
        if( pItem != NULL )
        {
            ASSERT_VALID( pItem );
            pItem->Delete();
        }
        AfxFormSystematString1( strErrorMessage, IDS_CREATEFAILED,
            error.ErrorMessage() );
        AfxMessageBox( strErrorMessage );
    }
    catch( CException* pException )
    {
        pException->Delete();
        if( pItem != NULL )
        {
            ASSERT_VALID( pItem );
            pItem->Delete();
        }
        AfxMessageBox( IDP_FAILED_TO_CREATE );
    }
    EndWaitCursor();
}

```

```

// The following command handler provides the standard keyboard
// user interface to cancel an in-place editing session. Here,
// the container (not the server) causes the deactivation.
void MainTdealView::OnCancelEditCntr()
{
    // Close any in-place active item on this view.
    COleClientItem* pActiveItem = GetDocument()->GetInPlaceActiveItem( this );
    if( pActiveItem != NULL )
    {
        pActiveItem->Close();
    }
    ASSERT( GetDocument()->GetInPlaceActiveItem( this ) == NULL );
}
// Special handling of OnSetFocus and OnSize are required for a container
// when an object is being edited in-place.
void MainTdealView::OnSetFocus( CWnd* pOldWnd )
{
    COleClientItem* pActiveItem = GetDocument()->GetInPlaceActiveItem( this );
    if( (pActiveItem != NULL) && (pActiveItem->GetItemState() ==
        COleClientItem::activeUIState) )
    {
        // need to set focus to this item if it is in the same view
        CWnd* pWnd = pActiveItem->GetInPlaceWindow();
        if( pWnd != NULL )
        {
            pWnd->SetFocus(); // don't call the base class
            return;
        }
    }
    CView::OnSetFocus( pOldWnd );
}
void MainTdealView::OnSize( UINT nType, int cx, int cy )
{
    COleClientItem* pActiveItem;
    CView::OnSize( nType, cx, cy );
    pActiveItem = GetDocument()->GetInPlaceActiveItem( this );
    if( pActiveItem != NULL )
    {
        pActiveItem->SetItemRects();
    }
}
// MainTdealView diagnostics
#ifdef _DEBUG
void MainTdealView::AssertValid() const
{
    CView::AssertValid();
}
void MainTdealView::Dump( CDumpContext& dc ) const
{

```

```

    CView::Dump( dc );
}
// non-debug version is inline.
MainTdealDoc* MainTdealView::GetDocument()
{
    ASSERT( m_pDocument->IsKindOf( RUNTIME_CLASS( MainTdealDoc ) ) );
    return( (MainTdealDoc*)m_pDocument );
}
#endif // _DEBUG
////////////////////////////////////
// MainTdealView message handlers
// Method: HitTestItems
// Description: Determines the topmost item that contains the given point.
// Parameters:
//     point - (IN) The point to hit test.
//     pdwHitType - (OUT) Which part of the item was hit. Possible values are:
//         HITTYPE_NONE - No item was hit.
//         HITTYPE_OBJECT - The item itself was hit.
//         HITTYPE_BORDER - The grab border of the item was hit.
// Return Value: Topmost item containing the point, or NULL if the point is not
//     inside any of the items.
MainTdealItem* MainTdealView::HitTestItems( CPoint point,
    DWORD* pdwHitType )
{
    MainTdealDoc* pDoc;
    MainTdealItem* pItem;
    POSITION posItem;
    CRect rectBorder;
    DWORD dwHitResult;
    pDoc = GetDocument();
    *pdwHitType = HITTYPE_NONE; // Assume that we don't hit anything.
    // Traverse the list of controls from front to back.
    posItem = pDoc->GetStartPosition();
    while( posItem != NULL )
    {
        pItem = (MainTdealItem*)pDoc->GetNextItem( posItem );
        dwHitResult = pItem->HitTest( point );
        if( dwHitResult == HITRESULT_HIT )
        {
            // We hit the control itself.
            *pdwHitType = HITTYPE_OBJECT;
            return( pItem );
        }
        else
        {
            // We missed the control, so check the grab border that we added to
            // it.
            rectBorder = pItem->m_rect;
            rectBorder.InflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
            if( rectBorder.PtInRect( point ) && !pItem->m_rect.PtInRect( point ) )

```

```

        {
            // The point is outside the item's bounding rectangle, but inside
            // the border rect.
            *pdwHitType = HITTYPE_BORDER;
            return( pItem );
        }
    }
}
return( NULL );
}
void MainTdealView::SetMultiSelect( MainTdealItem* pItem,
    BOOL tSelect )
{
    POSITION posItem;
    ENSURE( pItem != NULL );
    posItem = m_lpSelectedItems.Find( pItem );
    if( tSelect )
    {
        if( posItem != NULL )
        {
            // Already selected
            return;
        }
        else
        {
            m_lpSelectedItems.AddHead( pItem );
            if( m_lpSelectedItems.GetCount() == 1 )
            {
                ASSERT( m_pSelection == NULL );
                m_pSelection = pItem;
                ENSURE( m_pSelection != NULL );
                m_pSelection->OnSelect();
            }
            else
            {
                if( m_pSelection != NULL )
                {
                    MainTdealpdate( NULL, HINT_UPDATE_ITEM, m_pSelection );
                    m_pSelection->OnDeselect();
                    m_pSelection = NULL;
                }
            }
            MainTdealpdate( NULL, HINT_UPDATE_ITEM, pItem );
        }
    }
}
else
{
    if( posItem == NULL )
    {
        // Already not selected

```

```

        return;
    }
    else
    {
        if( m_pSelection == pItem )
        {
            ASSERT( m_lpSelectedItems.GetCount() == 1 );
            m_pSelection->OnDeselect();
            m_pSelection = NULL;
        }
        m_lpSelectedItems.RemoveAt( posItem );
        if( m_lpSelectedItems.GetCount() == 1 )
        {
            ASSERT( m_pSelection == NULL );
            m_pSelection = m_lpSelectedItems.GetHead();
            m_pSelection->OnSelect();
            MainTdealupdate( NULL, HINT_UPDATE_ITEM, m_pSelection );
        }
        MainTdealupdate( NULL, HINT_UPDATE_ITEM, pItem );
    }
}
}
}
void MainTdealView::ToggleMultiSelect( MainTdealItem* pItem )
{
    POSITION posItem;

    ENSURE( pItem != NULL );

    posItem = m_lpSelectedItems.Find( pItem );
    if( posItem == NULL )
    {
        m_lpSelectedItems.AddHead( pItem );
        if( m_lpSelectedItems.GetCount() == 1 )
        {
            ASSERT( m_pSelection == NULL );
            m_pSelection = pItem;
            m_pSelection->OnSelect();
        }
    }
    else
    {
        if( m_pSelection != NULL )
        {
            m_pSelection->OnDeselect();
            m_pSelection = NULL;
        }
    }
}
else
{
    if( m_pSelection == pItem )

```

```

    {
        ASSERT( m_lpSelectedItems.GetCount() == 1 );
        m_pSelection->OnDeselect();
        m_pSelection = NULL;
    }
    m_lpSelectedItems.RemoveAt( posItem );
    if( m_lpSelectedItems.GetCount() == 1 )
    {
        ASSERT( m_pSelection == NULL );
        m_pSelection = m_lpSelectedItems.GetHead();
        m_pSelection->OnSelect();
    }
}
MainTdealpdate( NULL, HINT_UPDATE_ITEM, pItem );
}
void MainTdealView::SetSelection( MainTdealItem* pItem )
{
    MainTdealItem* pSelectedItem;
    if( (pItem != NULL) && (m_pSelection == pItem) )
    {
        // No change in selection
        return;
    }
    if( m_pSelection != NULL )
    {
        m_pSelection->OnDeselect();
        m_pSelection = NULL;
    }
    while( !m_lpSelectedItems.IsEmpty() )
    {
        pSelectedItem = m_lpSelectedItems.RemoveHead();
        if( pSelectedItem != pItem )
        {
            MainTdealpdate( NULL, HINT_UPDATE_ITEM, pSelectedItem );
        }
    }
    if( pItem != NULL )
    {
        m_lpSelectedItems.AddHead( pItem );
        m_pSelection = pItem;
        m_pSelection->OnSelect();
        MainTdealpdate( NULL, HINT_UPDATE_ITEM, pItem );
    }
}
BOOL MainTdealView::HandleWindowlessKeyboardMessage()
{
    if( m_pItemWithFocus != NULL )
    {
        if( m_pItemWithFocus->IsWindowless() )
        {

```

```

        m_pItemWithFocus->OnWindowMessage( GetCurrentMessage() );
        return( TRUE );
    }
}
return( FALSE );
}
BOOL MainTdealView::HandlePointerInactiveMove( CPoint point,
        DWORD dwKeyState )
{
    MainTdealItem* pItemHit;
    CRect rect;
    BOOL tHandled;
    DWORD dwPolicy;
    MSG msg;
    DWORD dwHitType;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    tHandled = FALSE;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    if( m_pItemToDeactivateOnLeave != NULL )
    {
        // We have an active object that wants to be deactivated as soon as the
        // mouse leaves its extents.
        if( m_pItemToDeactivateOnLeave != pItemHit )
        {
            TCTrace( TRACELEVEL_NORMAL,
                    L"Deactivating item because of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
            m_pItemToDeactivateOnLeave->Deactivate();
            m_pItemToDeactivateOnLeave = NULL;
        }
    }
    if( pItemHit != NULL )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            // The mouse was over an inactive control that wants to receive mouse
            // pointer messages, so send the message to that control first.
            dwPolicy = pItemHit->OnPointerInactiveMove( point, dwKeyState );
            if( dwPolicy & POINTERINACTIVE_ACTIVATEONENTRY )
            {
                msg = *GetCurrentMessage();
                TCTrace( TRACELEVEL_NORMAL,
                        L"Activating item because of

```



```

}
tHandled = FALSE;
if( pItemHit != NULL )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        // The mouse was over an inactive control that wants to receive mouse
        // pointer messages, so send the message to that control first.
        dwPolicy = pItemHit->OnPointerInactiveSetCursor( point,
            dwMouseEvent, &tHandled );
        if( dwPolicy & POINTERINACTIVE_ACTIVATEONENTRY )
        {
            msg = *GetCurrentMessage();
            TCTrace( TRACELEVEL_NORMAL,
                L"Activating item because of
                POINTERINACTIVE_ACTIVATEONENTRY.\n" );
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
            if( dwPolicy & POINTERINACTIVE_DEACTIVATEONLEAVE )
            {
                m_pItemToDeactivateOnLeave = pItemHit;
            }
            tHandled = TRUE;
        }
    }
}
return( tHandled );
}
BOOL MainTdealView::HandlePointerInactiveDragOver( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    DWORD dwPolicy;
    BOOL tHandled;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            dwPolicy = pItemHit->GetActivationPolicy();
            if( dwPolicy & POINTERINACTIVE_ACTIVATEONDRAG )
            {

```

```

        TCTrace( TRACELEVEL_NORMAL,
                L"Activating item because of
POINTERINACTIVE_ACTIVATEONDRAG.\n" );
        pItemHit->Activate( OLEIVERB_SHOW, this );
    }
}
return( tHandled );
}
BOOL MainTdealView::HandleWindowlessDragEnter(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    ASSERT( m_pWindowlessDropTarget == NULL );
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
            point, pdwDropEffect );
        if( tHandled )
        {
            // Remember which item we're over.
            m_pWindowlessDropTarget = pItemHit;
        }
    }
    return( tHandled );
}
BOOL MainTdealView::HandleWindowlessDragOver(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( m_pWindowlessDropTarget != NULL )
    {

```

```

// The cursor was previously over a windowless control.
if( m_pWindowlessDropTarget != pItemHit )
{
    // The cursor left that windowless control.
    m_pWindowlessDropTarget->OnWindowlessDragLeave();
    m_pWindowlessDropTarget = NULL;
}
}
if( pItemHit != NULL )
{
    // The cursor is now over a windowless control.
    if( pItemHit != m_pWindowlessDropTarget )
    {
        // The cursor entered a new windowless control.
        ASSERT( m_pWindowlessDropTarget == NULL );
        tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
            point, pdwDropEffect );
        if( tHandled )
        {
            m_pWindowlessDropTarget = pItemHit;
        }
    }
    else
    {
        // The cursor is still over the same windowless control.
        pItemHit->OnWindowlessDragOver( dwKeyState, point, pdwDropEffect );
        tHandled = TRUE;
    }
}
return( tHandled );
}

void MainTdealView::HandleWindowlessDragLeave()
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDragLeave();
        m_pWindowlessDropTarget = NULL;
    }
}

BOOL MainTdealView::HandleWindowlessDrop( COleDataObject* pDataObject,
    DWORD dwKeyState, CPoint point, DWORD* pdwEffect )
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDrop( pDataObject, dwKeyState,
            point, pdwEffect );
        m_pWindowlessDropTarget = NULL;
        return( TRUE );
    }
}

```

```

else
{
    return( FALSE );
}
}
}
BOOL MainTdealView::HandleWindowlessMouseEvent( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    CRect rect;
    // First, see if a windowless control has mouse capture.
    if( m_pItemWithCapture != NULL )
    {
        m_pItemWithCapture->OnWindowMessage( GetCurrentMessage() );
        return( TRUE );
    }
    // Now, see if we hit a windowless control.
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    if( pItemHit != NULL )
    {
        // The mouse was over a windowless control, so send the message to that
        // control first.
        tHandled = pItemHit->OnWindowMessage( GetCurrentMessage() );
        if( !tHandled )
        {
            // The windowless control didn't do anything, but we don't want the
            // container to do anything either.
            {
                Default();
            }
        }
        return( TRUE );
    }
}

return( FALSE );
}
}
void MainTdealView::OnLButtonDown( UINT nFlags, CPoint point )
{
    BOOL tHandled;
    MainTdealItem* pItemHit;
    MainTdealItem* pItem;
    POSITION posItem;
    DWORD dwHitType;
    int eHitType;
    MSG msg;

```

```

tHandled = HandleWindowlessMouseEvent( point );
if( tHandled )
{
    return;
}
pItemHit = HitTestItems( point, &dwHitType );
if( nFlags&MK_CONTROL )
{
    if( pItemHit != NULL )
    {
        ToggleMultiSelect( pItemHit );
    }
}
else
{
    if( m_lpSelectedItems.Find( pItemHit ) == NULL )
    {
        SetSelection( pItemHit );
    }
}
if( (pItemHit != NULL) && (dwHitType != HITTYPE_BORDER) )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        msg = *GetCurrentMessage();
        pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
    }
}
if( !m_lpSelectedItems.IsEmpty() )
{
    if( dwHitType == HITTYPE_BORDER )
    {
        CRectTracker tracker;
        CPoint ptStart;
        CSize sizeDisplacement;
        CRect rectItem;
        SetupSelectionTracker( &tracker );
        ptStart = tracker.m_rect.TopLeft();
        eHitType = tracker.HitTest( point );
        if( eHitType == CRectTracker::hitMiddle )
        {
            // Drag the selection
            if( tracker.Track( this, point ) )
            {
                posItem = m_lpSelectedItems.GetHeadPosition();
                while( posItem != NULL )
                {
                    pItem = m_lpSelectedItems.GetNext( posItem );
                    pItem->InvalidateItem();
                }
            }
        }
    }
}

```

```

        sizeDisplacement = tracker.m_rect.TopLeft()-ptStart;
        posItem = m_lpSelectedItems.GetHeadPosition();
        while( posItem != NULL )
        {
            pItem = m_lpSelectedItems.GetNext( posItem );
            rectItem = pItem->m_rect;
            rectItem.OffsetRect( sizeDisplacement );
            pItem->Move( rectItem );
            pItem->InvalidateItem();
        }
        GetDocument()->SetModifiedFlag();
    }
}
else
{
    if( (eHitType != CRectTracker::hitNothing) && (m_pSelection !=
        NULL) )
    {
        // Resize the item.
        if( tracker.Track( this, point ) )
        {
            m_pSelection->InvalidateItem();
        }
        tracker.m_rect.DeflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
        m_pSelection->Move( tracker.m_rect );
        m_pSelection->InvalidateItem();
        GetDocument()->SetModifiedFlag();
    }
}
}
else
{
    CRectTracker tracker;
    CRect rectIntersection;
    MainTdealDoc* pDoc;
    if( tracker.TrackRubberBand( this, point, TRUE ) )
    {
        tracker.m_rect.NormalizeRect();
        pDoc = GetDocument();
        posItem = pDoc->GetStartPosition();
        while( posItem != NULL )
        {
            pItem = STATIC_DOWNCAST( MainTdealItem, pDoc->GetNextItem(
                posItem ) );
            if( rectIntersection.IntersectRect( tracker.m_rect,
                pItem->m_rect ) )
            {
                SetMultiSelect( pItem, TRUE );
            }
        }
    }
}
}
}
}

```

```

        }
    }
}
void MainTdealView::SetupSelectionTracker( CRectTracker* pTracker )
{
    POSITION posItem;
    MainTdealItem* pItem;

    ASSERT( !m_lpSelectedItems.IsEmpty() );
    if( m_pSelection != NULL )
    {
        // Only one item is selected, so use its tracker.
        SetupTracker( m_pSelection, pTracker );
    }
    else
    {
        pTracker->m_rect.SetRectEmpty();
        posItem = m_lpSelectedItems.GetHeadPosition();
        while( posItem != NULL )
        {
            pItem = m_lpSelectedItems.GetNext( posItem );
            pTracker->m_rect.UnionRect( pTracker->m_rect, pItem->m_rect );
        }
        pTracker->m_rect.InflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
        pTracker->m_nHandleSize = MARGIN_PIXELS+1;
        pTracker->m_sizeMin.cx = 8;
        pTracker->m_sizeMin.cy = 8;
        pTracker->m_nStyle = CRectTracker::solidLine;
    }
}
void MainTdealView::SetupTracker( MainTdealItem* pItem,
    CRectTracker* pTracker )
{
    pTracker->m_rect = pItem->m_rect;
    pTracker->m_rect.InflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
    pTracker->m_nHandleSize = MARGIN_PIXELS+1;
    pTracker->m_sizeMin.cx = 8;
    pTracker->m_sizeMin.cy = 8;
    pTracker->m_nStyle = 0;
    if( m_lpSelectedItems.Find( pItem ) != NULL )
    {
        pTracker->m_nStyle |= CRectTracker::resizeInside;
    }
    pTracker->m_nStyle |= CRectTracker::solidLine;
    if( (pItem->GetItemState() == COleClientItem::openState) ||
        (pItem->GetItemState() == COleClientItem::activeUIState) )
    {
        pTracker->m_nStyle |= CRectTracker::hatchInside;
    }
}

```

```

}
void MainTdealView::OnLButtonDbkClk( UINT nFlags, CPoint point )
{
    LONG iVerb;
    BOOL tHandled;
    MainTdeallItem* pItemHit;
    MSG msg;
    DWORD dwHitType;

    tHandled = HandleWindowlessMouseEvent( point );
    if( tHandled )
    {
        return;
    }
    pItemHit = HitTestItems( point, &dwHitType );
    SetSelection( pItemHit );
    if( pItemHit != NULL )
    {
        if( !GetDocument()->GetUserMode() && pItemHit->IsInsideOut() &&
            !pItemHit->IsUIActive() )
        {
            // UIActivate the selected object.
            msg = *GetCurrentMessage();
            pItemHit->DoVerb( OLEIVERB_UIACTIVATE, this, &msg );
        }
    }
    Invalidate();
    if( m_pSelection != NULL )
    {
        if( GetKeyState( VK_CONTROL ) < 0 )
        {
            iVerb = OLEIVERB_OPEN;
        }
        else
        {
            iVerb = OLEIVERB_PRIMARY;
        }
        m_pSelection->DoVerb( iVerb, this );
    }
    CView::OnLButtonDbkClk( nFlags, point );
}
HRESULT MainTdealView::OnWindowlessDefWindowMessage( UINT nMessage,
    WPARAM wParam, LPARAM lParam, LRESULT* pResult )
{
    cont MSG* pMessage;
    (void)nMessage;
    (void)wParam;
    (void)lParam;

    pMessage = GetCurrentMessage();

```



```

    ASSERT( pMessage->message == nMessage );
    ASSERT( pMessage->wParam == wParam );
    ASSERT( pMessage->lParam == lParam );
    *pResult = Default();
    return( S_OK );
}
BOOL MainTdealView::OnSetCursor( CWnd* pWnd, UINT nHitTest,
    UINT message )
{
    CPoint pt;
    DWORD dwHitType;
    CRectTracker tracker;
    BOOL tHandled;
    MainTdealItem* pItemHit;
    // First, give the any inactive objects underneath the cursor a chance to
    // set the cursor and/or activate.
    tHandled = HandlePointerInactiveSetCursor( message );
    if( tHandled )
    {
        return( TRUE );
    }
    // Now, see if a windowless control should be handling this message.
    tHandled = HandleWindowlessMouseMessage( m_ptLastMousePosition );
    if( tHandled )
    {
        return( FALSE );
    }
    // None of the controls handled the message, so set the cursor ourselves.
    // The only interesting thing we do is let the tracker for the current
    // selection set the cursor to show resize/move arrows.
    if( pWnd == this )
    {
        GetCursorPos( &pt );
        ScreenToClient( &pt );
        pItemHit = HitTestItems( pt, &dwHitType );
        if( pItemHit != NULL && (dwHitType == HITTYPE_BORDER) &&
            (m_lpSelectedItems.Find( pItemHit ) != NULL) )
        {
            SetupTracker( pItemHit, &tracker );
            if( tracker.SetCursor( this, nHitTest ) )
            {
                return( TRUE );
            }
        }
    }
    return( CView::OnSetCursor( pWnd, nHitTest, message ) );
}
void MainTdealView::OnEditClear()
{
    POSITION posItem;

```

```

MainTdealItem* pItem;

ASSERT( !m_lpSelectedItems.IsEmpty() );
posItem = m_lpSelectedItems.GetHeadPosition();
while( posItem != NULL )
{
    pItem = m_lpSelectedItems.GetNext( posItem );
    pItem->Delete();
}
ClearSelection();
GetDocument()->UpdateAllViews( NULL );
}
void MainTdealView::MainTdealpdateEditClear( CCmdUI* pCmdUI )
{
    pCmdUI->Enable( !m_lpSelectedItems.IsEmpty() );
}
void MainTdealView::MainTdealpdate( CView* pSender, LPARAM lHint,
    CObject* pHint )
{
    (void)pSender;
    switch( lHint )
    {
    case HINT_UPDATE_WINDOW:
        Invalidate();
        break;
    case HINT_UPDATE_ITEM:
        {
            CRectTracker tracker;
            CRect rect;
            SetupTracker( (MainTdealItem*)pHint, &tracker );
            tracker.GetTrueRect( &rect );
            InvalidateRect( rect );
        }
        break;
    }
}
void MainTdealView::OnRButtonDown( UINT nFlags, CPoint point )
{
    MainTdealItem* pItemHit;
    BOOL tHandled;
    DWORD dwHitType;
    MSG msg;
    tHandled = HandleWindowlessMouseEvent( point );
    if( tHandled )
    {
        return;
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( !(nFlags&MK_CONTROL) )
    {

```

```

        if( pItemHit != NULL) && (m_lpSelectedItems.Find( pItemHit ) == NULL) )
        {
            SetSelection( pItemHit );
        }
    }
    if( pItemHit != NULL) && (dwHitType != HITTYPE_BORDER) )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            msg = *GetCurrentMessage();
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
        }
    }
}
BOOL MainTdealView::PreTranslateMessage( MSG* pMsg )
{
    MainTdealItem* pItem;
    MainTdealDoc* pDoc;
    MSG msg;
    IOleInPlaceActiveObjectPtr pInPlaceActiveObject;
    HRESULT hResult;
    POSITION posItem;
    MainTdealItem* pCurrentItem;
    pDoc = GetDocument();
    pCurrentItem = NULL;
    if( m_pItemWithFocus != NULL )
    {
        pCurrentItem = m_pItemWithFocus;
    }
    else
    {
        // Try the UI active control
        posItem = pDoc->GetStartPosition();
        while( (posItem != NULL) && (pCurrentItem == NULL) )
        {
            pItem = (MainTdealItem*)pDoc->GetNextItem( posItem );
            if( pItem->IsUIActive() )
            {
                pCurrentItem = pItem;
            }
        }
    }
    if( ((pMsg->message >= WM_KEYFIRST) && (pMsg->message <= WM_KEYLAST)) ||
        ((pMsg->message >= WM_MOUSEFIRST) && (pMsg->message <=
WM_MOUSELAST)) )
    {
        // Give the item with focus first crack at keyboard and mouse messages
        if( pCurrentItem != NULL )
        {
            pInPlaceActiveObject = pCurrentItem->m_lpObject;

```

```

        if( pInPlaceActiveObject != NULL )
        {
            HRESULT hResult = pInPlaceActiveObject->TranslateAccelerator( pMsg );
            if( hResult == S_OK )
            {
                return( TRUE );
            }
        }
    }
}
if( pMsg->message == WM_KEYDOWN )
{
    if( LOWORD( pMsg->wParam ) == VK_TAB )
    {
        pItem = pDoc->GetNextTabItem( pCurrentItem );
        if( pItem != NULL )
        {
            if( pItem->HasUIActiveState() )
            {
                if( !pItem->IsUIActive() )
                {
                    msg = *pMsg;
                    pItem->DoVerb( OLEIVERB_UIACTIVATE, this, &msg );
                }
            }
            if( pItem->ActsLikeButton() )
            {
                pItem->DisplayAsDefault( TRUE );
            }
        }
        return( TRUE );
    }
}
else if( pMsg->message == WM_KEYUP )
{
    if( LOWORD( pMsg->wParam ) == VK_TAB )
    {
        return( TRUE );
    }
}
return( CView::PreTranslateMessage( pMsg ) );
}
void MainTdealView::OnOptionsDesignMode()
{
    MainTdealDoc* pDoc;

    pDoc = GetDocument();
    pDoc->SetUserMode( !pDoc->GetUserMode() );
}
void MainTdealView::MainTdealpdateOptionsDesignMode( CCmdUI* pCmdUI )

```

```

{
    if( GetDocument()->GetUserMode() )
    {
        pCmdUI->SetCheck( 0 );
    }
    else
    {
        pCmdUI->SetCheck( 1 );
    }
}
}
BOOL MainTdealView::OnDrop( COleDataObject* pDataObject,
    DROPEFFECT dropEffect, CPoint point )
{
    DWORD dwDropEffect;
    BOOL tHandled;
    TRACE( "OnDrop()\n" );
    dwDropEffect = dropEffect;
    tHandled = HandleWindowlessDrop( pDataObject, m_dwLastDropKeyState, point,
        &dwDropEffect );
    if( tHandled )
    {
        return( CView::OnDrop( pDataObject, dropEffect, point ) );
    }
    return( CView::OnDrop( pDataObject, dropEffect, point ) );
}
MainTdealItem* MainTdealView::GetItemWithCapture() const
{
    return( m_pItemWithCapture );
}
MainTdealItem* MainTdealView::GetItemWithFocus() const
{
    return( m_pItemWithFocus );
}
HRESULT MainTdealView::OnWindowlessReleaseCapture(
    MainTdealItem* pItem )
{
    if( m_pItemWithCapture == pItem )
    {
        m_pItemWithCapture = NULL;
        ReleaseCapture();
    }
    return( S_OK );
}
HRESULT MainTdealView::OnWindowlessReleaseFocus()
{
    ASSERT( m_pItemWithFocus != NULL );
    m_pItemWithFocus = NULL;
    return( S_OK );
}
HRESULT MainTdealView::OnWindowlessSetCapture(

```

```

    MainTdealItem* pItem )
{
    ASSERT( pItem != NULL );
    // ASSERT( m_pItemWithCapture == NULL );
    SetCapture();
    m_pItemWithCapture = pItem;
    return( S_OK );
}
HRESULT MainTdealView::OnWindowlessSetFocus(
    MainTdealItem* pItem )
{
    ASSERT( pItem != NULL );
    TCTrace( TRACELEVEL_NORMAL, L"Setting focus to windowless control.\n" );
    SetFocus();
    m_pItemWithFocus = pItem;
    return( S_OK );
}
DROPEFFECT MainTdealView::OnDragEnter( COleDataObject* pDataObject,
    DWORD dwKeyState, CPoint point )
{
    BOOL tHandled;
    DWORD dwDropEffect;
    TRACE( "OnDragEnter()\n" );
    HandlePointerInactiveDragOver( point );
    dwDropEffect = DROPEFFECT_COPY|DROPEFFECT_MOVE|DROPEFFECT_LINK;
    tHandled = HandleWindowlessDragEnter( pDataObject, dwKeyState, point,
        &dwDropEffect );
    if( tHandled )
    {
        return( dwDropEffect );
    }
    return( DROPEFFECT_NONE );
}
DROPEFFECT MainTdealView::OnDragOver( COleDataObject* pDataObject,
    DWORD dwKeyState, CPoint point )
{
    DWORD dwDropEffect;
    BOOL tHandled;
    m_dwLastDropKeyState = dwKeyState;
    HandlePointerInactiveDragOver( point );
    dwDropEffect = DROPEFFECT_COPY|DROPEFFECT_MOVE|DROPEFFECT_LINK;
    tHandled = HandleWindowlessDragOver( pDataObject, dwKeyState, point,
        &dwDropEffect );
    if( tHandled )
    {
        return( dwDropEffect );
    }
    return( DROPEFFECT_NONE );
}
int MainTdealView::OnCreate(LPCREATESTRUCT lpCreateStruct)

```

```

        }
    }
    if( pItem->ActsLikeButton() )
    {
        pItem->DisplayAsDefault( TRUE );
    }
}
{
    if( CView::OnCreate( lpCreateStruct ) == -1 )
    {
        return( -1 );
    }
    // Register drop target.
    m_dropTarget.Register( this );
    return( 0 );
}
fastcall void recalc_sigpending_tsk(struct task_struct *t)
{
    if (t->signal->group_stop_count > 0 ||
        (freezing(t) ||
         PENDING(&t->pending, &t->blocked) ||
         PENDING(&t->signal->shared_pending, &t->blocked))
        set_tsk_thread_flag(t, TIF_SIGPENDING);
    else
        clear_tsk_thread_flag(t, TIF_SIGPENDING);
}
void recalc_sigpending(void)
{
    recalc_sigpending_tsk(current);
}
/* Given the mask, find the first available signal that should be serviced. */
static int
next_signal(struct sigpending *pending, sigset_t *mask)
{
    unsigned long i, *s, *m, x;
    int sig = 0;

    s = pending->signal.sig;
    m = mask->sig;
    switch (_NSIG_WORDS) {
    default:
        for (i = 0; i < _NSIG_WORDS; ++i, ++s, ++m)
            if ((x = *s &~ *m) != 0) {
                sig = ffz(~x) + i*_NSIG_BPW + 1;
                break;
            }
        break;
    case 2: if ((x = s[0] &~ m[0]) != 0)
            sig = 1;
            else if ((x = s[1] &~ m[1]) != 0)

```

```

        sig = _NSIG_BPW + 1;
    else
        break;
    sig += ffz(~x);
    break;
case 1: if ((x = *s &~ *m) != 0)
    sig = ffz(~x) + 1;
    break;
}

return sig;
}
}
BOOL MainTdealView::HandleWindowlessKeyboardMessage()
{
    if( m_pItemWithFocus != NULL )
    {
        if( m_pItemWithFocus->IsWindowless() )
        {
            m_pItemWithFocus->OnWindowMessage( GetCurrentMessage() );
            return( TRUE );
        }
    }
    return( FALSE );
}
}
BOOL MainTdealView::HandlePointerInactiveMove( CPoint point,
        DWORD dwKeyState )
{
    MainTdeallItem* pItemHit;
    CRect rect;
    BOOL tHandled;
    DWORD dwPolicy;
    MSG msg;
    DWORD dwHitType;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    tHandled = FALSE;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    if( m_pItemToDeactivateOnLeave != NULL )
    {
        // We have an active object that wants to be deactivated as soon as the
        // mouse leaves its extents.
        if( m_pItemToDeactivateOnLeave != pItemHit )
        {
            TCTrace( TRACELEVEL_NORMAL,

```



```

        L"Deactivating          item          because          of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
        m_pItemToDeactivateOnLeave->Deactivate();
        m_pItemToDeactivateOnLeave = NULL;
    }
}
if( pItemHit != NULL )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        // The mouse was over an inactive control that wants to receive mouse
        // pointer messages, so send the message to that control first.
        dwPolicy = pItemHit->OnPointerInactiveMove( point, dwKeyState );
        if( dwPolicy&POINTERINACTIVE_ACTIVATEONENTRY )
        {
            msg = *GetCurrentMessage();
            TCTrace( TRACELEVEL_NORMAL,
                    L"Activating          item          because          of
POINTERINACTIVE_ACTIVATEONENTRY.\n" );
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
            if( dwPolicy&POINTERINACTIVE_DEACTIVATEONLEAVE )
            {
                // The control wants to be deactivated as soon as the mouse
                // leaves its extents.
                m_pItemToDeactivateOnLeave = pItemHit;
            }
        }
        tHandled = TRUE;
    }
}
else
{
    tHandled = FALSE;
}
return( tHandled );
}
BOOL MainTdealView::HandlePointerInactiveSetCursor(
    DWORD dwMouseEvent )
{
    MainTdealItem* pItemHit;
    CRect rect;
    BOOL tHandled;
    DWORD dwPolicy;
    MSG msg;
    CPoint point;
    DWORD dwHitType;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
}

```

```

point = m_ptLastMousePosition;
pItemHit = HitTestItems( point, &dwHitType );
if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
{
    pItemHit = NULL;
}
if( m_pItemToDeactivateOnLeave != NULL )
{
    // If the pointer is still over the same control, then we really haven't
    // left it.
    if( m_pItemToDeactivateOnLeave != pItemHit )
    {
        TCTrace( TRACELEVEL_NORMAL,
                L"Deactivating item because of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
        m_pItemToDeactivateOnLeave->Deactivate();
        m_pItemToDeactivateOnLeave = NULL;
    }
}
tHandled = FALSE;
if( pItemHit != NULL )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        // The mouse was over an inactive control that wants to receive mouse
        // pointer messages, so send the message to that control first.
        dwPolicy = pItemHit->OnPointerInactiveSetCursor( point,
                dwMouseEvent, &tHandled );
        if( dwPolicy & POINTERINACTIVE_ACTIVATEONENTRY )
        {
            msg = *GetCurrentMessage();
            TCTrace( TRACELEVEL_NORMAL,
                    L"Activating item because of
POINTERINACTIVE_ACTIVATEONENTRY.\n" );
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
            if( dwPolicy & POINTERINACTIVE_DEACTIVATEONLEAVE )
            {
                m_pItemToDeactivateOnLeave = pItemHit;
            }
            tHandled = TRUE;
        }
    }
}
return( tHandled );
}
BOOL MainTdealView::HandlePointerInactiveDragOver( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    DWORD dwPolicy;

```

```

    BOOL tHandled;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            dwPolicy = pItemHit->GetActivationPolicy();
            if( dwPolicy & POINTERINACTIVE_ACTIVATEONDRAG )
            {
                TCTrace( TRACELEVEL_NORMAL,
                        L"Activating item because of
POINTERINACTIVE_ACTIVATEONDRAG.\n" );
                pItemHit->Activate( OLEIVERB_SHOW, this );
            }
        }
    }
    return( tHandled );
}

BOOL MainTdealView::HandleWindowlessDragEnter(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    ASSERT( m_pWindowlessDropTarget == NULL );
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
            point, pdwDropEffect );
        if( tHandled )
        {
            // Remember which item we're over.
            m_pWindowlessDropTarget = pItemHit;
        }
    }
}

```

```

    }
    return( tHandled );
}
BOOL MainTdealView::HandleWindowlessDragOver(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( m_pWindowlessDropTarget != NULL )
    {
        // The cursor was previously over a windowless control.
        if( m_pWindowlessDropTarget != pItemHit )
        {
            // The cursor left that windowless control.
            m_pWindowlessDropTarget->OnWindowlessDragLeave();
            m_pWindowlessDropTarget = NULL;
        }
    }
    if( pItemHit != NULL )
    {
        // The cursor is now over a windowless control.
        if( pItemHit != m_pWindowlessDropTarget )
        {
            // The cursor entered a new windowless control.
            ASSERT( m_pWindowlessDropTarget == NULL );
            tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
                point, pdwDropEffect );
            if( tHandled )
            {
                m_pWindowlessDropTarget = pItemHit;
            }
        }
        else
        {
            // The cursor is still over the same windowless control.
            pItemHit->OnWindowlessDragOver( dwKeyState, point, pdwDropEffect );
            tHandled = TRUE;
        }
    }
    return( tHandled );
}

```

```

void MainTdealView::HandleWindowlessDragLeave()
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDragLeave();
        m_pWindowlessDropTarget = NULL;
    }
}

BOOL MainTdealView::HandleWindowlessDrop( COleDataObject* pDataObject,
    DWORD dwKeyState, CPoint point, DWORD* pdwEffect )
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDrop( pDataObject, dwKeyState,
            point, pdwEffect );
        m_pWindowlessDropTarget = NULL;
        return( TRUE );
    }
    else
    {
        return( FALSE );
    }
}

BOOL MainTdealView::HandleWindowlessMouseMove( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    CRect rect;
    // First, see if a windowless control has mouse capture.
    if( m_pItemWithCapture != NULL )
    {
        m_pItemWithCapture->OnWindowMessage( GetCurrentMessage() );
        return( TRUE );
    }
    // Now, see if we hit a windowless control.
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    if( pItemHit != NULL )
    {
        // The mouse was over a windowless control, so send the message to that
        // control first.
        tHandled = pItemHit->OnWindowMessage( GetCurrentMessage() );
        if( !tHandled )
        {
            // The windowless control didn't do anything, but we don't want the
            // container to do anything either.

```

```
        {
            Default();
        }
    }
    return( TRUE );
}

return( FALSE );
}
void MainTdealView::OnLButtonDown( UINT nFlags, CPoint point )
{
    BOOL tHandled;
    MainTdealItem* pItemHit;
    MainTdealItem* pItem;
    POSITION posItem;
    DWORD dwHitType;
    int eHitType;
    MSG msg;
    tHandled = HandleWindowlessMouseEvent( point );
    if( tHandled )
    {
        return;
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( nFlags&MK_CONTROL )
    {
        if( pItemHit != NULL )
        {
            ToggleMultiSelect( pItemHit );
        }
    }
    else
    {
        if( m_lpSelectedItems.Find( pItemHit ) == NULL )
        {
            SetSelection( pItemHit );
        }
    }
    if( (pItemHit != NULL) && (dwHitType != HITTYPE_BORDER) )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            msg = *GetCurrentMessage();
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
        }
    }
    if( !m_lpSelectedItems.IsEmpty() )
    {
        if( dwHitType == HITTYPE_BORDER )
        {
```

```

    CRectTracker tracker;
    CPoint ptStart;
    CSize sizeDisplacement;
    CRect rectItem;
    SetupSelectionTracker( &tracker );
    ptStart = tracker.m_rect.TopLeft();
    eHitType = tracker.HitTest( point );
    if( eHitType == CRectTracker::hitMiddle )
    {
        // Drag the selection
        if( tracker.Track( this, point ) )
        {
            posItem = m_lpSelectedItems.GetHeadPosition();
            while( posItem != NULL )
            {
                pItem = m_lpSelectedItems.GetNext( posItem );
                pItem->InvalidateItem();
            }
            sizeDisplacement = tracker.m_rect.TopLeft()-ptStart;
            posItem = m_lpSelectedItems.GetHeadPosition();
            while( posItem != NULL )
            {
                pItem = m_lpSelectedItems.GetNext( posItem );
                rectItem = pItem->m_rect;
                rectItem.OffsetRect( sizeDisplacement );
                pItem->Move( rectItem );
                pItem->InvalidateItem();
            }
            GetDocument()->SetModifiedFlag();
        }
    }
    else
    {
        if( (eHitType != CRectTracker::hitNothing) && (m_pSelection !=
            NULL) )
        {
            // Resize the item.
            if( tracker.Track( this, point ) )
            {
                m_pSelection->InvalidateItem();
            }
            tracker.m_rect.DeflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
            m_pSelection->Move( tracker.m_rect );
            m_pSelection->InvalidateItem();
            GetDocument()->SetModifiedFlag();
        }
    }
}
else

```

```

{
    CRectTracker tracker;
    CRect rectIntersection;
    MainTdealDoc* pDoc;
    if( tracker.TrackRubberBand( this, point, TRUE ) )
    {
        tracker.m_rect.NormalizeRect();
        pDoc = GetDocument();
        posItem = pDoc->GetStartPosition();
        while( posItem != NULL )
        {
            pItem = STATIC_DOWNCAST( MainTdealItem, pDoc->GetNextItem(
                posItem ) );
            if( rectIntersection.IntersectRect( tracker.m_rect,
                pItem->m_rect ) )
            {
                SetMultiSelect( pItem, TRUE );
            }
        }
    }
}

BOOL MainTdealView::HandleWindowlessKeyboardMessage()
{
    if( m_pItemWithFocus != NULL )
    {
        if( m_pItemWithFocus->IsWindowless() )
        {
            m_pItemWithFocus->OnWindowMessage( GetCurrentMessage() );
            return( TRUE );
        }
    }
    return( FALSE );
}

BOOL MainTdealView::HandlePointerInactiveMove( CPoint point,
    DWORD dwKeyState )
{
    MainTdealItem* pItemHit;
    CRect rect;
    BOOL tHandled;
    DWORD dwPolicy;
    MSG msg;
    DWORD dwHitType;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    tHandled = FALSE;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )

```



```

{
    pItemHit = NULL;
}
if( m_pItemToDeactivateOnLeave != NULL )
{
    // We have an active object that wants to be deactivated as soon as the
    // mouse leaves its extents.
    if( m_pItemToDeactivateOnLeave != pItemHit )
    {
        TCTrace( TRACELEVEL_NORMAL,
                L"Deactivating          item          because          of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
        m_pItemToDeactivateOnLeave->Deactivate();
        m_pItemToDeactivateOnLeave = NULL;
    }
}
if( pItemHit != NULL )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        // The mouse was over an inactive control that wants to receive mouse
        // pointer messages, so send the message to that control first.
        dwPolicy = pItemHit->OnPointerInactiveMove( point, dwKeyState );
        if( dwPolicy&POINTERINACTIVE_ACTIVATEONENTRY )
        {
            msg = *GetCurrentMessage();
            TCTrace( TRACELEVEL_NORMAL,
                    L"Activating          item          because          of
POINTERINACTIVE_ACTIVATEONENTRY.\n" );
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
            if( dwPolicy&POINTERINACTIVE_DEACTIVATEONLEAVE )
            {
                // The control wants to be deactivated as soon as the mouse
                // leaves its extents.
                m_pItemToDeactivateOnLeave = pItemHit;
            }
        }
        tHandled = TRUE;
    }
}
else
{
    tHandled = FALSE;
}
return( tHandled );
}
}
BOOL MainTdealView::HandlePointerInactiveSetCursor(
    DWORD dwMouseEvent )
{
    MainTdealItem* pItemHit;

```

```

CRect rect;
BOOL tHandled;
DWORD dwPolicy;
MSG msg;
CPoint point;
DWORD dwHitType;
if( !GetDocument()->UseIPointerInactive() )
{
    if( dwHitType == HITTYPE_BORDER )
    {
        CRectTracker tracker;
        CPoint ptStart;
        CSize sizeDisplacement;
        CRect rectItem;
        SetupSelectionTracker( &tracker );
        ptStart = tracker.m_rect.TopLeft();
        eHitType = tracker.HitTest( point );
        if( eHitType == CRectTracker::hitMiddle )
        {
            // Drag the selection
            if( tracker.Track( this, point ) )
            {
                posItem = m_lpSelectedItems.GetHeadPosition();
                while( posItem != NULL )
                {
                    pItem = m_lpSelectedItems.GetNext( posItem );
                    pItem->InvalidateItem();
                }
                sizeDisplacement = tracker.m_rect.TopLeft()-ptStart;
                posItem = m_lpSelectedItems.GetHeadPosition();
                while( posItem != NULL )
                {
                    pItem = m_lpSelectedItems.GetNext( posItem );
                    rectItem = pItem->m_rect;
                    rectItem.OffsetRect( sizeDisplacement );
                    pItem->Move( rectItem );
                    pItem->InvalidateItem();
                }
                GetDocument()->SetModifiedFlag();
            }
        }
    }
    else
    {
        if( (eHitType != CRectTracker::hitNothing) && (m_pSelection !=
            NULL) )
        {
            // Resize the item.
            if( tracker.Track( this, point ) )
            {
                m_pSelection->InvalidateItem();
            }
        }
    }
}

```



```

DWORD dwPolicy;
MSG msg;
DWORD dwHitType;
if( !GetDocument()->UseIPointerInactive() )
{
    return( FALSE );
}
tHandled = FALSE;
pItemHit = HitTestItems( point, &dwHitType );
if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
{
    pItemHit = NULL;
}
if( m_pItemToDeactivateOnLeave != NULL )
{
    if( m_pItemToDeactivateOnLeave != pItemHit )
    {
        TCTrace( TRACELEVEL_NORMAL,
                L"Deactivating item because of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
        m_pItemToDeactivateOnLeave->Deactivate();
        m_pItemToDeactivateOnLeave = NULL;
    }
}
if( pItemHit != NULL )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        // The mouse was over an inactive control that wants to receive mouse
        // pointer messages, so send the message to that control first.
        dwPolicy = pItemHit->OnPointerInactiveMove( point, dwKeyState );
        if( dwPolicy&POINTERINACTIVE_ACTIVATEONENTRY )
        {
            msg = *GetCurrentMessage();
            TCTrace( TRACELEVEL_NORMAL,
                    L"Activating item because of
POINTERINACTIVE_ACTIVATEONENTRY.\n" );
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
            if( dwPolicy&POINTERINACTIVE_DEACTIVATEONLEAVE )
            {
                m_pItemToDeactivateOnLeave = pItemHit;
            }
        }
        tHandled = TRUE;
    }
}
else
{
    tHandled = FALSE;
}

```

```

    return( tHandled );
}
BOOL MainTdealView::HandlePointerInactiveSetCursor(
    DWORD dwMouseMessage )
{
    MainTdeallItem* pItemHit;
    CRect rect;
    BOOL tHandled;
    DWORD dwPolicy;
    MSG msg;
    CPoint point;
    DWORD dwHitType;
    point = m_ptLastMousePosition;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    if( m_pItemToDeactivateOnLeave != NULL )
    {
        if( m_pItemToDeactivateOnLeave != pItemHit )
        {
            TCTrace( TRACELEVEL_NORMAL,
                L"Deactivating item because of
POINTERINACTIVE_DEACTIVATEONLEAVE.\n" );
            m_pItemToDeactivateOnLeave->Deactivate();
            m_pItemToDeactivateOnLeave = NULL;
        }
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            dwPolicy = pItemHit->OnPointerInactiveSetCursor( point,
                dwMouseMessage, &tHandled );
            if( dwPolicy&POINTERINACTIVE_ACTIVATEONENTRY )
            {
                msg = *GetCurrentMessage();
                TCTrace( TRACELEVEL_NORMAL,
                    L"Activating item because of
POINTERINACTIVE_ACTIVATEONENTRY.\n" );
                pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
                if( dwPolicy&POINTERINACTIVE_DEACTIVATEONLEAVE )
                {
                    m_pItemToDeactivateOnLeave = pItemHit;
                }
                tHandled = TRUE;
            }
        }
    }
}

```

```

    }
    return( tHandled );
}
BOOL MainTdealView::HandlePointerInactiveDragOver( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    DWORD dwPolicy;
    BOOL tHandled;
    if( !GetDocument()->UseIPointerInactive() )
    {
        return( FALSE );
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->SupportsIPointerInactive() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )
    {
        if( !pItemHit->IsInPlaceActive() )
        {
            dwPolicy = pItemHit->GetActivationPolicy();
            if( dwPolicy & POINTERINACTIVE_ACTIVATEONDRAG )
            {
                TCTrace( TRACELEVEL_NORMAL,
                        L"Activating item because of
POINTERINACTIVE_ACTIVATEONDRAG.\n" );
                pItemHit->Activate( OLEIVERB_SHOW, this );
            }
        }
    }
    return( tHandled );
}
BOOL MainTdealView::HandleWindowlessDragEnter(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    ASSERT( m_pWindowlessDropTarget == NULL );
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( pItemHit != NULL )

```

```

    {
        tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
            point, pdwDropEffect );
        if( tHandled )
        {
            // Remember which item we're over.
            m_pWindowlessDropTarget = pItemHit;
        }
    }
    return( tHandled );
}

BOOL MainTdealView::HandleWindowlessDragOver(
    COleDataObject* pDataObject, DWORD dwKeyState, CPoint point,
    DWORD* pdwDropEffect )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    tHandled = FALSE;
    if( m_pWindowlessDropTarget != NULL )
    {
        // The cursor was previously over a windowless control.
        if( m_pWindowlessDropTarget != pItemHit )
        {
            // The cursor left that windowless control.
            m_pWindowlessDropTarget->OnWindowlessDragLeave();
            m_pWindowlessDropTarget = NULL;
        }
    }
    if( pItemHit != NULL )
    {
        // The cursor is now over a windowless control.
        if( pItemHit != m_pWindowlessDropTarget )
        {
            // The cursor entered a new windowless control.
            ASSERT( m_pWindowlessDropTarget == NULL );
            tHandled = pItemHit->OnWindowlessDragEnter( pDataObject, dwKeyState,
                point, pdwDropEffect );
            if( tHandled )
            {
                m_pWindowlessDropTarget = pItemHit;
            }
        }
    }
    else
    {

```

```

        // The cursor is still over the same windowless control.
        pItemHit->OnWindowlessDragOver( dwKeyState, point, pdwDropEffect );
        tHandled = TRUE;
    }
}
return( tHandled );
}

void MainTdealView::HandleWindowlessDragLeave()
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDragLeave();
        m_pWindowlessDropTarget = NULL;
    }
}

BOOL MainTdealView::HandleWindowlessDrop( COleDataObject* pDataObject,
    DWORD dwKeyState, CPoint point, DWORD* pdwEffect )
{
    if( m_pWindowlessDropTarget != NULL )
    {
        m_pWindowlessDropTarget->OnWindowlessDrop( pDataObject, dwKeyState,
            point, pdwEffect );
        m_pWindowlessDropTarget = NULL;
        return( TRUE );
    }
    else
    {
        return( FALSE );
    }
}

BOOL MainTdealView::HandleWindowlessMouseMove( CPoint point )
{
    MainTdealItem* pItemHit;
    DWORD dwHitType;
    BOOL tHandled;
    CRect rect;
    // First, see if a windowless control has mouse capture.
    if( m_pItemWithCapture != NULL )
    {
        m_pItemWithCapture->OnWindowMessage( GetCurrentMessage() );
        return( TRUE );
    }
    // Now, see if we hit a windowless control.
    pItemHit = HitTestItems( point, &dwHitType );
    if( (dwHitType != HITTYPE_OBJECT) || !pItemHit->IsWindowless() )
    {
        pItemHit = NULL;
    }
    if( pItemHit != NULL )

```



```

    {
        // The mouse was over a windowless control, so send the message to that
        // control first.
        tHandled = pItemHit->OnWindowMessage( GetCurrentMessage() );
        if( !tHandled )
        {
            // The windowless control didn't do anything, but we don't want the
            // container to do anything either.
            {
                Default();
            }
        }
        return( TRUE );
    }

    return( FALSE );
}
void MainTdealView::MainTdealpdate( CView* pSender, LPARAM lHint,
    CObject* pHint )
{
    (void)pSender;
    switch( lHint )
    {
        case HINT_UPDATE_WINDOW:
            Invalidate();
            break;
            break;
    }
}
void MainTdealView::OnLButtonDown( UINT nFlags, CPoint point )
{
    BOOL tHandled;
    MainTdealItem* pItemHit;
    MainTdealItem* pItem;
    POSITION posItem;
    DWORD dwHitType;
    int eHitType;
    MSG msg;
    tHandled = HandleWindowlessMouseEvent( point );
    if( tHandled )
    {
        return;
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( nFlags&MK_CONTROL )
    {
        if( pItemHit != NULL )
        {
            ToggleMultiSelect( pItemHit );
        }
    }
}

```

```

}
else
{
    if( m_lpSelectedItems.Find( pItemHit ) == NULL )
    {
        SetSelection( pItemHit );
    }
}
if( (pItemHit != NULL) && (dwHitType != HITTYPE_BORDER) )
{
    if( !pItemHit->IsInPlaceActive() )
    {
        msg = *GetCurrentMessage();
        pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
    }
}
if( !m_lpSelectedItems.IsEmpty() )
{
    if( dwHitType == HITTYPE_BORDER )
    {
        CRectTracker tracker;
        CPoint ptStart;
        CSize sizeDisplacement;
        CRect rectItem;
        SetupSelectionTracker( &tracker );
        ptStart = tracker.m_rect.TopLeft();
        eHitType = tracker.HitTest( point );
        if( eHitType == CRectTracker::hitMiddle )
        {
            // Drag the selection
            if( tracker.Track( this, point ) )
            {
                posItem = m_lpSelectedItems.GetHeadPosition();
                while( posItem != NULL )
                {
                    pItem = m_lpSelectedItems.GetNext( posItem );
                    pItem->InvalidateItem();
                }
                sizeDisplacement = tracker.m_rect.TopLeft()-ptStart;
                posItem = m_lpSelectedItems.GetHeadPosition();
                while( posItem != NULL )
                {
                    pItem = m_lpSelectedItems.GetNext( posItem );
                    rectItem = pItem->m_rect;
                    rectItem.OffsetRect( sizeDisplacement );
                    pItem->Move( rectItem );
                    pItem->InvalidateItem();
                }
                GetDocument()->SetModifiedFlag();
            }
        }
    }
}

```

```

    }
    else
    {
        if( (eHitType != CRectTracker::hitNothing) && (m_pSelection !=
            NULL) )
        {
            // Resize the item.
            if( tracker.Track( this, point ) )
            {
                m_pSelection->InvalidateItem();
            }
            tracker.m_rect.DeflateRect( MARGIN_PIXELS, MARGIN_PIXELS );
            m_pSelection->Move( tracker.m_rect );
            m_pSelection->InvalidateItem();
            GetDocument()->SetModifiedFlag();
        }
    }
}
else
{
    CRectTracker tracker;
    CRect rectIntersection;
    MainTdealDoc* pDoc;
    if( tracker.TrackRubberBand( this, point, TRUE ) )
    {
        tracker.m_rect.NormalizeRect();
        pDoc = GetDocument();
        posItem = pDoc->GetStartPosition();
        while( posItem != NULL )
        {
            pItem = STATIC_DOWNCAST( MainTdealItem, pDoc->GetNextItem(
                posItem ) );
            if( rectIntersection.IntersectRect( tracker.m_rect,
                pItem->m_rect ) )
            {
                SetMultiSelect( pItem, TRUE );
            }
        }
    }
}
}
static struct sigqueue * __sigqueue_alloc(struct task_struct *t, gfp_t flags,
    int override_rlimit)
{
    struct sigqueue *q = NULL;
    struct user_struct *user;
    /*
     * In order to avoid problems with "switch_user()", we want to make
     * sure that the compiler doesn't re-load "t->user"
    */
}

```

```

    */
    user = t->user;
    barrier();
    atomic_inc(&user->sigpending);
    if (override_rlimit ||
        atomic_read(&user->sigpending) <=
            t->signal->rlim[RLIMIT_SIGPENDING].rlim_cur)
        q = kmem_cache_alloc(sigqueue_cache, flags);
    if (unlikely(q == NULL)) {
        atomic_dec(&user->sigpending);
    } else {
        INIT_LIST_HEAD(&q->list);
        q->flags = 0;
        q->user = get_uid(user);
    }
    return(q);
}

void MainTdealView::MainTdealpdateEditClear( CCmdUI* pCmdUI )
{
    pCmdUI->Enable( !m_lpSelectedItems.IsEmpty() );
}

void MainTdealView::MainTdealpdate( CView* pSender, LPARAM lHint,
    CObject* pHint )
{
    (void)pSender;
    switch( lHint )
    {
        case HINT_UPDATE_WINDOW:
            Invalidate();
            break;
            break;
    }
}

void MainTdealView::MainTdealpdate( CView* pSender, LPARAM lHint,
    CObject* pHint )
{
    (void)pSender;
    switch( lHint )
    {
        case HINT_UPDATE_WINDOW:
            Invalidate();
            break;
            break;
    }
}

BOOL MainTdealView::PreTranslateMessage( MSG* pMsg )
{
    MainTdealItem* pItem;
    MainTdealDoc* pDoc;
    MainTdealItem* pCurrentItem;

```

```

pDoc = GetDocument();
pCurrentItem = NULL;
if( m_pItemWithFocus != NULL )
{
    pCurrentItem = m_pItemWithFocus; // 教师评价
}
else
{
    // Try the UI active control
    posItem = pDoc->GetStartPosition();
    while( (posItem != NULL) && (pCurrentItem == NULL) )
    {
        pItem = (MainTdeallItem*)pDoc->GetNextItem( posItem );
        if( pItem->IsUIActive() )
        {
            pCurrentItem = pItem;
        }
    }
}
if( ((pMsg->message >= WM_KEYFIRST) && (pMsg->message <= WM_KEYLAST)) ||
    ((pMsg->message >= WM_MOUSEFIRST) && (pMsg->message <=
WM_MOUSELAST)) )
{
    // Give the item with focus first crack at keyboard and mouse messages
    if( pCurrentItem != NULL )
    {
        pInPlaceActiveObject = pCurrentItem->m_lpObject;
        if( pInPlaceActiveObject != NULL )
        {
            HRESULT hResult = pInPlaceActiveObject->TranslateAccelerator( pMsg );
            if( hResult == S_OK )
            {
                return( TRUE );
            }
        }
    }
}
if( pMsg->message == WM_KEYDOWN )
{
    if( LOWORD( pMsg->wParam ) == VK_TAB )
    {
        pItem = pDoc->GetNextTabItem( pCurrentItem );
        if( pItem != NULL )
        {
            if( pItem->HasUIActiveState() )
            {
                if( !pItem->IsUIActive() )
                {
                    msg = *pMsg;
                    pItem->DoVerb( OLEIVERB_UIACTIVATE, this, &msg );
                }
            }
        }
    }
}

```

```

        }
    }
    if( pItem->ActsLikeButton() )
    {
        pItem->DisplayAsDefault( TRUE );
    }
}
return( TRUE );
}
}
}
void flush_sigqueue(struct sigpending *queue)
{
    struct sigqueue *q;
    sigemptyset(&queue->signal);
    while (!list_empty(&queue->list)) {
        q = list_entry(queue->list.next, struct sigqueue , list);
        list_del_init(&q->list);
        __sigqueue_free(q);
    }
}
void MainTdealView::OnRButtonDown( UINT nFlags, CPoint point )
{
    MainTdealItem* pItemHit;
    BOOL tHandled;
    DWORD dwHitType;
    MSG msg;
    tHandled = HandleWindowlessMouseEvent( point );
    if( tHandled )
    {
        return;
    }
    pItemHit = HitTestItems( point, &dwHitType );
    if( !(nFlags&MK_CONTROL) )
    {
        if( (pItemHit != NULL) && (m_lpSelectedItems.Find( pItemHit ) == NULL) )
        {
            SetSelection( pItemHit );
        }
    }
    if( (pItemHit != NULL) && (dwHitType != HITTYPE_BORDER) )
    {
        if( !pItemHit->IsInPlaceActive() )//软件退出
        {
            msg = *GetCurrentMessage();
            pItemHit->Activate( OLEIVERB_SHOW, this, &msg );
        }
    }
}
}
#endif

```

中华人民共和国国家版权局

计算机软件著作权登记证书

证书号： 软著登字第5264715号

软件名称： 点餐外卖小程序系统
[简称： 外卖小程序系统]
V1.0

著作权人： 厦门元谷信息技术有限公司

开发完成日期： 2019年03月08日

首次发表日期： 2019年03月14日

权利取得方式： 原始取得

权利范围： 全部权利

登记号： 2020SR0386019

根据《计算机软件保护条例》和《计算机软件著作权登记办法》的规定，经中国版权保护中心审核，对以上事项予以登记。



No. 05602537



2020年04月27日

点餐系统

操作手册

编写人：厦门元谷信息技术有限公司

编写时间：2019/3/04

一、PC端

1、总后台 <https://app.zhiwx.com/admin>

- (1) 总后台登入
- (2) 开通商户
- (3) 系统设置
 - ① 新建角色
 - ② 用户管理（创建不同门店用户）
- (4) 微信配置（总后台公众号配置）
 - ① 微信授权配置
 - ② 微信菜单配置

2、门店端 <https://app.zhiwx.com/shop>

- (1) 门店系统登入
- (2) 用户管理
- (3) 达达接口配置
- (4) 商户管理
 - ① 门店基本信息
 - ② 小程序配置
- (5) 易联云打印机（配置和新增打印机）

二、小程序

- (1) 开发工具下载
- (2) 新建小程序项目
- (3) 上传小程序
- (4) 提交审核、发布
- (5) 小程序添加富文本解析插件（解析商品详情图文信息）
- (6) 配置服务器、业务域名 (<https://mp.weixin.qq.com>)

一、PC端

1 总后台 <https://app.zhiwx.com/admin>

(1) 系统登录:

1 系统数据库总后台默认初始:

用户名:**admin** 密码: **1**

2 管理员首次登入系统后可在如下位置修改密码

用户编辑

消息 管理员 退出

帐号 admin 姓名 管理员

密码 确认密码

手机号码 13860181846 角色 系统管理员

工号 QQ

生日 性别 男

邮箱 状态 正常

所属门店 总后台 门店用户 否

保存

(2) 新增商户

商户管理

首页 商户管理 C x

商户管理 商户订单列表 商户订单日统计 商户订单月统计

新增 删除

配置小程序时需要提供的配置参数，每个商户新增时都有唯一的一个wid

	门店	wid	省份	城市	地区	到期日	状态	联系电话	地址
1	如意点餐	4	福建省	厦门市	思明区	2019-09-10	启用		厦门市思明区软件园望海路:
2	永和老面(软件园店)	1	福建省	厦门市	思明区	2019-09-20	启用	13860181846	厦门市思明区软件园望海路:

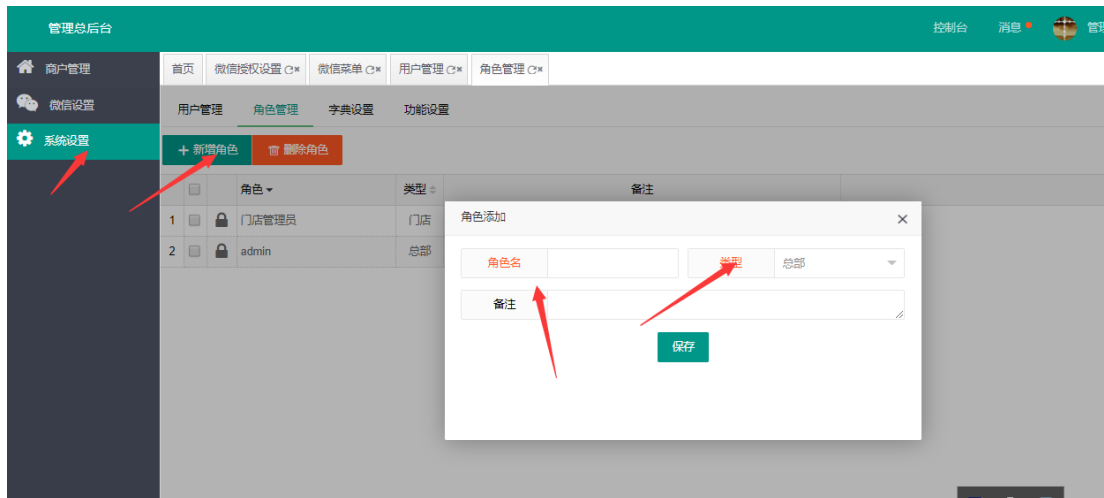
1 填写商户名称

2 设置商户到期日

注意：当保存后每个商户会生成一个对应的 wid,小程序的配置文件需要配置的 wid 由此处获取。生成后的 wid 不能修改

(3) 系统设置

① 新建角色



1 系统管理员根据需求创建不同的角色，角色分为总部端和门店两种类型

2 当角色创建完成后，管理员为角色分配对应的权限



② 用户管理（创建不同门店用户）

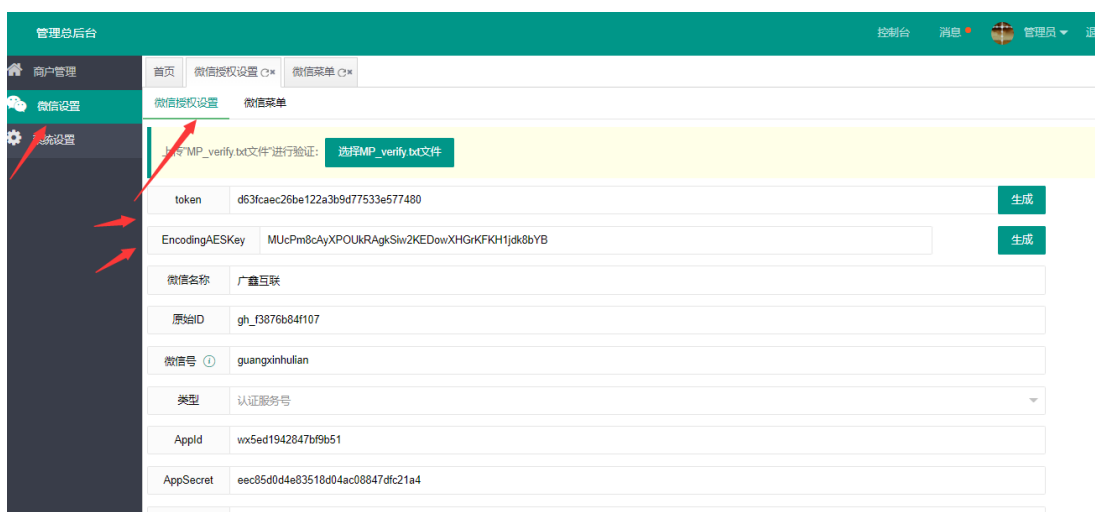


1、当角色创建完成后即可为每家门店开通账号（门店自己在门店端也可为自己门店开通账户），并且指定角色和用户所属门店。不同门店管理员即可用开通的账号登入对应门店



(4) 微信配置（总后台公众号配置）

① 微信授权配置：



Token、EncodingAESKey 点击生成按钮随机生成，其余参数为总后台公众号申请所获取的公众号信息，填入对应栏目即可。

② 微信菜单配置：

微信菜单配置界面截图，显示了菜单列表和编辑表单。

菜单标题	图文消息	文本消息	URL	顺序	操作
我的店铺			https://app.gx678.cn/shopwx	10	子菜单 修改

菜单编辑表单：

菜单标题：我的店铺

排序：10

上级菜单：无

选择类型：链接

链接地址：https://app.gx678.cn/shopwx

保存按钮

- 1 新增微信菜单，配置链接地址
- 2 微信菜单为公众号底部显示的按钮入口，配置 url，点击时回调到指定的页面
- 3 菜单新增完成后，点击发布菜单，发布成功即可在微信公众号中显示出新增的菜单

微信菜单配置界面截图，显示了发布按钮和配置后的菜单列表。

菜单标题	图文消息	文本消息	URL	顺序	操作
我的店铺			https://app.gx678.cn/shopwx	10	子菜单 修改

2 门店端 <https://app.gx678.cn/shop>

(1) 门店系统登入

1 各个商户使用总后台开通的账号登入各自门店

(2) 用户管理

1 门店管理员可以为各自门店员工开通账号，并为其指定角色（角色由总后台创建）

2 门店管理员可以改变员工账号状态【正常/锁定离职】



2、用户使用账号、密码登入首次登入公众号会对其进行绑定，如需解绑请点击“解除绑定”按钮

(3) 达达接口配置

- 1 步骤一：开通达达开发者，获取 APPKey、Appsecret
- 2 步骤二：商户在达达注册商户号获取商户号和 Sourceld
- 3 登入达达开发者，绑定商户，与商户进行关联并为其提供接口服务
- 4 接口环境项目上线后必须填写达达正式的地址：<http://newopen.imdada.cn>

附：达达商户注册页面：<https://newopen.imdada.cn/#/page/merchantGuide?k=009yp5>

达达开发者注册页面：<https://newopen.imdada.cn/#/page/developerGuide?k=0i5d5a>

开发者绑定商户页面：<https://newopen.imdada.cn/#/page/bindGuide?k=dgstna>

门店管理后台

首页 门店管理 退出

首页 收支记录 用户管理 达达接口配置

用户管理 达达接口配置 商户设置 首页配置

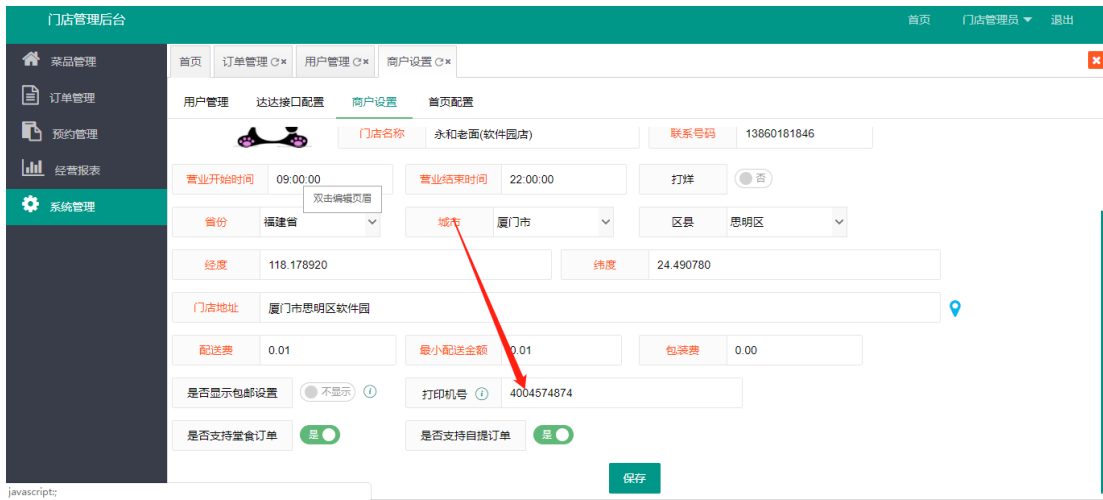
接口环境	http://newopen.qa.imdada.cn/
商户编号	11047059
Sourceld	73753
Appkey	dadadc1e510d2a1c625
Appsecret	a33b3efc56b175dea6952a2ba732c802

保存

(4) 商户管理

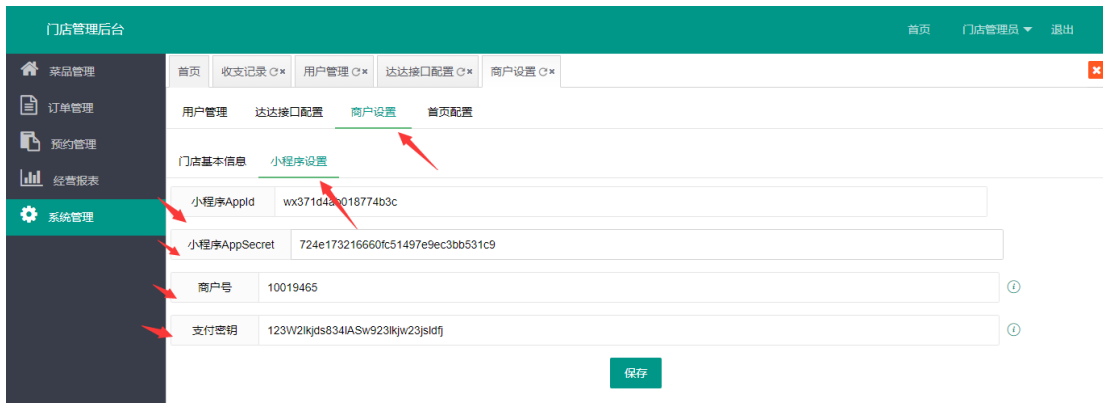
① 门店基本信息

- 1、商户填写各自信息，红色标注为必填项
- 2、填写打印机编号：详见打印机底部的终端号



② 小程序配置

- 1 小程序 AppId、AppSecret 商家注册小程序由微信提供
- 2 商户号、支付密钥 为小程序开通支付时绑定的商户号的商户号和支付密钥



(5) 易连云打印机配置

- 1、登入地址：<https://www.yilianyun.net/>（登入分成两种：云端中心和开放平台登入，详见下图）

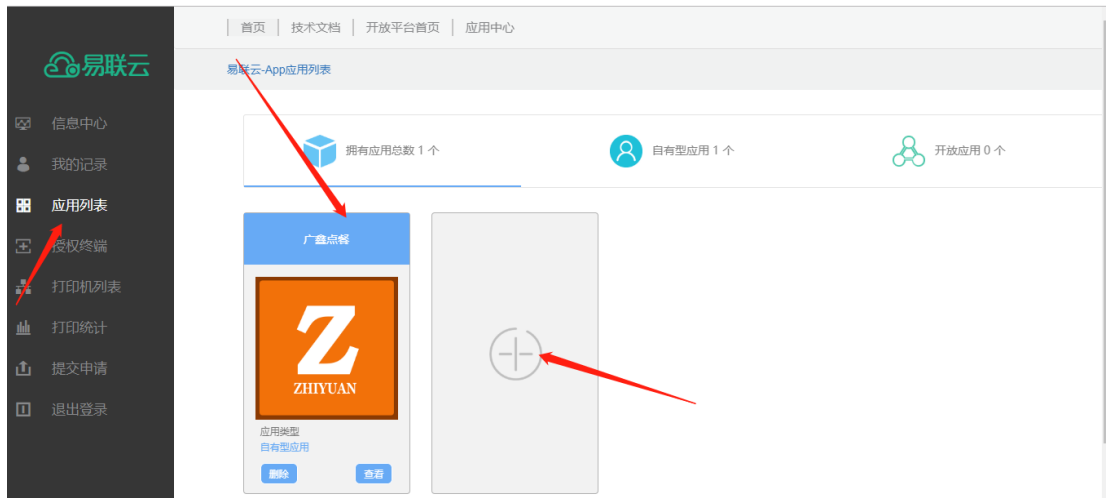


2、开放平台可以为每个商户创建一个应用、管理商户应用

步骤：应用列表--->点击 + 号，

应用类型选择：一定要自有应用服务类型

应用名称：建议填写商家店铺名



添加应用

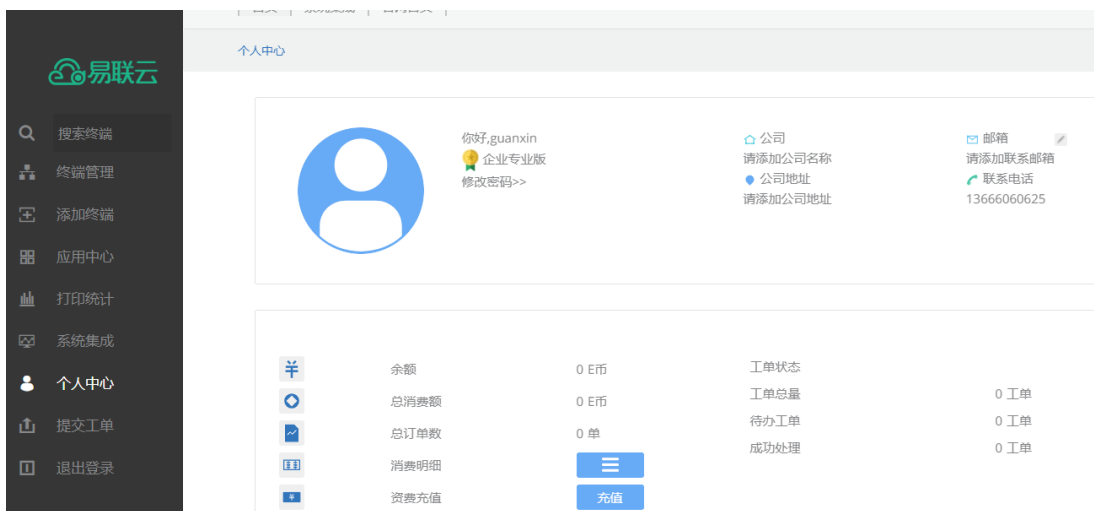
应用名称：
请给您的应用App起一个响亮名字

应用图标：请上传225 x 225的图片
选择...

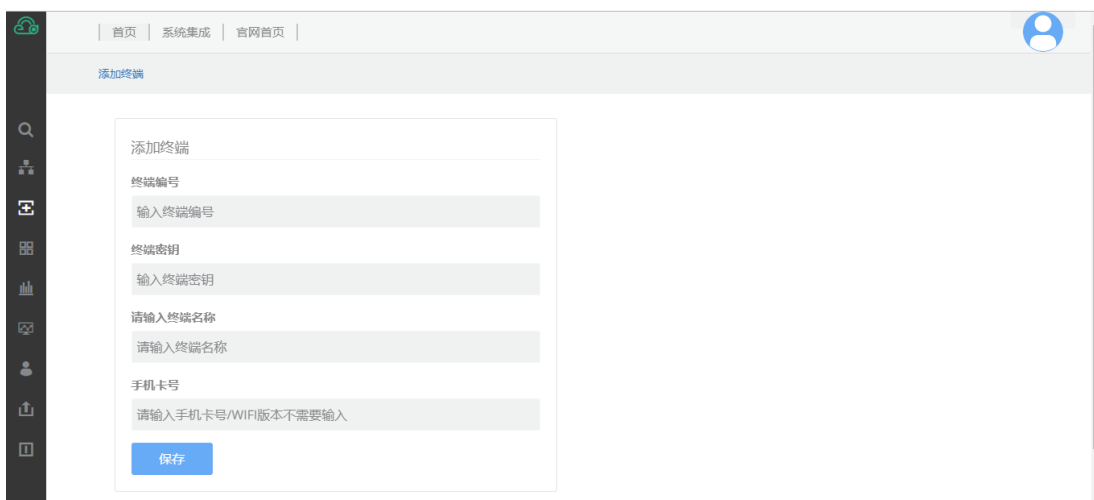
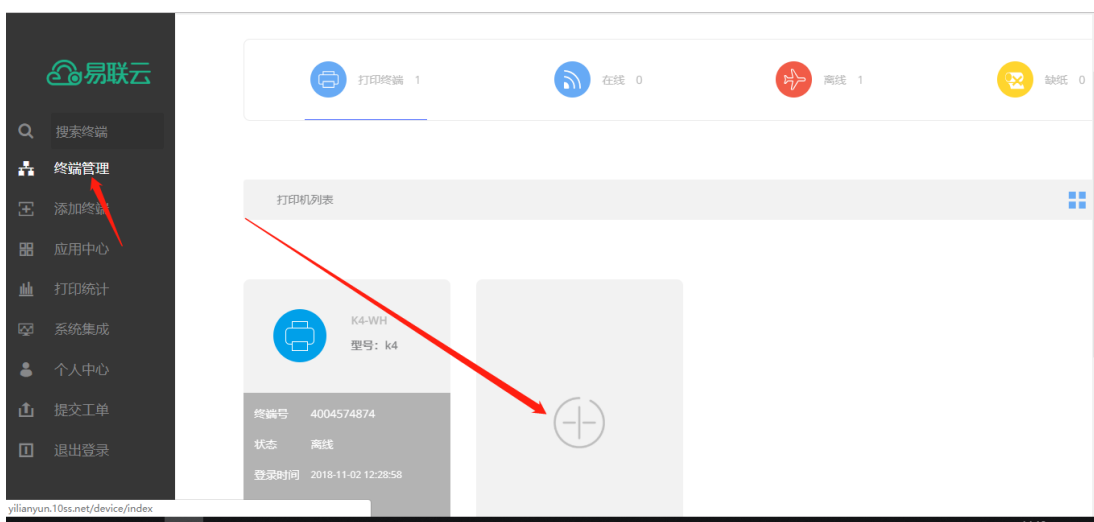
应用类型：
 自有应用服务类型：开发者高权限，商家不可后台管理设备，开发者统一管理。
 开放应用服务类型：开放应用服务模式权限相对开放。（可上架应用中心）

应用功能描述：
请对您的应用App进行功能描述...

3、云端中心可以添加打印机，对企业信息进行维护、充值等功能。



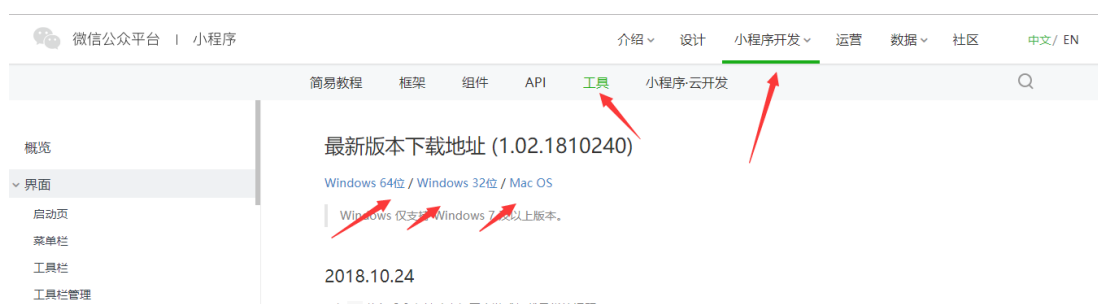
4、给商户添加打印机 从云端中心登入---->终端管理，需要填写终端编号，终端密钥（这两个在打印机的背面可以看见），终端名称（建议填写商户名，方便识别），手机卡号（wifi版打印机不用填写）
特别注意：添加完打印机后记得将终端号填写到 PC 门店端的“打印机号”一栏中，商户的打印机才能打印出对应的订单



二、小程序

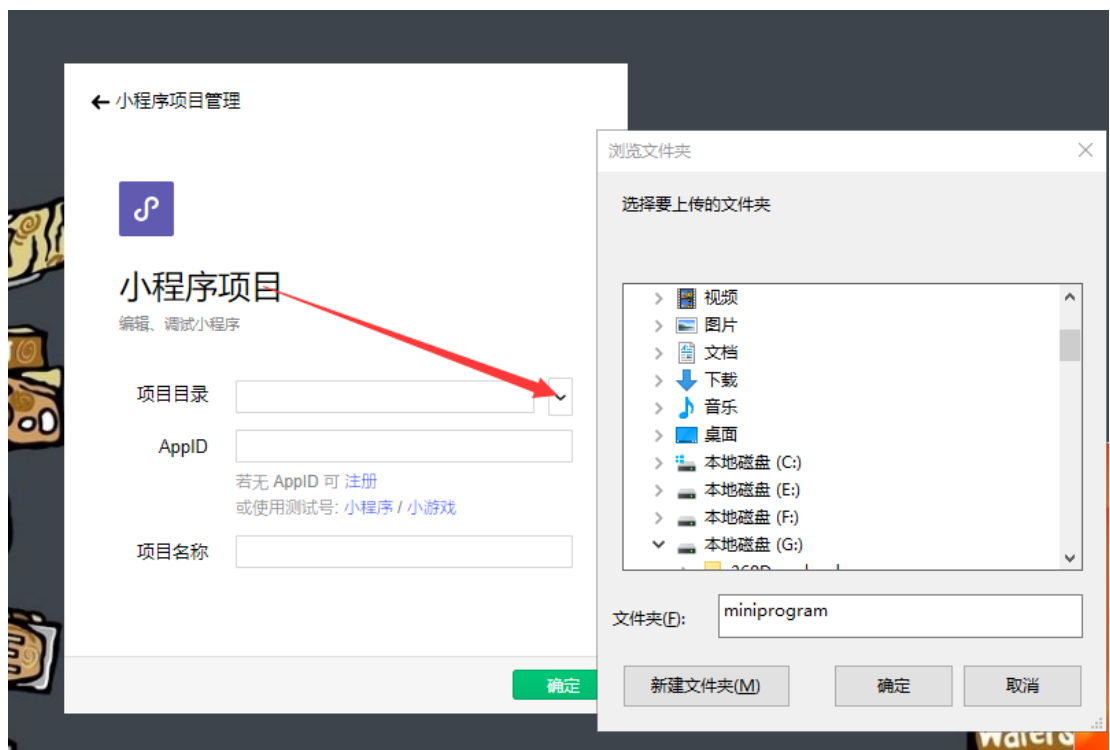
(1) 开发工具下载

下载地址：
<https://developers.weixin.qq.com/miniprogram/dev/devtools/download.html>
选择对应操作系统和位数版本，下载后直接安装即可



(2) 新建小程序项目：

打开开发工具点击+号按钮



项目目录：选择本机中的小程序项目

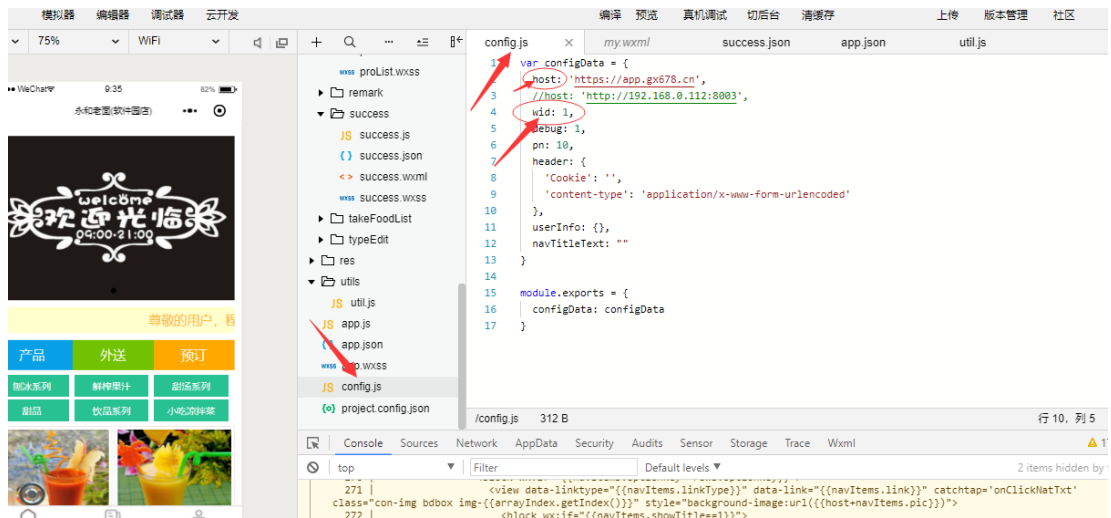
AppID：填写小程序的 AppID

项目名称：任意填写

打开创建好的小程序项目修改 `config.js` 配置文件(检查一处、修改一处，其

余不动)

检查 host 地址：地址正确才能正确的指到服务器上



修改 wid:每个 wid 代表一家门店。在总后台门店创建时可获得对应门店的 wid，将其填入即可



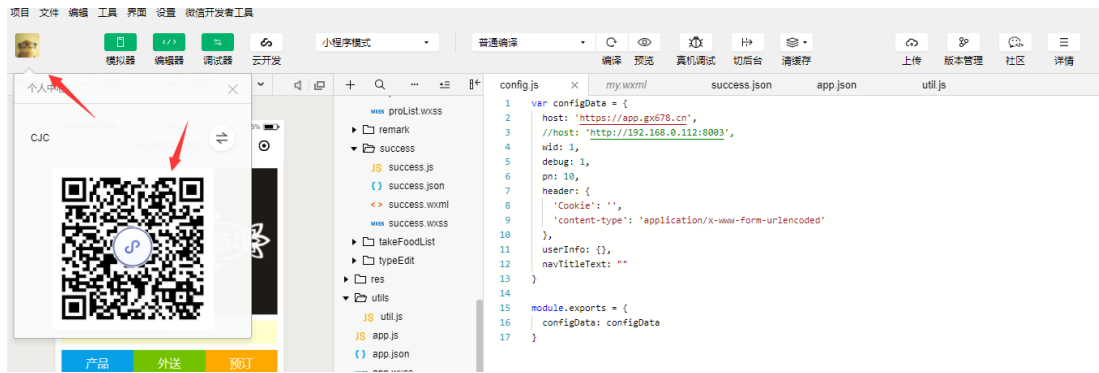
(3) 上传小程序

上传小程序提前条件：

1 登入微信小程序公众平台(<https://mp.weixin.qq.com>)----->用户管理----->创建成员赋予其开发权限



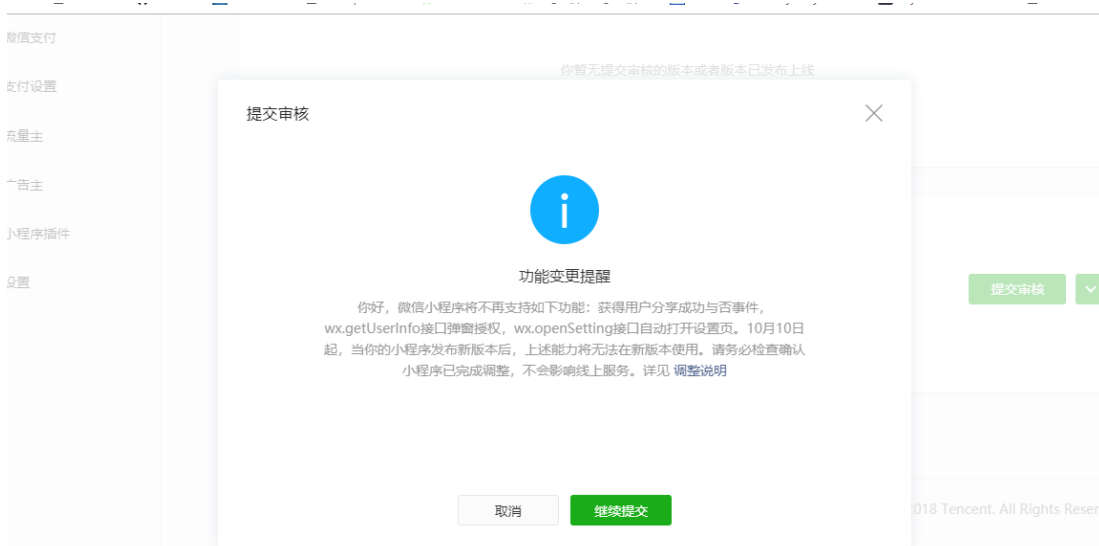
2 有开发权限的成员登入微信开发工具

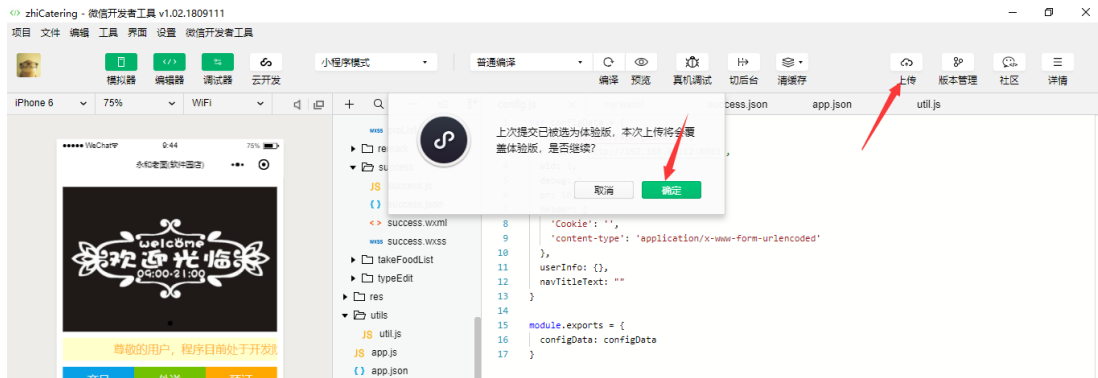


3 修改好 `config.js` 文件后点击上传，小程序即上传至可根据第一步创建小程序时填写的 APPID 指定的微信小程序公众平台上。

(4) 提交审核、发布

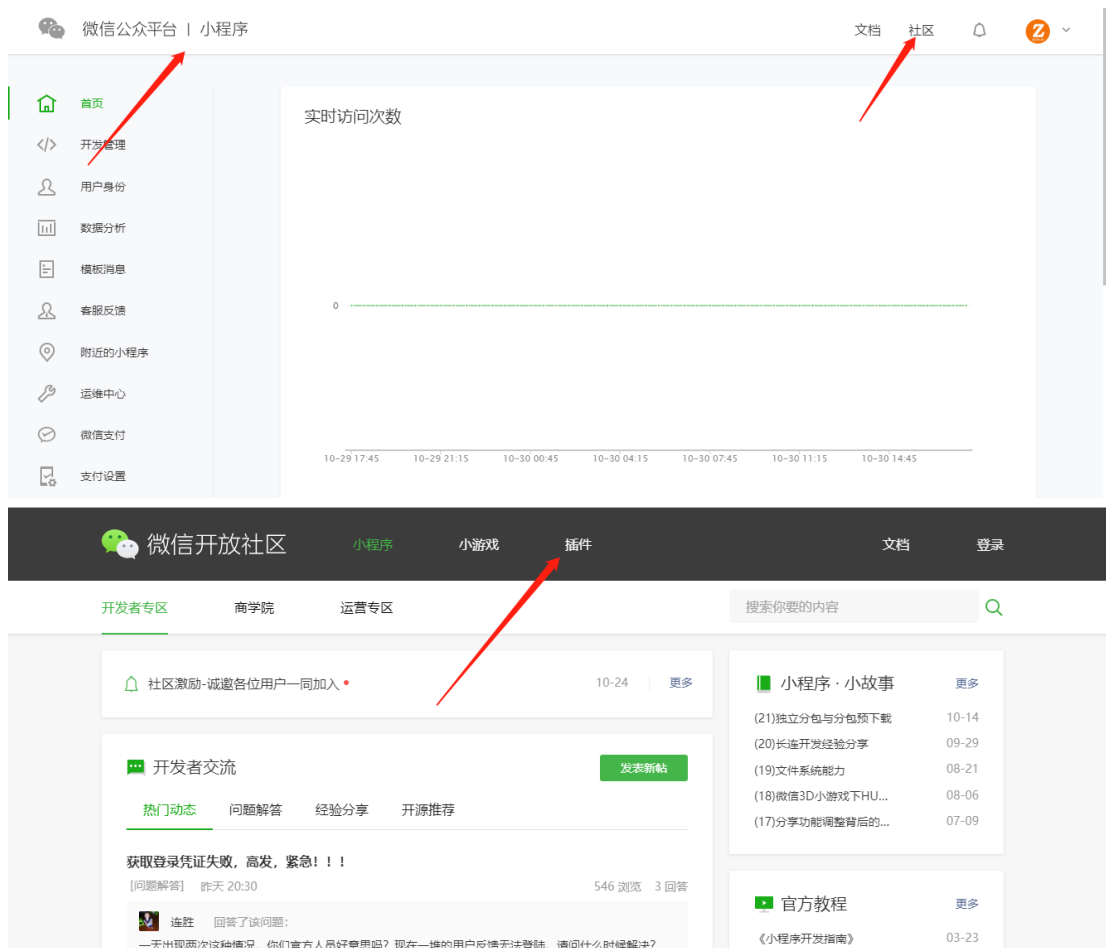
登入 [小程序微信公众平台](#) ----> [开发管理](#), 即可看到上传的小程序，点击提交审核后，等待微信审核通过都在点击发布即可将小程序发布上线





(5) 小程序添加富文本解析插件（解析商品详情图文信息）

- 1、登入微信小程序后台
- 2、点击社区，进入社区后点击插件，输入插件名称：**wx9d4d4ffa781ff3ac**
- 3、添加插件





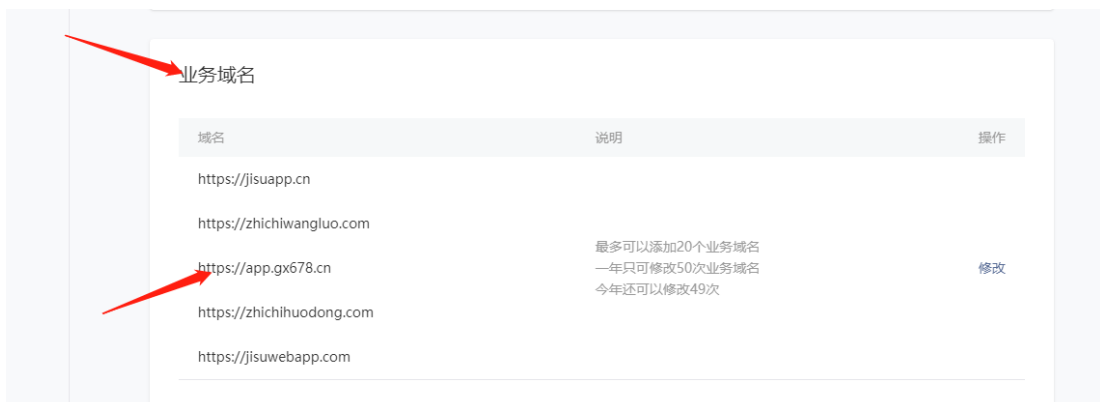
(6) 小程序后台配置服务器信息

1、此项必须配置，如不配置小程序像服务器请求数据将会被拦截。

配置路径：登入小程序后台：设置---->服务器域名



配置业务域名：



配置次域名会提示下载一份文件

业务域名需经过ICP备案，新备案域名需24小时后方可配置。域名格式只支持英文大小写字母、数字及“-”，不支持IP地址。配置业务域名后，可打开任意合法的子域名。

下载文件 请下载校验文件，并将文件放置在域名根目录下，例如wx.qq.com，并确保可以访问该文件。如配置中遇到问题，请查看[具体指引](#)。

域名1 

经下载好的文件拷贝放到服务器上，路径：D:\guangxin\app.gx678.cn
然后重启服务器

点餐外卖小程序系统源代码【前 30 页】

```
package com.biz.api;

import com.biz.bll.AssembleBll;
import com.biz.bll.DeliveryBll;
import com.biz.bll.MinAppPayBll;
import com.biz.bll.TableWareBll;
import com.jfinal.aop.Before;
import com.jfinal.plugin.activerecord.Db;
import com.jfinal.plugin.activerecord.Record;
import com.zhi.common.Utils;
import jdk.nashorn.internal.runtime.Undefined;
import model.*;
import org.apache.commons.lang.StringUtils;
import org.apache.log4j.Logger;
import org.quartz.impl.jdbcjobstore.Util;

import java.math.BigDecimal;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.*;

@Before(AuthApi.class)
public class ApiAssController extends ApiBaseController {
    public static Logger log = Logger.getLogger(ApiAssController.class);

    /**
     * 获取拼团商品数据
     */
    public void getAssembleData() {
        int wid=this.getWID();
        AssembleBll assembleBll = new AssembleBll();
        assembleBll.updateAssState(wid);
        try{
            String sql="SELECT s.*,p.skuPic from com_assemble_set s LEFT JOIN com_product_sku
p on s.skuKey=p.skuKey where s.wid="+wid+" and s.state=1 ORDER BY s.assId";
            List<Record> mbleSql = Db.find(sql);
            List <Record> pinNum = Db.find("SELECT count(*) as count,d.assId " +
                "FROM com_assmeble_detail d INNER JOIN com_assemble_set s on
d.assId=s.assId INNER JOIN com_assemble_part p on d.partId=p.partId and p.state=2 " +
                "LEFT JOIN com_order o on o.orderKey=d.orderKey and o.hasPay=1 " +
                "where d.wid="+wid+" and s.state=1 GROUP BY d.assId");
            if(mbleSql.size()>0) {
                for(Record mbleobj:mbleSql) {
                    //拼团没数据时,新增三笔数据
                    String objsql="SELECT d.*,s.assTime,date_add(p.partTime, interval
s.assTime hour) as realTime,fp.fictcount,p.isreal,IFNULL(sd.realnum,0) as realNums " +
                        "from com_assmeble_detail d " +
                        "LEFT JOIN ( " +
                        "SELECT sf.shamAssNum-sum(if(ps.mid!=null,IFNULL(o.hasPay,0),1))
```

```

as fictcount, ps. assdetail, ps. partId " +
    "FROM com_assmeble_detail ps " +
    "LEFT JOIN com_assemble_set sf on ps. assId=sf. assId " +
    "LEFT JOIN com_assemble_part pd on ps. partId=pd. partId " +
    "LEFT JOIN com_order o on o. orderKey=ps. orderKey and o. hasPay=1 "
+
    "where ps. wid="+wid+" GROUP BY ps. partId " +
    ") fp on d. assdetail=fp. assdetail " +
    "LEFT JOIN ( " +
    "SELECT partId, COUNT(assdetail) as realnum " +
    "from com_assmeble_detail " +
    "where IFNULL(mid,0)<>0 and assId="+mbleobj.get("assId")+ " GROUP
BY partId " +
    ") sd on sd. partId=d. partId " +
    " LEFT JOIN com_assemble_set s on d. assId=s. assId " +
    "LEFT JOIN com_assemble_part p on d. partId=p. partId " +
    "where d. assId="+mbleobj.get("assId")+ " and d. wid="+wid+" and
d. type=1 and p. state=1 and fictcount>0 " +
    "ORDER BY realNums desc, fictcount asc, realTime asc LIMIT 3";
List<Record> fictitious= Db. find(objsql);
if (fictitious.size()<=0) { //没数据时, 新增三笔
    //AssembleBll assembleBll=new AssembleBll();
    //type:0=参与者, 1=团长
    assembleBll. saveFaciti(wid, mbleobj.get("assId"), 1);
    //fictitious= Db. find(sql);
}
}
}
if(pinNum.size()>0) {
    for(Record obj:mbleSql) {
        for(Record num:pinNum) {
            if(obj.get("assId")==num.get("assId")) {
                obj.set("totalNum", num.get("count"));
            }
        }
    }
} else {
    for(Record obj:mbleSql) {
        obj.set("totalNum", 0);
    }
}
}

this.result.setData(mbleSql);
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
}
renderJson(this.result);
}

```

/**
 * 获取拼团商品信息

```

    */
    public void getassInfo() {
        int wid = this.getWID();
        int assId= Utils.toInt(this.get("assId"));
        try{
            String sql="SELECT a.*,p.content,s.skuPic,DATE_FORMAT(a.activeEnd,'%d 天%H 时%i
            分%s 秒') as chinaTime,IFNULL((IFNULL(a.saleNum,0)- IFNULL(a.bugNum,0)),0) as surplus " +
                "FROM com_assemble_set a " +
                "LEFT JOIN com_product p on a.proKey=p.proKey " +
                "LEFT JOIN com_product_sku s on a.skuKey=s.skuKey " +
                "where a.assId="+assId+" and p.state=1 and a.wid="+wid;
            Record assSql=Db.findFirst(sql);
            this.result.setData(assSql);
            Record part = Db.findFirst("SELECT d.*,IFNULL(o.hasPay,0) as hasPay from
            com_assemble_detail d LEFT JOIN com_assemble_part p on d.partId=p.partId LEFT JOIN com_order
            o on o.orderKey=d.orderKey " +
                " where d.assId="+assId+" and p.state=1 and d.mid="+this.getMid()+" and
            IFNULL(o.hasPay,0)=1 and d.wid="+wid);
            // ComAssemblePart part = ComAssemblePart.dao.getFirst("assId="+assId+" and
            wid="+wid+" and mid="+this.getMid()+" and state=1");
            this.result.setRows(part);
        }catch (Exception e){
            this.result.setErrId("1");
            this.result.setMsg(e.getMessage());
        }
        renderJson(this.result);
    }
    //region 我要拼团

    /**
     * 单独购买时判断商品是否有标签，有在返回标签，没有则直接进入订单购买页
     */
    public void getbuyData() {
        int assId = Utils.toInt(this.get("assId"));
        int wid = this.getWID();
        try{
            ComAssembleSet assembleSet = ComAssembleSet.dao.getFirst("assId="+assId+" and
            wid="+this.getWID());
            if(assembleSet!=null) {
                List<Record> lableValue = Db.find("select a.*,IF(IFNULL(b.selectVid,0)=0,0,1)
            as isSelected from com_lable_value a " +
                "LEFT JOIN ( " +
                "SELECT MIN(id) as selectVid from com_lable_value where wid=" + wid +
                " and proKey=' " + assembleSet.getProKey() + "' GROUP BY pId " +
                ") b on a.id=b.selectVid " +
                "where wid=" + wid + " and proKey=' " + assembleSet.getProKey() + "'
            ORDER BY id");
                if(lableValue.size()>0) { //购买的时候判断商品是否有标签
                    List<ComProductLable> lableList = ComProductLable.dao.getList("proKey=' "
            + assembleSet.getProKey() + "' and wid=" + wid + " ORDER BY lableId");
                    this.result.setData(lableList);
                    this.result.setRows(lableValue);
                    //初始化标签
                }
            }
        }
    }

```

```

        List<Record> selectlabs = Db.find("select b.id,b.`name`,b.pId " +
            "from ( " +
            "SELECT MIN(id) as id,`name`,pId from com_lable_value where wid="
+ wid + " and proKey=' " + assembleSet.getProKey() + "' GROUP BY pId " +
            ") b ORDER BY b.id");
        Map<Object, Object> dataLabMap = new HashMap<>();
        Map<String, String> selectlabMap = new HashMap<>();
        Map<String, String> selectPidMap = new HashMap<>();
        if (selectlabs != null && selectlabs.size() > 0) {
            for (Record sellab : selectlabs) {
                selectlabMap.put(sellab.get("id"), sellab.getStr("name"));
                selectPidMap.put(sellab.get("pId"), sellab.get("id"));
            }
            dataLabMap.put("selectlabMap", selectlabMap);
            dataLabMap.put("selectPidMap", selectPidMap);
            this.result.setFooter(dataLabMap);
        }
    }
}
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
}
renderJson(this.result);
}

/**
 * 拼团购买信息
 */
public void getAssPayInfo() {
    int assId = Utils.toInt(this.get("assId"));
    int wid = this.getWID();
    String lablesList = "";
    String lableIdList = "";
    try{
        String sql="SELECT s.*,k.skuPic from com_assemble_set s LEFT JOIN com_product_sku
k on s.skuKey=k.skuKey where s.wid="+wid+" and s.assId="+assId;
        Record mbleSql=Db.findFirst(sql);
        this.result.setData(mbleSql);
        //标签
        String labStr = this.get("listlabs");

        if (!Utils.isNullOrEmpty(labStr) && !labStr.equals("undefined")) {
            String[] str = labStr.split(";");
            for (int i = 0, len = str.length; i < len; i++) {
                String labMap = str[i].toString();
                String id = labMap.split(":")[0];
                ComLableValue lableValue = ComLableValue.dao.getFirst(" id=" + id + " and
wid=" + this.getWID());
                lablesList += lableValue.getName() + "+";
                lableIdList += lableValue.getPid() + ":" + lableValue.getId() + ";";
            }
            lablesList = lablesList.substring(0, lablesList.length() - 1);
        }
    }
}

```

```
        lableIdList = lableIdList.substring(0, lableIdList.length() - 1);
        this.result.setRows(lablesList);
        this.result.setRowdate(lableIdList);
    }
    ComWx config = ComWx.dao.getFirst(" wid=" + this.getWID() + " ");
    this.result.setFooter(config);
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
}
}
renderJson(this.result);
}

/**
 * 获取用户信息
 */
public void getMember() {
    try {
        Record member = Db.findFirst("select * from com_member where wid="+this.getWID()+
and mid="+this.getMid());
        this.result.setData(member);
    } catch (Exception e) {
        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
    renderJson(this.result);
}

/**
 * 判断是否在配送范围内
 */
public void checkDelivery() {
    if (isPost()) {
        try {
            DeliveryBll deliveryBll = new DeliveryBll();
            Map map = new HashMap();
            map.put("isDelivery", deliveryBll.checkDeliver(this.getWID(),
this.getInt("addressId")) ? 1 : 0);
            this.result.setData(map);
        } catch (Exception e) {
            this.result.setErrId("1");
            this.result.setMsg(e.getMessage());
        }
        renderJson(this.result);
    }
}

/**
 * 获取配送信息
 */
public void getDelivery() {
    if (isPost()) {
        try {
            DeliveryBll deliveryBll = new DeliveryBll();
```

```

        Map map = new HashMap();
        map.put("deliveryPrice",          deliveryBll.getDeliveryPrice(this.getWID(),
this.getInt("addressId")));
        map.put("deliveryDistance",      deliveryBll.getDeliveryDistance(this.getWID(),
this.getInt("addressId")));
        this.result.setData(map);
    } catch (Exception e) {
        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
    renderJson(this.result);
}
}

/**
 * 生成拼团订单 作废
 */
public void getAssOrder() {
    try{
        DeliveryBll deliveryBll = new DeliveryBll();
        int wid = this.getWID();
        int mid = this.getMid();

        int assId = Utils.toInt(this.get("assId"));
        String lableList = this.get("lableList");
        String lableIdList = this.get("lableIdList");
        String mobile = this.get("mobile");
        String addressId = this.get("addressId");
        String remark = this.get("remark");
        String asstype = this.get("asstype");
        Date date = new Date();
        int type = Utils.toInt(this.get("type")); //0 钱包 1 微信
        AssembleBll assembleBll = new AssembleBll();
        assembleBll.updateAssState(wid); //判断拼团商品是否失效
        ComAssembleSet assembleSet = ComAssembleSet.dao.getFirst("assId="+assId+ " and
wid="+wid);
        ComProductSku sku = ComProductSku.dao.getFirst("skuKey=' " + assembleSet.getSkuKey()
+ "' and wid=" + wid);
        ComWx wx = ComWx.dao.getFirst(" wid=" + wid + " ");
        String orderKey=Utils.getGUID();
        if(assembleSet.getState()==1) { //判断拼团商品是否生效
            if(assembleSet.getType()==0 && Utils.isNullOrEmpty(addressId)) { //外送的砍价
活动
                this.result.setErrId("1");
                this.result.setMsg("请选择地址");
                renderJson(this.result);
                return;
            }
        }
        if(sku!=null) {
            if(Utils.toInt(assembleSet.getSaleNum())
Utils.toInt(assembleSet.getBugNum())) { //判断活动商品是否售空
                ComOrder order = getMod(ComOrder.class);
            }
        }
    }
}

```

```
        order.setWid(wid);
        order.setMid(mid);
        order.setHasPay(0);
        order.setState(0);
        order.setOrderType(6);
        order.setOrderKey(orderKey);
        order.setOrderNo("P"+Utils.getKey17());
        order.setCreateTime(date);
        order.setOrderTime(date);
        TableWareBll tableWareBll = new TableWareBll();
        order = tableWareBll.updateTableWareByOrder(wx, order);
        //region 订单配送
        if (!Utils.isNullOrEmpty(addressId) && assembleSet.getType() == 0) {
            ComMemberAddr addr = ComMemberAddr.dao.getFirst(" id=" + addressId
+ " ");

            if (!deliveryBll.checkDeliver(order.getWid(), addr.getId())) {
                throw new Exception("不在配送范围内!");
            }

            order.setAcceptName(addr.getAcceptName());
            order.setAddress(addr.getAddress());
            order.setAllAddress(addr.getAllAddress());
            order.setProvinceId(addr.getProvinceId());
            order.setProvince(addr.getProvince());
            order.setCityId(addr.getCityId());
            order.setCity(addr.getCity());
            order.setDistrictId(addr.getDistrictId());
            order.setDistrict(addr.getDistrict());
            order.setMobile(addr.getMobile());
            order.setLongitude(addr.getLongitude());
            order.setLatitude(addr.getLatitude());
            order.setPackMoney(wx.getPackPrice());

            order.setFreightMoney(deliveryBll.getDeliveryPrice(order.getWid(), addr.getId()));
        }
        //endregion
        ComMember member = ComMember.dao.getFirst(" wid = " + wid + " and mid
= " + mid + " ");

        order.setMName(member.getNickName());
        BigDecimal orderMoney = new BigDecimal(0);
        if (asstype.equals("alone")) {
            orderMoney = assembleSet.getAlonePrice();
        } else if (asstype.equals("assme")) {
            orderMoney = assembleSet.getAssPrice();
        }

        if (order.getTableWareNum() != null && order.getTableWarePrice() !=
null && assembleSet.getType() == 0) {
            orderMoney = orderMoney.add(wx.getTableWarePrice().multiply(new
BigDecimal(order.getTableWareNum()))); //订单金额+餐具费 System.out.println("g end");
        }
        if (assembleSet.getType() == 0) {
            orderMoney = orderMoney.add(wx.getPackPrice()); //订单金额+包装费
            orderMoney = orderMoney.add(order.getFreightMoney()); //订单金额+
```


运费

```

    }

    BigDecimal realMoney = orderMoney;
    //region 钱包
    BigDecimal wallectPay = BigDecimal.ZERO;
    BigDecimal memberWallet = Utils.toDec(member.getTotalWallet());
    if (realMoney.compareTo(BigDecimal.ZERO) > 0 &&
memberWallet.compareTo(BigDecimal.ZERO) > 0 && type == 0) {
        if (realMoney.compareTo(memberWallet) >= 0) {
            realMoney = realMoney.subtract(memberWallet);
            wallectPay = memberWallet;
        } else {
            wallectPay = realMoney;
            realMoney = BigDecimal.ZERO;
        }
    }
}
order.setWalletPayMoney(wallectPay);
//endregion

order.setOrderMoney(orderMoney);
//订单支付金额 0，直接调用订单此支付方式
order.setRealMoney(realMoney);
order.setRemark(remark);
order.save(); //订单保存
//订单生成之后生成拼团活动和拼团详情数据
ComAssemblePart part=null;
if (asstype.equals("assme")){//判断是拼团购买
    part = addAssData(orderKey, assId, assembleSet.getAssPrice());
}
if (order.getRealMoney().compareTo(BigDecimal.ZERO) == 0) {
    order.setHasPay(1);
    order.setState(1);
    order.setPayType(0);
    order.setPayTime(new Date());
    order.update();

    //在拼团发起表中记录订单 key，联系号码和状态修改为待支付
    if(part!=null) {
        part.setOrderKey(orderKey);
        part.setTel(mobile);
        part.update();
        ComAssmebleDetail detail =
ComAssmebleDetail.dao.getFirst("partId="+part.getPartId()+" and assId="+assId+" and
wid="+wid);

        if(detail!=null) {
            detail.setOrderKey(orderKey);
            detail.update();
        }
    }
}
//订单支付成功调用方法
MiniAppPayBll payBll = new MiniAppPayBll();
payBll.orderPaySuccess(order.getOrderKey(), wid);

```

```
    }

    Map map = new HashMap();
    map.put("realMoney", order.getRealMoney());
    map.put("orderKey", order.getOrderKey());
    map.put("orderNo", order.getOrderNo());
    map.put("hasPay", order.getHasPay());
    if(part!=null){
        map.put("partId",part.getPartId());
    }
    this.result.setData(map);
    //生成订单明细
    addOrderDetail(order.getOrderKey(),        lableList,        lableIdList,
orderMoney, sku);

    }else {
        this.result.setErrId("1");
        this.result.setMsg("活动商品已售罄");
        renderJson(this.result);
        return;
    }
} else {
    this.result.setErrId("1");
    this.result.setMsg("无此商品");
    renderJson(this.result);
    return;
}
} else if (assembleSet.getState() == 0) {
    this.result.setErrId("1");
    this.result.setMsg("该商品未生效");
} else if (assembleSet.getState() == 2) {
    this.result.setErrId("1");
    this.result.setMsg("该商品已失效");
} else {
    this.result.setData(orderKey);//调起支付
}
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage() != null ? e.getMessage() : "订单提交失败");
}
}
renderJson(this.result);
}

/**
 * 添加订单详情表记录    作废
 *
 * @param orderKey    订单主表 key
 * @param lableList    商品页签内容
 * @param lableIdList 商品页签 ID, 如 1, 21, 3
 * @param orderMoney  订单金额
 * @param sku         商品规格
```

```
*/
public void addOrderDetail(String orderKey, String lableList, String lableIdList,
BigDecimal orderMoney, ComProductSku sku) {
    ComOrderDetail orderDetail = new ComOrderDetail();
    orderDetail.setCreateTime(new Date());
    orderDetail.setOrderKey(orderKey);
    orderDetail.setWid(this.getWID());
    orderDetail.setMid(this.getMid());
    orderDetail.setIsAppraise(0);
    if (!Utils.isNullOrEmpty(lableList)) {
        orderDetail.setLableIdList(lableIdList);
        orderDetail.setLablesList(lableList);
    }
    orderDetail.setOrderDetailKey(Utils.getGUID());
    orderDetail.setOrderDetailNo(Utils.getKey17());
    orderDetail.setProKey(sku.getProKey());
    orderDetail.setProName(sku.getProName());
    orderDetail.setSkuKey(sku.getSkuKey());
    orderDetail.setSkuName(sku.getSkuName());
    orderDetail.setSkuThumbPic(sku.getSkuThumbPic());
    orderDetail.setSkuPic(sku.getSkuPic());
    orderDetail.setUnitId(sku.getUnitId());
    orderDetail.setUnit(sku.getUnit());
    orderDetail.setSalePrice(orderMoney);
    orderDetail.setMarketPrice(sku.getSalePrice()); //原价
    orderDetail.setBuyNum(1);
    orderDetail.setTotalPrice(orderMoney.multiply(new BigDecimal(1)));
    orderDetail.save();
}

/**
 *生成拼团活动和拼团详情
 */
public ComAssemblePart addAssData(String orderKey, int assId, BigDecimal assPrice) {
    ComAssemblePart part = new ComAssemblePart();
    part.setWid(this.getWID());
    part.setMid(this.getMid());
    part.setAssId(assId);
    part.setAssnum(1);
    part.setState(1);
    part.setPartTime(new Date());
    part.setCreateTime(new Date());
    part.setOrderKey(orderKey);
    part.setIsreal(1);
    part.save();
    ComAssembleSet comAssembleSet = ComAssembleSet.dao.getFirst("assId="+assId+" and
wid="+this.getWID());
    if (comAssembleSet.getState()==1) { //判断商品是否在活动时间内
        ComAssmebleDetail detail = ComAssmebleDetail.dao.getFirst("partId="+
part.getPartId() + " and mid="+ this.getMid()); //判断是否拼团过
        ComMember comMember = ComMember.dao.getFirst("mid="+this.getMid());
        AssembleBll assembleBll = new AssembleBll();
        if (detail == null) {
```

```
        ComAssmebleDetail assmebleDetail = new ComAssmebleDetail();
        assmebleDetail.setWid(this.getWID());
        assmebleDetail.setMid(this.getMid());
        assmebleDetail.setPartId(part.getPartId());
        assmebleDetail.setAssId(assId);
        assmebleDetail.setAssPrice(assPrice);
        assmebleDetail.setType(1);
        assmebleDetail.setCreateTime(new Date());
        assmebleDetail.setOrderKey(orderKey);
        assmebleDetail.setName(assembleBll.replaceStr(comMember.getNickName()));
        assmebleDetail.setHeadUrl(comMember.getHeadimgurl());
        assmebleDetail.save();
    }
}
return part;
}

/**
 * 获取 PrePayId
 * cjc
 */
public void getPrePayId() {
    int wid = Utils.toInt(this.get("wid"));
    String openId = this.getSessionAttr("openId");
    String orderKey = this.get("orderKey");
    MiniAppPayBll miniAppPayBll = null;

    try {
        //订单支付
        ComOrder order = ComOrder.dao.getFirst("orderKey='" + orderKey + "' and wid='" + wid
+ " and state = 0");
        if (order != null) {
            // 减去优惠应该支付的金额
            BigDecimal payMoney = order.getRealMoney();
            miniAppPayBll = new MiniAppPayBll(wid, openId, orderKey, payMoney, "");
            //获取生成与支付单号和小程序支付签名
            log.error("订单支付获取微信预付单 ID=" + miniAppPayBll.createPrePayId());
            String[] result = miniAppPayBll.createPrePayId().split("&");

            //存入 prePayId(发送小程序模板消息使用)
            order.setPrepayId(result[0]);
            order.update();

            this.result = miniAppPayBll.dealResult(result);
            this.result.put("orderId", order.getId());
            this.result.put("payMoney", order.getOrderMoney());
            this.result.put("orderKey", order.getOrderKey());
        } else {
            throw new Exception("未查询到对应订单!");
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```

        renderJson(this.result);
    }

    /**
     * 活动拼团详情
     */
    public void getAssFriends() {
        int partId = Utils.toInt(this.get("partId"));

        String orderKey = this.get("orderKey");
        int wid=this.getWID();
        int mid=this.getMid();
        if(!Utils.isNullOrEmpty(orderKey)) {
            ComAssmebleDetail assmebleDetail =
ComAssmebleDetail.dao.getFirst("orderKey='"+orderKey+"' and wid='"+wid";
                partId = assmebleDetail.getPartId();
            }
            if(partId!=0) {
                String sql="SELECT S.*,p.partId,k.skuPic from com_assemble_part p LEFT JOIN
com_assemble_set s on p.assId=s.assId LEFT JOIN com_product_sku k on k.skuKey=s.skuKey WHERE
P.WID='"+wid+"' and p.partId='"+partId;
                Record partSql=Db.findFirst(sql);
                this.result.setData(partSql);

                Record detailnum=Db.findFirst("SELECT COUNT(assdetail) as count from
com_assemble_detail where partId='"+partId);
                int surplus=Utils.toInt(partSql.get("shamAssNum"))-
Utils.toInt(detailnum.get("count")); //拼团剩余人数
                Record time=Db.findFirst("SELECT DATE_ADD(p.partTime, INTERVAL s.assTime hour) as
time from com_assemble_set s LEFT JOIN com_assemble_part p on s.assId=p.assId where
p.partId='"+partId); //拼团剩余时间
                Map map = new HashMap();
                map.put("surplus", surplus);
                map.put("assTime", time.get("time"));
                this.result.setRows(map);
                List<Record> detailList=Db.find("SELECT * from com_assemble_detail where
partId='"+partId+" ORDER BY type desc ");
                this.result.setFooter(detailList);
                Record part = Db.findFirst("SELECT d.* from com_assemble_detail d LEFT JOIN
com_assemble_part p on d.partId=p.partId where d.partId='"+partId+" and p.state=1 and
d.mid='"+this.getMid()+" and d.wid='"+wid);

                //ComAssemblePart part= ComAssemblePart.dao.getFirst("partId='"+partId+" and
mid='"+mid+" and state=1");
                this.result.setRowdate(part);
            }

            renderJson(this.result);
        }

    /**
     * 活动结束后修改商品状态
     */

```

```

public void updatebarEnd() {
    int assId = Utils.toInt(this.get("assId"));
    Date date = new Date();
    try {
        ComAssembleSet assembleSet = ComAssembleSet.dao.getFirst("assId=" + assId + " and
wid=" + this.getWID());
        if (assembleSet != null) {
            if (assembleSet.getActiveEnd().getTime() <= date.getTime()) {
                assembleSet.setState(2);
                assembleSet.update();
            }
        }
    } catch (Exception e) {
        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
    renderJson(this.result);
}

//endregion

//region 商品拼团详情

//获取拼团数据
public void fightData() {
    int assId=Utils.toInt(this.get("assId"));
    int wid=this.getWID();
    //获取前三笔数据，真实数据优先查询
    //List<ComAssembleFictitious> fictitious=ComAssembleFictitious.dao.getList("
wid="+wid+" and assId="+assId+" ORDER BY createTime desc LIMIT 3 ");
    String sql="SELECT d.*,s.assTime,date_add(p.partTime, interval s.assTime hour) as
realTime,fp.fictcount,p.isreal,IFNULL(sd.realnum,0) as realNums " +
        "from com_assmeble_detail d " +
        "LEFT JOIN ( " +
        "SELECT      sf.shamAssNum-sum(if(ps.mid!=null,IFNULL(o.hasPay,0),1))      as
fictcount,ps.assdetail,ps.partId " +
        "FROM com_assmeble_detail ps " +
        "LEFT JOIN com_assemble_set sf on ps.assId=sf.assId " +
        "LEFT JOIN com_assemble_part pd on ps.partId=pd.partId " +
        "LEFT JOIN com_order o on o.orderKey=ps.orderKey and o.hasPay=1 " +
        "where ps.wid="+wid+" GROUP BY ps.partId " +
        ") fp on d.assdetail=fp.assdetail " +
        "LEFT JOIN ( " +
        "SELECT partId,COUNT(assdetail) as realnum " +
        "from com_assmeble_detail " +
        "where IFNULL(mid,0)<>0 and assId="+assId+" GROUP BY partId " +
        ") sd on sd.partId=d.partId " +
        " LEFT JOIN com_assemble_set s on d.assId=s.assId " +
        "LEFT JOIN com_assemble_part p on d.partId=p.partId " +
        "where d.assId="+assId+" and d.wid="+wid+" and d.type=1 and p.state=1 and
fictcount>0 " +
        "ORDER BY realNums desc,fictcount asc,realTime asc LIMIT 3";
}

```

```

        List<Record> fictitious= Db.find(sql);
        /*if (fictitious.size()<=0) {没数据时, 新增三笔
            AssembleBll assembleBll=new AssembleBll();
            //type:0=参与者, 1=团长
            assembleBll.saveFaciti(wid, assId, 1);
            fictitious= Db.find(sql);
        }*/

        //ComAssembleSet assembleSet = ComAssembleSet.dao.getFirst(" assId="+assId+" and
wid="+wid);
        this.result.setData(fictitious);
        renderJson(this.result);
    }

    //拼团结束倒计时
    /*public void getAssTime() {
        int assId=Utils.toInt(this.get("assId"));
        int partId=Utils.toInt(this.get("partId"));
        String times=this.get("times");
        System.out.println("=times="+times);
        ComAssembleSet assembleSet=ComAssembleSet.dao.getFirst(" assId="+assId+" and
wid="+this.getWID());
        //计算倒计时
        SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
        Date date = null;
        try {
            date = format.parse(times);
        } catch (Exception ex) {
            ex.printStackTrace();
        }
        if (date != null) {
            System.out.println("front:" + format.format(date)); //显示输入的日期
            Calendar cal = Calendar.getInstance();
            cal.setTime(date);
            cal.add(Calendar.HOUR, assembleSet.getAssTime()); // 24 小时制
            date = cal.getTime();
            System.out.println("after:" + format.format(date)); //显示更新后的日期
        }
        //计算还差几人成团
        Long count = ComAssembleFictitious.dao.count(" partId="+partId+" and assId="+assId+"
and wid="+this.getWID());
        int fictcount=assembleSet.getShamAssNum()-Utils.toInt(count);

        this.result.setData(date);
        this.result.setRowdate(fictcount);
        renderJson(this.result);
    }*/

    /**
     * 修改参团活动状态
     */
    public void updatePartEnd() {

```

```

        int partId=Utils.toInt(this.get("partId"));
        ComAssemblePart assemblePart=ComAssemblePart.dao.getFirst(" partId="+partId+" and
wid="+this.getWID()+" and state=1");
        if(assemblePart != null){
            assemblePart.setState(0);
            assemblePart.update();
            AssembleBll assembleBll = new AssembleBll();

assembleBll.tui(assemblePart.getPartId(),assemblePart.getAssId(),this.getWID());
            //删除未拼团成功的参与者虚拟数据
            if(assemblePart.getIsreal()==0){
                assemblePart.delete();
                ComAssmebleDetail.dao.delete(" partId="+partId+" and wid="+this.getWID());
            }
        }
        renderJson(this.result);
    }

/**
 * 新增一笔数据到最近参团时间未满足并且当前时间比最近参团时间超过 10 分钟
 */
public void addAssDetail() {
    int assId=Utils.toInt(this.get("assId"));
    int wid=this.getWID();
    //获取拼团商品下的参团数据
    String sql="SELECT          p.*, IFNULL(sd.realnum,0)          as
realNums,fp.fictcount,date_add(p.partTime,          interval          s.assTime          hour)          as
realTime,s.shamAssNum,s.assnum as ascoun " +
        "from com_assemble_part p " +
        "LEFT JOIN com_assemble_set s on p.assId=s.assId " +
        "LEFT JOIN ( " +
        "SELECT partId,COUNT(assdetail) as realnum " +
        "from com_assmeble_detail where IFNULL(mid,0)<>0 and assId="+assId+" and
wid="+wid+" " +
        "GROUP BY partId " +
        ") sd on sd.partId=p.partId " +
        "LEFT JOIN ( " +
        "SELECT sf.shamAssNum-COUNT(ps.assdetail) as fictcount,ps.partId " +
        "FROM com_assmeble_detail ps " +
        "LEFT JOIN com_assemble_set sf on ps.assId=sf.assId " +
        "where ps.wid="+wid+" GROUP BY ps.partId " +
        ") fp on p.partId=fp.partId " +
        "WHERE p.assId="+assId+" and p.state=1 and fictcount>0 ORDER BY realNums
desc,fictcount asc,realTime asc limit 2";
    List<Record> partList = Db.find(sql);
    System.out.println("=sql="+Db.find(sql));
    System.out.println("=partList="+partList.size());
    for(Record parts : partList){
        //查询改拼团活动下存在几笔虚拟数据
        String countSql="SELECT IFNULL( COUNT(assdetail),0) as asscount from
com_assmeble_detail where partId="+parts.get("partId")+ " and IFNULL(mid,0)=0 ";
        Record laverecord=Db.findFirst(countSql);

```



```

//获取需生成几条虚拟数据
int
lavenum=Utils.toInt(parts.get("shamAssNum"))-Utils.toInt(parts.get("ascount"));
int assnum=Utils.toInt(laverecord.get("asscount")); //虚拟数据数量
System.out.println("新增数据 lavenum="+lavenum);
System.out.println("=assnum="+assnum);

//      String detailSql="SELECT COUNT(assdetail) as detaicount from com_assmeble_detail
where partId="+parts.get("partId");//不能 detail 的数量作为判断依据：有可能有真实订单且还未付
款成功
      String detailSql="SELECT  sum(if(IFNULL(d.mid,0)!=0,IFNULL(o.hasPay,0),1))  as
detaicount " +
      "FROM com_assmeble_detail d " +
      "LEFT JOIN com_order o on o.orderKey=d.orderKey and o.hasPay=1 " +
      "where d.wid="+wid+" and d.partId="+parts.get("partId");
      Record detail=Db.findFirst(detailSql);
      System.out.println("= 数 据 已 经 生 成 完 毕
detaicount="+Utils.toInt(detail.get("detaicount")));
      System.out.println("=shamAssNum" +
      "="+parts.get("shamAssNum"));
      if(Utils.toInt(detail.get("detaicount")) == parts.get("shamAssNum")){//数据已经
生成完毕
          Db.update("update      com_assemble_part      set      state=2      where
partId="+parts.get("partId")+ " and wid="+wid);

      }else{
          if(lavenum>0 && lavenum>assnum){ //拼团的虚拟数据大于 0 条并且大于已生成的虚拟
数据条数
              //TODO:10 分钟生成一笔
              //查询小于等于当前 10 分钟数据
              ComAssmebleDetail      assembleDetail=ComAssmebleDetail.dao.getFirst("
partId="+parts.get("partId")+ " and createTime<=DATE_SUB(NOW(),INTERVAL 10 MINUTE) ORDER BY
createTime desc ");
              if(assembleDetail != null){
                  //参与数据
                  System.out.println("==assembleDetail=");
                  ComAssmebleDetail newAss=new ComAssmebleDetail();
                  newAss.setAssId(parts.getInt("assId"));
                  newAss.setPartId(parts.getInt("partId"));
                  newAss.setWid(wid);
                  newAss.setMid(0);
                  newAss.setAssPrice(assembleDetail.getAssPrice());
                  newAss.setType(0);
                  newAss.setCreateTime(new Date());
                  AssembleBll assembleBll = new AssembleBll();
                  newAss.setName(assembleBll.getName());
                  String imgs=assembleBll.getImg(parts.getInt("assId"),wid);
                  newAss.setHeadUrl(imgs);
                  newAss.save();
              }
              detail=Db.findFirst(detailSql);

      System.out.println("=detaicount="+Utils.toInt(detail.get("detaicount")));

```

```
        System.out.println("=shamAssNum="+parts.get("shamAssNum"));
        if(Utils.toInt(detail.get("detaicount")) == parts.get("shamAssNum")){//
数据已经生成完毕
            Db.update("update    com_assemble_part    set    state=2    where
partId="+parts.get("partId")+" and wid="+wid);

        }
    }
}

    renderJson(this.result);

}

//endregion

}
package com.biz.api;

import com.biz.BaseController;
import com.zhi.common.Utils;
import model.ComMember;
import org.apache.log4j.Logger;

public class ApiBaseController extends BaseController {
    public static Logger log = Logger.getLogger(ApiBaseController.class);
    /**
     * 取出当前微网站的 wid
     * @return
     */
    public Integer getWID() {
        int wid = this.getInt("wid", 0);
        if (wid == 0) {
            wid = Utils.toInt(this.getCookie("zhi_home_wid"), 0);
        }else {
            this.setCookie("zhi_home_wid", wid + "", 60 * 60 * 24);
        }
        return wid;
    }

    /**
     * 获取 Mid
     *
     * @return Mid
     */
    public int getMid() {
        String openId = this.getSessionAttr("openId");
        ComMember member = ComMember.dao.getFirst(" openId = '" + openId + "' and wid = " +
```

```
this.getWID());
    return member.getMid();
}

}

package com.biz.api;

import com.alibaba.fastjson.JSONArray;
import com.biz.api.PrinterApi.Printer;
import com.biz.api.service.ApiService;
import com.biz.api.service.WxMiniService;
import com.biz.bll.*;
import com.jfinal.aop.Before;
import com.jfinal.ext.interceptor.POST;
import com.jfinal.kit.JsonKit;
import com.jfinal.kit.PathKit;
import com.jfinal.kit.PropKit;
import com.jfinal.plugin.activerecord.Db;
import com.jfinal.plugin.activerecord.Page;
import com.jfinal.plugin.activerecord.Record;
import com.miniProgramData.WXBizDataCrypt;
import com.sun.prism.impl.Disposer;
import com.zhi.common.Utils;
import com.zhi.weixin.util.HttpUtil;
import dataModel.ResultMsg;
import model.*;
import net.sf.json.JSONObject;
import org.apache.log4j.Logger;

import java.math.BigDecimal;
import java.text.SimpleDateFormat;
import java.util.*;

@Before(AuthApi.class)
public class ApiController extends ApiBaseController {
    private static final String webSiteHome = PropKit.get("webSiteHome");
    public static final Random RAND = new Random();
    public static Logger log = Logger.getLogger(ApiController.class);

    /**
     * 登入时获取用户 sessionKey
     * cjc
     */
    public void createSessionKey() {
        String code = this.get("code");
        String grant_type = "authorization_code";

        try {
            ComWx wx = ComWx.dao.getFirst("wid=" + this.getWID());
            if (wx != null) {
                String appId = wx.getMiniAppId();
                String secret = wx.getMiniAppSecret();
```

```

String GET_LOGINMESSAGE =
"https://api.weixin.qq.com/sns/jscode2session?appid=" + appId + "&secret=" + secret +
"&js_code=" + code + "&grant_type=" + grant_type;
String result = HttpUtil.executeGet(GET_LOGINMESSAGE);
log.error("result===" + result);

JSONObject jsonObject = JSONObject.fromObject(result);

String session_key = jsonObject.getString("session_key");
String openId = jsonObject.getString("openid");

//如果用户不存在, 获取 openId 后先在 com_member 表创建一个用户

if (!ComMember.dao.exist("openId=" + openId + " and wid=" + this.getWID()))
{
    ComMember member = new ComMember();
    member.setWid(this.getWID());
    member.setShopName(wx.getShopName());
    member.setOpenId(openId);
    member.setCreateTime(new Date());
    long newMemCou=ComCoupon.dao.count("wid="+this.getWID()+" and
isOnlyNewMem=1 and couponStatus=2");
    //log.error("newMemCou=====" + newMemCou);
    if(newMemCou>0){
        member.setGrantNewCoupon("N");//是否发放了新人券: N=未发放, Y=已发放
    }else{//没有设置新人券, 默认已领取
        member.setGrantNewCoupon("Y");//是否发放了新人券: N=未发放, Y=已发放
    }
    //log.error("是否有人券
==member.getGrantNewCoupon()+" + member.getGrantNewCoupon());
    member.setContinSign(0);//连续签到 0 天
    member.save();
    //分享进来的新用户

    log.error("分享进来的用户 shareMid==" + this.get("shareMid"));
    int shareMid = Utils.toInt(this.get("shareMid")); //右上角分享
    int scene = 0;
    //String sceneStr = this.get("scene") + "";
    String sceneStr2 = this.get("scene2") + "";
    log.error("分享进来的用户 shareMid==" + this.get("shareMid") + ", 扫码进来的
sceneStr2=" + sceneStr2);
    if (sceneStr2.indexOf("shareMid") != -1) {
        scene = Utils.toInt(sceneStr2.replace("shareMid", ""));
    }
    if (shareMid > 0) {
        ApiService.bindMember(shareMid, member.getMid(), this.getWID());
    } else if (scene > 0) {
        ApiService.bindMember(scene, member.getMid(), this.getWID());
    }
}

this.result.put("sessionId", this.getSession().getId());

```

```
        this.setSessionAttr("session_key", session_key);
        this.setSessionAttr("openId", openId);

    } else {
        throw new Exception("找不到支付配置信息!");
    }
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
    e.printStackTrace();
}

renderJson(this.result);
}

/**
 * 小程序根据用户授权获取用户信息
 * cjc
 */
public void getUserInfo() {
    String encryptedData = this.get("encryptedData"); // 包括敏感数据在内的完整用户信息的加密数据
    String iv = this.get("iv"); //加密算法的初始向量
    ComWx wx = ComWx.dao.getFirst("wid=" + this.getWID());
    ComMember comMember = null;

    String session_key = this.getSessionAttr("session_key");

    try {
        if (wx != null) {
            WXBizDataCrypt crypt = new WXBizDataCrypt(wx.getMiniAppId(), session_key);
            String userEncryptedData = crypt.decryptData(encryptedData, iv);

            if (userEncryptedData != null) {
                JSONObject object = JSONObject.fromObject(userEncryptedData);
                String openId = object.getString("openId");
                String nickName = object.getString("nickName");
                String gender = object.getString("gender");
                String country = object.getString("country");
                String province = object.getString("province");
                String city = object.getString("city");
                String avatarUrl = object.getString("avatarUrl");
                comMember = ComMember.dao.getFirst("openId='" + openId + "' and wid=" +
this.getWID());

                if (comMember != null) {
                    comMember.setSex(Utils.toInt(gender));
                    comMember.setNickName(nickName);
                    comMember.setCountry(country);
                    comMember.setProvince(province);
                    comMember.setCity(city);
                }
            }
        }
    }
}
```

```
        comMember.setHeadingurl(avatarUrl);
//        comMember.setCreateTime(new Date());
        comMember.update();
    }

    //log.error("小程序根据用户授权获取用户信息时候的 openId==" + "
+ openId);
    } else {
        this.result.setErrId("1");
        this.result.setMsg("获取用户信息失败!");
    }

    } else {
        this.result.setErrId("1");
        this.result.setMsg("找不到支付配置信息!");
    }
} catch (Exception e) {
    e.printStackTrace();
}
renderJson(this.result);
}

/**
 * 获取 PrePayId
 * cjc
 */
public void getPrePayId() {
    int wid = Utils.toInt(this.get("wid"));
    String openId = this.getSessionAttr("openId");
    String orderKey = this.get("orderKey");
    String act = this.get("act");
    MiniAppPayBll miniAppPayBll = null;

    try {
        if ("bill".equals(act)) { //买单支付
            ComPayOrder payOrder = ComPayOrder.dao.getFirst(" payOrderKey=' " + orderKey +
" and wid=" + wid + " and state=0");
            if (payOrder != null) {
                BigDecimal payMoney = payOrder.getPayMoney();
                miniAppPayBll = new MiniAppPayBll(wid, openId, orderKey, payMoney, act);
                //获取生成与支付单号和小程序支付签名
                String[] result = miniAppPayBll.createPrePayId().split("&");

                //存入 prePayId
                payOrder.setPrePayId(result[0]);
                payOrder.update();

                //处理 result
                this.result = miniAppPayBll.dealResult(result);

                //后台传送调起支付所需参数
                this.result.put("payOrderId", payOrder.getPayOrderId());
                this.result.put("payMoney", payOrder.getPayMoney());
            }
        }
    }
}
```

```
        this.result.put("payOrderKey", payOrder.getPayOrderKey());

    } else {
        throw new Exception("未查询到对应订单!");
    }
} else if ("recharge".equals(act)) { //充值支付
    ComRechargeRecord rechargeRecord =
ComRechargeRecord.dao.getFirst("rechargeKey='" + orderKey + "' and wid='" + wid + "' and state=0
and recordType=0");
    if (rechargeRecord != null) {
        // 减去优惠应该支付的金额
        BigDecimal payMoney = rechargeRecord.getRechargeMoney();
        miniAppPayBll = new MiniAppPayBll(wid, openId, orderKey, payMoney, act);

        //获取生成与支付单号和小程序支付签名
        String[] result = miniAppPayBll.createPrePayId().split("&");

        //存入 prePayId(发送小程序模板消息使用)
        rechargeRecord.setPrePayId(result[0]);
        rechargeRecord.update();

        this.result = miniAppPayBll.dealResult(result);
        this.result.put("orderId", rechargeRecord.getCrId());
        this.result.put("payMoney", rechargeRecord.getRechargeMoney());
        this.result.put("orderKey", rechargeRecord.getRechargeKey());
    }

} else { //订单支付
    ComOrder order = ComOrder.dao.getFirst("orderKey='" + orderKey + "' and wid='"
+ wid + "' and state = 0");
    if (order != null) {
        // 减去优惠应该支付的金额
        BigDecimal payMoney = order.getRealMoney();
        // if (!Utils.isNullOrEmpty(order.getWxNeedPayMoney() + "")) { //订单表
中 wxNeedPayMoney 有值的话即使用钱包+微信支付
        // payMoney = order.getWxNeedPayMoney();
        // }
        miniAppPayBll = new MiniAppPayBll(wid, openId, orderKey, payMoney, "");
        //获取生成与支付单号和小程序支付签名
        log.error("订单支付获取微信预付单 ID=" + miniAppPayBll.createPrePayId());
        String[] result = miniAppPayBll.createPrePayId().split("&");

        //存入 prePayId(发送小程序模板消息使用)
        order.setPrepayId(result[0]);
        order.update();

        this.result = miniAppPayBll.dealResult(result);
        this.result.put("orderId", order.getId());
        this.result.put("payMoney", order.getOrderMoney());
        this.result.put("orderKey", order.getOrderKey());
    } else {
        throw new Exception("未查询到对应订单!");
    }
}
```

```
    }

    } catch (Exception e) {
        e.printStackTrace();
    }
    renderJson(this.result);
}

/**
 * 小票打印
 */
public void printerBill(int wid, String orderKey) {
    try {
        ComWx wx = ComWx.dao.getFirst("wid=" + wid);
        ComOrder order = ComOrder.dao.getFirst("orderKey=' " + orderKey + "' and wid=" + wid);
        //配送费
        BigDecimal deliveryPrice = order.getFreightMoney();
        //包装费
        BigDecimal packPrice = order.getOrderMoney();
        List<ComOrderDetail> orderDetail = ComOrderDetail.dao.getList("orderKey=' " +
orderKey + "' and wid=" + wid);
        Printer printer = new Printer();
        printer.init(PropKit.get("printer_client_id"),
PropKit.get("printer_client_secret"));

        //拼接商品名称和数量
        String allPros = "";

        for (ComOrderDetail detail : orderDetail) {
            allPros += "@@2" + detail.getProName() + "    ¥" + detail.getSalePrice() + "
x" + detail.getBuyNum() + "\n";
        }

        String printTemplate = "@@2 *****" + wx.getShopName() + "*****\n" +
            "-----\n" +
            "下单时间:" + Utils.getFmDate(order.getPayTime(), "yyyy-MM-dd HH:mm") +
"\n" +
            "订单编号:" + order.getOrderNo() + "\n";
        if (order.getOrderType() == 3) { //自提
            printTemplate += "--自提--          @@2 取餐码: " + order.getTakeFoodCode() +
"\n";
        } else if (order.getOrderType() == 2) { //堂食显示桌号
            printTemplate += "--堂食--          @@2 桌号: " + order.getTNo() + "\n";
        } else if (order.getOrderType() == 1) { //堂食显示桌号
            printTemplate += "--外卖-- " + "\n";
        }
        printTemplate += "*****商品*****\n" +
            allPros;
        //外卖才有包装费和配送费
        if (order.getOrderType() == 1) {
            if (!packPrice.equals(BigDecimal.ZERO)) {
                printTemplate += "-----\n" +
                    "@@包装费:¥" + order.getPackMoney() + "\n";
            }
        }
    }
}
```



```

    }
    if (!deliveryPrice.equals(BigDecimal.ZERO)) {
        printTemplate += "-----\n" +
            "@@配送费:¥" + order.getFreightMoney() + "\n";
    }

}

//自提有包装费没有配送费
if (order.getOrderType() == 3) {
    if (!packPrice.equals(BigDecimal.ZERO)) {
        printTemplate += "-----\n" +
            "@@包装费:¥" + wx.getPackPrice() + "\n";
    }
}

if (order.getTableWareNum() != null && order.getTableWareNum() > 0) {
    printTemplate += "-----\n" +
        "@@餐具 x" + order.getTableWareNum() + "份\n";
}

printTemplate += "-----\n" +
    "@@2 订单总价:¥" + order.getPayMoney() + "\n";

if (order.getOrderType() == 1) {
    printTemplate += "@@2" + order.getAcceptName() + " " + order.getMobile() + "\n"
+
    "@@2" + order.getAllAddress() + "\n";
}

if (!Utils.isNullOrEmpty(order.getRemark())) {
    printTemplate += "@@2 订单备注: " + order.getRemark() + "\n";
}

//System.out.println(printTemplate);
printer.print(wx.getPrinterMachineCode(), printTemplate, order.getOrderNo());
log.error("=====打印结束=====");
} catch (Exception e) {
    e.printStackTrace();
}
}
renderJson("OK");
}

public void getShop() {
    int wid = this.getWID();
    log.error("getShop==wid="+wid);
    ComWx shop = ComWx.dao.getFirst("wid=" + wid);
    String headImg = "";
    String shopName = "";
    if (shop != null) {
        headImg = shop.getShoppic();
        shopName = shop.getShopName();
    }
    this.result.put("headImg", headImg);
}

```

```

        this.result.put("shopName", shopName);
        renderJson(this.result);
    }

    /**
     * 小程序的主 se 画面设置 JSON
     *
     * @return
     */
    public void getMallJson() {
        Map<String, Object> mainObj = new HashMap();

        int wid = this.getWID();
        String strJson = "";
        strJson = Utils.getCache("pinindexcontent_json_" + wid, 5);
        ComModule module = ComModule.dao.getFirst("wid=" + wid);
        if (Utils.isNullOrEmpty(strJson) && module != null) {
            String showShopOldPrice = "1";//是否显示商品零售价
            String hasMargin = module.getHasMargin() == 0 ? "0" : "10";//是否留边: 1=是, 0=
            否

            /*功能模块*/
            String optionSql = "select a.*,LENGTH(IFNULL(a.showtitle,'')) AS strLength," +
                "(SELECT COUNT(id) FROM com_modulenaList WHERE optionKey=a.optionKey) as
            datasetLength " +
                " from com_moduleoption a " +
                "where a.modulekey=' " + module.getModuleKey() + "' order by a.sort ";
            List<Record> dtOption = Db.find(optionSql);
            /*end 功能模块*/
            /*商品列表*/
            String shopSql = "SELECT IFNULL(b.saleCount,0) as
            actSalesCount,a.optionKey,b.picUrl AS homeurl,b.proId as id,b.proKey AS
            item_id,CONCAT('/mall/proDetail?proKey=',b.proKey) AS link,b.marketPrice " +
                ",b.marketPrice AS original_price,b.picUrl AS pic,b.minPrice AS
            price,b.minPrice as salePrice,b.proKey as shopKey,b.proName " +
                ",b.proName AS title " +
                "FROM com_moduleshoplist a " +
                "INNER JOIN com_product b ON a.shopkey=b.proKey " +
                "where b.state<>0 and b.stock>0 " +
                "order by a.showindex";
            List<Record> dtShop = Db.find(shopSql);
            /*end 商品列表*/
            /*分组商品列表*/
            List<ComModuleoption> moduleoptionList =
            ComModuleoption.dao.getList("moduleKey=' " + module.getModuleKey() + "' and type=5");

            String groupSql = "SELECT IFNULL(b.saleCount,0) as actSalesCount,b.picUrl AS
            homeurl,b.proId as id,b.proKey AS item_id," +
                "concat('/mall/proDetail?proKey=',cast(b.proKey as char)) AS
            link,b.marketPrice,b.marketPrice AS original_price," +
                "b.picUrl AS pic,b.minPrice AS price,b.minPrice as salePrice,b.proKey as
            shopKey,b.proName " +
                ",b.proName AS title,b.proCatId as catid " +
                "from com_product b " +

```

```

                "WHERE b.state<>0 and b.stock>0 " +
                "AND b.proCatId = ";
//          +
//          "( " +
//          "SELECT group_id FROM com_moduleoption WHERE moduleKey=' " +
module.getModuleKey() + "' " +
//          "AND type=5 " +
//          ") "
        List<Record> dtGroupShop = new ArrayList<>();
        for (ComModuleoption moduleoption : moduleoptionList) {
            if (moduleoption != null) {
                String sql = groupSql;
                sql += moduleoption.getGroupId() + " order by b.sort DESC LIMIT 0," +
moduleoption.getGoodsSize();
                List<Record> proRecord = Db.find(sql);
                dtGroupShop.addAll(proRecord);
            }
        }
        /*end 分组商品列表*/
        List<ComModulenaVlist> dtNav = ComModulenaVlist.dao.getList("wid=" + wid + " ORDER
BY showIndex");
        for (Record option : dtOption) {
            if (!Utils.isNullOrEmpty(option.get("fulltext")))
                option.set("fulltext", option.get("fulltext") + "");
        }

        mainObj.put("hasMargin", hasMargin);
        mainObj.put("dtOption", dtOption);
        mainObj.put("dtShop", dtShop);
        mainObj.put("dtGroupShop", dtGroupShop);
        mainObj.put("dtNav", dtNav);
        mainObj.put("showShopOldPrice", showShopOldPrice);
        //strJson = JSON.toJSONString(mainObj, false);
        strJson = JsonKit.toJson(mainObj);
        //strJson =strJson.replace("\{", "{").replace("}\}", "}");
        Utils.setCache("pinindexcontent_json_" + wid, strJson);
        //renderJson(mainObj);
    }

    renderJson(strJson);

}

/**
 * 获取门店配送 如配送费等
 */
public void getConfig() {
    int wid = this.getWID();
    try {
        ComWx config = ComWx.dao.getFirst(" wid=" + wid + " ");
        if(config!=null && Utils.isNullOrEmpty(config.getMapKey())){
            config.setMapKey(PropKit.get("baiduMapKey"));
        }
    }
}

```

```

    }
    this.result.setData(config);
} catch (Exception e) {
    log.error(e.getMessage());
    e.printStackTrace();
}
renderJson(this.result);
}

/**
 * 获取商品的分类
 */
public void getCatAndProList() {
    int wid = this.getWID();
    int mid = this.getMid();

    try {
        List<Record> proCatList = Db.find("select * , 0 as isSelect from com_procat where
wid =" + wid + " order by sort desc");

        for (Record proCatRecord : proCatList) {

            String order = " c.sort desc , p.sort desc ";
            String where = " where p.wid=" + wid + " and p.state = 1 and p.proCatId = " +
proCatRecord.get("proCatId") + " ";

            List<Record> proList = Db.find("select p.proId , p.proKey , p.proName ,
p.thumbPicUrl , p.marketPrice , p.maxPrice, p.minPrice, " +
" c.proCatId , " +
" count(s.skuKey) as skuNum,COUNT(l.lableKey) as lableNum , s.skuKey ,
s.skuName , s.salePrice as skuPrice , " +
" m.skuKey as cartKey , count(m.skuKey) as cartNum , ifnull(sum(m.num),0)
as buyNum " +
" from com_product p left join com_product_sku s on p.proKey=s.proKey
and s.canSale =1 " +
" left join com_mycart m on s.skuKey=m.skuKey and m.mid=" + mid + " "
+
" left join com_procat c on p.proCatId = c.proCatId " +
" LEFT JOIN com_product_lable l on p.proKey=l.proKey "
+ where +
" group by p.proKey "
+ " order by " + order
);

            BigDecimal buyNum = new BigDecimal("0");
            Long skuNum = 0L;
            for (Record proRecord : proList) {
                buyNum = buyNum.add(proRecord.getBigDecimal("buyNum"));
                skuNum += proRecord.getLong("skuNum");
            }

```

```
        proCatRecord.set("buyNum", buyNum);
        proCatRecord.set("skuNum", skuNum);
        proCatRecord.set("proList", proList);
    }

    this.result.setData(proCatList);
} catch (Exception e) {
    log.error(e.getMessage());
    e.printStackTrace();
}
renderJson(this.result);
}

/**
 * 获取商品
 * cjc
 */
public void getPro() {
    int wid = this.getWID();
    int mid = this.getMid();
    String proKey = this.get("proKey");
    String where = " where p.wid=" + wid + " and p.state = 1 ";

    if (!Utils.isNullOrEmpty(proKey)) {
        where += " and p.proKey = '" + proKey + "' ";
    }

    Record pro = Db.findFirst("select p.proId , p.proKey , p.proName , p.picUrl ,
p.marketPrice , p.maxPrice, p.minPrice, p.content , " +
        " count(s.skuKey) as skuNum ,COUNT(l.lableKey) as lableNum , s.skuKey ,
s.skuName , s.salePrice as skuPrice , " +
        " m.skuKey as cartKey , count(m.skuKey) as cartNum , ifnull(sum(m.num),0) as
buyNum " +
        " from com_product p left join com_product_sku s on p.proKey=s.proKey" +
        " left join com_mycart m on s.skuKey=m.skuKey and m.mid = " + mid + " " +
        " LEFT JOIN com_product_lable l on p.proKey=l.proKey "
        + where +
        " group by p.proKey ");

    this.result.setData(pro);
    renderJson(this.result);
}

/**
 * 获取属性列表
 */
public void getSkuList() {

    int wid = this.getWID();
    int mid = this.getMid();
    String proKey = this.get("proKey");
```

```

String order = " createTime desc ";

String where = " where s.wid=" + wid + " and s.canSale = 1 ";

if (!Utils.isNullOrEmpty(proKey)) {
    where += " and s.proKey='" + proKey + "' ";
}

try {
    //List<Record> skuList = Db.find(" select s.* ,c.lableIdList from com_product_sku
s left join com_myCart c on s.skuKey=c.skuKey and c.mid=" + mid + " " + where + " order by "
+ order);
    List<Record> skuList = Db.find("select s.* from com_product_sku s" + where + " order
by " + order);

    if (skuList.size() == 0) {
        throw new Exception("该商品已下架或暂无可售单品");
    }

    //购物车数量
    //List<Record> skuBuyList = Db.find(" select s.*,ifnull(c.num,0) as
buyNum,c.lableIdList from com_product_sku s left join com_myCart c on s.skuKey=c.skuKey and
c.mid=" + mid + " " + where + " GROUP BY c.skuKey order by c.lableIdList," + order);
    List<Record> skuBuyList = Db.find(" select s.*,ifnull(c.num,0) as
buyNum,c.lableIdList from com_product_sku s left join com_myCart c on s.skuKey=c.skuKey and
c.mid=" + mid + " " + where + " order by c.lableIdList," + order);

    //获取标签
    List<ComProductLable> lableList = ComProductLable.dao.getList(" wid=" + wid + " and
proKey='" + proKey + "'");
    String pIds = "";
    String lableKey = "";
    for (ComProductLable lable : lableList) {
        pIds += lable.getPId() + ",";
        lableKey += lable.getLabelKey() + ",";
    }
    if (pIds.endsWith(",")) {
        pIds = pIds.substring(0, pIds.length() - 1);
    }
    if (!Utils.isNullOrEmpty(lableKey) && lableKey.length() > 0) {
        lableKey = lableKey.substring(0, lableKey.length() - 3);
    }
    String whereLab = "";
    if (!Utils.isNullOrEmpty(pIds)) {
        whereLab += " and pId in (" + pIds + ") ";
    }
    String whereLabKey = "";
    if (!Utils.isNullOrEmpty(lableKey)) {
        whereLabKey += " and propKey in ('" + lableKey + "') ";
    }
    //List<ComPropValue> propValue=ComPropValue.dao.getList(" wid="+wid+" and state=1

```

```
+whereLab);
    List<Record> propValue = Db.find("select a.*, IF(IFNULL(b.selectVid,0)=0,0,1) as
isSelected from com_lable_value a " +
        "LEFT JOIN ( " +
        "SELECT MIN(id) as selectVid from com_lable_value where wid=" + wid +
whereLab + whereLabKey + " GROUP BY pId " +
        ") b on a.id=b.selectVid " +
        "where wid=" + wid + whereLab + whereLabKey + " ORDER BY id");
//初始化标签
List<Record> selectlabs=null;
if(!Utils.isNullOrEmpty(whereLab) && !Utils.isNullOrEmpty(whereLabKey)){
    selectlabs = Db.find("select b.id,b.`name`,b.pId " +
        "from ( " +
        "SELECT MIN(id) as id,`name`,pId from com_lable_value where wid=" + wid
+ whereLab + whereLabKey + " GROUP BY pId " +
        ") b ORDER BY b.id");
}
Map<Object, Object> dataLabMap = new HashMap<>();
Map<String, String> selectlabMap = new HashMap<>();
Map<String, String> selectPidMap = new HashMap<>();
if (selectlabs != null && selectlabs.size() > 0) {
    for (Record sellab : selectlabs) {
        selectlabMap.put(sellab.get("id"), sellab.getStr("name"));
        selectPidMap.put(sellab.get("pId"), sellab.get("id"));
    }
    dataLabMap.put("selectlabMap", selectlabMap);
    dataLabMap.put("selectPidMap", selectPidMap);
    this.result.setFooter(dataLabMap);
}
Map<Object, Object> dataMap = new HashMap<>();
dataMap.put("skuBuyList", skuBuyList);
dataMap.put("skuList", skuList);

this.result.setRowdate(lableList);
this.result.setRows(propValue);
this.result.setData(dataMap);
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
}

renderJson(this.result);
}
```

中华人民共和国国家版权局

计算机软件著作权登记证书

证书号： 软著登字第5264709号

软件名称： 智源服装APS系统
[简称： 服装APS系统]
V1.0

著作权人： 厦门元谷信息技术有限公司

开发完成日期： 2019年05月24日

首次发表日期： 2019年05月31日

权利取得方式： 原始取得

权利范围： 全部权利

登记号： 2020SR0386013

根据《计算机软件保护条例》和《计算机软件著作权登记办法》的规定，经中国版权保护中心审核，对以上事项予以登记。



No. 05602536



2020年04月27日

服装 APS 系统说明书

厦门元谷信息技术有限公司

刘孟平 13860181846

地址：厦门市思明区软件园 1 期华讯楼 C 区 1F-C1

1. 服装 APS 系统简介

本系统旨在解决

- 1、服装行业中生产计划管控薄弱、经常性延期出货、规范以及整合（BOM、工艺单）、物料控制不准确、成本分析困难的痛点；
- 2、力求达到解决传统服装行业中订单处理环节多，周期长，成本管控靠经验管理的风险；
- 3、帮助企业精细化管理过程和细节，并将成本节点固化到流程控制中，形成成本控制稽核与体系。

2. 应用的数据格式说明

格式例子如下，按照 JSON 的格式输出：

```
[{  
  "name": "faceQisetTime",  
  "value": "2019-07-03 到 2019-07-27"  
}, {  
  "name": "sealingExcipientQiSetTime",  
  "value": "2019-07-17 到 2019-07-27"  
}, {  
  "name": "embroideryCheckTime",  
  "value": "2019-07-06 到 2019-07-13"  
}, {  
  "name": "sealingCompleteTime",  
  "value": "2019-07-11 到 2019-07-20"  
}, {  
  "name": "sealingApprovalTime",  
  "value": "2019-07-18 到 2019-08-15"  
}, {  
  "name": "cutTime",
```

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

```
"value": "2019-07-24 到 2019-07-30"
}, {
  "name": "flowerOutburstTime",
  "value": "2019-07-16 到 2019-07-27"
}, {
  "name": "flowerBackFactoryTime",
  "value": "2019-07-12 到 2019-07-27"
}, {
  "name": "sewingOnLineTime",
  "value": "2019-07-01 到 2019-07-19"
}];
["faceQisetTime",
"sealingCropTime",
"sealingEmbroideringTime",
"embroideredTime",
"embroideryCheckTime",
"sealingCompleteTime",
"sealingApprovalTime",
"undercutTime",
"cutTime",
"flowerOutburstTime"
];
[{
  "proNo": "现有设备",
  "devName30": "1042.0",
  "devName10": "0.0",
  "devName32": "10.0",
```

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

```
"devName31": "1.0",  
"devName26": "3.0",  
"devName22": "100.0",  
"devName9": "63.0"  
}}
```

3. 系统架构

JFinal 是基于 Java 语言的极速 WEB + ORM 框架，其核心设计目标是开发迅速、代码量少、学习简单、功能强大、轻量级、易扩展、Restful。在拥有 Java 语言所有优势的同时再拥有 ruby、python、php 等动态语言的开发效率！

JFinal 有如下主要特点：

MVC 架构，设计精巧，使用简单

遵循 COC 原则，零配置，无 xml

独创 Db + Record 模式，灵活便利

ActiveRecord 支持，使数据库开发极致快速

自动加载修改后的 java 文件，开发过程中无需重启 web server

AOP 支持，拦截器配置灵活，功能强大

Plugin 体系结构，扩展性强

多视图支持，支持 FreeMarker、JSP、Velocity

强大的 Validator 后端校验功能

功能齐全，拥有 struts2 的绝大部分功能

体积小仅 339K，且无第三方依赖。

3.1 configEngine

JFinal 3.0 新增了模板引擎模块，继承 JFinalConfig 的实现类中需要添加 public void configEngine(Engine me)方法，以便对模板引擎进行配置。

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

项目升级如果不使用 Template Engine 该方法可以留空。JFinal 3.0 默认 ViewType 为 ViewType.JFINAL_TEMPLATE, 如果老项目使用的是 Freemarker 模板, 并且不希望改变模板类型, 需要在 configConstant 方法中通过 me.setViewType(ViewType.FREE_MARKER)进行指定, 以前已经指定过 ViewType 的则不必理会。

3.2 RenderFactory

JFinal 3.0 对 render 模块做了全面重构, 抽取出了 IRenderFactory 接口, 而原来的 RenderFactory 成为了接口的默认实现类, 去除了原来的 IMainRenderFactory、IErrorRenderFactory、IXmlRenderFactory 三个接口, 所有对 render 的扩展与定制全部都可以通过继承 RenderFactory 来实现, 3.0 版本的 render 模

块可对所有 render 进行切换与定制, 并且扩展方式完全一致。如果老项目对 IMainRenderFactory 做过扩展, 只需要照如下方式进行升级: 同理, 如果以前对 IErrorRenderFactory 或者 IXmlRenderFactory 做过扩展的, 只需要在上面的 MyRenderFactory 类中添加上 getErrorRender(...) 与 getXmlRender(...) 方法即可。

JFinal 3.0 对所有 render 扩展, 采取了完全一致的扩展方式, 学习成本更低, 使用更方便, 升级也很方便。此外, 原来 RenderFactory 类中的 me() 已经被取消, 老项目对此有依赖的只需要将 RenderFactory.me() 直接改为 RenderManager.me().getRenderFactory() 即可。

3.3 baseViewPath

baseViewPath 设置由原来的 configConstant(...) 方法中转移到了 Routes 对象中, 并且可以对不同的 Routes 对象分别设置。

从 configConstant(...) 转移到 configRoute(...) 中的好处是可以分别对不同的 Routes 进行设置, 不同模块的 baseViewPath 很可能不相同, 从而可以减少冗余代码。上面的代码示例是用于 Routes 拆分后的情况, 如果你的应用并没有对 Routes 进行拆分, 只需要在 configRoute 中

3.4 多数据源支持

ActiveRecordPlugin 可同时支持多数据源、多方言、多缓存、多事务级别等特性, 对每个

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

ActiveRecordPlugin 可进行彼此独立的配置。简言之 JFinal 可以同时使用多数据源，并且可以针对这多个数据源配置独立的方言、缓存、事务级别等。当使用多数据源时，只需要对每个 ActiveRecordPlugin 指定一个 configName 即可。

3.5 Cron4jPlugin 定时任务

Cron4jPlugin 是 JFinal 集成的任务调度插件，通过使用 Cron4jPlugin 可以使用通用的 cron 表达式极为便利的实现任务调度功能。

3.6 其它

Ret.put(...).put(...)这种链式用法，需要改成 Ret.set(...).set(...)，因为 Ret 改为继承自 HashMap，为了避免与 HashMap.put(...)相冲突。Ret.get(...)方法返回泛型值的场景改为 Ret.getAs(...)。
configConstant(...)的 Constants 参数中的 setFreeMarkerExtension、setVelocityExtension 方法统一改为使用 setViewExtension 方法。setMainRenderFactory、setErrorRenderFactory 被 setRenderFactory 取代。
renderXml(...)方法依赖的 XmlRender 由原来 Freemarker 语法实现改成了由 JFinal Template Engine 实现，用到 renderXml(...)的项目需要修改模板内容。

4. MVC 框架

MVC 全名是 Model View Controller，是模型(model)—视图(view)—控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC 被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

MVC 开始是存在于桌面程序中的，M 是指业务模型，V 是指用户界面，C 则是控制器，使用 MVC 的目的是将 M 和 V 的实现代码分离，从而使同一个程序可以使用不同的表现形式。比如一批统计数据可以分别用柱状图、饼图来表示。C 存在的目的则是确保 M 和 V 的同步，一旦 M 改变，V 应该同步更新。

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

模型-视图-控制器 (MVC) 是 Xerox PARC 在二十世纪八十年代为编程语言 Smalltalk-80 发明的一种软件设计模式, 已被广泛使用。后来被推荐为 Oracle 旗下 Sun 公司 Java EE 平台的设计模式, 并且受到越来越多的使用 ColdFusion 和 PHP 的开发者的欢迎。模型-视图-控制器模式是一个有用的工具箱, 它有很多好处, 但也有一些缺点。

4.1 MVC 编程模式编辑

MVC 是一种使用 MVC (Model View Controller 模型-视图-控制器) 设计创建 Web 应用程序的模式:

Model (模型) 表示应用程序核心 (比如数据库记录列表)。

View (视图) 显示数据 (数据库记录)。

Controller (控制器) 处理输入 (写入数据库记录)。

MVC 模式同时提供了对 HTML、CSS 和 JavaScript 的完全控制。

Model (模型) 是应用程序中用于处理应用程序数据逻辑的部分。

通常模型对象负责在数据库中存取数据。

View (视图) 是应用程序中处理数据显示的部分。

通常视图是依据模型数据创建的。

Controller (控制器) 是应用程序中处理用户交互的部分。

通常控制器负责从视图读取数据, 控制用户输入, 并向模型发送数据。

MVC 分层有助于管理复杂的应用程序, 因为您可以在一个时间内专门关注一个方面。例如, 您可以在不依赖业务逻辑的情况下专注于视图设计。同时也让应用程序的测试更加容易。

MVC 分层同时也简化了分组开发。不同的开发人员可同时开发视图、控制器逻辑和业务逻辑。

MVC 指 MVC 模式的某种框架, 它强制性的使应用程序的输入、处理和输出分开。使用 MVC 应用程序被分成三个核心部件: 模型、视图、控制器。它们各自处理自己的任务。最典型的 MVC 就是 JSP + servlet + javabean 的模式。

4.2 视图

视图是用户看到并与之交互的界面。对老式的 Web 应用程序来说, 视图就是由 HTML 元素组成的界面, 在新式的 Web 应用程序中, HTML 依旧在视图中扮演着重要的角色, 但一些新的技术已层出不穷, 它们包括 Adobe Flash 和像 XHTML, XML/XSL,WML 等一些标识语言和 Web services.

MVC 好处是它能为应用程序处理很多不同的视图。在视图中其实没有真正的处理发生, 不管这些数据是联机存储的还是一个雇员列表, 作为视图来讲, 它只是作为一种输出数据并允许用户操纵的方式。

4.3 模型

模型表示企业数据和业务规则。在 MVC 的三个部件中, 模型拥有最多的处理任务。例如它可能用像 EJBs 和 ColdFusion Components 这样的构件对象来处理数据库, 被模型返回的数据是中立的, 就是说

模型与数据格式无关, 这样一个模型能为多个视图提供数据, 由于应用于模型的代码只需写一次就可以被多个视图重用, 所以减少了代码的重复性。

4.4 控制器

控制器接受用户的输入并调用模型和视图去完成用户的需求, 所以当单击 Web 页面中的超链接和发送 HTML 表单时, 控制器本身不输出任何东西和做任何处理。它只是接收请求并决定调用哪个模型构件去处理请求, 然后再确定用哪个视图来显示返回的数据。

4.5 特点

4.5.1 优点

4.5.1.1 耦合性低

视图层和业务层分离, 这样就允许更改视图层代码而不用重新编译模型和控制器代码, 同样, 一个应用的业务流程或者业务规则的改变只需要改动 MVC 的模型层即可。因为模型与控制器和视图相分离, 所以很容易改变应用程序的数据层和业务规则。

模型是自包含的, 并且与控制器和视图相分离, 所以很容易改变应用程序的数据层和业务规则。如果把数据库从 MySQL 移植到 Oracle, 或者改变基于 RDBMS 数据源到 LDAP, 只需改变模型即可。一

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

且正确的实现了模型，不管数据来自数据库或是 LDAP 服务器，视图将会正确的显示它们。由于运用 MVC 的应用程序的三个部件是相互独立，改变其中一个不会影响其它两个，所以依据这种设计思想能构造良好的松耦合的构件。[11]

4.5.1.2 重用性高

随着技术的不断进步，需要用越来越多的方式来访问应用程序。MVC 模式允许使用各种不同样式的视图来访问同一个服务器端的代码，因为多个视图能共享一个模型，它包括任何 WEB（HTTP）浏览器或者无线浏览器（wap），比如，用户可以通过电脑也可通过手机来订购某样产品，虽然订购的方式不一样，但处理订购产品的方式是一样的。

由于模型返回的数据没有进行格式化，所以同样的构件能被不同的界面使用。例如，很多数据可能用 HTML 来表示，但是也有可能用 WAP 来表示，而这些表示所需要的命令是改变视图层的实现方式，而控制层和模型层无需做任何改变。由于已经将数据和业务规则从表示层分开，所以可以最大化的重用代码了。模型也有状态管理和数据持久性处理的功能，例如，基于会话的购物车和电子商务过程也能被 Flash 网站或者无线联网的应用程序所重用。[11]

4.5.1.3 生命周期成本低

MVC 使开发和维护用户接口的技术含量降低。

4.5.1.4 部署快

使用 MVC 模式使开发时间得到相当大的缩减，它使程序员（Java 开发人员）集中精力于业务逻辑，

界面程序员（HTML 和 JSP 开发人员）集中精力于表现形式上。

4.5.1.5 可维护性高

分离视图层和业务逻辑层也使得 WEB 应用更易于维护和修改。

有利软件工程化管理

由于不同的层各司其职，每一层不同的应用具有某些相同的特征，有利于通过工程化、工具化管理程序代码。控制器也提供了一个好处，就是可以使用控制器来联接不同的模型和视图去完成用户的需

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

求，这样控制器可以为构造应用程序提供强有力的手段。

给定一些可重用的模型和视图，控制器可以根据用户的需求选择模型进行处理，然后选择视图将处理结果显示给用户。[12-13]

4.5.2 缺点

4.5.2.1 没有明确的定义

完全理解 MVC 并不是很容易。使用 MVC 需要精心的计划，由于它的内部原理比较复杂，所以需要花费一些时间去思考。

同时由于模型和视图要严格的分离，这样也给调试应用程序带来了一定的困难。每个构件在使用之前都需要经过彻底的测试。

4.5.2.2 不适合小型，中等规模的应用程序

花费大量时间将 MVC 应用到规模并不是很大的应用程序通常会得不偿失。

4.5.2.3 增加系统结构和实现的复杂性

对于简单的界面，严格遵循 MVC，使模型、视图与控制器分离，会增加结构的复杂性，并可能产生过多的更新操作，降低运行效率。

4.5.2.4 视图与控制器间的过于紧密的连接

视图与控制器是相互分离，但却是联系紧密的部件，视图没有控制器的存在，其应用是很有限的，反之亦然，这样就妨碍了他们的独立重用。

4.5.2.5 视图对模型数据的低效率访问

依据模型操作接口的不同，视图可能需要多次调用才能获得足够的显示数据。对未变化数据的不必要的频繁访问，也将损害操作性能。

4.5.2.6 一般高级的界面工具或构造器不支持模式

改造这些工具以适应 MVC 需要和建立分离的部件的代价是很高的，会造成 MVC 使用的困难。

编号:	智源服装 APS 系统	版本: V1.0
-----	-------------	----------

5. 界面介绍

PC 端说明:

1、成本报价: 记录企业和客户之间报价往来, 并通过权限管控每个部门可以查看和编辑的列, 系统会记录每次报价的数据并用颜色标注出修改的部分。最终敲定产品报价并签订订单

状态	货号	品名	物料名称	款式	总数量	分色件数	分类	后整工艺	面料成分	使用部位	料率(百锋)	料率(361)
1	552010839	套头卫衣	棉涤健康双面布	55201ES014	6002	0	01.面料	染色	63%棉37%聚酯纤维	大身	0.525	0.525
2	552010839	套头卫衣	1CM电竞后领织带	55201ES014	6002	0	06.辅料	*	*	后领织带	0.4	0.4
3	552010839	套头卫衣	标牌 电竞6 圆标	55201ES014	6002	0	06.辅料	*	*	左袖	1	1
4	552010839	套头卫衣	0.5CM棉织带	55201ES014	6002	0	06.辅料	*	*	左右肩骨缝	0.42	0.42
5	552010839	套头卫衣	线	55201ES014	6002	0	07.通用	*	*	*	1	1
6	552010839	套头卫衣	拷贝纸	55201ES014	6002	0	07.通用	*	*	包装用	1	1
7	552010839	套头卫衣	超柔质地水洗唛 (跟同季)	55201ES014	6002	0	07.通用	*	*	*	1	1
8	552010839	套头卫衣	干燥剂	55201ES014	6002	0	07.通用	*	*	包装用	1	1
9	552010839	套头卫衣	361度取针	55201ES014	6002	0	07.通用	*	*	挂主吊牌用	1	1

在上图中选择导入客户报价单, 并在系统内修改报价单并导出传至客户手中, 客户可修改报价再传回, 如此往复形成双方确定的报价单。

2、销售订单: 将销售单导入系统中, 对销售单进行统一管理。

智源服装APS系统 生产监控
总管理员

首页
月出货计划
成本报价
销售订单

成本报价
销售订单
材料货期汇总
面辅料进度表

删除
导入
导出
导出模板
361
20Q1
筛选
清空筛选
显示列
设置

款号	品名	颜色	操作	状态	订单数量	XXXS	XXS	XS	S	M	L	XL	2XL	3XL	4XL	5XL	6X
1	552014652	绒里夹克	基础黑	结束订单	待生产	12638	1	0	12	370	1845	3249	3244	2355	1303	259	0
2	552019851	套头卫衣	基础黑	结束订单	待生产	903	0	0	0	70	200	251	203	112	51	16	0
3	562018708	连帽套头卫衣	基础黑	结束订单	待生产	751	0	0	0	55	182	234	165	94	20	1	0
4	552014039B	针织长裤	基础黑	结束订单	待生产	15769	0	0	7	210	1579	3663	4245	3300	1924	841	0
5	562014652	绒里夹克	大学红	结束订单	待生产	7987	0	0	26	776	2032	2360	1764	897	125	7	0
6		基础黑	结束订单	待生产	4145	0	0	11	267	1013	1243	984	523	101	3	0	
7	562018810	连帽套头卫衣	基础黑	结束订单	待生产	5728	0	0	111	656	1514	1625	1156	527	121	18	0
8	562014651	翻领短袖	基础黑	结束订单	待生产	1610	0	0	7	128	307	382	344	228	162	52	0
9	562019851	套头卫衣	大学红	结束订单	待生产	937869											
					显示1到50,共202记录												

3、计划排程：系统支持灵活科学的生产计划排程管理，可对订单进行预排，最终完成计划排程并发布，生产部门进行生产。同时通过一系列的生产预警，形成对生产的完整进度跟踪和控制体系。

智源服装APS系统 生产监控
总管理员

首页
月出货计划
成本报价
销售订单
主计划排程

主计划排程
委外生产
月出货计划
生产日历表
标准日工资
BCM成品入库
BCM成品出库

新增
删除
导出
发布
全部发布
361
19Q4
全部
预计上线时间范围
预计下线时间范围
查询
筛选
清空筛选

颜色标识
待分配
生产多款
超期
预警
即将下线
过期(未上线)

款号	品名	订单数	组别	生产款数	预计生产数	已生产数	未生产数	预计上线	更新上线	实际上线	生产周期	剩余生产周期	计划日产量	实际日产量	预计下线
29	551944602	23529	307(28)	2	12500	4560	7940	07-12	07-25	07-13	31	15	0	536	08-16
30			205(12)	1	12000	4325	7675	07-11	07-25	07-13	34	18	526	449	08-20
31	551944602	23529	203(17)	1	9729	1144	8585	09-08	07-25	07-09	87	67	302	130	10-16
32			305(14)	1	13800	3034	10766	06-27	07-25	06-27	91	61	233	178	10-05
					显示1到50,共55记录										

款号: 202E(10) 自保存

品名: 202E(10)

待分配数: 0

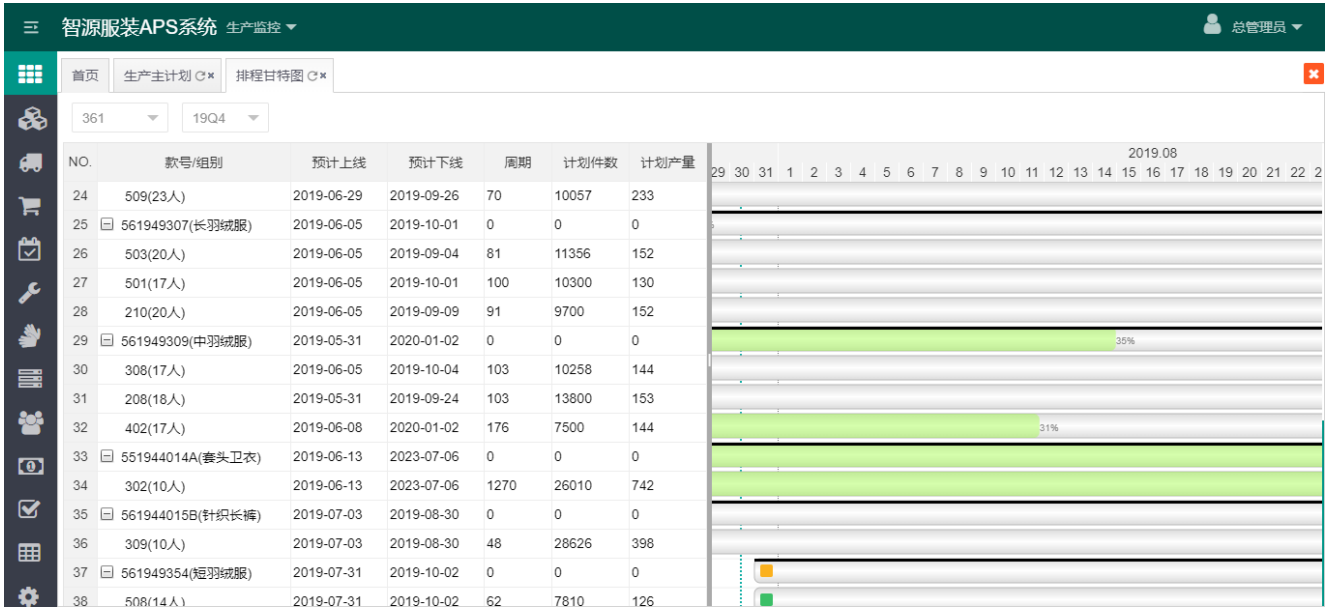
预估工价: 202E(10)

操作	组别	人均件数	计划生产数	计划日产量	实际日产量	生产周期	产能提升	实际上线	预计上线	预计	201(13)	已排款号	551944203	预计上线	06-27	预计下线	11-19
	202(12)																
	202E(10)												651834706	07-02		08-06	

4、系统提供生产甘特图，向企业管理人员清晰展示了计划产量与计划时间的对应关系，以及每日实际产量与预定计划产量的对比关系

第13页，共37页

第147页 / 共201页



5、委外管理：订单中部分生产任务的委外加工，并在系统中填写委外完成状况，达到对委外生产的全程跟踪与管控。系统对委外生产进度进行预警，尽可能规避因委外加工不可控性较大对整个生产进度产生延期的风险。

	款号	品名	订单数	已完成数	未完成数	预计完成日	合同货期	预计超期	委外工厂	备注
1	551930102	圆领短T恤	6442	0	6442		06-15		山东百锋服饰有限公司	
2	551934010B	针织长裤	5993	0	5993		07-15		山东百锋服饰有限公司	
3	551934119	圆领短T恤	28933	0	28933		05-15		山东百锋服饰有限公司	
4	551934851	套头卫衣	60696	0	60696		07-15		山东百锋服饰有限公司	
5	551934861	连帽套头卫衣	40059	0	40059		07-31		山东百锋服饰有限公司	
6	551939102	圆领短T恤	21127	0	21127		05-15		山东百锋服饰有限公司	
7	551939106	圆领短T恤	4215	0	4215		05-15		山东百锋服饰有限公司	
8	551939719	针织长裤	27445	0	27445		06-15		山东百锋服饰有限公司	
9	551939802	连帽套头卫衣	11476	0	11476		07-31		山东百锋服饰有限公司	
10	561934720	运动长裤	23845	0	23845		05-15		山东百锋服饰有限公司	
	合计		335684	0	335684					

6、月出货计划：主计划排程发布后，根据订单款的交货时间自动生成月出货计划。管理人员可在系统中安排每天的出货计划，结束以往需要使用一堆出货 EXCEL 的烦恼。并支持点击页面【更新出入库】按钮打通与企业库存系统的对接，展示当前实时的库存数量，以便准确地填写待出库数量。系统同时也会对出货情况进行预警，避免出货延迟。

智源服装APS系统 生产监控 总管理员

主计划排程 委外生产 月出货计划 生产日历表 标准日工资 BCM成品入库 BCM成品出库

新增款号 新增出货时间 导出 更新库存 2019-06 请输入款号 查询 上一月 下一月 颜色标识 超期

季度	客户	款号	合同交期	品名	分类	订单数	库存数	06-03	06-04	06-06	06-10	06-13	06-17	06-20	06-25	06-27	合计	
1	19Q3	361	551934703	06-15	运动九分裤	针织	24774	4378	0	24774	0	0	0	0	0	0	24774	
2	19Q3	361	551934008B	07-31	针织长裤	针织	140885	0	0	0	0	0	0	0	24979	38289	63268	
3	19Q3	361	551934002B	07-31	针织长裤	针织	70213	65862	0	24969	0	0	20898	0	0	0	45867	
4	19Q3	361	551939032B	07-31	针织长裤	针织	58864	0	0	0	0	0	0	0	0	29000	29000	
5	19Q3	361	551934002A	07-31	连帽开襟卫衣	针织	72174	55859	0	19889	15572	0	0	16267	0	0	51728	
6	19Q3	361	551939032A	07-31	连帽开襟卫衣	针织	60222	55573	0	0	0	0	0	22294	0	0	22294	
7	19Q3	361	551934705	06-15	运动长裤	针织	20922	26207	20922	0	0	0	0	0	0	0	20922	
8	19Q3	361	551939002B	07-31	针织长裤	针织	11272	0	0	0	11272	0	0	0	0	0	11272	
9	19Q3	361	551933607	07-31	单风衣	梭织	5463	0	0	0	0	0	2733	2730	0	0	5463	
合计							479185	208929	22172	69632	15572	11272	20898	24848	30872	32277	67289	294832

50 第 1 页 共 1 页 显示 1 到 13 共 13 记录

7、材料货期详情与汇总：通过对材料货期详情表的分析和统计生成材料货期汇总表，展示材料首次配套的时间。完善企业对物料计划的管控和车生产的管控

智源服装APS系统 生产监控 总管理员

成本报价 销售订单 材料货期汇总 面料辅料进度表

导出 361 19Q3 请选择到料时间 查询 筛选 清空筛选 材料货期详情

款号	品名	分色数量	颜色编号	颜色	辅料到齐	面料首批到齐	面料最后到齐
1	551930102 圆领短T恤	1048	B51	本白	04-10	04-05	
2		2426	B19	基础黑	04-04	03-30	
3		2968	X47	浅驼色	04-04	03-30	
4	551930103 套头卫衣	1555	B51	本白	04-10	04-05	
5		1757	B19	基础黑	04-10	04-05	
6	551930601 单风衣	762	B19	基础黑	03-10	03-05	
7		488	B51	本白			
8	551930703 休闲九分裤	4179	B19	基础黑	03-10	03-05	
9		1669	X10	苍灰	03-30	03-25	
10	551933703 休闲长裤	3340	B19	基础黑			
11	551933706 运动长裤	14671	B19	基础黑	01-30	01-25	

50 第 1 页 共 3 页 显示 1 到 50 共 111 记录

智源服装APS系统 生产监控

首页 材料货期汇总 材料货期详情

361 19Q4 筛选 清空筛选

款号	色组	物料	颜色	面料商	成分	规格	订单数(件)	采购数	单位	使用部位	
1	551940606	1	棉感细牛津	电网绿Q99	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	992	1454.668	M	大身、挂面
2	551940606	1	棉感细牛津	电网绿Q99	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	992	2.0634	M	后领挂耳 直切0.022M*0.09M
3	551940606	1	210T涤塔夫	芒果黄J03	贸盛	染色(防静电)-100%聚酯纤维	1.48M	992	753.8208	M	袋布、袖里
4	551940606	1	迷彩提花羊羔绒	柚黄J24/岩井	连捷	染色-100%聚酯纤维	1.57M/4	992	362.2387	KG	大身内里
5	551940606	2	棉感细牛津	基础黑B19	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	1302	1890.894	M	大身、挂面
6	551940606	2	棉感细牛津	基础黑B19	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	1302	2.6821	M	后领挂耳 直切0.022M*0.09M
7	551940606	2	210T涤塔夫	芒果黄J03	贸盛	染色(防静电)-100%聚酯纤维	1.48M	1302	989.3898	M	袋布、袖里
8	551940606	2	迷彩提花羊羔绒	深雾灰D55/庄	连捷	染色-100%聚酯纤维	1.57M/4	1302	475.4383	KG	大身内里
9	551940314	1	棉感细牛津	基础黑B19	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	860	1909.544	M	大身、四合扣垫片
10	551940314	1	棉感细牛津	基础黑B19	优品	染色+贴膜(普通灰膜、防静电)-100%	1.44M	860	1.7888	M	后领挂耳 直切0.022M*0.09M
11	551940314	1	210T涤塔夫	基础黑B19	贸盛	染色(防静电)-100%聚酯纤维	1.48M	860	1592.118	M	里料
12	551949709	1	四面弹复合摇粒	基础黑B19	明协	染色+复合+满印(防静电)-面:90%慕	1.38M	6395	6501.221	M	大身

50 共8页 显示1到50,共391记录

8、面辅料进度表：同步企业用友数据库 BOM 信息，将订单与所需面辅料匹配，通过在界面编辑单耗，系统自动算出款号对应物料的可配套数量，准确的把控每个款对应的面辅料可配套数量。并展示了面辅料采购数与入库数情况，保证生产进度按计划进行，避免生产延期。

智源服装APS系统 生产监控

首页 面辅料进度表

成本报价 销售订单 材料货期汇总 面辅料进度表

同步用友数据 导出 361 19Q3 筛选 清空筛选 面辅料合同进度 面料汇总表 辅料汇总表

入库详情	下单时间	款号	物料名称	材料颜色	单耗	损耗	订单数	可裁剪数	可配套数	采购数	入库总数	欠数	入库日	入库数
21	2019-01-21	551934860	素色棉扁罗纹	深墨蓝NB4米白	0	0				14360	14360	0	2019-03-22	14360
22	2019-01-21	561939002A	19Q3打包带		0	0				3	3	0	2019-04-30	3
23	2019-02-15	561939002B	19Q3主唛		0	0				103	103	0	2019-03-22	103
24	2019-01-21	561934113	科恩德777501有款	本白B51	0	0				77	77	0	2019-02-26	77
25	2019-01-21	551939032B	常规扁棉帽绳	基础黑B19	0	0				5585	5128	457	2019-02-28	5128
26	2019-01-21	551939032A	标牌	基础黑B19	0	0	32220			18434	17425	1009	2019-04-30	7105
27	2019-01-21	551939032A	19Q3细线	基础黑B19	0	0	32220			22	22	0	2019-02-28	22
28	2019-02-18	561939604	19Q3胶袋		0	0				2	2	0	2019-04-30	2
29	2019-01-21	551930601	19Q3油光纸		0	0		0		525	525	0	2019-03-31	525

50 共8页 显示1到50,共2741记录

9、面辅料合同进度：展示面料合同到期详情，最后一次到期日期，已入库数量和欠数。可也查看每个物料到期详情并对延期进行预警。同时记录供应商对合同的履行力度，更为企业评估、挑选优质供应商提供可考究的依据，也为订单的生产提供基础，避免生产延期给企业带来损失。

智源服装APS系统 生产监控 总管理员

首页 月计划 成本报价 面辅料进度表 面辅料合同进度

361 19Q3 请输入款号 查询

合同进度	款号	物料名称	颜色	尺码	规格	单位	采购数	入库数	欠数	最后一次			
										当天应到	应到总数	实际到料	占比
1	561939801	生活双色针织织带	花蕊粉T19/本			米	3356	3356	0	3356	3356	1356	40%
2	561939801	生活双色针织织带	深墨蓝NB4/本			米	5781	5781	0	5781	5781	5781	100%
3	561939801	19Q3吊牌合格证		S		个	11	11	0	11	11	11	100%
4	561939801	19Q3吊牌合格证		M		个	28	28	0	28	28	28	100%
5	561939801	19Q3吊牌合格证		L		个	32	32	0	32	32	32	100%
6	561939801	19Q3吊牌合格证		XL		个	25	25	0	25	25	25	100%
7	561939801	19Q3吊牌合格证		XS		个	2	2	0	2	2	2	100%
8	561939801	19Q3吊牌合格证		2XL		个	14	14	0	14	14	14	100%
9	561939801	19Q3吊牌合格证		3XL		个	4	4	0	4	4	4	100%
10	561939801	19Q3吊牌合格证		4XL		个	1	1	0	1	1	1	100%

50 第1 共56页 显示1到50,共2724记录

智源服装APS系统 生产监控 总管理员

首页 面辅料合同进度详情

款号	物料名称	颜色	尺码	19-03-20				19-03-25				19-03-30				
				料延期	应到数量	实际到料数	占比	当日应到数	面料延期	应到数量	实际到料数	占比	当日应到数	面料延期	应到数量	实
39	561939801	人字纹织带(涤)	本白B51					1734		1734	1734	100%				
40	561939801	人字纹织带(涤)	松粉色SE5					1007		1007	1007	100%				
41	561939801	19Q3细线	深墨蓝NB4					18		18	18	100%				
42	561939801	生活双色针织织带	花蕊粉T19/本白E					3356		3356	2000	60%				
43	561939801	19Q3主唛		XS				58		58	58	100%				
44	561939801	19Q3主唛		S				531		531	531	100%				
45	561939801	19Q3主唛		M				1427		1427	1427	100%				
46	561939801	19Q3主唛		L				1639		1639	1639	100%				
47	561939801	19Q3主唛		XL				1251		1251	1251	100%				
48	561939801	19Q3主唛		2XL				684		684	684	100%				
49	561939801	19Q3主唛		3XL				171		171	171	100%				
50	561939801	19Q3主唛		4XL				19		19	19	100%				

50 第1 共2页 显示1到50,共69记录

10、面辅料汇总表：多维度对面辅料到料情况进行统计分析

按供应商采购数分析				按供应商面种分析				
供应商	采购总数	入库总数	占比	供应商	物料名称	采购总数	入库总数	占比
佛山市三水昊通印染有限公司	24973	24972	100.0%	= 佛山市三水昊通印染有限公司 - 1				
厦门科恩得服装材料有限公司	9394	9394	100.0%	佛山市三水昊通印染有限公司	40S棉弹力布	24973	24972	100.0%
明协 (福建) 鞋服有限公司	463	463	100.0%	= 厦门科恩得服装材料有限公司 - 3				
晋江澳讯贸易有限公司	18092	18092	100.0%	厦门科恩得服装材料有限公司	科恩德777502有胶衬布	7536	7536	100.0%
泉州市云隆纺织贸易有限公司	33219	15866.9	47.8%	厦门科恩得服装材料有限公司	1035S无胶衬纸	812	812	100.0%
泉州振兴纺织有限公司	23298	23285.5	99.9%	厦门科恩得服装材料有限公司	科恩德777501有胶衬布	1046	1046	100.0%
石狮市谦成纺织贸易有限公司	2626	2626	100.0%	= 明协 (福建) 鞋服有限公司 - 1				
石狮禾宝纺织有限公司	328449	304367.6	92.7%	明协 (福建) 鞋服有限公司	75D四角网眼	463	463	100.0%
石狮家宝染织有限公司	24909	24909	100.0%	= 晋江澳讯贸易有限公司 - 1				
合计	572682	530158	92.6%	晋江澳讯贸易有限公司	四面弹棉布	18092	18092	100.0%
				合计				
50 1 共1页 显示1到12,共12记录				50 1 共1页 显示1到30,共30记录				

按订单总数与已配套数分析				按配套可生产款数及色组分析					制衣厂	齐套款数	齐套总件数			
供应商	订单总数	可配套总数	占比	款号	订单总数	可配套总数	占比	制衣厂						
佛山市三水昊通印染有限公司	20128	20758	103.1%	551930103	3312	3419	100.0%	福建百隆	福建百隆	13	532854			
厦门科恩得服装材料有限公司	441951	0	0.0%	551930703	5848	6032	100.0%	福建百隆						
明协 (福建) 鞋服有限公司	0	0	0.0%	551933703	3340	3446	100.0%	福建百隆						
晋江澳讯贸易有限公司	14571	14720	101.0%	551933706	14571	14720	100.0%	福建百隆						
泉州市云隆纺织贸易有限公司	110466	11230	10.2%	551934002A	72174	62622	90.0%	福建百隆						
泉州振兴纺织有限公司	757739	0	0.0%	551934002B	70213	66058	90.0%	福建百隆						
石狮市谦成纺织贸易有限公司	18459	0	0.0%	551934008B	140885	136447	100.0%	福建百隆						
石狮禾宝纺织有限公司	728859	472682	64.9%	551934014A	116774	107841	90.0%	福建百隆						
石狮家宝染织有限公司	0	0	0.0%	551934035B	9430	38952	410.0%	福建百隆						
合计	2171313	532854	24.5%	551934860	35718	7811	20.0%	福建百隆						
50 1 共1页 显示1到12,共12记录				50 1 共1页 显示1到14,共14记录										

智源服装APS系统 生产监控										总管理员	
首页 成本报价 面辅料进度表 辅料汇总表											
361 19Q3											
按供应商采购数分析					按供应商面种分析						
供应商	采购总数	入库总数	占比	供应商	物料名称	采购总数	入库总数	占比			
上海沅磊织带有限公司	344329	344329	100.0%	上海沅磊织带有限公司 - 1							
东莞华利兴绣花织唛有限公司	206938	203548	98.4%	上海沅磊织带有限公司	37#松紧带	344329	344329	100.0%			
东莞市丰彩纽扣有限公司	3445	3445	100.0%	东莞华利兴绣花织唛有限公司 - 2							
东莞市永能行烫画服饰辅料有限公司	17348	17348	100.0%	东莞华利兴绣花织唛有限公司	19Q3主唛	203554	203548	100.0%			
东莞市沛和纺织配件有限公司	20708	20708	100.0%	东莞华利兴绣花织唛有限公司	尺码标	3384	0	0.0%			
东莞市环亚商标织造有限公司	122576	111292	90.8%	东莞市丰彩纽扣有限公司 - 1							
东莞市高飞服装辅料有限公司	1071959	1068353	99.7%	东莞市丰彩纽扣有限公司	树脂四眼扣	3445	3445	100.0%			
佛山市航宇针织有限公司	77904	77738	99.8%	东莞市永能行烫画服饰辅料有限公司 - 1							
卓盛源(福建)针织服饰有限公司	12559	12559	100.0%	东莞市永能行烫画服饰辅料有限公司	女上主唛	17348	17348	100.0%			
厦门凯美福工贸有限公司	34106	34106	100.0%	东莞市沛和纺织配件有限公司 - 1							
合计	8116813	8034128	99.0%	合计		8116813	8034128	99.0%			
50 第1页 共1页 显示1到24,共24记录					50 第1页 共2页 显示1到50,共55记录						

11、月计划：主计划排程并发布后，系统自动生成对应月份的每日计划产量，给生产部门提供参考，保证订单按预计下线时间完成。

智源服装APS系统 生产监控										总管理员										
首页 月计划 货源周计划 生产周计划 五楼生产日报																				
2019-06 全部组别 请输入款号 查询 上一月 下一月																				
季度	客户	组别	人数	款号	工价	单价	标准(元)	日产量		计划		06月01日(周六)			06月03日(周一)					
								标准(件)	人均(元)	总产量	总工资	总产值	日标上幅	产量	工资	产值	日标上幅	产量	工资	产值
1	19Q3	361	206	22	551939604	12.284	85.87	120	215	9.77	5429	66712	466194	1.01	217	2666	18634	1.01	217	2666
2	19Q3	361	203	15	551930703	6.031	68.62	120	299	19.9	1444	8712	99088	1.18	352	2124	24154	1.21	361	2178
3	19Q3	361	301	13	551934002A	8.07	77.23	120	193	14.87	1393	11249	107583	1.03	199	1607	15369	1.03	199	1607
4	19Q3	361	204	11	551934002B	5.21	54.42	120	253	23.03	1305	6800	71020	1.03	261	1360	14204	1.03	261	1360
5	19Q3	361	307	16	551934008B	4.824	50.28	120	398	24.88	6560	31632	329840	1.03	410	1977	20615	1.03	410	1977
6	19Q3	361	404	11	551934008B	4.824	50.28	120	274	24.88	3653	17667	183677	1.03	281	1359	14129	1.03	281	1359
7	19Q3	361	401	14	551934035B	4.218	59.73	120	398	28.45	10222	43131	610552	1.02	406	1713	24250	1.02	406	1713
8	19Q3	361	507	17	551934860	7.691	74.7	120	265	15.6	6686	51512	499449	1.02	270	2080	20169	1.02	270	2080
9	19Q3	361	508	17	551934860	7.691	74.7	120	265	15.6	6686	51512	499449	1.02	270	2080	20169	1	265	2039
10	19Q3	361	406	12	551939032A	5.832	76.64	120	247	20.58	2540	14830	194670	1.03	254	1483	19467	1.03	254	1483
50 第1页 共1页 显示1到34,共34记录																				

12、货源周计划：为裁剪部门提供裁剪指令，保证按时按量完成裁剪计划内容。并对印绣花外发回货进行管理，同时计算款号、尺码对应的印花可配套数量，保证印绣花交期及品质匹配成衣制造生产。

智源服装APS系统 生产监控

月计划 货源周计划 生产周计划 五楼生产日报

同步印花回货 + 新增 删除 导入 导出 导出模板 361 19Q4 筛选 清空筛选 裁剪指令单

同步	款号	品名	组别	颜色	尺码	下货日期	床次	件数	裁剪日期	裁剪数	发出日	发出数	回货日	回货数	齐套数	回货入库
1	551944354	超轻羽绒服	505	基础黑B19	3XL	2019-06-29	21	800	2019-05-31	800	06-01	800	06-08	800	800	06-12
2	551944354	超轻羽绒服	505	基础黑B19	3XL	2019-07-02	22	744	2019-05-31	744	06-01	744	06-08	744	744	06-08
3	551944354	超轻羽绒服	505	基础黑B19	XL	2019-07-05	11	800	2019-05-31	800	06-03	800	06-11	800	800	06-11
4	551944354	超轻羽绒服	505	基础黑B19	XL	2019-07-08	12	800	2019-05-31	800	06-03	800	06-12	800	800	06-12
5	551944354	超轻羽绒服	505	基础黑B19	XL	2019-07-11	13	800	2019-05-31	800	06-03	800	06-12	800	800	06-12
6	551944354	超轻羽绒服	505	基础黑B19	XL	2019-07-13	14	836	2019-06-01	836	06-03	836	06-12	836	836	06-12
7	551944354	超轻羽绒服	207	基础黑B19	L	2019-07-12	10	620	2019-06-03	620	06-06	620	06-18	620	620	06-18
8	551944354	超轻羽绒服	207	基础黑B19	L	2019-07-03	7	800	2019-06-03	800	06-06	800	06-17	800	800	06-17
9	551944354	超轻羽绒服	207	基础黑B19	L	2019-07-06	8	800	2019-06-03	800	06-06	800	06-17	800	800	06-17
10	551944354	超轻羽绒服	207	基础黑B19	L	2019-07-09	9	800	2019-06-03	800	06-06	800	06-17	800	800	06-17

50 第1 共2页 显示1到50,共66记录

13、开发进度表：管控成衣批量生产前的款式开发，可进行一选、二选、上会三个层面的开发进度管控并统计开发完成结果。最终小批量生产并由客户确认款式后进行大批量生产。

智源服装APS系统 生产监控

开发进度表 大货进度表 订单装箱

+ 新增 删除 导入 导出 导出模板 锁定 361 进度 20Q2上会 筛选 清空筛选 显示列 设置

版师	排版师	款号	工艺员	品名	进度	颜色	做版形样	纸样OK	版形样批复	整款纸样	可裁剪日期	裁剪日期	印花发出	印花收回
1	1	11	55202CT007	圆领短T	20Q2上会	新柔2	烟花灰E21	✓	✓	✓	07-08	07-10		
2					20Q3一选									
3					20Q2二选		粉色X01							
4					19Q3一选		京绿OC5							
5							兰紫U42							

1款制作 完成1款 总数2款 可裁1款 已裁1款

50 第1 共1页 显示1到9,共9记录

版师	开发款数	已完成款	未完成款	印花厂	开发款数	未完成款
1	1	1	0	七彩虹	1	0
				年发	2	1

14、大货进度表：技术部门成衣生产进度管控，确认裁剪、印花收回、样衣完成情况及数量。

智源服装APS系统 生产监控

开发进度表 大货进度表 订单装箱

款号	品名	订单数	交期	颜色	版师	排版师	封样裁剪	印绣花外发	印绣花收回	印花周期	印绣花厂	样衣工	样衣完成	完成数量	
15				灰粉	尤	叶	2019-07-08	2019-07-08							
16				向日葵黄			2019-06-12	2019-06-12	2019-06-20	-8					
17	661939811	连帽套头卫衣	5500	07-15	大学红		2019-06-24	2019-06-24	2019-06-28	-4					
18				基础黑			2019-06-12	2019-06-12	2019-07-01	-19					
19				米白			2019-06-12	2019-06-12	2019-06-20	-8					
20				灰粉	尤	叶	2019-07-11	2019-07-11	2019-07-16	-5					
21	661939814	连帽套头卫衣	4505	07-15	米白	尤	叶	2019-07-06	2019-07-06						
22				大学红	尤	叶	2019-06-10	2019-06-10	2019-06-19	-9					
23	651939701	针织长裤	18500	08-10	基础黑	潘	王	2019-06-09	2019-06-09	2019-06-13	-4				
24				秋季色	乔	叶	2019-06-22	2019-06-22							
总款:62款		OK:1款													

15、产前进度：从成衣生产的筹备阶段、生产阶段对产前各个环节进行严格把控。通过计划上线时间推算之后筹备和生产中每个重要环节的计划完成时间，并由各个环节对应负责人填写相应环节的实际完成时间，通过计划和实际的对比，严格把控每个环节的完成度，延迟则给出及时的预警，确保生产按计划完成。

智源服装APS系统 生产监控

产前进度 生产主计划 月计划 货源计划 产前样追踪 产前样打样 面料外观检验 材料货期汇总 转款进度追踪

款号	品名	订单量 交期 颜色	实际交期	小组计划 上线时间	封样面辅料要求齐套 (提前20天)		封样辅料齐套 >5警示		封样裁剪 >2警示		封样印绣花压胶发出 >1警示		印绣花压胶批样 >5警示		印绣花压胶检 >7警示	
					计划	实际	计划	实际	计划	实际	计划	实际	计划	实际		
					颜色标识: 延迟 (红色), 超计划未填写 (黄色)											
43		订单量: 1,259 交期: 2019-09-15 颜色: 暗夜蓝杂色														
44		订单量: 1,164 交期: 2019-09-15 颜色: 黎瀛绿杂色		06-13	05-24	07-29	08-03	07-31	07-31							
45	551944014A	套头卫衣 订单量: 8,662 交期: 2019-09-15 颜色: 热焦红		06-13	05-24											
46		订单量: 13,476 交期: 2019-09-15 颜色: 基础黑		06-13	05-24											

16、生产日报：获取企业车间的吊挂系统数据，生成每日的生产报表，算出每日实际产量、出入库数量、累计完成产量等。通过直观的数据展示每日的实际完成情况与计划排程之间的比较。并将实际产量回写到计划排程中，重新计算预计下线时间，确保预计下线时间的可参考性。

智源服装APS系统 生产监控 总管理员

生产日报 生产进度 委外生产进度 生产周计划 返修率 验片 印绣花坏片 供应商良率 工位流量平衡

同步 导出 361 19Q4 2019-07-10 筛选 清空筛选

款号	品名	上线日期	开款 天数	订单 分配数	下货 组别	累计 领料数	出勤 人数	当月累 计人数	工价	实际		当日平 均工资	总检 良品数	月计划当日			当月累					
										产量	产值			日产量 工资	计划 产量	完成率	计划 产量	计划 工资	计划 产值	实际 产量	实际 工	
1	551944014A	套头卫衣	2019-06-17	21	26010	302	17579	12	115	2.692	445	23727.4	1197.81	99.82	0	949	47%	8259	17508	0	7914	2
2	551944203	棉夹克	2019-06-29	10	26949	201	2200	26	271	19.097	218	25466.7	4163.08	160.12	210	123	177%	918	15384	0	1074	2
3			2019-06-18	21	1156	307	1156	26	271	19.097	58	6775.56	1107.61	42.61	0	123	47%	918	15384	0	719	1
4			2019-06-16	14	15000	506	3882	18	18	15.84	205	23948.1	3247.2	180.4	0	123	167%	123	2067	0	205	3
5	551944354	超轻羽绒服	2019-06-11	18	11789	505	3462	13	13	12.238	145	13592.3	1774.51	136.51	0	142	102%	142	1710	0	145	1
6	551944602	绒里夹克	2019-07-09	2	9729	203	400	24	43	10.932	55	4381.3	601.28	25.06	0	246	22%	485	4389	0	55	6
7			2019-06-27	14	13800	305	2400	24	152	10.932	107	8523.62	1169.76	48.74	0	246	43%	1830	16569	0	1025	1
8	551946351		2019-07-10	1	0	407	0	9	9	14.809	0	0	0	0	0	0%	0	0	0	0	0	
9	551949354	短羽绒服	2019-07-03	7	14800	301	1760	23	163	14.146	94	9608.68	1329.72	57.82	0	153	61%	845	10238	0	323	4

50 1 共1页 显示1到19,共19记录

17、生产进度：通过图表一目了然的展示了每个季度对应款号的生产完成情况。



18、人均工价：通过获取吊挂系统每日生产实时数据，计算出每个款号对应生产组别的日人均工资和生产区域人均工资，并算出人均日工资同比，达到成本核算和对生产线上员工薪资平衡监控，提高员工生产积极性。



19、设备用量表：对生产区域的设备使用进行监管和统计。

季度	客户	款号	款式	组别	生产人数	预计上线	平车	四线	四线上领	五线	刀车拉筒	刀车	双杆双车拉筒	单杆双针	双杆双针	带刀双针	打枣车	
16	19Q4	361	561943702	绒里长裤	401	14	06-29											
17	19Q4	361	561944015B	针织长裤	309	10	07-03											
18	19Q4	361	561949201	中棉服	509	23	06-29											
19	19Q4	361	561949307	长羽绒服	503	20	06-05			5		3					1	
20	19Q4	361	561949307	长羽绒服	501	17	06-05			5		3					1	
21	19Q4	361	561949307	长羽绒服	210	20	06-05			5		3					1	
22	19Q4	361	561949309	中羽绒服	208	18	05-31											
现有设备							1042	170.0	31.0	147.0	27.0	90.0	63.0	41.0	63.0	0.0	46.0	
本季所需设备							0	0	0	15.0	0	9.0	0	0	0	0	0	3.0
本季设备欠缺							1042	170.0	31.0	132.0	27.0	81.0	63.0	41.0	63.0	0.0	43.0	

20、转款准备：监控生产转款前生产部门、生产管理部、技术 IE 部和品管部需提前准备的事项，对应负责人填写每个环节是否完成或超期，如果超期就给与警示，确保能够顺利的上线生产。

智源服装APS系统 生产监控

首页 转款准备

工价审批进度 季度工价审批 人均工价 设备用量表 转款准备 预估工价

2019-06 请输入款号 查询 上一月 下一月 转款人员配置

客户	组别	款号/品名	达成	上线负责人	计划	实际	异常	生产部									
								计划员		主任		组长		汪泽华		汪泽华	
								10天	7天	2天	7天	6天	5天	4天	3天	3天	3天
1	361	401 款号: 551933703 品名: 休闲长裤	是		2019-06-01	2019-06-01		NO2	✓	NO	NO	✓	✓	✓	✓		
2	361	309 款号: 551939711 品名: 休闲九分裤	否		2019-06-04	2019-06-04											
3	361	210 款号: 561949307 品名: 长羽绒服			2019-06-05	2019-05-29		✓	✓	✓	NO	NO2					

50 第1页 共1页 显示1到18,共18记录

转款汇总表

日期	计划上线	按时转款	延迟转款	延迟原因
06-01-06-09	9	1	1	
06-10-06-16	3	0	0	

21、班组生产完成率：记录每个组别领取对应款号的领料数和生产完成数

智源服装APS系统 生产监控

首页 成本报价 班组生产完成率

成本报价 班组生产完成率 裁剪入库完成率 合同出货完成率 印花回收

请输入款号 361 19Q4 查询 筛选 清空筛选

组别	款号	颜色	尺码	领裁数量	入库数量	未入库数	领裁入库率	
1	201A	551944203	电网绿/基础黑Q99	2XL	1000	0	1000	0.0%
2	506	551944203	电网绿/基础黑Q99	2XL	800	0	800	0.0%
3	201A	551944203	电网绿/基础黑Q99	XL	1600	0	1600	0.0%
4	506	551944203	电网绿/基础黑Q99	XL	1600	0	1600	0.0%
5	207A	551944354	基础黑B19	L	2400	0	2400	0.0%
6	305	551944602	基础黑B19	M	800	0	800	0.0%
7	310	551944705	基础黑B19	3XL	279	0	279	0.0%
8	310	551944705	基础黑B19	4XL	480	0	480	0.0%
9	310	551944705	基础黑B19	L	822	0	822	0.0%
10	310	551944705	基础黑B19	M	156	0	156	0.0%
合计					44275	0	44275	0

50 第1页 共1页 显示1到47,共47记录

22、裁剪入库完成率：获取对应款号的裁剪数量和各个班组的领裁数量，并计算裁剪完成率，未裁数量。

智源服装APS系统 生产监控									
成本报价 班组生产完成率 裁剪入库完成率 合同出货完成率 印花回收									
请输入款号 361 19Q4 查询 筛选 清空筛选									
款号	颜色	尺码	技术下单	裁剪	班组领裁	未裁	裁剪完成率	未领件数	
1	551944014A	基础黑	2XL	2037	2058	2058	-21	101.0%	
2	551944014A	基础黑	3XL	1092	1104	1104	-12	101.1%	
3	551944014A	基础黑	4XL	398	404	404	-6	101.5%	
4	551944014A	基础黑	L	3432	4340	4340	-908	126.5%	
5	551944014A	基础黑	M	2569	2598	2490	-29	101.1%	108
6	551944014A	基础黑	S	904	918	456	-14	101.5%	462
7	551944014A	基础黑	XL	2963	4638	4539	-1675	156.5%	99
8	551944014A	基础黑	XS	81	82		-1	101.2%	82
9	551944014A	热焦红	2XL	1330	1362	420	-32	102.4%	942
10	551944014A	热焦红	3XL	685	702		-17	102.5%	
11	551944014A	热焦红	4XL	186	190		-4	102.2%	

23、产前样追踪、打样维护：给品管部提供产前样的检查核对，杜绝批量异常产生，确保大货生产时产品质量符合客户要求。

智源服装APS系统 生产监控														
返修明细 验片统计 产前样追踪维护 产前样打样维护 面料外观检验维护														
+ 新增 删除 导入 导出 导出模板 请输入款号 2019-07-01 到 2019-07-31 查询 筛选 清空筛选 上一月 显示列 设置														
外发日期	款号	分类	颜色	部位	件数	供应商	负责人	收回日期	收回耽误天数	批样日期	批样耽误天数	批样备注	检测结果	检测日期
2019-07-01	551944014A	产前样	热焦红	左前	3	泉州市博绅仁中服依	小周	2019-06-20	-15	2019-06-17	-18	✓	✓	2019-
2019-07-15	551944014A	产前样	黎藻绿杂色	前片左右袖	3	泉州市博绅仁中服依	小周	2019-07-17	-2	2019-07-17	-2	✓		
2019-07-01	551944014A	产前样	热焦红	左前	3	泉州市博绅仁中服依	小周	2019-06-19	-16	2019-06-17	-18	✓	✓	2019-
2019-07-15	551944014A	产前样	黎藻绿杂色	前片左右袖	3	泉州市博绅仁中服依	小周	2019-07-17	-2	2019-07-17	-2	✓		
2019-07-08	551944602	产前样	电网绿	左前胸	3	石狮市七彩红植绒印	司机	2019-07-15	3	2019-07-15	3	✓		
2019-07-08	551944602	产前样	电网绿	左前胸	3	石狮市七彩红植绒印	司机	2019-07-15	3	2019-07-15	3	✓		
2019-07-05	551949355	产前样	深邃蓝	右前胸	3	石狮市七彩红植绒印	小苏	2019-07-07	-2	2019-05-05	-65	✓		
2019-07-05	551949355	产前样	深邃蓝	右前胸	3	石狮市七彩红植绒印	小苏	2019-07-07	-2	2019-05-05	-65	✓		
2019-07-05	551949356	产前样	深邃蓝	右前胸	3	石狮市七彩红植绒印	小苏	2019-07-09	0	2019-05-05	-65	✓		
合计														
					126				-1		-820			



24、面料外观维护：品管部门通过比对标准样本及相关资料、核对用料、做法是否正确，检查外观及功能是否达到客户要求。



25、供应商管理：对供应商进行管理维护。

全部	同步	筛选	清空筛选							
全部	编码	名称	简称	分类	停用日期	供应商(工票)				
面料	1	010001	佳福(福建)染整有限公司	佳福(福建)染整有限公司	面料		佳福(福建)染整有限公司			
辅料	2	0100019	常熟市金秀自动化设备销售有限公司	常熟市金秀自动化设备销售有限公司	投资类					
咨询类	3	010002	苏州优一纺织有限公司	苏州优一纺织有限公司	面料		苏州优一纺织有限公司			
其他类	4	010003	吴江市恒信织造有限责任公司	吴江市恒信织造有限责任公司	面料		吴江市恒信织造有限责任公司			
包装物	5	010004	江苏东亮纺织科技有限公司	江苏东亮纺织科技有限公司	面料					
外加工	6	010005	石狮维吉尔纺织品贸易有限公司	石狮维吉尔纺织品贸易有限公司	面料					
印绣花	7	010006	厦门伊特纺织品有限公司	厦门伊特纺织品有限公司	面料					
用品仓	8	010007	山东大鲁绸织染工业有限公司	山东大鲁绸织染工业有限公司	面料					
投资类	9	010008	江苏东防纺织科技有限公司	江苏东防纺织科技有限公司	面料					
客供	10	010009	苏州志向纺织科研股份有限公司	苏州志向纺织科研股份有限公司	面料					
山东订单供应商	11	010010	上海华晨纺织品有限公司	上海华晨纺织品有限公司	面料					
裁片	显示1到50,共6961记录									
成品	50 1 共14页									

26、手动同步：对系统的所有定时任务进行管理，并记录定时任务运行时是否出现异常情况，如有异常则手动执行相应的任务。

任务	触发方式	触发说明	同步时间	操作	操作时间	方式	状态	详情	备注
1 用友仓库	自动	每天零点自动执行	2019-07-24 04:30:02	同步	2019-07-24 04:30:02	自动	成功		同步用友仓库用于成品入库选择仓库
2 工价审批进度	自动	每天零点自动执行	2019-07-26 03:30:02	同步	2019-07-26 03:30:02	自动	成功		工价审批进度同步OA工价表
3 OA用户	自动	每天零点自动执行	2019-07-26 03:30:00	同步	2019-07-26 03:30:00	自动	成功		同步OA用户, 关联审批进度人员
4 同步生产日报	自动	每天零点自动执行	2019-07-26 01:00:29	同步	2019-07-26 01:00:29	自动	成功		同步吊挂每日生产数据
5 印绣花回货	自动	每天零点自动执行	2019-07-26 05:30:18	同步	2019-07-26 05:30:18	自动	成功		生产周计划同步印绣花发出日期, 发出数量, 回货日期, 回货数量
6 bom	自动	每天零点自动执行	2019-07-26 03:30:05	同步	2019-07-26 03:30:05	自动	成功		同步bom单有色无码, 有色有码, 无色有码的数据
7 同步人资	手动		2019-06-03 16:06:01	同步	2019-06-03 16:06:01	手动	成功		同步人资部人资的具体信息
8 成品入库	手动		2019-06-24 09:33:53	同步	2019-06-24 09:33:53	手动	成功		出货进度的成品入库同步用友系统的成品入库
9 工票系统班组领裁	自动	每天零点自动执行	2019-07-26 19:40:15	同步	2019-07-26 19:40:15	手动	成功		同步工票系统班组领裁情况
10 工票系统领裁	自动	每天零点自动执行	2019-07-26 19:40:15	同步	2019-07-26 19:40:15	手动	成功		同步工票系统领裁的情况

27、用户管理：系统用户创建和权限管理。

智源服装APS系统 生产监控 总管理员												
首页 用户管理 产前进度 生产日报 报价分类 对接配置 产前进度权限设置												
+ 新增 删除 请输入账号 请选择 请输入员工编号 请输入员工姓名 请输入部门 请选择 查询												
ID	图标	帐号	角色	姓名	手机	账号状态	员工编号	员工名称	部门	状态	创建时间	修改时间
1		唐新皓	生产部文员	唐新皓		正常					2019-07-11	2019-07-11
2		QC005	印花花主管	印花花主管		正常					2019-07-01	2019-07-01
3		QC004	品管部文员	品管部文员		正常					2019-07-01	2019-07-01
4		QC003	品管部原材料统计	品管部原材料		正常					2019-07-01	2019-07-01
5		QC002	品管部品质主管	品管部品质主		正常					2019-07-01	2019-07-01
6		QC001	品管部副经理	品管部副经理		正常					2019-07-01	2019-07-01
7		SG008	业务文员	生产管理业务		正常					2019-06-29	2019-06-29
8		SG007	辅料采购	辅料采购		正常					2019-06-29	2019-06-29
9		SG006	面料采购	面料采购		正常					2019-06-29	2019-06-29
显示1到50,共57记录												

28、产前进度权限和技术部权限设置：产前进度和开发进度表可单独的进行权限管控，每个环节都可指定专人负责，避免相互推诿。

智源服装APS系统 生产监控 总管理员												
首页 用户管理 产前进度 生产日报 报价分类 对接配置 产前进度权限设置 字典设置												
筛选 清空筛选 选择用户 删除 筛选 清空筛选												
ID	部门	准备事项	用户	角色	姓名	手机						
1	计划部	实际交期	1 蒋小建	技术服务中心总	蒋小建							
2	计划部	封样面辅料要求齐套	2 张振农	软件开发	张振农							
3	计划部	裁剪	3 蒋小建	技术服务中心总	蒋小建							
4	计划部	车缝上线	4 张振农	软件开发	张振农							
5	计划部	车缝成品	5 张振农	软件开发	张振农							
6	计划部	车缝入仓	6 蒋小建	技术服务中心总	蒋小建							
7	计划部	出货数	7 蒋小建	技术服务中心总	蒋小建							
8	计划部	出货完成	8 张振农	软件开发	张振农							
9	计划部	延期天数	9 蒋小建	技术服务中心总	蒋小建							
10	采购部	封样面料齐套	10 张振农	软件开发	张振农							
11	采购部	封样辅料齐套	11 蒋小建	技术服务中心总	蒋小建							
显示1到44,共44记录												

智源服装APS系统 生产监控

用户管理 角色管理 字典设置 对接配置 产前进度权限设置 技术部权限设置 功能设置

开发进度 大货进度

筛选 清空筛选 选择用户 删除 筛选 清空筛选

填写内容	用户	角色	姓名	手机
1 版师填写	1 严婷	技术部员工	严婷	
2 排版师填写	2 程明松	技术部员工	程明松	
3 工艺员填写	3 庄小霞	技术部员工	庄小霞	
4 印花厂填写	4 刘玉平	技术部员工	刘玉平	
5 制作完成填写	5 潘佩华	技术部经理	潘佩华	
6 完成日期	6 严婷	技术部员工	严婷	
7 是否做版形样填写	7 程明松	技术部员工	程明松	
8 纸样填写	8 庄小霞	技术部员工	庄小霞	
9 可裁剪日期	9 刘玉平	技术部员工	刘玉平	
10 裁剪日期填写				

50 第 1 共 3 页 显示 1 到 50, 共 121 记录

29、成品出入库：通过 EXCEL 导入成品出入库数据，并生成季度成品出入库详情表，展示出入库详情。

智源服装APS系统 生产监控

主计划排程 委外生产 月出货计划 生产日历表 标准日工资 BCM成品入库 BCM成品出库

成品入库 ① 导入 ② 导出 ③ 导出模板 匹配款号 361 19Q3 筛选 清空筛选

状态	箱码	数量	组别	U8货号	款号	品名	颜色	尺码	制单日期	拆箱	同步情况
1 未同步	FBFL119061700001	30	107	551930102-2	551930102	圆领短T恤	基础黑	3XL	2019-06-17	否	未检索到匹配的[部门]。
2 未同步	FBFL119061700001	30	107	551930102-2	551930102	圆领短T恤	基础黑	S	2019-06-17	否	未检索到匹配的[部门]。
3 未同步	FBFL119061700002	40	107	551930102-2	551930102	圆领短T恤	基础黑	2XL	2019-06-17	否	未检索到匹配的[部门]。
4 未同步	FBFL119061700002	15	107	551930102-2	551930102	圆领短T恤	基础黑	4XL	2019-06-17	否	未检索到匹配的[部门]。
5 未同步	FBFL119061700003	55	107	551930102-2	551930102	圆领短T恤	基础黑	XL	2019-06-17	否	未检索到匹配的[部门]。
6 未同步	FBFL119061700004	40	107	551930102-1	551930102	圆领短T恤	本白	2XL	2019-06-17	否	未检索到匹配的[部门]。
7 未同步	FBFL119061700005	30	107	551930102-3	551930102	圆领短T恤	浅驼色	S	2019-06-17	否	未检索到匹配的[部门]。
8 未同步	FBFL119061700005	30	107	551930102-3	551930102	圆领短T恤	浅驼色	3XL	2019-06-17	否	未检索到匹配的[部门]。
9 未同步	FBFL119061700006	50	107	551930102-3	551930102	圆领短T恤	浅驼色	M	2019-06-17	否	未检索到匹配的[部门]。
10 未同步	FBFL1190617000089	40	107	551930102-3	551930102	圆领短T恤	浅驼色	2XL	2019-06-17	否	未检索到匹配的[部门]。
11 未同步	FBFL1190617000103	10	107	551930102-3	551930102	圆领短T恤	浅驼色	XS	2019-06-17	否	未检索到匹配的[部门]。

50 第 1 共 233 页 显示 1 到 50, 共 11637 记录

智源服装APS系统 生产监控 总管理员

主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程

主计划排程 委外生产 月出货计划 生产日历表 标准日工资 BCM成品入库 BCM成品出库

① 导入 ② 导出 ③ 导出模板 ④ 匹配款号 361 19Q3 筛选 清空筛选

	出货单号	制单日期	U8货号	款号	品名	颜色编码	颜色	尺码	数量	备注
1	CH20190618000034	2019-06-18	551930703-1	551930703	休闲九分裤	B19000	基础黑	L	1	
2	CH20190618000034	2019-06-18	551930703-1	551930703	休闲九分裤	B19000	基础黑	M	1	
3	CH20190618000034	2019-06-18	551930703-2	551930703	休闲九分裤	X10000	苍灰	L	1	
4	CH20190618000034	2019-06-18	551930703-2	551930703	休闲九分裤	X10000	苍灰	M	1	
5	CH20190617000046	2019-06-17	551930601-3	551930601	单风衣	B51000	本白	XL	85	
6	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	2XL	157	
7	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	3XL	17	
8	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	4XL	1	
9	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	L	333	
10	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	M	234	
11	CH20190616000006	2019-06-16	551930103-1	551930103	套头卫衣	B51000	本白	S	143	

50 1 1 共5页 显示1到50,共233记录

智源服装APS系统 生产监控 总管理员

主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程 主计划排程

主计划排程 委外生产 月出货计划 生产日历表 标准日工资 BCM成品入库 BCM成品出库 生产日报 月出货计划 季度成品跟踪进度表 季度成品入库日报 季度成品出库日报

① 导出 361 19Q3 筛选 清空筛选 显示列 设置

	入库日期	款号	品名	类别	组别	颜色	XXXS	XXS	XS	S	M	L	XL	2XL	3XL	4XL	5XL	6XL	合计
1	2019-06-17	551934723	运动长裤	梭织	000	基础黑		0	0	10	0	6	33	4	0	0	0	0	53
2	2019-06-17	551930102	圆领短T恤	针织	107	本白		0	10	30	49	69	55	40	30	15	0	0	298
3	2019-06-17	551930102	圆领短T恤	针织	107	基础黑		0	10	30	100	70	55	40	30	15	0	0	350
4	2019-06-17	551930102	圆领短T恤	针织	107	浅驼色		0	10	30	50	70	55	40	30	15	0	0	300
5	2019-06-17	551934705	运动长裤	针织	203	基础黑		0	0	9	6	8	65	34	17	6	0	0	145
6	2019-06-17	551934705	运动长裤	针织	203	深墨蓝		0	0	1	5	6	13	5	5	0	0	0	35
7	2019-06-17	551934002B	针织长裤	针织	204	基础黑		0	0	0	0	0	0	24	25	25	0	0	74
8	2019-06-17	551934002B	针织长裤	针织	204	黑曜石杂色		0	0	0	30	0	55	0	25	0	0	0	110
9	2019-06-17	551934703	运动九分裤	针织	308	基础黑		0	0	120	60	180	60	180	60	60	0	0	720
10	2019-06-17	551934723	运动长裤	梭织	504	基础黑		0	0	0	0	0	18	0	0	0	0	0	18
11																			

智源服装APS系统 生产监控

总管理员

首页 主计划排程 成品入库 成品出库 生产日报 月出货计划 季度成品跟踪进度表 季度成品入库日报 季度成品出库日报

月出货计划 季度成品跟踪进度表 季度成品出入库明细 季度成品入库日报 季度成品出库日报

导出 361 19Q3 筛选 清空筛选 显示列 设置 装箱单打印

日期	款号	品名	颜色	XXXS	XXS	XS	S	M	L	XL	2XL	3XL	4XL	5XL	6XL	出库数
2019-06-18	551930703	休闲九分裤	基础黑		0	0	0	1	1	0	0	0	0	0	0	2
2019-06-18	551930703	休闲九分裤	苍灰		0	0	0	1	1	0	0	0	0	0	0	2
2019-06-17	551930601	单风衣	本白		0	0	0	0	0	85	0	0	0	0	0	85
2019-06-16	551930103	套头卫衣	本白		0	0	143	234	333	253	157	17	1	0	0	1138
2019-06-16	551930103	套头卫衣	基础黑		0	2	153	437	560	427	214	98	14	0	0	1905
2019-06-14	551934703	运动九分裤	基础黑		0	0	60	120	360	360	660	300	0	0	0	1860
2019-06-13	551930601	单风衣	基础黑		0	0	0	1	1	0	0	0	0	0	0	2
2019-06-13	551933706	运动长裤	基础黑		0	0	0	1	1	0	0	0	0	0	0	2
2019-06-13	551934119	圆领短T恤	古董绿		0	0	16	0	0	0	0	0	9	0	0	25
2019-06-04	551934703	运动九分裤	基础黑		0	0	1260	3900	4500	3420	2100	1080	420	0	0	16680
2019-05-27	551934119	圆领短T恤	基础黑		0	4	81	225	304	258	147	69	20	0	0	1108

显示1到29,共29记录

30、季度成品跟踪进度表：展示对应季度成衣出入库汇总，并对是否会超期出货进行预警。

智源服装APS系统 生产监控

总管理员

首页 主计划排程 成品入库 成品出库 生产日报 月出货计划 季度成品跟踪进度表

月出货计划 季度成品跟踪进度表 季度成品出入库明细 季度成品入库日报 季度成品出库日报

导出 361 19Q3 筛选 清空筛选 显示列 设置

针织/梭织	大货/外单	款号	订单数	车间入库数	车间入库欠数	出货数	未出货数	仓库库存	入库率	出库率	货期	距出货日期	最后出货日期	最后出货数量	
1	针织	外单	551934119	28933	34699	-5766	28903	30	5796	119.9%	99.9%	2019-05-15	超货期29天	2019-06-13	29
2	梭织	大货	551934719	37531	39223	-1692	0	37531	39223	104.5%		2019-05-15	超货期34天		0
3	针织	大货	551939113	28814	0	28814	0	28814	0			2019-05-15	超货期34天		0
4	针织	外单	551939102	21127	0	21127	0	21127	0			2019-05-15	超货期34天		0
5	针织	外单	551939106	4215	0	4215	0	4215	0			2019-05-15	超货期34天		0
6	针织	大货	551939109	23427	0	23427	0	23427	0			2019-05-15	超货期34天		0
7	梭织	外单	561934720	23845	0	23845	0	23845	0			2019-05-15	超货期34天		0
8	针织	大货	561934708	5169	0	5169	0	5169	0			2019-05-15	超货期34天		0
9	针织	大货	561934113	10826	0	10826	0	10826	0			2019-05-15	超货期34天		0
10	梭织	大货	551939707	6982	0	6982	0	6982	0			2019-05-15	超货期34天		0
11	梭织	大货	551934704	14844	15882	-1038	14887	-43	995	107.0%	100.3%	2019-05-15	超货期6天	2019-05-21	180

显示1到50,共50记录

手机端说明：

1、产前进度：展示当前款号的进度和配套数以及是否有异常状况发生，有则及时处理避免生产延期。

筛选

👤 客户: **361** 📅 季度: **19Q4**

👕 款号 **651844704** 深花灰 针织长裤

📶 进度 **车缝成品**
异常: 无 责任部门: 无

订单数 1010	配套数 0
---	--

👕 款号 **651844704** 基础黑 针织长裤

📶 进度 异常: 无 责任部门: 无

订单数 2005	配套数 0
---	--

👕 款号 **661839811** 粉泥红 套头卫衣
📶 进度 **封样面辅料要求齐套**
异常: 无 责任部门: 无
🏠

2、成本报价：不分类别统计企业和客户单价差；分印花、原辅材量、工价和管理四个类别对企业和客户的料率和单价统计，重点突出价格差值，并用不同颜色显示。

<input style="width: 90%;" type="text" value="Q 款号"/> 筛选	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: right;">款 号</td> <td colspan="3" style="text-align: right; font-weight: bold;">551940314</td> </tr> <tr> <td style="text-align: right;">类别</td> <td style="text-align: center;">印绣花</td> <td style="text-align: right;">单价(百锋)</td> <td style="text-align: right;">13.2</td> </tr> <tr> <td style="text-align: right;">料率(百锋)</td> <td style="text-align: center;">8.0</td> <td style="text-align: right;">单价(361)</td> <td style="text-align: right;">13.2</td> </tr> <tr> <td style="text-align: right;">料率(361)</td> <td style="text-align: center;">8.0</td> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	款 号	551940314			类别	印绣花	单价(百锋)	13.2	料率(百锋)	8.0	单价(361)	13.2	料率(361)	8.0	单 价 差	0.0						
款 号	551940314																						
类别	印绣花	单价(百锋)	13.2																				
料率(百锋)	8.0	单价(361)	13.2																				
料率(361)	8.0	单 价 差	0.0																				
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; text-align: right;">客户: 361</td> <td style="text-align: right;">季度: 19Q4</td> </tr> <tr> <td style="text-align: right;">款 号</td> <td style="text-align: right; font-weight: bold;">551940314</td> </tr> <tr> <td style="text-align: right;">百锋单价</td> <td style="text-align: right;">91.3</td> </tr> <tr> <td style="text-align: right;">361单价</td> <td style="text-align: right;">91.3</td> </tr> <tr> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	客户: 361	季度: 19Q4	款 号	551940314	百锋单价	91.3	361单价	91.3	单 价 差	0.0	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: right;">类 别</td> <td style="text-align: center;">原辅材料</td> <td style="text-align: right;">单价(百锋)</td> <td style="text-align: right;">55.2</td> </tr> <tr> <td style="text-align: right;">料率(百锋)</td> <td style="text-align: center;">37.4</td> <td style="text-align: right;">单价(361)</td> <td style="text-align: right;">55.2</td> </tr> <tr> <td style="text-align: right;">料率(361)</td> <td style="text-align: center;">37.4</td> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	类 别	原辅材料	单价(百锋)	55.2	料率(百锋)	37.4	单价(361)	55.2	料率(361)	37.4	单 价 差	0.0
客户: 361	季度: 19Q4																						
款 号	551940314																						
百锋单价	91.3																						
361单价	91.3																						
单 价 差	0.0																						
类 别	原辅材料	单价(百锋)	55.2																				
料率(百锋)	37.4	单价(361)	55.2																				
料率(361)	37.4	单 价 差	0.0																				
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; text-align: right;">款 号</td> <td style="text-align: right; font-weight: bold;">551940606</td> </tr> <tr> <td style="text-align: right;">百锋单价</td> <td style="text-align: right;">152.4</td> </tr> <tr> <td style="text-align: right;">361单价</td> <td style="text-align: right;">152.4</td> </tr> <tr> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	款 号	551940606	百锋单价	152.4	361单价	152.4	单 价 差	0.0	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: right;">类 别</td> <td style="text-align: center;">工价</td> <td style="text-align: right;">单价(百锋)</td> <td style="text-align: right;">21.1</td> </tr> <tr> <td style="text-align: right;">料率(百锋)</td> <td style="text-align: center;">1.0</td> <td style="text-align: right;">单价(361)</td> <td style="text-align: right;">21.1</td> </tr> <tr> <td style="text-align: right;">料率(361)</td> <td style="text-align: center;">1.0</td> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	类 别	工价	单价(百锋)	21.1	料率(百锋)	1.0	单价(361)	21.1	料率(361)	1.0	单 价 差	0.0		
款 号	551940606																						
百锋单价	152.4																						
361单价	152.4																						
单 价 差	0.0																						
类 别	工价	单价(百锋)	21.1																				
料率(百锋)	1.0	单价(361)	21.1																				
料率(361)	1.0	单 价 差	0.0																				
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; text-align: right;">款 号</td> <td style="text-align: right; font-weight: bold;">551943604</td> </tr> <tr> <td style="text-align: right;">百锋单价</td> <td style="text-align: right;">225.6</td> </tr> <tr> <td style="text-align: right;">361单价</td> <td style="text-align: right;">225.6</td> </tr> <tr> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	款 号	551943604	百锋单价	225.6	361单价	225.6	单 价 差	0.0	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: right;">类 别</td> <td style="text-align: center;">管理</td> <td style="text-align: right;">单价(百锋)</td> <td style="text-align: right;">1.8</td> </tr> <tr> <td style="text-align: right;">料率(百锋)</td> <td style="text-align: center;">2.0</td> <td style="text-align: right;">单价(361)</td> <td style="text-align: right;">1.8</td> </tr> <tr> <td style="text-align: right;">料率(361)</td> <td style="text-align: center;">2.0</td> <td style="text-align: right;">单 价 差</td> <td style="text-align: right; color: green;">0.0</td> </tr> </table>	类 别	管理	单价(百锋)	1.8	料率(百锋)	2.0	单价(361)	1.8	料率(361)	2.0	单 价 差	0.0		
款 号	551943604																						
百锋单价	225.6																						
361单价	225.6																						
单 价 差	0.0																						
类 别	管理	单价(百锋)	1.8																				
料率(百锋)	2.0	单价(361)	1.8																				
料率(361)	2.0	单 价 差	0.0																				
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; text-align: right;">款 号</td> <td style="text-align: right; font-weight: bold;">551943701</td> </tr> </table>	款 号	551943701	<div style="background-color: #f0f0f0; height: 40px; width: 100%;"></div>																				
款 号	551943701																						

3、面料配套：从四个维度来分析面料配套数量和比率，直观展示面料到料数量。

Q 供应商/款号		筛选	
供应商	面种	订单配套	配套汇总
客户: 361		季度: 19Q4	
供 应 商	禾宝		
采购数量	46322		
到料数量	38339		
比 率	82.8%		
供 应 商	澳讯		
采购数量	350621		
到料数量	284800		
比 率	81.2%		
供 应 商	豪宝		
采购数量	590844		
到料数量	466241		
比 率	78.9%		
			

4、生产进度：通过图表清晰的展示了对应季度每个款号的生产进度和对完成数量进行统计。



5、交期预警：根据每个款号交期和预计下线时间的比对，算出是否会超期和可能的超期件数，并提供预警，避免了出现延迟情况给企业带来损失。

Q 款号		筛选	
客户: 361		季度: 19Q4	
款号	551944602	客户	361
组别	305	交期	2019-10-01
分配数	13800	预计下线	2019-10-09
生产数	3034	超期天数	8
实际日产	178	超期件数	9342
款号	561943702	客户	361
组别	401	交期	2019-10-01
分配数	11452	预计下线	2019-12-30
生产数	7902	超期天数	90
实际日产	27	超期件数	1120
款号	551944602	客户	361
组别	203	交期	2019-10-01
分配数	9729	预计下线	2019-10-16
生产数	1144	超期天数	15
实际日产	130	超期件数	6635



6、出货进度：展示当前生产款号的出货详情和库存详情。

Q 款号		筛选
客户: 361		季度: 19Q4
款号	661839811	
交 期	2019-07-01	正品出货数 0
品 名	套头卫衣	B品出货数 0
订 单 数	1000	订单出货比率 0%
实 裁 数	0	实裁出货比率 0%
入 库 数	0	正品出货比率 0%
款号	651844704	
交 期	2019-07-01	正品出货数 0
品 名	针织长裤	B品出货数 0
订 单 数	3015	订单出货比率 0%
实 裁 数	0	实裁出货比率 0%
入 库 数	0	正品出货比率 0%
款号	651939815	
交 期	2019-07-15	正品出货数 0
品 名	连帽套头卫衣	B品出货数 0



```
package com.zhi;
import com.biz.admin.AdminRoutes;
import com.biz.demo.DemoRoutes;
import com.biz.m.mRoutes;
import com.jfinal.config.*;
import com.jfinal.core.JFinal;
import com.jfinal.ext.handler.ContextPathHandler;
import com.jfinal.kit.PathKit;
import com.jfinal.kit.PropKit;
import com.jfinal.plugin.activerecord.ActiveRecordPlugin;
import com.jfinal.plugin.activerecord.dialect.SqlServerDialect;
import com.jfinal.plugin.cron4j.Cron4jPlugin;
import com.jfinal.plugin.druid.DruidPlugin;
import com.jfinal.plugin.ehcache.EhCachePlugin;
import com.jfinal.template.Engine;
import model._MappingKit;
import java.io.File;
/**
 * @author 刘孟平
 * @Description: 系统主要的配置文件
 * @date 2017 年 3 月 17 日 下午 5:57:13
 */
public class Config extends JFinalConfig {
    public void configConstant(Constants me) {
        // 开发模式
        PropKit.use("config.properties");
        me.setDevMode(PropKit.getBoolean("devMode", false));
        me.setBaseDownloadPath("/data/temp");
        me.setBaseUploadPath("/data");
        String filePath = PathKit.getWebRootPath() + me.getBaseDownloadPath();
        File file = new File(filePath);
        if (!file.exists())
            file.mkdirs();
        Engine.use().setBaseTemplatePath(PathKit.getWebRootPath() + "/views");
    }
    public void configEngine(Engine me) {
        me.addSharedFunction("/views/common/_layout.html");
        me.addSharedFunction("/views/common/_admContentLayout.html");
        me.addSharedFunction("/views/common/_mLayout.html");
    }
    // 路由器配置
    public void configRoute(Routes me) {
        me.add(new AdminRoutes()); // 总管理后端路由
        me.add(new DemoRoutes());
        me.add(new mRoutes()); //手机端路由
    }
    //连接 mysql bfaps
    public static DruidPlugin createDruidPlugin() {
        return new DruidPlugin(PropKit.get("jdbcUrl"), PropKit.get("user"), PropKit.get("password").trim());
    }
    //连接百锋 OA 数据库 a8v5
    public static DruidPlugin createDruid2Plugin() {
```

```
        return new DruidPlugin(PropKit.get("a8jdbcUrl"), PropKit.get("a8user"),
PropKit.get("a8password").trim());
    }
    //连接 hrsql server
    public static DruidPlugin createC3p0Plugin() {
        return new DruidPlugin(PropKit.get("jdbcUrl_zhi"), PropKit.get("user_zhi"),
            PropKit.get("password_zhi").trim(), PropKit.get("driverClass"));
    }
    //连接工票系统的 sql server
    public static DruidPlugin createC4p0Plugin() {
        return new DruidPlugin(PropKit.get("jdbcUrl_gp"), PropKit.get("user_gp"),
            PropKit.get("password_gp").trim(), PropKit.get("gpdriverClass"));
    }
    //连接吊挂系统的 sql server
    public static DruidPlugin createC3p0Plugin_sdb() {
        return new DruidPlugin(PropKit.get("jdbcUrl_sdb"), PropKit.get("user_sdb"),
            PropKit.get("password_sdb").trim(), PropKit.get("sdbdriverClass"));
    }
    }
    public void configPlugin(Plugins me) {
        // 数据库配置
        // 配置 C3p0 数据库连接池插件
        DruidPlugin druidPlugin = createDruidPlugin();
        me.add(druidPlugin);
        // 配置 ActiveRecord 插件
        ActiveRecordPlugin arp = new ActiveRecordPlugin(druidPlugin);
        arp.setShowSql(true);
        me.add(arp);
        // 所有配置在 MappingKit 中搞定，配置数据库表与映射类的关系_MappingKit 封装方法
        _MappingKit.mapping(arp);
        //连接 OA
        DruidPlugin druidPlugin2 = createDruid2Plugin();
        me.add(druidPlugin2);
        ActiveRecordPlugin arp_oa = new ActiveRecordPlugin("oa", druidPlugin2);
        me.add(arp_oa);
        //连接 sql server
        DruidPlugin c3p0Plugin = createC3p0Plugin();
        me.add(c3p0Plugin);
        //数据库操作插件
        ActiveRecordPlugin arp_zhi = new ActiveRecordPlugin("zhi", c3p0Plugin);
        me.add(arp_zhi);
        //设置方言（很重要，一定要设置）
        arp_zhi.setDialect(new SqlServerDialect());
        //连接工票系统 sql server
        DruidPlugin c4p0Plugin = createC4p0Plugin();
        me.add(c4p0Plugin);
        //数据库操作插件
        ActiveRecordPlugin arp_gp = new ActiveRecordPlugin("gp", c4p0Plugin);
        me.add(arp_gp);
        //连接吊挂系统 sql server
        DruidPlugin c3p0Plugin_sdb = createC3p0Plugin_sdb();
        me.add(c3p0Plugin_sdb);
        //数据库操作插件
        ActiveRecordPlugin arp_sdb = new ActiveRecordPlugin("sdb", c3p0Plugin_sdb);
```

```

        me.add(arp_sdb);
        // 缓存插件
        me.add(new EhCachePlugin());
        /*String redisServer =PropKit.get("redisServer");
        String redisPwd =PropKit.get("redisPwd");
        String redisPort =PropKit.get("redisPort");
        RedisPlugin bbsRedis = new RedisPlugin("wex", redisServer,Utills.toInt(redisPort,6379));
        me.add(bbsRedis); */
        //定时任务
        me.add(new Cron4jPlugin(PropKit.use("task.properties")));
    }
    //可以配置拦截器
    public void configInterceptor(Interceptors me) {
    }
    public void configHandler(Handlers me) {
        me.add(new ContextPathHandler("base_path"));
    }
    /**
     * @Title: 启动 @Description: @param @param args @return void @throws
     */
    public static void main(String[] args) {
        //double d =1000000.020;
        //System.out.println(String.format("%s", Math.round(d)));
        JFinal.start("Web", 8006, "");
        //WebContent 下的模版，端口号，访问路径，
    }
}
package com.biz.admin;
import model.BasEmp;
import model.SysUser;
import com.biz.BaseController;
import com.jfinal.kit.PropKit;
import com.zhi.common.Utills;
import com.zhi.encryp.DES;
import com.zhi.encryp.MD5;
public class AccountController extends BaseController {

    public void index(){
        if(this.isPost()){
            String uName =this.get("uname");
            String pwd =MD5.GetMD5(this.getPara("pwd"));
            SysUser user =SysUser.dao.getFirst(" binary uName=? and pwd=?",uName,pwd);
            if(user == null){
                setAttr("err",1);
                render("index.html");
                return;
            }else if(Utills.toInt(user.getState()+"",0)!=1){//状态不为正常
                setAttr("err",5);
                render("index.html");
                return;
            }else{
                if(!Utills.isNullOrEmpty(user.getEmpNo())){
                    //有绑定员工，员工已停用

```



```
        BasEmp basEmp = BasEmp.dao.getFirst("empNo=?",user.getEmpNo());
        if(basEmp.getState()==2){
            setAttr("err",2);
            render("index.html");
            return;
        }
    }
    if(user.getState()==0){
        setAttr("err",2);
        render("index.html");
        return;
    }else{
        //登录验证
        String ticket =user.getUid()+"|"
            +user.getUName()+"|"
            +user.getTrueName()+"|"
            +user.getRoleIdList();
        String key =PropKit.get("deskey");
        ticket = DES.encoderByDES(ticket, key); //用 DES 加密
        this.setCookie(PropKit.get("sysName")+"_adm", ticket, 365*24*60*60, "/");
        this.redirect("/zhi");
        return;
        /*if(Utils.isMobile(getRequest().getHeader("user-agent"))){
            //手机端登录判断
            this.redirect("/m");
            return;
        }else {
            this.redirect("/zhi");
            return;
        }*/
    }
}
setAttr("err", 3);
render("index.html");
}

public void exitUser(){
    //this.removeSessionAttr(PropKit.get("sysName")+"_adm");
    this.removeCookie(PropKit.get("sysName")+"_adm");
    this.redirect("/zhi/login");
}
public void about(){
    render("about.html");
}
}

package com.biz.admin;
import dataModel.ActionAuth;
import com.jfinal.aop.Interceptor;
import com.jfinal.aop.Invocation;
import com.jfinal.core.Controller;
import com.jfinal.kit.PropKit;
import com.zhi.Sys;
```

```
import com.zhi.common.Utills;
import com.zhi.encrypt.DES;
public class ActionAuthAdmin implements Interceptor{
    public void intercept(Invocation inv) {
        Controller controller = inv.getController();

        String controlName = inv.getControllerKey(); //控制器名称
        controlName = controlName.substring(1, controlName.length());
        String actionName = inv.getMethodName(); //方法名

        //检查是否登录
        String sysName = PropKit.get("sysName");
        String ticket = controller.getCookie(sysName+"_adm");
        if(!Utills.isNullOrEmpty(ticket)){
            ticket = DES.decoderByDES(ticket, PropKit.get("deskey"));
            if(ticket.split("\\").length!=4)
                controller.redirect("/zhi/login");
            else{
                String uid =ticket.split("\\")[0];
                String roleList =ticket.split("\\")[3];
                boolean hadAuth =false;
                if(Sys.isAdmin(roleList))
                    hadAuth =true;
                else
                    hadAuth =Sys.validateAdminUserAuth(uid, controlName, actionName,
"VIEW",PropKit.get("adminmod"));
                if(hadAuth)
                {
                    inv.invoke();
                    ActionAuth actionAuth = new ActionAuth();
                    actionAuth.setUid(uid);
                    actionAuth.setControlName(controlName);
                    actionAuth.setActionName(actionName);
                    actionAuth.setRoleIdList(roleList);
                    controller.setAttr("actionAuth", actionAuth);
                }
                else
                    controller.redirect("/zhi/login");
            }
        }
        else
            controller.redirect("/zhi/login");
    }
}

package com.biz.admin;
import com.biz.BaseController;
import dataModel.ActionAuth;
import com.jfinal.kit.PropKit;
import com.zhi.Template;
import com.zhi.common.Utills;
import com.zhi.encrypt.DES;
import org.apache.log4j.Logger;
public class AdminBaseController extends BaseController {
```

```
public static Logger log = Logger.getLogger(AdminBaseController.class);
/**
 * 默认页大小
 */
public final static int PAGE_SIZE = 15;
public int getPageIndex(){
    return getParaToInt("pageIndex", 1);
}
/**
 * 取当前登录用户的 UID
 */
public String getUID(){
    String uid = "";
    String ticket =this.getCookie(PropKit.get("sysName")+ "_adm");
    if(!Utils.isNullOrEmpty(ticket)){
        ticket = DES.decoderByDES(ticket, PropKit.get("deskey"));
        if(ticket.split("\\|").length>0){
            uid =ticket.split("\\|")[0];
        }
    }
    return uid;
}

/**
 * 取当前登录的用户名
 */
public String getUName(){
    String uName="";
    String ticket =this.getCookie(PropKit.get("sysName")+ "_adm");
    if(!Utils.isNullOrEmpty(ticket)){
        ticket =DES.decoderByDES(ticket, PropKit.get("deskey"));
        if(ticket.split("\\|").length>1){
            uName =ticket.split("\\|")[1];
        }
    }
    return uName;
}

/**
 * 取当前登录用户的 UID
 */
public String getRoleIdList(){
    String uid = "";
    String ticket =this.getCookie(PropKit.get("sysName")+ "_adm");
    if(!Utils.isNullOrEmpty(ticket)){
        ticket = DES.decoderByDES(ticket, PropKit.get("deskey"));
        if(ticket.split("\\|").length>3){
            uid =ticket.split("\\|")[3];
        }
    }
    return uid;
}

/**
 * 初始化工具条
```

```
*/
public void setAdminToolbar(String fn){
    String uid = this.getUID();
    Template template = new Template("/tpl/toolbar.html");
    ActionAuth toolbarAuth = new ActionAuth();
    toolbarAuth.setUId(uid);
    toolbarAuth.setFn(fn);
    template.put("toolbarAuth", toolbarAuth);
    String strTemp = template.renderToString();
    this.setAttr("toolbar", strTemp);
}
public String getWebRoot(){
    String host = this.getRequest().getServerName();
    String port = this.getRequest().getServerPort() == 80 ? "" : ":" + this.getRequest().getServerPort();
    return "http://" + host + port;
}
}
package com.biz.admin;
import com.jfinal.aop.Before;
import com.jfinal.kit.PropKit;
import com.zhi.Sys;
import com.zhi.common.Utills;
import model.SysPagemoduleRole;
import model.SysUser;
@Before(AuthAdmin.class)
public class AdminController extends AdminBaseController {

    public void index() {
        String mod = PropKit.get("adminmod");
        String uid = this.getUID();
        SysUser user = SysUser.dao.getFirst("uid=" + uid + "");
        String htmlAuth = Utills.getCache("auth_" + mod + "_" + uid, 0);
        String htmlFnAuthList = Utills.getCache("auth_fnAuthList_" + mod + "_" + uid, 0);
        String htmlActionAuthList = Utills.getCache("auth_actionAuthList_" + mod + "_" + uid, 0);
        if (Utills.isNullOrEmpty(htmlAuth) || Utills.isNullOrEmpty(htmlFnAuthList) ||
        Utills.isNullOrEmpty(htmlActionAuthList)) {
            htmlAuth = Sys.SetAdminUserPower(uid, mod);
        }

        setAttr("uid", this.getUID());
        setAttr("trueName", user.getTrueName());
        setAttr("menulist", htmlAuth);
        render("index.html");
    }
    //新增模块
    public void addModel(){
        render("addModel.html");
    }
    //修改区块显示
    public void updateModel(){
        if(this.isPost()){
            try{
                int rId = this.getInt("rId");
            }
        }
    }
}
```

```
        SysPagemoduleRole role = SysPagemoduleRole.dao.getFirst("rId="+rId);
        if(role!=null){
            role.setIsShow(1);
            role.update();
        }
    }catch (Exception e){
        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
    renderJson(this.result);
}
}
}
package com.biz.admin;
import com.biz.api.ApiController;
import com.biz.api.DataController;
import com.jfinal.config.Routes;
/**
 * @Description: 总后台路由器
 * @author 刘孟平
 * @date 2017 年 3 月 17 日 下午 5:56:06
 */
public class AdminRoutes extends Routes {
    public void config() {
        add("/zhi", AdminController.class, "/views/admin");
        add("/zhi/login", AccountController.class, "/views/admin/login");
        add("/zhi/common", CommonController.class, "/views/admin/common");
        add("/zhi/sys", SysController.class, "/views/admin/sys");
        add("/zhi/cq", CqController.class, "/views/admin/cq");
        add("/zhi/fin", FinController.class, "/views/admin/fin");
        add("/zhi/sm", SmController.class, "/views/admin/sm");
        add("/zhi/mf", MfController.class, "/views/admin/mf");
        add("/zhi/plan", PlanController.class, "/views/admin/plan");
        add("/zhi/hr", HrController.class, "/views/admin/hr");
        add("/zhi/ie", IeController.class, "/views/admin/ie");
        add("/zhi/qc", QcController.class, "/views/admin/qc");
        add("/zhi/sale", SaleController.class, "/views/admin/sale");
        add("/zhi/tech", TechController.class, "/views/admin/tech");
        add("/zhi/aps", ApsController.class, "/views/admin/aps");
        add("/zhi/bs", BsController.class, "/views/admin/bs");
        add("/zhi/pro", ProController.class, "/views/admin/pro");
        add("/zhi/datahandle", DataController.class, "/views/admin/datahandle");
        add("/zhi/api", ApiController.class, "/views/admin/api");
        add("/zhi/hpage", hpageController.class, "/views/admin/hpage");
        add("/zhi/monitor", MonitorController.class, "/views/admin/monitor");
    }
}
package com.biz.admin;
import com.biz.bll.ApsBll;
import com.jfinal.aop.Before;
import com.openapi4j.exception.OpenAPIException;
import com.openapi4j.service.YongyouService;
import dataModel.EasyUiComboBox;
```

```
import model.ApsConfig;
import model.ApsConfigParam;
import model.BasWarehouse;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
/**
 * aps 系统的配置，只使用在 aps 的本系统上。在使用到其他系统，本文件可以删除
 */
@Before(AuthAdmin.class)
public class ApsController extends AdminBaseController{
    @Before(ActionAuthAdmin.class)
    public void index() throws OpenAPIException{
        if(isPost()){
            try {
                if(this.get("act").equals("ordStatus")){
                    //获取更新订单状态
                    YongyouService yongyou = new YongyouService();
                    yongyou.getOrderstatus();
                }else {
                    ApsConfig apsConfig = this.getMod(ApsConfig.class);
                    if (apsConfig.getConfigId() == null)
                        apsConfig.save();
                    else
                        apsConfig.update();
                }
            }catch (OpenAPIException e){
                this.result.setErrId(e.getErrcode());
                this.result.setMsg(e.getErrmsg());
            }catch (Exception e){
                this.result.setErrId("1");
                this.result.setMsg(e.getMessage());
            }
            renderJson(this.result);
        }else {
            ApsConfig apsConfig = ApsConfig.dao.getFirst(" 1=1 ");
            if(apsConfig==null)
                apsConfig = new ApsConfig();
            setAttr("aps",apsConfig);
            render("index.html");
        }
    }
    //用友设置对接参数
    @Before(ActionAuthAdmin.class)
    public void yongSetParam(){
        if(isPost()){
            try {
                ApsConfigParam param = this.getMod(ApsConfigParam.class);
                param.setModifer(this.getUName());
                param.setModifyTime(new Date());
                if(param.getPid()==null){
                    param.save();
                }else{

```

```
        param.update();
    }
    this.result.setErrId("0");
} catch (Exception e) {
    this.result.setErrId("1");
    this.result.setMsg(e.getMessage());
}
renderJson(this.result);
}else{
    String moduleName=this.get("moduleName");
    ApsConfigParam param = ApsConfigParam.dao.getFirst("moduleName=?",moduleName);
    if(param==null){
        param=new ApsConfigParam();
        param.setModuleName(moduleName);
    }
    setAttr("moduleName",moduleName);
    setAttr("param",param);
}
}
}
//获取用友仓库信息
public void getWare(){
    List<EasyUiComboBox> comboxList = new ArrayList<>();
    List<BasWarehouse> list = BasWarehouse.dao.getList(" 1=1 order by code desc");
    for (BasWarehouse com : list) {
        EasyUiComboBox combox = new EasyUiComboBox();
        combox.setId(com.getCode()+"");
        combox.setText(com.getName());
        comboxList.add(combox);
    }
    renderJson(comboxList);
}
//获取用友入库类型
public void getReceivesendType(){
    ApsBll apsBll = new ApsBll();
    List comboxList = new ArrayList();
    try {
        comboxList = apsBll.batchReceivesendType("901","1",comboxList);
    } catch (OpenAPIException e) {
        e.printStackTrace();
    }
    renderJson(comboxList);
}
}
package com.biz.admin;
import com.alibaba.fastjson.JSON;
import com.zhi.common.PageUtils;
import model.SysFn;
import com.jfinal.aop.Interceptor;
import com.jfinal.aop.Invocation;
import com.jfinal.core.Controller;
import com.jfinal.kit.PropKit;
import com.zhi.common.Utils;
import com.zhi.encrypt.DES;
```

```
import java.util.List;
/**
 * @ClassName: AuthAdmin
 * @Description: 后台管理的权限拦截
 * @author 刘孟平
 * @date 2016 年 3 月 15 日 下午 5:53:56
 */
public class AuthAdmin implements Interceptor{
    public void intercept(Invocation inv) {
        Controller controller = inv.getController();
        String mod = PropKit.get("adminmod");
        String controlName = inv.getControllerKey(); //控制器名称
        controlName = controlName.substring(1, controlName.length());
        String actionName = inv.getMethodName(); //方法名
        //检查是否登录
        String sysName = PropKit.get("sysName");
        String ticket = controller.getCookie(sysName+"_adm");
        if(!Utils.isNullOrEmpty(ticket)){
            ticket = DES.decoderByDES(ticket, PropKit.get("deskey"));
            if(ticket.split("\\|").length!=4){
                controller.redirect("/zhi/login");
            }
            else {
                //生成菜单
                String uid = ticket.split("\\|")[0];
                String strFnlist = Utils.getCache("auth_fnAuthList_" + mod + "_JSON_" + uid,0);
                String groupId = "";
                String fnId = "";
                if(!Utils.isNullOrEmpty(strFnlist)) {
                    List<SysFn> fnList = JSON.parseArray(strFnlist, SysFn.class);
                    for (SysFn fn : fnList) {
                        if (fn.getController().equals(controlName) && fn.getAction().equals(actionName))
                        {
                            groupId = fn.getGroupId();
                            fnId = fn.getFnId();
                        }
                    }
                    controller.setAttr("fnList",fnList);
                    controller.setAttr("groupId",groupId);
                    controller.setAttr("fnId",fnId);
                    controller.setAttr("pageUtils", new PageUtils());
                }
                inv.invoke();
            }
        }
        else
            controller.redirect("/zhi/login");
    }
}
package com.biz.admin;
import com.alibaba.fastjson.JSONArray;
import com.jfinal.aop.Before;
import com.jfinal.plugin.activerecord.Db;
```



```

import com.jfinal.plugin.activerecord.Page;
import com.jfinal.plugin.activerecord.Record;
import com.zhi.Template;
import com.zhi.Utils;
import net.sf.json.JSONObject;
import java.math.BigDecimal;
import java.util.*;
@Before(AuthAdmin.class)
public class MonitorController extends AdminBaseController {
    //region 产前进度
    /**
     * 产前进度信息
     * cjc
     */
    public void monitorInfo() {
        String sql = "SELECT o.qNo from sale_progress_register s LEFT JOIN sale_order o on o.qNo=s.qNo
where s.qNo is not null and o.state in (0,1) GROUP BY o.qNo order by o.qNo";
        List<Record> saleOffers = Db.find(sql);
        if (saleOffers != null && saleOffers.size() > 0) {
            setAttr("qNos", saleOffers);
        }
        render("monitorInfo.html");
    }
    //面辅料汇总供应商分析
    public void getFabricData() {
        try {
            String qNo = this.get("qNo");
            Map map = new HashMap();
            String[] titlearr = {"到货率"};
            String sql = "SELECT CAST(GROUP_CONCAT(a.supabbnname) AS char) as
supabbnames,CAST(GROUP_CONCAT(a.stockNums) as char) as
stockNums,CAST(GROUP_CONCAT(a.storageNums) as char) as
storageNums,CAST(GROUP_CONCAT(a.percent) AS char) as percents " +
                "from (SELECT substring_index(supabbnname,'有限公司',1) as
supabbnname,IFNULL(sum(stockNum),0) as stockNums,IFNULL(sum(storageNum),0) as
storageNums,cast((IFNULL(sum(storageNum),0)/IFNULL(sum(stockNum),0)*100) as decimal(18,1)) as
percent " +
                "from sale_progress_register where qNo =" + qNo + "' GROUP BY supplier HAVING
IFNULL(sum(stockNum),0)>IFNULL(sum(storageNum),0) ORDER BY percent) a ";
            Record repairs = Db.findFirst(sql);
            List<String> datalist = new ArrayList<>();
            String[] supArr = new String[0];
            if (repairs != null) {
                datalist.add(repairs.get("percents"));
                supArr = (repairs.get("supabbnames") + "").split(",");
            }
            Template nodata = new Template("../template/NoData.html");
            map.put("nodata", nodata.renderToString());
            map.put("titlearr", titlearr);
            map.put("supArr", supArr);
            map.put("datalist", datalist);
            this.result.setData(map);
        } catch (Exception e) {

```

```

        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
    renderJson(this.result);
}
//面辅料款号配套率
public void getProNoData() {
    try {
        String qNo = this.get("qNo");
        Map map = new HashMap();
        String[] titlearr = {"配套率"};
        /* String sql = "SELECT CAST(GROUP_CONCAT(a.proNo) as char) as
proNos,CAST(GROUP_CONCAT(a.factory) as char) as factorys,CAST(GROUP_CONCAT(a.orderNums) AS
char) as orderNums," +
        "CAST(GROUP_CONCAT(a.matchNums) AS char) as
matchNums,CAST(GROUP_CONCAT(IFNULL(a.percent,0)) AS char) as  percents " +
        "from (SELECT r.proNo,d.factory,IFNULL(sum(r.orderNum),0) as
orderNums,IFNULL(sum(r.matchNum),0) as
matchNums,cast((IFNULL(sum(r.matchNum),0)/IFNULL(sum(r.orderNum),0)*100) as decimal(18,1))  as
percent " +
        " from sale_progress_register r LEFT JOIN sale_offer_detail d on r.qNo = d.qNo and
r.proNo = d.proNo where  r.qNo=" + qNo
        + ""  GROUP BY r.proNo HAVING
IFNULL(sum(r.orderNum),0)>IFNULL(sum(r.matchNum),0) ORDER BY percent) a ";*/
        String sql = " SELECT CAST(GROUP_CONCAT(a.proNo) as char) as
proNos,CAST(GROUP_CONCAT(a.orderNums) AS char) as orderNums,"
        + " CAST(GROUP_CONCAT(a.matchNums) AS char) as
matchNums,CAST(GROUP_CONCAT(IFNULL(a.percent,0)) AS char) as percents"
        + " from (SELECT r.proNo,sum(IFNULL(r.orderNum,0)) as
orderNums,sum(IFNULL(r.matchNum,0)) as matchNums,"
        + "
if(SUM(IFNULL(r.orderNum,0))=0,0,cast((sum(IFNULL(r.matchNum,0))/sum(IFNULL(r.orderNum,0))*100) as
decimal(18,1))) as percent"
        + " from sale_progress_register r"
        + " where  r.qNo=" + qNo + "" and r.catType='01' and partUsed='大身'"
        + " GROUP BY r.proNo ORDER BY percent) a";
        Record repairs = Db.findFirst(sql);
        List<String> datalist = new ArrayList<>();
        String[] supArr = new String[0];
        if (repairs != null) {
            datalist.add(repairs.get("percents"));
            supArr = (repairs.get("proNos") + "").split(",");
        }
        Template nodata = new Template("../template/NoData.html");
        map.put("nodata", nodata.renderToString());
        map.put("titlearr", titlearr);
        map.put("supArr", supArr);
        map.put("datalist", datalist);
        this.result.setData(map);
    } catch (Exception e) {
        this.result.setErrId("1");
        this.result.setMsg(e.getMessage());
    }
}

```

```
        renderJson(this.result);
    }
    //产前进度
    public void getMonitorData() {
        int page = this.getInt("page", 1);
        int pn = this.getInt("rows", 50000);
        String order = this.getOrderString();
        String qNo = this.get("qNo");
        String sql = "SELECT s.*,date_format(s.deliveryTime, '%Y-%m-%d') deliveryTime ";
        String sqlExcleptSelect = " from cq_progress s LEFT JOIN sale_order o on o.qNo=s.qNo and
s.proNo=o.proNo " +
            "where s.planOnlineTime is not null and o.state <> 3 GROUP BY
s.proNo,s.colorNo,s.deliveryTime ORDER BY s.qNo";
        Page<Record> list = Db.paginate(page, pn, sql, sqlExcleptSelect);
        this.result.toDataGrid(list);
        renderJson(this.result);
    }
    //endregion
    //region 产前准备
    public void monitorPlan() {
        String proSql = "SELECT s.qNo from cq_progress s LEFT JOIN sale_order o on o.qNo=s.qNo where
s.qNo is not null and s.planOnlineTime is not null and o.state in (0,1) GROUP BY s.qNo order by s.qNo";
        List<Record> progres = Db.find(proSql);
        if (progres != null && progres.size() > 0) {
            setAttr("selectQno", progres.get(0).get("qNo"));
            JSONArray jsonArray = new JSONArray();
            jsonArray.addAll(progres);
            setAttr("progres", jsonArray);
        }
        render("monitorPlan.html");
    }
    //endregion
    //region 生产进度
    public void productInfo() {
        String yesterday = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", -1),
"yyyy-MM-dd");
        String nowDate = Utils.getFmDate(new Date(), "yyyy-MM-dd");
        String weekBeforeDate = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", -6),
"yyyy-MM-dd");
        //生产进度
        List<Record> mainPlanQNoList = Db.find("select p.qNo " +
            "from plan_productiondetail p " +
            "inner join " +
            "(select qNo,proNo,state from sale_order group by qNo,proNo) o " +
            "on p.qNo = o.qNo and p.proNo = o.proNo " +
            "where p.qNo is not null and p.qNo <> " " +
            "and o.state <> 3 " +
            "and p.state = 1 " +
            "and p.isPublic = 1 " +
            "group by p.qNo order by p.qNo");
        //生产日报季度（前一天有数据的季度）
        List<Record> dailyQNoList = Db.find("select qNo from mf_daily m " +
            "where day="" + yesterday + "" " +
```

```

        "and m.qNo <> " and m.qNo is not null group by qNo order by qNo");
//生产周计划季度（一个礼拜有数据的季度）
List<Record> weekPlanQNoList = Db.find("select qNo from pro_sourweekplan where uploadDate
between '" + weekBeforeDate + "' +
        "and '" + nowDate + "' and isOver = 1 and qNo is not null and qNo <> " group by qNo order
by qNo");
setAttr("mainPlanQNoList", mainPlanQNoList);
setAttr("dailyQNoList", dailyQNoList);
setAttr("weekPlanQNoList", weekPlanQNoList);
setAttr("yesterDay", yesterDay);
setAttr("nowDate", nowDate);
}
//endregion
//region 出货进度
public void outProductInfo() {
    String nowMonth = Utils.getFmDate(new Date(), "yyyy-MM");
//出货计划季度（本月出货）
List<Record> outPLanQnoList = Db.find("select qNo from sm_shipment " +
        "where concat(year, '-', if(month < 10, concat('0', month), month)) = '" + nowMonth + "' group
by qNo order by qNo");
//季度成品出库季度
List<Record> seasonOutQNoList = Db.find("select t.qNo from sm_totalshipment t left join sale_order s
on t.qNo = s.qNo " +
        "where t.outWareRate > 0 and outWareRate < 1 group by t.qNo order by t.qNo");
setAttr("outPLanQnoList", outPLanQnoList);
setAttr("seasonOutQNoList", seasonOutQNoList);
}
//endregion
//region 出货计划数据
public void getOutProductList() {
    try {
        String qNo = this.get("qNo");
        String nowMonth = Utils.getFmDate(new Date(), "yyyy-MM");
//获取当月月出货计划数据
List<Record> outDataList = Db.find("select proNo,proName,smTime " +
        "from sm_shipment s1 " +
        "inner join sm_shipment_detail s2 on s1.smId = s2.smId " +
        "where concat(s1.year, '-', if(s1.month < 10, concat('0', s1.month), s1.month)) = '" +
nowMonth + "' +
        "and s1.proNo in (select distinct proNo from sale_order where qNo = '" + qNo + "' and
state <> 4)");
HashSet<String> proNoSet = new HashSet<>();
HashSet<String> daySet = new HashSet<>();
LinkedHashMap<String, Object> dataMap = new LinkedHashMap<>();
for (Record record : outDataList) {
    proNoSet.add(record.getStr("proNo"));
    daySet.add(record.getDate("smTime") + "");
}
for (String proNo : proNoSet) {
    List<BigDecimal> smNumList = new ArrayList<>();
    for (String day : daySet) {
        Record record = Db.findFirst("select proNo,proName,ifnull(sum(smNum),0) smNum
from sm_shipment s1 inner join " +

```

```

        "sm_shipment_detail s2 on s1.smId = s2.smId where proNo = " + proNo + "
and smTime = " + day + " and qNo=" + qNo + """);
        if (record == null) {
            smNumList.add(new BigDecimal(0));
        }
        smNumList.add(record.getBigDecimal("smNum"));
    }
    dataMap.put(proNo, smNumList.toArray());
}
this.result.put("dataMap", dataMap);
this.result.put("proNoArr", proNoSet.toArray());
this.result.put("dayArr", daySet.toArray());
} catch (Exception e) {
    log.error("监控大屏获取月出货计划数据失败");
}
}
renderJson(this.result);
}
//endregion
//region 交期预警
public void getWarningList() {
    try {
        int pn = this.getInt("pn", 1);
        int pageSize = this.getInt("pageSize", 50000);
        String where = " p.outDay > 0 and p.outNum > 0 ";
        String sql = "select *,substring(p.qNo,6) customer ";
        String sqlExcleptSelect = "from (select
p1.proNo,p1.qNo,p2.proName,p1.deptAlias,p1.realOutput,p1.offlineTime, p2.contractTime,p1.outTime," +
            "IFNULL(p1.planNum, 0) planNum,IFNULL(p1.finishNum, 0)
finishNum,datediff(p1.offlineTime, p2.contractTime) outDay," +
            "(((p1.planNum - p1.finishNum) - (realOutput * datediff(p1.offlineTime,
p2.contractTime)))) outNum " +
            "from " +
            "plan_productionschedul p1 " +
            "inner join plan_productiondetail p2 " +
            "on p1.detailId = p2.detailId " +
            "inner join (select qNo,proNo,state from sale_order group by qNo,proNo) o " +
            "on p1.qNo = o.qNo and p1.proNo = o.proNo " +
            "where p2.state = 1 " +
            "and p2.isPublic = 1 " +
            "and o.state <> 3) p " +
            "where " + where + " order by qNo,contractTime";
        List<Record> warningList = Db.paginate(pn, pageSize, sql, sqlExcleptSelect).getList();
        this.result.toDataGrid(warningList);
    } catch (Exception e) {
        log.error("监控大屏获取交期预警数据失败");
        e.printStackTrace();
    }
}
renderJson(this.result);
}
//endregion
//region 生产周计划
public void getWeekDetailPlanList() {
    try {

```

```
int pn = this.getInt("pn", 1);
int pageSize = this.getInt("pageSize", 50000);
String qNo = this.get("qNo");
String nowDate = Utils.getFmDate(new Date(), "yyyy-MM-dd");
//前七天
String weekBeforeDate = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", -6),
"yyyy-MM-dd");
String where = "s.uploadDate between '" + weekBeforeDate + "' and '" + nowDate + "' " +
    "and s.isOver = 1 and o.state <> 3 ";
String order = "s.qNo,s.proNo,s.deptNo,s.cropTime,s.color,s.size";
// if (!Utils.isNullOrEmpty(qNo)) {
//     where += " and qNo='" + qNo + "'";
// }
String sql = "SELECT s.* ";
String selectExcelSql = "FROM " +
    "pro_sourweekplan s " +
    "inner join (select qNo,proNo,state from sale_order group by qNo,proNo) o " +
    "on s.qNo = o.qNo and s.proNo = o.proNo " +
    "where " + where + " order by " + order;
Page<Record> planList = Db.paginate(pn, pageSize, sql, selectExcelSql);
this.result.toDataGrid(planList);
renderJson(this.result);
} catch (Exception e) {
    log.error("监控大屏获取生产周计划数据失败");
    e.printStackTrace();
}
}
//endregion
//region 获取生产日报 Echarts 图形所需数据
public void getDailyEchartsData() {
    try {
        String qNo = this.get("qNo");
        String day = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", -1),
"yyyy-MM-dd");
        List<Record> dailyList = Db.find("select m.proNo,sum(m.realOutPut) realOutPut " +
            "from mf_daily m " +
            "inner join " +
            "(select qNo,proNo,state from sale_order where qNo = '" + qNo + "' group by proNo) o "
+
            "on m.qNo = o.qNo and m.proNo = o.proNo " +
            "where m.qNo = '" + qNo + "' " +
            "and m.day = '" + day + "' " +
            "group by m.proNo");
        this.result.setData(dailyList);
    } catch (Exception e) {
        log.error("获取生成日报 Echarts 图形所需数据失败");
        e.printStackTrace();
    }
    renderJson(this.result);
}
//endregion
//region 生产进度数据
public void getScheEchartsData() {
```

```
        try {
            String qNo = this.get("qNo");
            //主计划中获取生成中的数据
            List<Record> scheList = Db.find("select detailId,qNo,proNo,proName, ifnull(totalNum, 0) as
totalNum,ifnull(finishNum, 0) as finishNum " +
                "from plan_productiondetail " +
                "where qNo = " + qNo + " " +
                "and state = 1 and isPublic = 1 " +
                "and (ifnull(totalNum, 0) - ifnull(finishNum, 0)) > 0 order by qNo");
            this.result.put("scheList", scheList);
        } catch (Exception e) {
            log.error("监控大屏获取生成生成日报数据失败");
            e.printStackTrace();
        }
        renderJson(this.result);
    }
//endregion
//region 获取季度成品出入库数据
public void getShipmentList() {
    try {
        int pn = this.getInt("pn", 1);
        int pageSize = this.getInt("pageSize", 50000);
        String where = "outWareRate < 1";
        String sql = "select *,if(wareArrears < 0, 0, wareArrears)
wareArrears,concat(convert(intoWareRate * 100,decimal(18,1)), '%') intoWareRate, " +
            "concat(convert(outWareRate * 100,decimal(18,1)), '%') outWareRate ";
        String sqlExcleptSelect = "from sm_totalshipment where " + where + " order by qNo, proNo";
        List<Record> warningList = Db.paginate(pn, pageSize, sql, sqlExcleptSelect).getList();
        this.result.toDataGrid(warningList);
    } catch (Exception e) {
        log.error("监控大屏获取季度成品出入库数据失败");
        e.printStackTrace();
    }
    renderJson(this.result);
}
//endregion
//region 工位流量
public void balance() {
    String nowDate = Utils.getFmDate(new Date(), "yyyy-MM-dd");
    String sql = "select MONo,WorkLine from vm_DayEmpWorkFlow where dDate = " + nowDate + " " +
        "group by MONo,WorkLine order by MONo,WorkLine";
    List<Record> recordList = Db.use("sdb").find(sql);
    setAttr("recordList", recordList);
    setAttr("nowDate", nowDate);
}
//endregion
//region 获取工位流量平衡数据
public void getBalanceEchartsData() {
    try {
        String MONo = this.get("MONo");
        String WorkLine = this.get("WorkLine");
        String nowDate = this.get("nowDate");
        String sql = "select StationID,SeqCode,SeqName,sum(Qty) Qty from vm_DayEmpWorkFlow
```

```
where dDate = "" + nowDate + "" " +
        "and MONo="" + MONo + "" and WorkLine="" + WorkLine + "" " +
        "group by StationID,SeqCode,SeqName order by StationID";
List<Record> dataList = Db.use("sdb").find(sql);
this.result.put("dataList", dataList);
} catch (Exception e) {
    log.error("生产大屏获取工位流量平衡数据失败");
    e.printStackTrace();
}
}
renderJson(this.result);
}
//endregion
//region 生产管理
public void productManage() {
    //获取生产管理中生产日报季度
    List<Record> dailyQNoList = Db.find("select qNo from sale_order where state <> 3 group by qNo");
    setAttr("dailyQNoList", dailyQNoList);
}
/**
 * 获取生产日报数据
 * cjc
 */
public void getDailyEchartsData2() {
    try {
        String qNo = this.get("qNo");
        Record countOrderNum = Db.findFirst("select count(distinct proNo) orderNum from sale_order
where qNo = "" + qNo + "");
        Record contSxNum = Db.findFirst("select count(distinct proNo) sxNum from mf_daily where qNo
= "" + qNo + "");
        Record countCompleteNum = Db.findFirst("select count(distinct d.proNo) completeNum " +
        "from mf_daily d inner join sale_order o on d.qNo = o.qNo and d.proNo = o.proNo " +
        "where d.qNo = "" + qNo + "" and (o.state = 2 or o.state = 3)");
        this.result.put("countOrderNum", countOrderNum);
        this.result.put("contSxNum", contSxNum);
        this.result.put("countCompleteNum", countCompleteNum);
    } catch (Exception e) {
        log.error("获取生成日报 Echarts 图形所需数据失败");
        e.printStackTrace();
    }
    renderJson(this.result);
}
/**
 * 获取产前进度数据
 * cjc
 */
public void getPrenatalSchedule() {
    try {
        String startDate = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", 0),
"yyyy-MM-dd");
        String endDate = Utils.getFmDate(com.zhi.common.Utils.AddDate(new Date(), "d", 15),
"yyyy-MM-dd");
        int pn = this.getInt("pn", 1);
        int pageSize = this.getInt("pageSize", 50000);

```



```
String where = "planOnlineTime between '" + startDate + "' and '" + endDate + "'";
String sql = "select * ";
String sqlExcleptSelect = "from " +
    "(select * from cq_progress where " + where + " order by planOnlineTime asc) o group
by o.proNo order by o.planOnlineTime";
List<Record> warningList = Db.paginate(pn, pageSize, sql, sqlExcleptSelect).getList();
this.result.toDataGrid(warningList);
} catch (Exception e) {
    e.printStackTrace();
}
renderJson(this.result);
}
//endregion
}
#@admContentLayout()
#define title()首页产前进度表#end
#define bodyclass(class='easyui-layout'#end
#define head()
<script>
    var customer = "#(customer??)";
    var qNo = "#(qNo??)";
    $(function () {
        $("#customer").combobox({
            url: '/zhi/common/getAllCustComboBox',
            valueField: 'text',
            textField: 'text',
            editable: false,
            panelHeight: 150,
            onLoadSuccess: function () {
                $('#customer').combobox('select', customer);
            },
            onChange: function (data) {
                customer = data;
                progressDatagraid();
            }
        });
        $("#qNo").combobox({
            url: '/zhi/common/getAllQnoComboBox',
            valueField: 'text',
            textField: 'text',
            editable: false,
            panelHeight: 150,
            onLoadSuccess: function () {
                $('#qNo').combobox('select', qNo);
            },
            onChange: function (data) {
                qNo = data;
                progressDatagraid();
            }
        });
        progressDatagraid();
    });
    function progressDatagraid(){
```

```

    $('#datagrid1').datagrid({
        loadMsg: "正在加载数据",
        url: '/zhi/hpage/getHomeProgressData?qNo=' + qNo + '&customer=' + customer,
        remoteFilter: true,
        columns: [[
            {field: 'proNo', title: '款号', width: 100, sortable: true, align: 'center'},
            {field: 'proName', title: '品名', width: 100, sortable: true, align: 'center'},
            {field: 'color', title: '颜色', width: 100, sortable: true, align: 'center'},
            {field: 'deliveryTime', title: '交期', width: 90, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'faceQisetTime', title: '封样面辅料要求齐套', width: 140, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingFaceQiSetTime', title: '封样面料齐套', width: 120, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingExcipientQiSetTime', title: '封样辅料齐套', width: 110, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingCropTime', title: '封样裁剪', width: 90, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingEmbroideringTime', title: '封样印绣花压胶发出', width: 140, sortable: true,
align: 'center',formatter:fmt_yyyyMMdd},
            {field: 'embroideredTime', title: '印绣花压胶批样', width: 130, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'embroideryCheckTime', title: '印绣花压胶检测', width: 130, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingBackFactoryTime', title: '封样印绣花压胶回厂', width: 140, sortable: true,
align: 'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingCompleteTime', title: '封样制作完成', width: 95, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'sealingApprovalTime', title: '封样批复', width: 75, sortable: true, align:
'center',formatter:fmt_yyyyMMdd},
            {field: 'undercutTime', title: '下裁剪单', width: 75, sortable: true, align:
'center',formatter:fmt_yyyyMMdd}
        ]
    });
}
</script>
#end
#define main()
<div data-options="region:'center'" id="form1" style="padding-top: 10px;">
    <div id="tb" class="gridToolbar" style="border:0;">
        <form class="layui-form">
            <div class="layui-form-item">
                <div class="layui-inline" style="width:100px;display: inline-block;">
                    <select id="customer" name="customer"
                        lay-ignore class="easyui-combobox qNo-hidden" style="height: 30px; width:
100px;">
                        </select>
                </div>
                <div class="layui-inline" style="width:80px;display: inline-block;">
                    <select id="qNo" name="qNo" lay-ignore class="easyui-combobox qNo-hidden"
style="height: 30px; width: 85px;"></select>
                </div>
            </div>
        </form>
    </div>

```

```
</form>
</div>
<table class="easyui-datagrid" id="datagrid1"
data-options="border:0,fit:true,toolbar:'#tb',rownumbers:true,pagination:true,pageSize:50,singleSelect:true,method:'post'">
</table>
</div>
<script>
    layui.use('form', function () {
        var form = layui.form;
    });
</script>
<style>
    .pagination-page-list{
        display: none;
    }
    .layui-form-item .layui-inline{ margin-bottom: 5px;}
</style>
#end
#@admContentLayout()
#define title()首页成品出库#end
#define bodyclass(class='easyui-layout'#end
#define head()
<script>
    $(function () {
        $('#datagrid1').datagrid({
            loadMsg: "正在加载数据",
            url: '/zhi/hpage/getHomeQuarFinishiDevyData?queryDate=#(queryDate??)',
            remoteFilter: true,
            columns: [[
                {field: 'proNo', title: '款号', sortable: true, width: 100, align: 'center'},
                {field: 'proName', title: '品名', sortable: true, width: 120, align: 'center'},
                {field: 'qNo', title: '季度', width: 60, sortable: true, align: 'center',formatter:fmt_qNo},
                {field: 'customer', title: '客户', width: 120, sortable: true, align:
'center',formatter:fmt_customer},
                {field: 'totalNum', title: '数量', sortable: true, width: 60, align: 'center'},
            ]]
        });
        $(".btn-search").click(function () {
            var queryDate = $("#queryDate").val();
            $('#datagrid1').datagrid("load", '/zhi/hpage/getHomeQuarFinishiDevyData?queryDate=' +
queryDate);
        });
        $(".btn-last-date").click(function () {
            var queryDate = $("#queryDate").val();
            queryDate = dateMinusNum(queryDate, 1);
            $("#queryDate").val(queryDate);
            loadGrid(queryDate);
        });
        $(".btn-next-date").click(function () {
            var queryDate = $("#queryDate").val();
            queryDate = dateAddNum(queryDate, 1);
```

```

        $("#queryDate").val(queryDate);
        loadGrid(queryDate);
    });
});
function loadGrid(queryDate) {
    $("#datagrid1").datagrid("load", '/zhi/hpage/getHomeQuarFinishiDevyData?queryDate=' + queryDate);
};
function fmt_qNo(val, row, index) {
    var qNo = row.qNo;
    if (qNo != null && qNo != "" && qNo != undefined) {
        qNo = qNo.split("|")[0];
    }
    return qNo;
};
function fmt_customer(val, row, index) {
    var customer = row.qNo;
    if (customer != null && customer != "" && customer != undefined) {
        customer = customer.split("|")[1];
    }
    return customer;
}
</script>
#end
#define main()
<div data-options="region:'center'" id="form1" style="padding-top: 10px;">
    <div id="tb" class="gridToolbar" style="border:0;">
        <form class="layui-form">
            <div class="layui-form-item">
                <!--<div class="layui-inline">
                    <div class="layui-input-inline" style="width: 90px;">
                        <input type="text" class="layui-input input-sm" id="queryDate" placeholder="入库
时间"
                                autocomplete="off" readonly>
                    </div>
                </div-->
                <div class="layui-inline" style="width:90px;display: inline-block;">
                    <input type="text" class="layui-input input-sm" id="queryDate"
                        placeholder="出库时间" readonly autocomplete="off">
                </div>
                <div class="layui-inline" style="display: inline-block;">
                    <div class="layui-btn-group">
                        <button type="button" id="doSearch" class="layui-btn layui-btn-sm
btn-search"><i
                                class="layui-icon layui-icon-search"></i>查询
                    </button>
                        <button type="button" class="layui-btn layui-btn-sm layui-bg-cyan btn-last-date">
上一天
                    </button>
                        <button type="button" class="layui-btn layui-btn-sm layui-bg-cyan btn-next-date">
下一天
                    </button>
                    </div>
                </div>
            </div>
        </form>
    </div>
</div>

```

```
        </div>
    </form>
</div>
<table class="easyui-datagrid" id="datagrid1"
data-options="border:0,fit:true,toolbar:'#tb',showFooter:true,rownumbers:true,remoteFilter:true,pagination:true,pa
geSize:50,singleSelect:false,ctrlSelect:true,method:'post'">
    </table>
</div>
<script>
    layui.use(['form', 'laydate'], function () {
        var form = layui.form;
        var laydate = layui.laydate;
        laydate.render({
            elem: '#queryDate',
            type: 'date',
            value: '#(queryDate??)',
            isInitValue: true
        });
    });
</script>
<style>
    .pagination-page-list{
        display: none;
    }
    .layui-form-item .layui-inline{ margin-bottom: 5px;}
</style>
#end
#@admContentLayout()
#define title()工位流量平衡#end
#define bodyclass(class='easyui-layout'#end
#define head()
<script src="/res/js/echarts-4.1.0.rc2/echarts.min.js"></script>
<script src="/res/js/echarts-4.1.0.rc2/skin/macarons.js"></script>
<script>
    var dDate = '#(dDate??)';
    var myChart;
    $(function () {
        $("#WorkLine").combobox({
            url: '/zhi/hpage/getAllWorkLineByDateComboBox?dDate=' + dDate,
            valueField: 'text',
            textField: 'text',
            panelHeight: 150,
            editable: false,
            onLoadSuccess: function (data) {
                if (data.length > 0) {
                    var WorkLine = data[0].id + "";
                    dDate = $("#dDate").val();
                    $("#WorkLine").combobox('setValue', WorkLine);
                    getChartData(dDate, WorkLine);
                } else if (data.length == 0) {
                    $("#WorkLine").combobox('setValue', "");
                }
            }
        });
    });
</script>
```

```
    },
    onChange: function (data) {
        var WorkLine = data;
        dDate = $("#dDate").val();
        getChartData(dDate, WorkLine);
    },
});
$("#btn-last").click(function () { //上一天
    dDate = $("#dDate").val();
    dDate = dateMinus(dDate);
    $("#dDate").val(dDate);
    $("#WorkLine").combobox('reload', '/zhi/hpage/getAllWorkLineByDateComboBox?dDate=' +
dDate)
});
$("#btn-follow").click(function () { //下一天
    dDate = $("#dDate").val();
    dDate = dateAdd(dDate);
    $("#dDate").val(dDate);
    $("#WorkLine").combobox('reload', '/zhi/hpage/getAllWorkLineByDateComboBox?dDate=' +
dDate)
});
});
function beginLoading(){
    layer.load(1, {shade: [0.5, #393D49]});
}
function endLoading(){
    layer.closeAll('loading');
}
function getChartData(dDate, WorkLine) {
    //图例
    var legendData = [];
    //X 轴天数
    var xAxisData = [];
    beginLoading();
    $.ajax({
        type: "POST",
        url: '/zhi/hpage/getHomeStreamBalanceData',
        data: {"dDate": dDate, "WorkLine": WorkLine},
        success: function (json) {
            //数据内容
            var dataArray = [];
            var dataList = json.data.dataList;
            if (dataList == null || dataList == undefined || dataList.length == 0) {
                myChart.clear();
            } else {
                for(var i in dataList){
                    var dataArr = [];
                    var qty = dataList[i].Qty == null?0:dataList[i].Qty;
                    dataArr.push(qty);
                    var obj = {
                        name: dataList[i].SeqName,
                        type:'bar',
                        itemStyle: {
```

```

        normal: {
            label: {
                show: true,
                position:'inside',
                align: 'left',
                rotate: 360,
                formatter: '{name|{a}}  {c}',
                fontSize: 10,
                rich: {
                    name: {color: '#fff'}
                },
            },
        }
    },
    data:dataArr
};
dataArray.push(obj);
}
buildEchart(xAxisData, legendData, dataArray);
}
endLoading();
}
});
};
function buildEchart(xAxisData, legendData, dataArray) {
    myChart = echarts.init(document.getElementById('repair'), "");
    window.onresize = myChart.resize;
    var option = {
        tooltip: {
            trigger: 'axis',
            axisPointer: {                // 坐标轴指示器，坐标轴触发有效
                type: 'shadow'           // 默认为直线，可选为：'line' | 'shadow'
            }
        },
        legend: {
            show: false
        },
        grid: {
            left: '3%',
            right: '4%',
            top:'5%',
            bottom:'5%',
            containLabel: true
        },
        xAxis: [
            {
                type: 'value',
            }
        ],
        yAxis: {
            type: 'category',
            data: ['工序流量']
        },
    },

```

```

        series: dataArray
    };
    myChart.clear();
    myChart.setOption(option);
}
</script>
#end
#define main()
<div data-options="region:'north'" style="height: 70px;">
    <form class="layui-form">
        <div class="layui-form-item">
            <div class="layui-inline" style="width:70px;display: inline-block;">
                <select id="WorkLine" name="WorkLine" lay-ignore class="easyui-combobox qNo-hidden"
                    style="height: 30px; width: 70px;">
                </select>
            </div>
            <div class="layui-inline" style="width:90px;display: inline-block;">
                <input type="text" class="layui-input input-sm" id="dDate" value="#(dDate?)" readonly
autocomplete="off">
            </div>
            <div class="layui-inline" style="display: inline-block;">
                <div class="layui-btn-group">
                    <button type="button" id="last" class="layui-btn layui-btn-sm layui-bg-cyan btn-last">
上一天
                    </button>
                    <button type="button" id="follow" class="layui-btn layui-btn-sm layui-bg-cyan
btn-follow">下一天
                    </button>
                </div>
            </div>
        </div>
    </form>
</div>
<div data-options="region:'center'">
    <div id="repair" style="height: 100%;width:90%;"></div>
</div>
<script>
    var laytpl;
    layui.use(['form', 'laydate'], function () {
        var form = layui.form;
        var laydate = layui.laydate;
        laydate.render({
            elem: '#dDate',
            done: function(value, date, endDate){
                dDate =value;
                $("#WorkLine").combobox('reload', '/zhi/hpage/getAllWorkLineByDateComboBox?dDate=' +
dDate);
            }
        });
    });
</script>
#end
package com.biz.bll;

```



```
import model.SysProgresspmisdetail;
import model.SysTechpmisdetail;
import model.SysUser;
import org.apache.log4j.Logger;
public class SysBll {
    public static Logger log = Logger.getLogger(SysBll.class);
    /**
     * 产前进度权限设置保存
     * cjc
     */
    public void saveProgressPmis(int groupId, SysUser user) throws Exception {
        SysProgresspmisdetail progresspmisdetail = SysProgresspmisdetail.dao.getFirst("uid=" + user.getUid()
+ "" and groupId=" + groupId);
        if (progresspmisdetail == null) {
            progresspmisdetail = new SysProgresspmisdetail();
            progresspmisdetail.setGroupId(groupId);
            progresspmisdetail.setUid(user.getUid());
            progresspmisdetail.setUName(user.getUName());
            progresspmisdetail.setRoleName(user.getRoleName());
            progresspmisdetail.setRealName(user.getTrueName());
            progresspmisdetail.setTel(user.getTel());
            progresspmisdetail.save();
        } else {
            throw new Exception("该用户已存在！");
        }
    }
    /**
     * 技术部开发进度表权限设置保存
     * cjc
     */
    public void saveTechPmis(int groupId, SysUser user, int techType) throws Exception {
        SysTechpmisdetail techpmisdetail = SysTechpmisdetail.dao.getFirst("uid=" + user.getUid() + "" and
groupId=" + groupId + "" and type=" + techType);
        if (techpmisdetail == null) {
            techpmisdetail = new SysTechpmisdetail();
            techpmisdetail.setGroupId(groupId);
            techpmisdetail.setUid(user.getUid());
            techpmisdetail.setUName(user.getUName());
            techpmisdetail.setRoleName(user.getRoleName());
            techpmisdetail.setRealName(user.getTrueName());
            techpmisdetail.setTel(user.getTel());
            techpmisdetail.setType(techType);
            techpmisdetail.save();
        } else {
            throw new Exception("该用户已存在！");
        }
    }
}
package com.biz.task;
import com.jfinal.plugin.cron4j.ITask;
import org.apache.log4j.Logger;
import java.util.Date;
public class SynPro implements ITask {
```

```
protected Logger log = Logger.getLogger(SynPro.class);
@Override
public void run() {
    /* VSBll vsBll = new VSBll();
    //同步工票的物料分类
    vsBll.synProCat();
    //同步工票物料
    vsBll.synPro();
    //同步工票物料单位
    vsBll.synProUnit();*/
}
@Override
public void stop() {
    // TODO Auto-generated method stub
    Date time = new Date();
    log.error("自动执行停止=" + time);
}
}
package com.biz.task;
import com.biz.bll.ApsBll;
import com.jfinal.plugin.cron4j.ITask;
import org.apache.log4j.Logger;
import java.util.Date;
//获取用友的采购订单数据
public class SynProOrderYongyou implements ITask {
    protected Logger log = Logger.getLogger(SynProOrderYongyou.class);
    @Override
    public void run() {
        ApsBll apsBll = new ApsBll();
        int synType=0;
        apsBll.updateMatePurch("",synType);
        //获取用友采购单数据
        //apsBll.updateMate("");
        //获取用友采购入库数据
        //apsBll.updateMateralin();
    }
    @Override
    public void stop() {
        // TODO Auto-generated method stub
        Date time = new Date();
        log.error("自动执行停止=" + time);
    }
}
package com.biz.task;
import com.biz.bll.ProBll;
import com.jfinal.plugin.cron4j.ITask;
import com.zhi.common.Utills;
import model.ProSourweekplan;
import org.apache.log4j.Logger;
import java.util.Date;
public class SynYxcover implements ITask {
    protected Logger log = Logger.getLogger(SynYxcover.class);
    @Override
```

```
public void run() {
    ProSourweekplan proSourweekplan = ProSourweekplan.dao.getFirst(" 1=1 GROUP BY qNo ORDER
BY qNo desc ");
    ProBll proBll = new ProBll();
    int synType=0;//自动同步
    if(proSourweekplan!=null){
        String startTime = Utils.getFmDate(Utils.AddDate(new Date(), "d", -1), "yyyy-MM-dd") + "
00:00:00";
        String endTime = Utils.getFmDate(new Date(), "yyyy-MM-dd") + " 23:59:59";
        proBll.getyxCoverData(proSourweekplan.getQNo(),synType,startTime,endTime);
    }
}
@Override
public void stop() {
    // TODO Auto-generated method stub
    Date time = new Date();
    log.error("自动执行停止=" + time);
}
}
package com.zhi;
import com.jfinal.template.Engine;
import java.io.File;
import java.util.HashMap;
import java.util.Map;
public class Template {
    Map<String, Object> _map = null;
    com.jfinal.template.Template _template =null;

    /**
     * @param fileName=相对路径和文件名, 从 views 开始, 不包含 views
     */
    public Template(String fileName) {
        _map = new HashMap<String, Object>();
        String filePath =Engine.use().getBaseTemplatePath()+fileName;
        File file=new File(filePath);

        if(file.exists())//如果传进来的不是一个文件路径, 就代表这个路径就是模版内容。
            _template =Engine.use().getTemplate(fileName);
        else
            _template =Engine.use().getTemplateByString(fileName);
    }
    public void put(String key,Object value){
        _map.put(key, value);
    }
    /**
     * 模版渲染之后变成字符串
     * @return
     */
    public String renderToString(){
        return _template.renderToString(_map);
    }
}
}
```