

# { Diana 轻量级开发框架 }


## 框架说明书


文件状态： <input type="checkbox"/> 草稿 <input checked="" type="checkbox"/> 正式发布 <input type="checkbox"/> 正在修改	文件标识：	华东交通大学系统工程与密码学研究所—— Diana 轻量级开发框架
	当前版本：	0.0.1
	作 者：	胡凯雨，张梦丽，艾美珍，易传佳，陈祚松，康文洋，张婷婷，王露，周庆，夏萍萍，陈兰兰
	完成日期：	2018.1.28


## 1. 框架简介


Diana 轻量级开发框架的思想是基于 MVC 框架，但去除 MVC 框架的系统结构和实现的复杂性。通过使用单入口模式，利用反射机制实现了模型(model)—视图(view)—控制器(controller)的逻辑、数据、界面显示的分离，使得框架加载性能相对于传统 MVC 框架较为高效。前台设计了两种页面风格(基于 MVVM 框架 Knockout 的 ACE 前端，基于 Bootstrap 开发的扁平化前端)。后台 ORM 采用 SQL Sugar 使是系统与数据库的操作更加灵活稳定。Diana 轻量级开发框架有强大的权限管理，并且集成 workflow 引擎组件，支持可视化操作，使业务流程灵活可控。框架本身是一个可二次开发的开发平台，开发者既能通过在本框架上完成大部分的基础开发工作，也可以根据框架预留接口进行再次开发，本框架结构清晰、性能高、速度快、扩展性强、界面简单。

## 2. 框架编写的目的

 **提高开发效率：**Diana 轻量级开整体框架都已经搭建好了，开发者可以使用该框架模板完成大部分的基础开发工作。

 **降低成本：**Diana 轻量级开发框架的核心思想就是在提高效率的同时就是在降低成本。为初级程序员甚至只要思路清晰的人就可以进行功能开发。开发周期变短。并且提供了通用 workflow，一般的业务系统均可适用。

 **提高客户满意度：**Diana 轻量级开发框架开发人员提供了两套目前较为流行的前端页面风格 Bootstrap 和 Ace。开发者可以自由切换前端页面，适用于不同客户的需求。

 **提供框架源代码，提供完整的授权：**我们具有稳定的开发团队，并全程提供开发框架源代码，可以放心地拿这套框架去开发自己的产品。

## 3 框架整体设计

Diana 开发框架提供两种前端模板，界面风格简洁、大气、操作便捷，一种是基于 MVVM 框架 Knockout 的 ACE 前端，另外一种是基于 Bootstrap 开发的扁平化前端。两套前端的切换只需在配置文件中直接更换即可。同时适用两种主流数据库 Sql Server 和 My sql，后台使用 Sql Sugar 组件，提高了运行效率，并且使框架更灵活。

Diana 开发框架提供了基于 rbac(role based access control)的形式的权限控制,按角色、按部门、按用户组都可以，操作权限细化到了界面上的每一个按钮，并且将权限细分为静态权限和动态权限，这样将资源和动作分离开，避免程序耦合度高，实现两者的独立运行，以适用于复杂的权限系统。

Diana 开发框架提供了通用 workflow 管理系统 (Workflow Management System, WfMS),主要功能是通过计算机技术的支持去定义、执行和管理 workflow，协调 workflow 执行过程中工作之间以及群体成员之间的信息交互。workflow 需要依靠 workflow 管理系统来实现。

**流程 (workflownodeaction)：** workflow 包含多个 workflow，处理时可任选一种流程进行处理，其包含步骤信息；

**节点 (workflownode)：** 流程中每一环节的名称，某一流程将包含多个节点，也就是状态 (其他 workflow 中也叫步骤)。

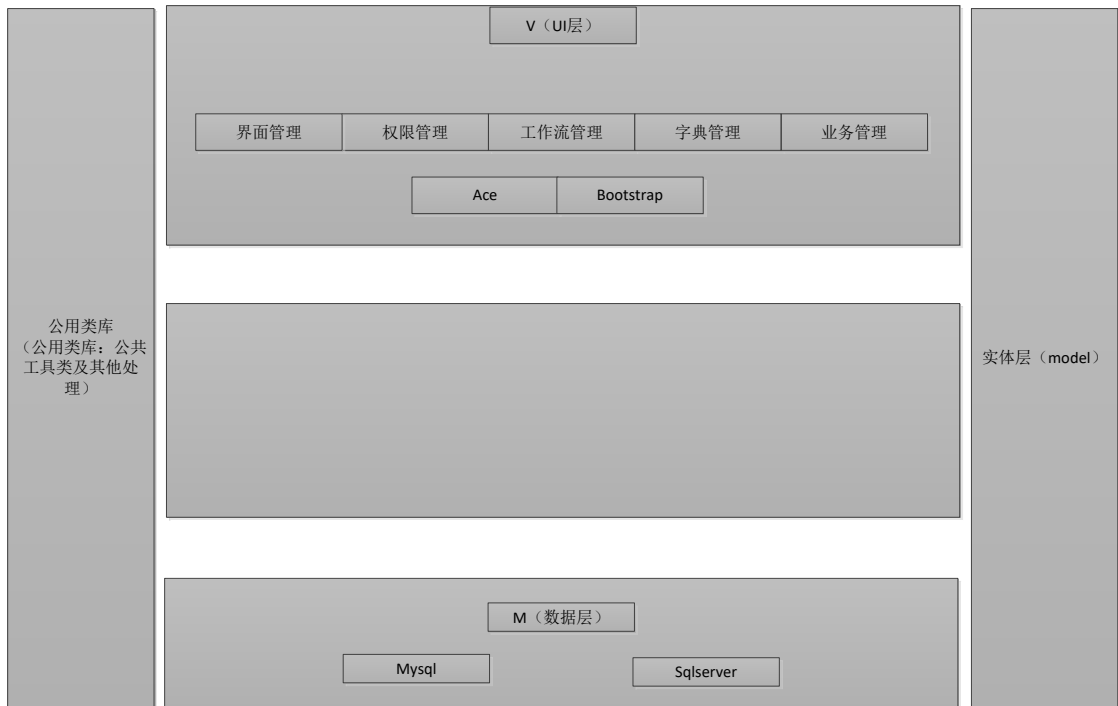


图 3.1 总体架构设计图

### 3.1. 控制器设计

设计的总体思想：使用单入口模式，利用反射机制实现了控制器到方法的控制操作。整个控制器的设计思想如下图：

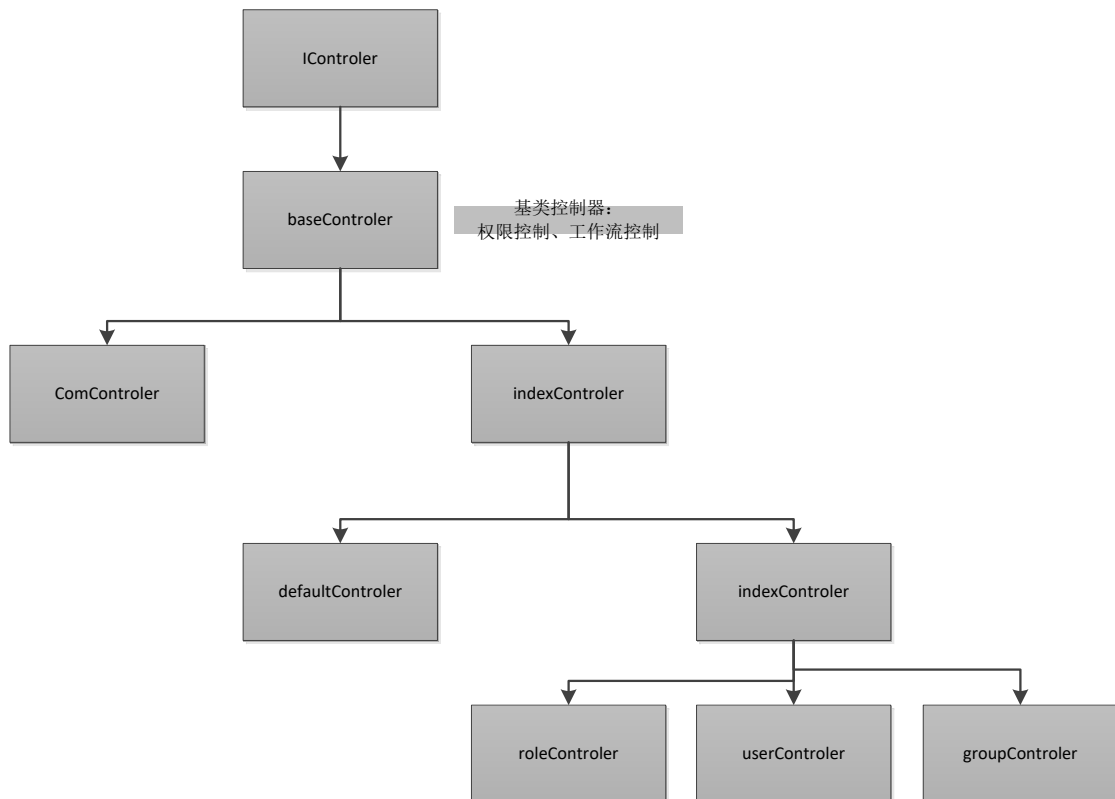


图 3.2 整个控制器的设计图

控制器设计思想介绍：IControler 通过接口类实现具体的基类方法中的执行；

baseControler: 单入口模式实现权限的控制和工作流方法的执行；comControler: 公共类库，包括存储过程，加载菜单、执行 sql、加载工具条、加载文件、加载脚本等

3.2. 系统权限设计

权限控制设计思想：良好的控制器的设计规则，可以为 Web 开发框架本身提供了很好地用户访问控制和权限控制，是的用户界面资源的控制（菜单、按钮、工具条、js、文件的呈现，动作的提交），均能在总体权限功能的分配和控制之下。

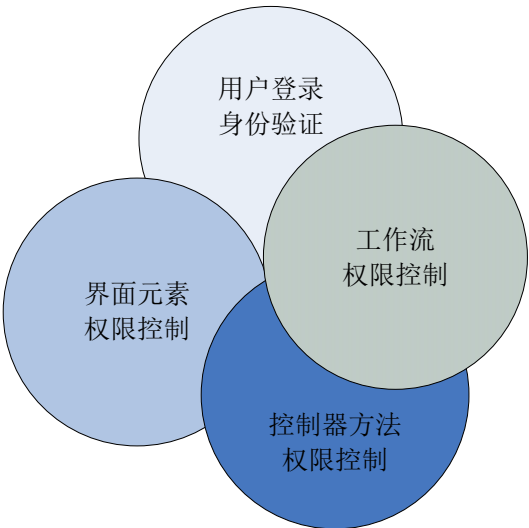


图 3.3 权限控制：四位一体

3.3 工作流设计

串行会签工作流：串行会签模式是指工作流流程中的任务是依照次序顺序执行，没有分支选择、循环等路由功能。



图 3.4 串行会签

分支/合并会签工作流：分支模式是指对工作流流程中的任务执行条件进行限定，在执行过程中，根据条件判断执行路径。

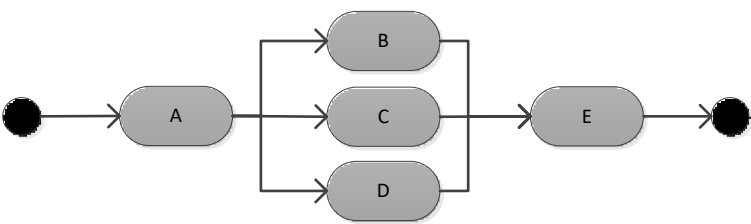


图 3.5 分支/合并会签

简单来说，在流程业务中，前者即是由一个领导签完再至另一领导签。后者表示几个领导同时进行签发，而不清楚最终是谁先签。

以上的解决方式有两大业务需求下是不能满足的：若会签的领导不是固定的，即可以由上一任务审批人提交前随意进行选择；另一种是对于会签业务中，要求若其中一部分领导审批通过，即直接往下走，不需要全部领导进行审批。另外，对于这种情况下，统计最终领导会签的结果也是比较困难的，即对审批单的意见是同意还是否决没有办法清楚。以上两种业务需求也是很常见的日常需求，但我们若采用了固定的流程节点，则不能实现。因此，我们

需要在会签节点上绑定我们的设计规则。

4. 框架具体实现

权限管理的思想是基于传统的rbac（role based access control）。以角色为中心，给对应的用户授权；授权项包括：动作授权、资源（菜单、工具条、按钮、脚本）授权。用户管理、分组管理、角色管理和资源管理、动作管理等功能管理，权限系统主界面如下所示：

- 🎨 用户管理：查询用户、新增用户、新增用户、删除用户、查看用户。  
查询用户：按条件查询，查询字段要求可配置：如，指定可按用户姓名、手机号可查询，则输入用户姓名、手机号即按关键字检索数据。  
新增用户：新增用户的基本信息：用户名、密码、并为该用户设置角色和分组  
修改用户：修改用户对应的信息：用户名、密码、并修改该用户角色和分组  
查看用户：用户信息包括：用户名、对于角色和分组  
查询授权对象：查看指定授权项的用户信息；

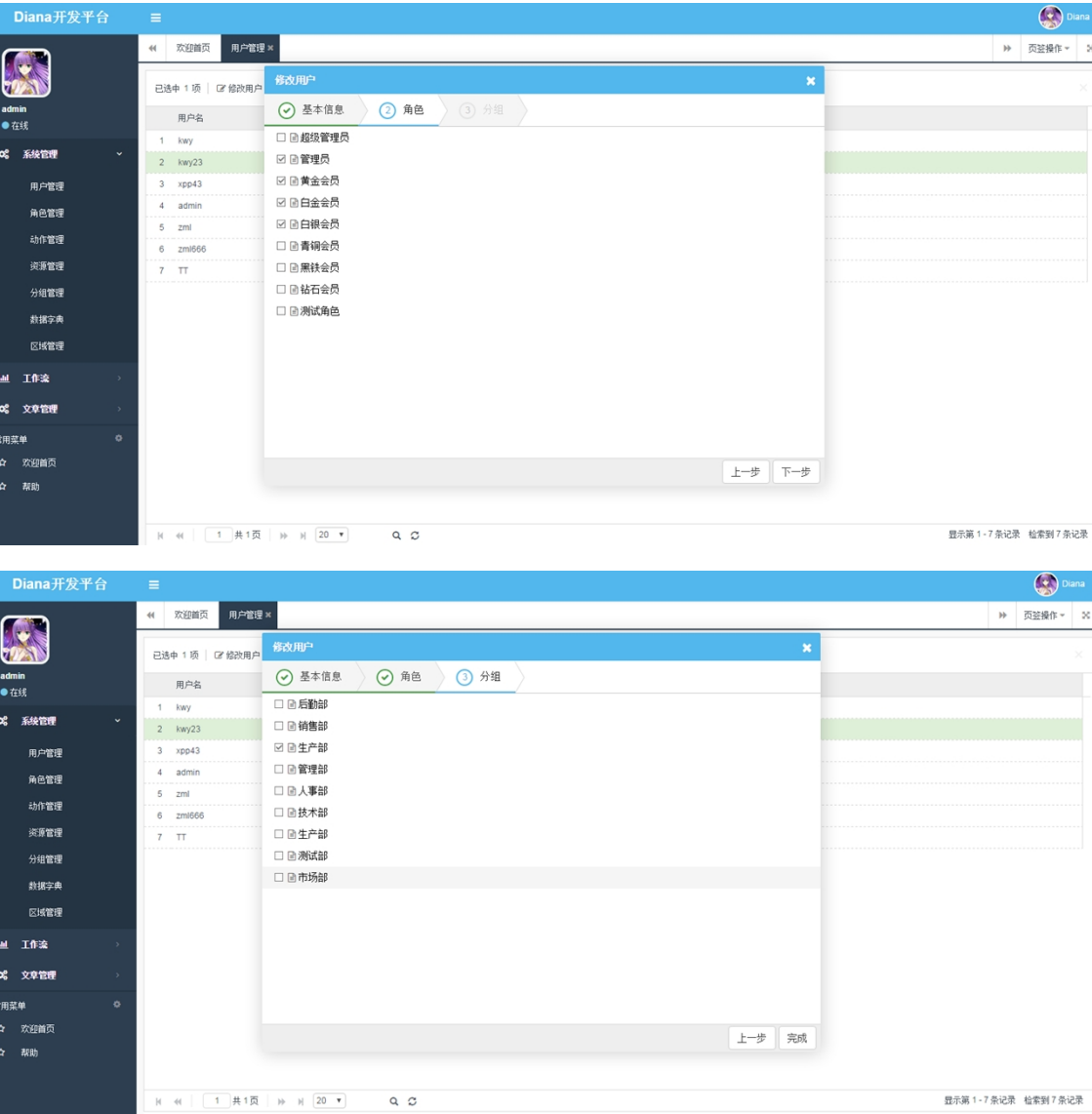


图 4.1 用户管理

- 🎨 角色管理：查询指定角色、修改角色、修改角色权限、删除角色、新增角色基本信

息及该角色所拥有的的权限、查看角色详情、查看角色拥有的权限。

角色类型分为：系统角色（超级管理员、其他角色）。超级管理员可以管理整个系统的人员及其相关信息，具有操作框架的最高权限，其他的权限由超级管理员去分配；业务角色由业务系统的具体业务来确定。

修改角色：修改如下图所对应的字段信息；

修改角色权限：角色权限包括资源（菜单、按钮、脚本等资源）的控制、动作的控制，以树的形式呈现，如下图所示：

查看角色详情：查看某个用角色详情，包括：角色基本信息，当前角色所拥有的的权限

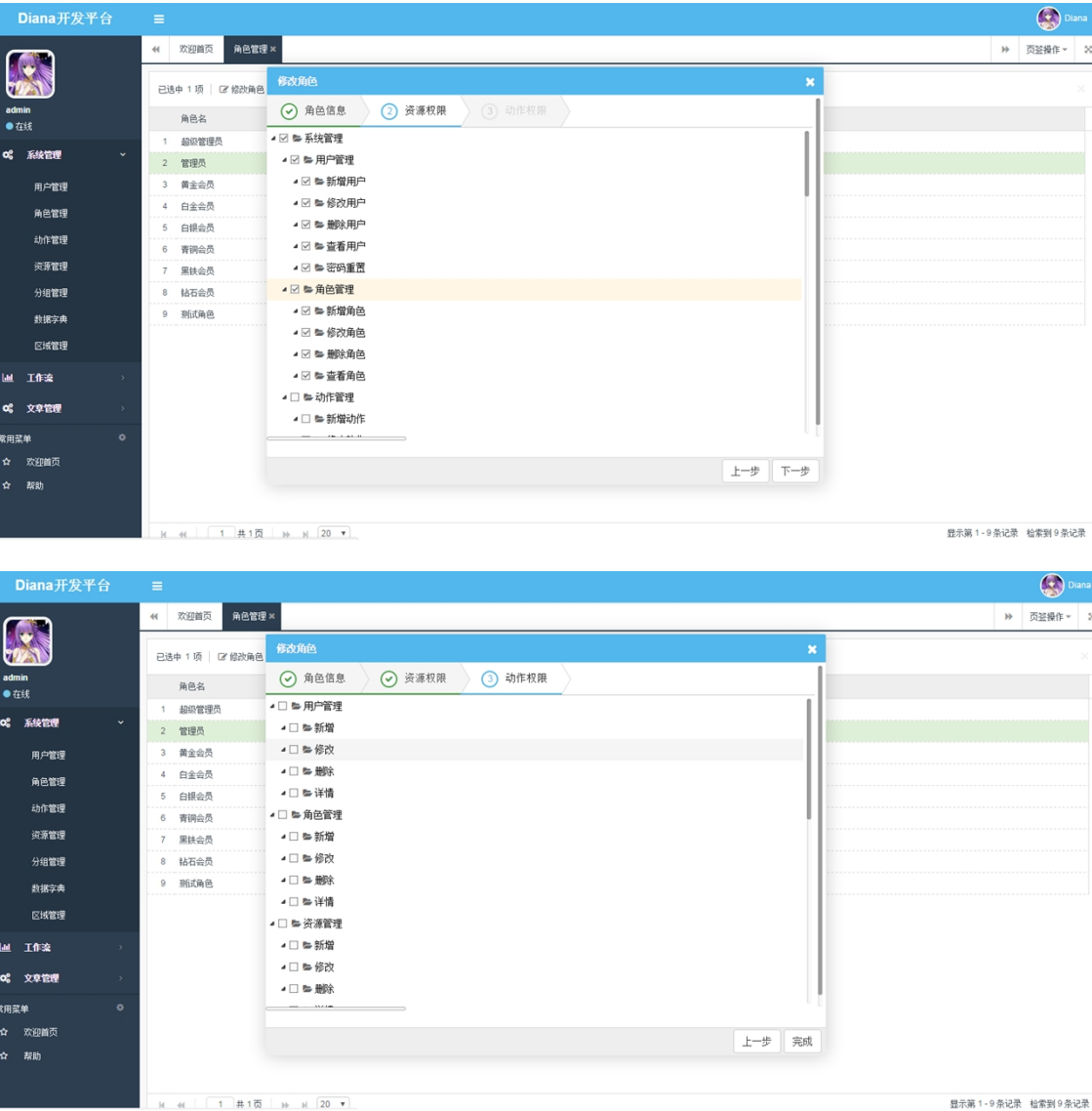



图 4.2 角色管理

 资源管理：菜单的查询、新增、修改、删除，对应表中的 resourcelevel 设置对应资源(1 菜单资源，2 工具条资源列表，3 按钮资源列表)

资源管理展示：数据表格树形分页呈现；

查询：可按用户指定数据查询相应数据；

新增：新选择相应的上一级菜单，添加资源名、资源描述、资源地址；

修改：修改的资源栏项，选择相应的上一级资源，修改资源名、资源描述；

删除：修的资源栏项，选择相应的数据删除。

名称	描述	路径	类型	级别	样式	左侧图标	右侧图标	预警数量	排序	脚本
1	系统管理		undefinec	1		fa fa-gears	fa arrow	0	0	
2	用户管理	index.ashx?ctrl=user	undefinec	2					0	
3	新增用户		undefinec	3	btn btn-primary dro	fa fa-plus	fa fa-plus	1		
4	修改用户		undefinec	3		fa fa-pencil-square	fa fa-pencil-square-o	1		
5	删除用户		undefinec	3		fa fa-trash-o	fa fa-trash-o	1		
6	查看用户		undefinec	3		fa fa-search-plus	fa fa-pencil-square-o	1		
7	密码重置		undefinec	3		fa fa-key	fa fa-pencil-square-o	1		
8	角色管理	index.ashx?ctrl=role	undefinec	2				0		
9	新增角色		undefinec	3	btn btn-primary dro	fa fa-plus	fa fa-plus	1		
10	修改角色		undefinec	3		fa fa-pencil-square	fa fa-pencil-square-o	1		
11	删除角色		undefinec	3		fa fa-trash-o	fa fa-trash-o	1		
12	查看角色		undefinec	3		fa fa-search-plus	fa fa-pencil-square-o	1		
13	动作管理	index.ashx?ctrl=action	undefinec	2				0		
14	新增动作		undefinec	3	btn btn-primary dro	fa fa-plus		1		
15	修改动作		undefinec	3		fa fa-pencil-square	fa fa-pencil-square-o	1		
16	删除动作		undefinec	3		fa fa-trash-o	fa fa-trash-o	1		
17	查看详情		undefinec	3		fa fa-search-plus	fa fa-pencil-square-o	1		
18	资源管理	index.ashx?ctrl=resource	undefinec	2				0		

图 4.3 资源管理

工作流管理：对工作流进行操作。

添加工作流（相应的工作流信息）——>添加该工作的节点（相应的节点信息）——>添加节点之间的触发动作（相应的动作信息）——>为该工作流配置业务数据；

工作流信息：工作流名字（唯一）、工作流描述、工作流对应的影子表（下拉选择）、工作流对应的业务表、工作流状态；

节点信息：所属工作流、节点名字、当点描述、节点操作者（用户、角色）、节点开始时间、节点结束时间、节点状态；

触发动作：在相应的两个节点之间添加触发动作信息，包括：动作名、动作描述。

工作流名称	启用时间	有效终止时间	工作流作用表	工作流状态	锁定状态	业务表/工作流字段
1 wfMaterial	2013-01-01 00:00:00	2017-01-01 00:00:00	MaterialInventory	未启用	未锁定	
2 tantest	2017-09-07 00:00:00	2017-09-07 00:00:00	tantest	未启用	未锁定	

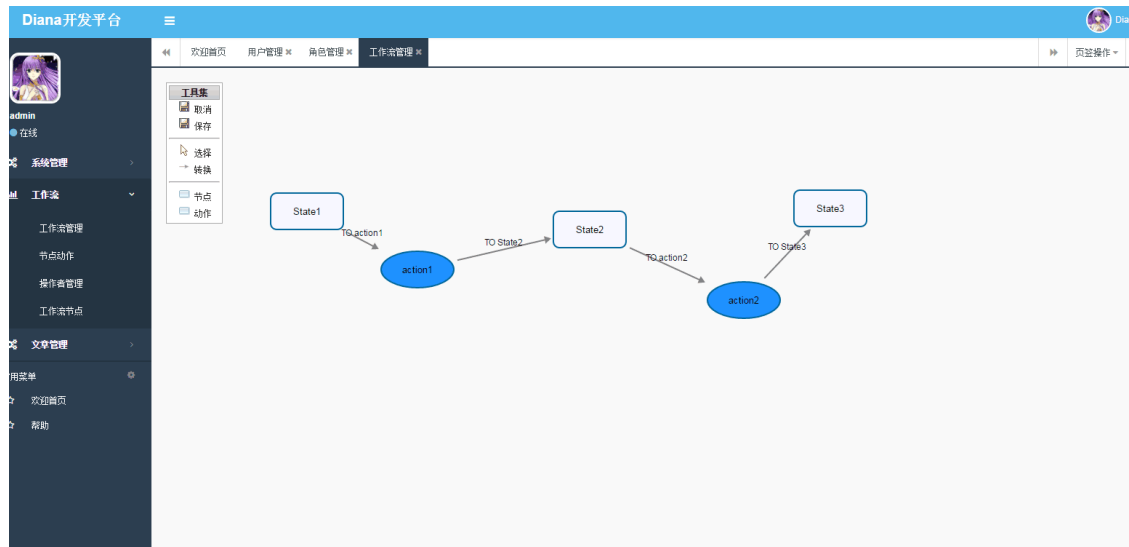


图 4.3 workflow 设计