

哈尔滨工业大学 计算学部
2023 年秋季学期 《开源软件开发实践》
Lab5：典型低代码平台使用

学号	姓名	联系方式
2023140022	雷文俊	514530706@qq.com/13394013296

目 录

1	实验要求.....	1
2	掌握 inBuilder 低代码开发平台的使用.....	1
3	利用 inBuilder 开发软件.....	41
3.1	场景描述和功能设计.....	41
3.2	界面截图.....	42
3.3	Demo 视频.....	43
4	***熟悉 UBML 项目	43
5	小结.....	43

[文档全部完成之后, 请更新上述区域]

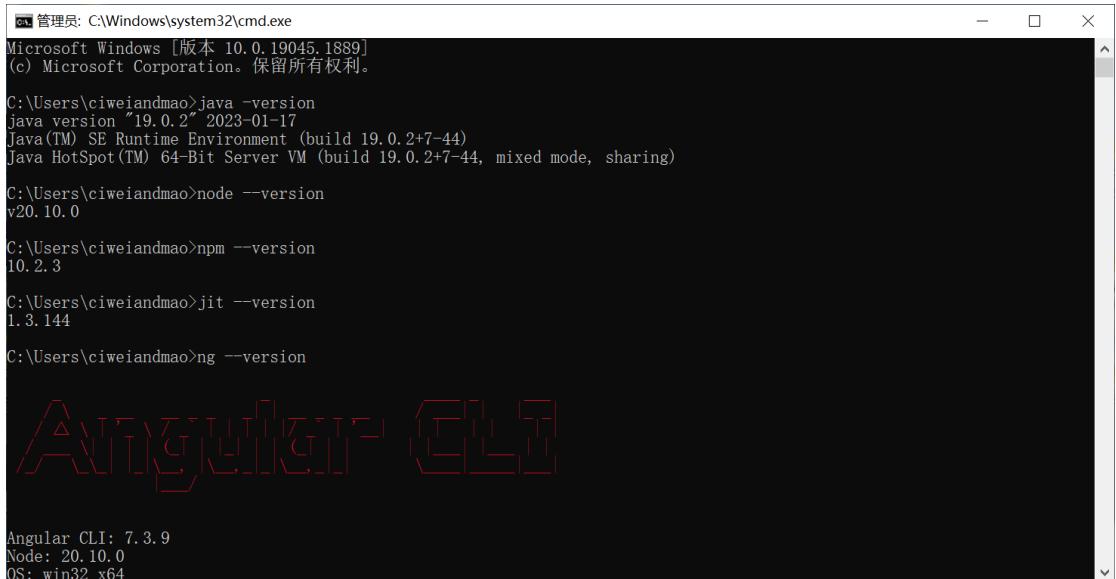
1 实验要求

本次实验训练典型低代码平台的使用和利用低代码平台开发程序，具体来说：

- 掌握典型低代码平台 inBuilder 的使用
- 能够利用 inBuilder 平台进行软件产品的开发
- 熟悉 UBML 开源项目

2 掌握 inBuilder 低代码开发平台的使用

完成 JDK, Node.js, Jit-Engin, ng 的安装



```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.19045.1889]
(c) Microsoft Corporation。保留所有权利。

C:\Users\ciweiandmao>java --version
java version "19.0.2" 2023-01-17
Java(TM) SE Runtime Environment (build 19.0.2+7-44)
Java HotSpot(TM) 64-Bit Server VM (build 19.0.2+7-44, mixed mode, sharing)

C:\Users\ciweiandmao>node --version
v20.10.0

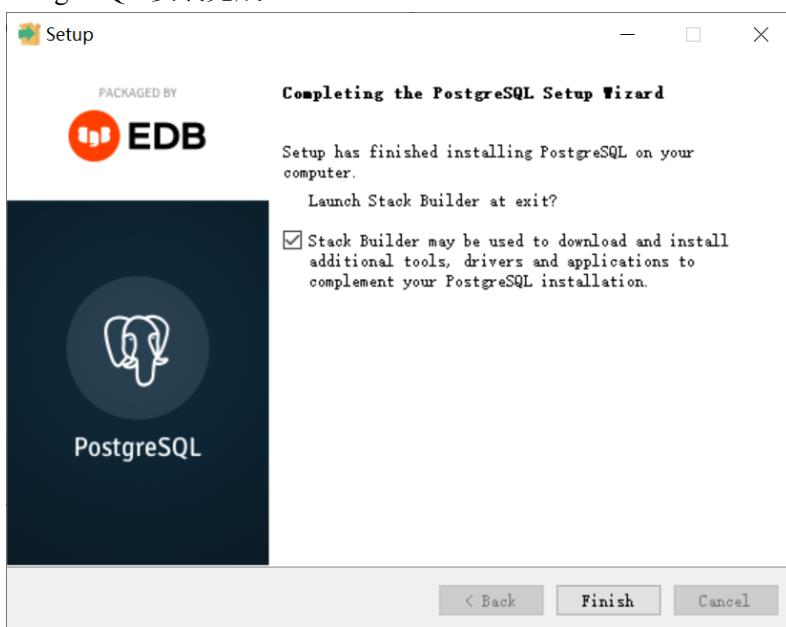
C:\Users\ciweiandmao>npm --version
10.2.3

C:\Users\ciweiandmao>jit --version
1.3.144

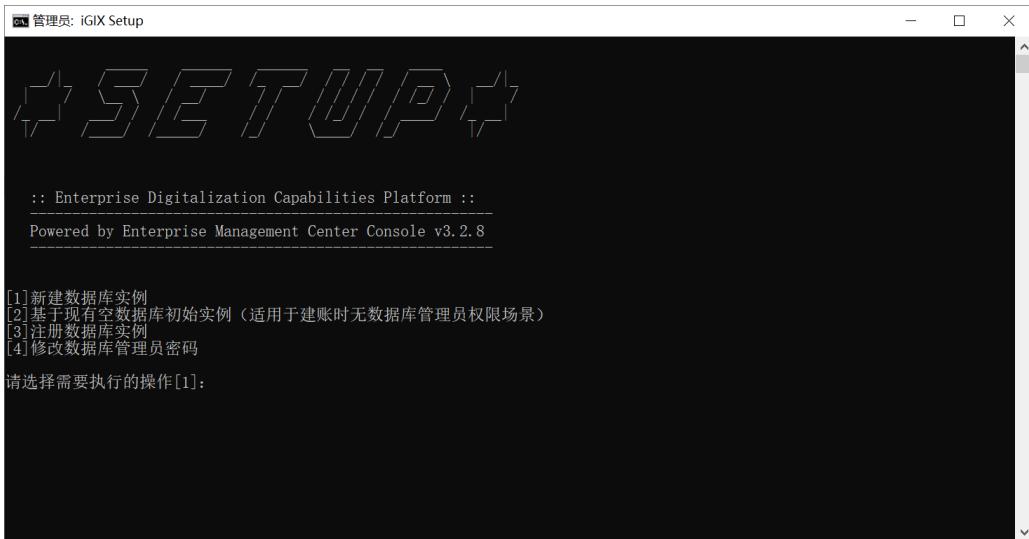
C:\Users\ciweiandmao>ng --version

Angular CLI: 7.3.9
Node: 20.10.0
OS: win32 x64
```

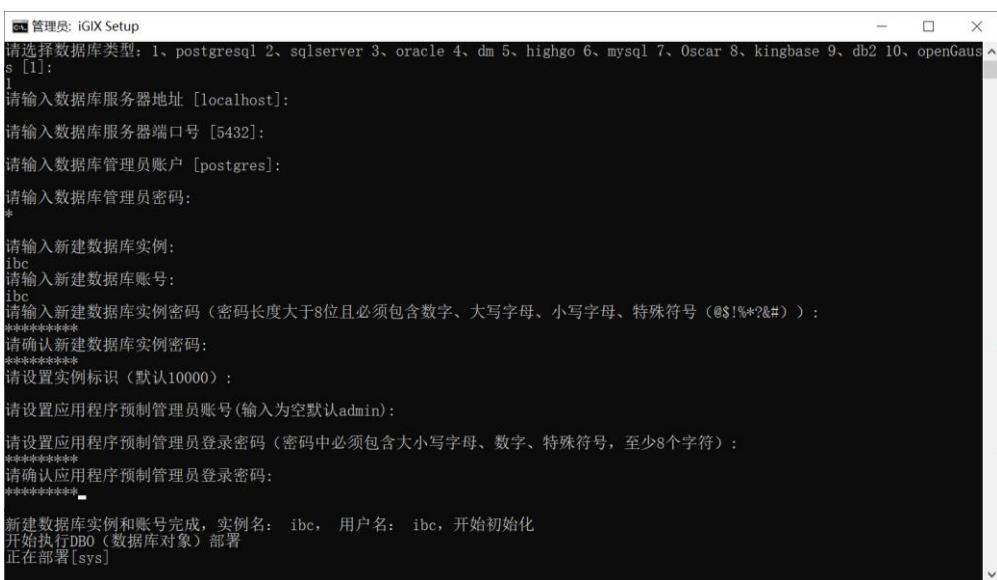
PostgreSQL 安装完成



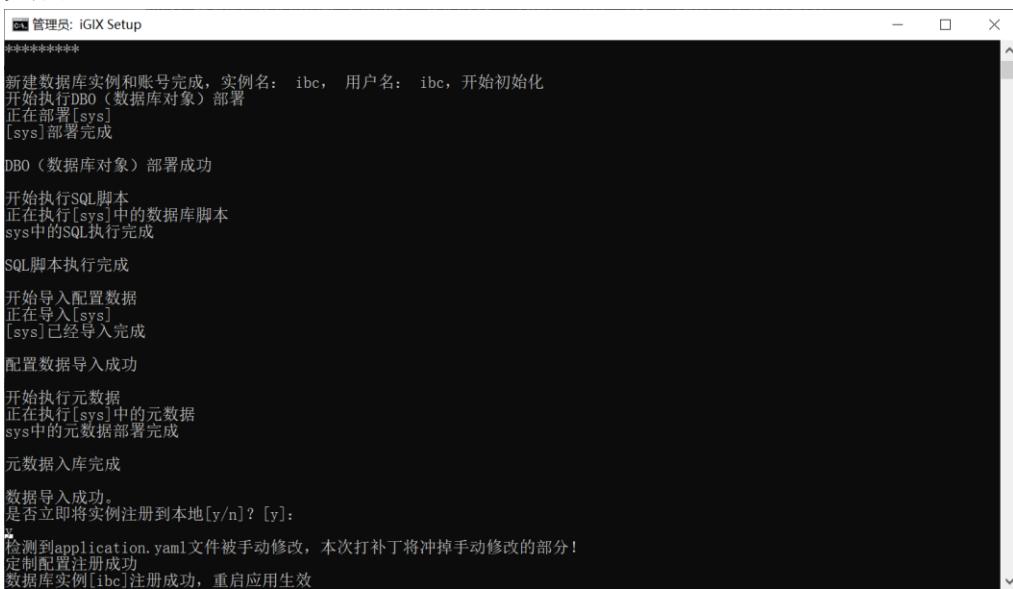
工具启动



建立数据库



完成配置



启动服务成功：

```

管理员: inBuilder Community Server
11:15:23.036 [main] INFO io.iec.edp.caf.common.runtime.msu.ServiceUnitConfigService - Enable service units: [DF, Order, p^
fcommon, main, aif, audit, Bcc, Chgdr, Prt, COMMENT, rtccommon, cs, csb, dfs, dip, ebs, epp, Ewp, Icc, Lcm, msg, nocode, portal, rri, Schd, sg, S
ys, task, Trace, wf]

:: Enterprise Digitalization Capabilities Platform ::

Powered by Cloud Application Foundation Server v6.0

PROCESSOR_ARCH: AMD64
JAVA_HOME: E:\Software\inbuilder_community_2305_x86_64_build20230703\server\runtime\java\AMD64-win
JAVA_VERSION: 1.8.0_222
IGIX_SERVER_HOME: E:\Software\inbuilder_community_2305_x86_64_build20230703\server
SPRING_BOOT: 2.4.13 (v2.4.13)
ACTIVE_PROFILES: prod
PROCESS_ID: 10436
TotalMemorySize ${TotalMemorySize}

2023-12-16 11:17:07.761 All rest endpoints have already been registered. Now running on: 5200
2023-12-16 11:17:09.278 The igIX server has started successfully in 107.637 seconds, Now listening on:5200 (http)

```

进入管理员界面，创建组织

编号	123456
名称	cwei
父级	cwei
类型	无
单位	无
地址	无

创建功能组

分组导航	含下级
所有列	名称
编号	666666
名称	开发者联盟
父级	无
功能权限	无
创建者	系统管理员
描述	996打工仔

分配功能权限

名称	AppOrder
GCloud	GCloud
SysPub	SysPub
Bcp	Bcp
BPMIS	BPMIS
HIP	HIP
Dev	Dev
Epp	Epp
Cms	Cms

新建岗位，设置功能组，设置用户和可管理范围

岗位信息

• 编号: dev
• 分组: 开发
• 名称: 开发岗
• 岗位类型: 组织岗
• 组织: cwei
• 描述:

授权功能组 授权业务组织 授权用户 可管理范围

序号	功能组编号	功能组名称
1	allfunc	全部功能权限
2	666666	开发岗权限

数据权限自动继承业务组织权限

授权功能组 授权业务组织 授权用户 可管理范围

可管理用户 可管理岗位 用户可授权岗位

关系	字段	运算符	取值方式	值
或	角色权限	等于	不变	用户所在组织和所有下级

新增管理员用户，授权给之前创造的岗位

用户信息

• 编号: admin02
• 分组: cwei
• 姓名: 管理员02
• 岗位: 开发
• 类型: 员工
• 称谓: 先生
• 职级: 高级
• 描述:

认证信息 个人信息 组织岗位 通用岗位 签名证书

认证方式: 用户名密码认证
• 密码:
必须包含大小写字母、数字、特殊符号，至少8个字符
• 确认密码:
有效期:

数据权限自动继承业务组织权限

授权功能组 授权业务组织 授权用户 可管理范围

序号	用户名	用户名
1	admin02	管理员02

新建岗位，设置功能组，授权用户

岗位信息

• 编号: devdev
• 分组: 开发
• 名称: 开发客户岗
• 岗位类型: 组织岗
• 组织: cwei
• 描述:

数据权限自动继承业务组织权限

授权功能组 授权业务组织 授权用户

序号	功能组编号	功能组名称
1	allfunc	全部功能权限

数据权限自动继承业务组织权限

授权功能组 授权业务组织 授权用户

序号	用户名	用户名
1	admin02	管理员02

新建业务用户并授权岗位

The screenshot shows a user creation form with fields for basic information, authentication, and roles. In the '组织岗位' tab, a new role 'devdev' is assigned to the user.

序号	岗位名称	岗位简称
1	devdev	开发客户端

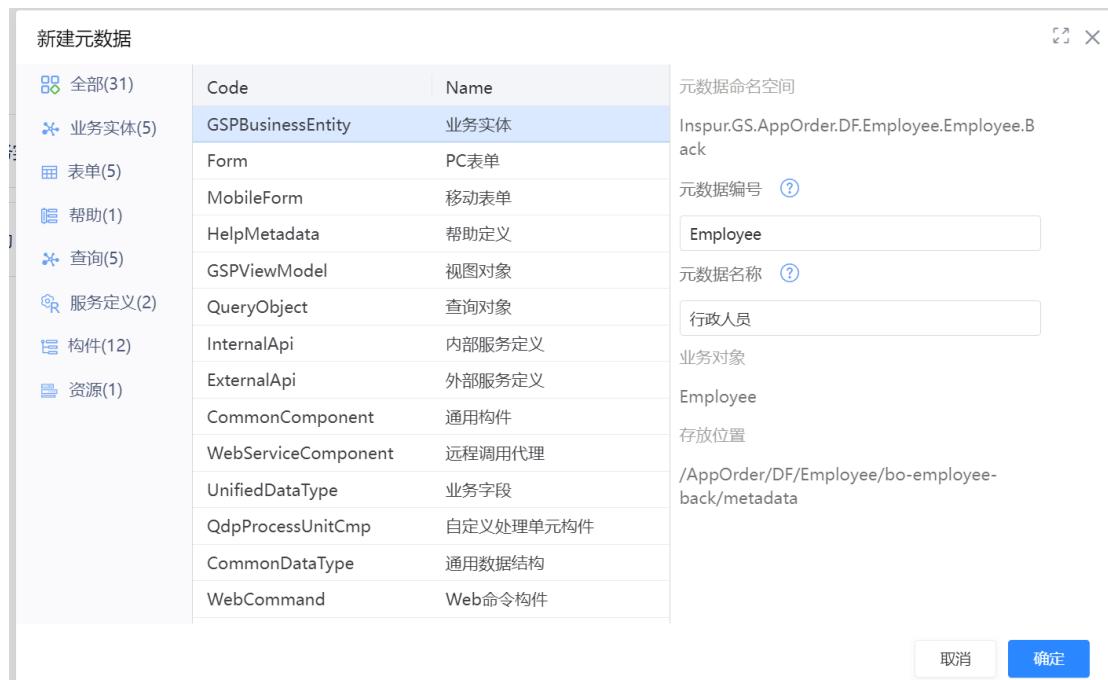
进入工作空间

The workbench has tabs for '开始' (Start) and '我的应用' (My Applications). It features a 'Hi 系统管理员, 欢迎一起来了解inBuilder, 加速企业数字化升级!' message. Below it are four cards: '初识inBuilder' (去完成), '了解inBuilder' (去完成), '创建第一个应用' (去完成), and '访问应用' (去完成).

关闭离线模式

The settings menu on the left includes '工作区' (Workspace), '元数据包管理' (Metadata Package Management), 'Maven配置' (Maven Configuration), '索引' (Index), '环境' (Environment), '源代码管理' (Source Code Management), 'Npm包管理' (Npm Package Management), and '开发过程与工具' (Development Process and Tools). The 'Npm包管理' section is currently selected. A modal dialog for '离线模式' (Offline Mode) is open, with the '开启' (Enable) checkbox unchecked. The '环境选择' (Environment Selection) dropdown is set to 'npm-gsp'. The 'npm-gsp' environment configuration shows an address of 'https://repos.iec.io/repository/npm-gsp/' and an update strategy of '编译前' (Before Compilation). Buttons at the bottom of the dialog are '取消' (Cancel), '应用' (Apply), and '确定' (Confirm).

创建新页面



导入 CMD，选择字段



序号	名称	数据类型	长度	对象类型
1	ID	文本	36	无
2	Version	日期时间	0	无
3	Code	文本	1000	无
4	Name	文本	1000	无
5	Remark	文本	1000	无
6	State	文本	22	无
7	Organization	文本	36	无
8	Gender	文本	22	无
9	CountryOrArea	文本	22	无
10	Location	文本	22	无
11	PostID	文本	36	无
12	Category	文本	36	无
13	DisplayName	文本	36	无
14	Telephone	文本	20	无
15	Secret	文本	10	无
16	Mailbox	文本	36	无

增加子对象表

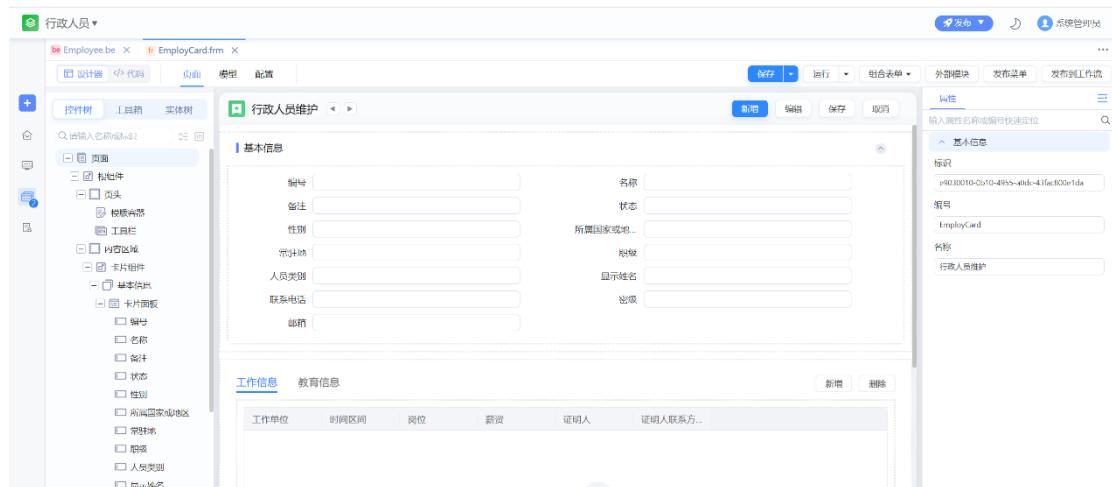
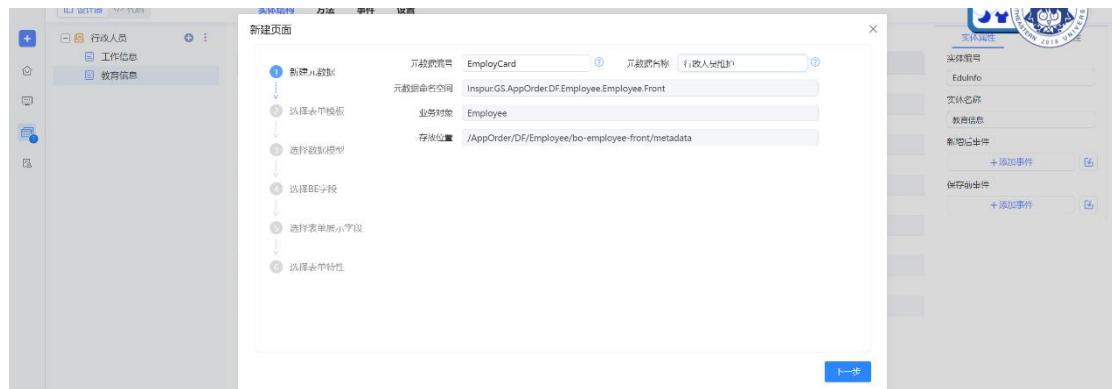
The screenshots illustrate the configuration of three entities in a low-code platform:

- Employee Entity Configuration:** Shows the entity structure with fields like序号 (ID), 版号 (Version), 编号 (Code), Name, Remark, State, Organization, Gender, CountryOrArea, Location, PostID, Category, DisplayName, and Telephone. A modal window titled "子级对象信息" (Sub-object Information) is open, showing two fields: *编号 JobInfo and *名称 工作信息.
- JobInfo Entity Configuration:** Shows the entity structure with fields like序号 (ID), 版号 (ParentID), Company, TimeInterval, Position, Salary, Certifier, and Telephone. Fields 3, 4, 5, 6, 7, and 8 are checked.
- EduInfo Entity Configuration:** Shows the entity structure with fields like序号 (ID), 版号 (ParentID), EmployeeID, SchoolName, TimeInterval, StudyType, Major, Education, EduSystem, Degree, IsFirstDegree, and IsHighest. Fields 3, 4, 5, 6, 7, 8, 9, 10, and 11 are checked.

保存发布

A success message "发布成功" (Published successfully) is displayed in a central modal window. The background shows the entity configuration interface for the Employee entity, with the "教育信息" (Education Information) section selected. The right panel displays the entity structure for the EduInfo entity.

新建页面

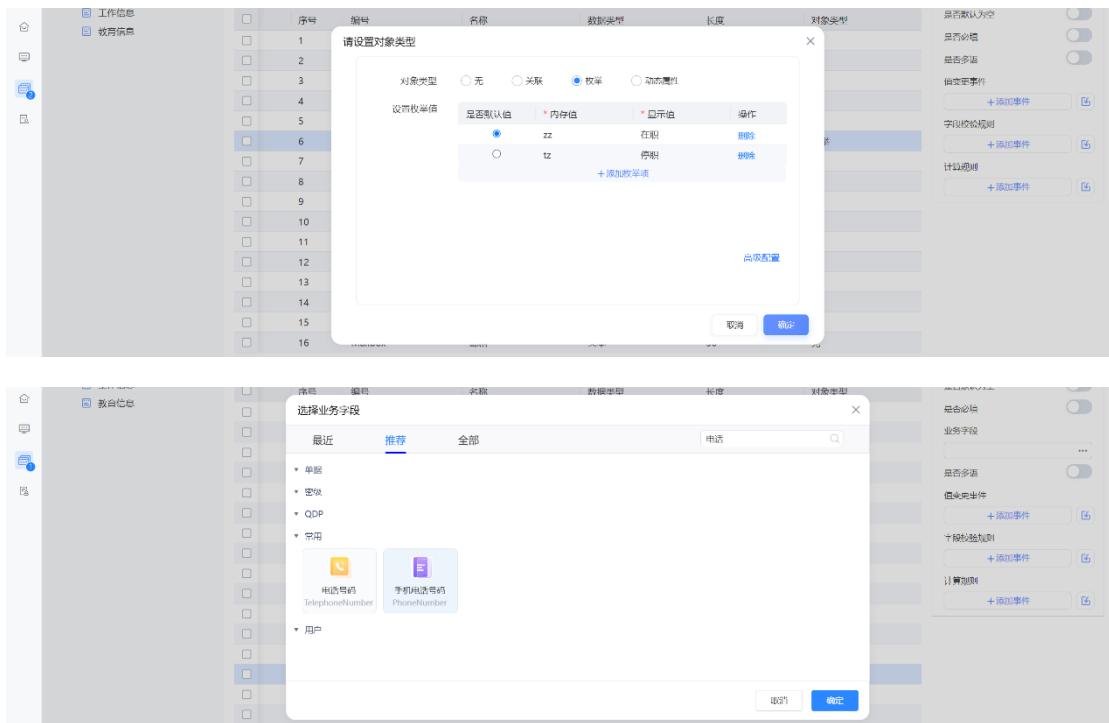


创建新功能

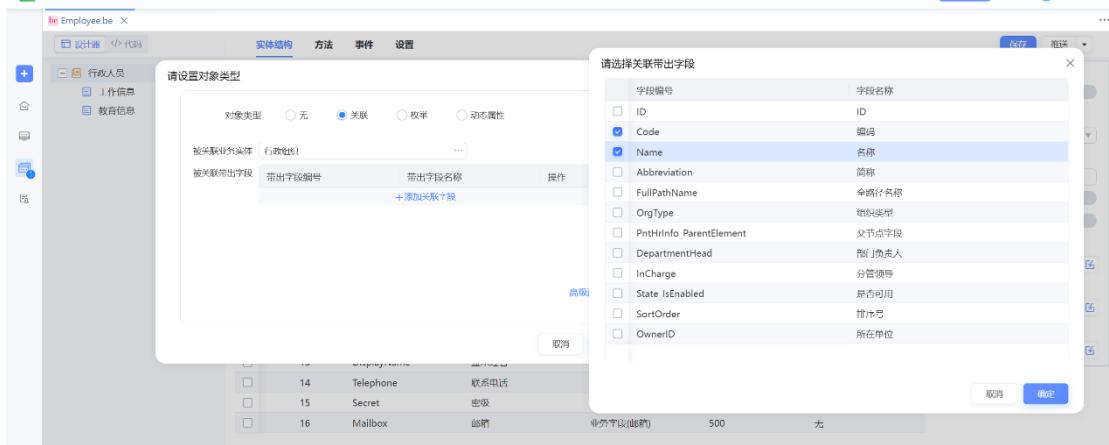
权限分配

名称	编号
GSCloud	GSCloud
系统公共	SysPub
控制平台	Bcp
流程平台	BPMMS
集成平台	HIP
开发平台	Dev
企业门户	Epp
内容管理	Cms
功能实验	AppOrder
基础数据	DF
基础数据	OrganizationGroup
人事管理	PersonnelManagementGroup
行政人员	Employee
订单基础数据	OrderBaseData
订单管理	Order

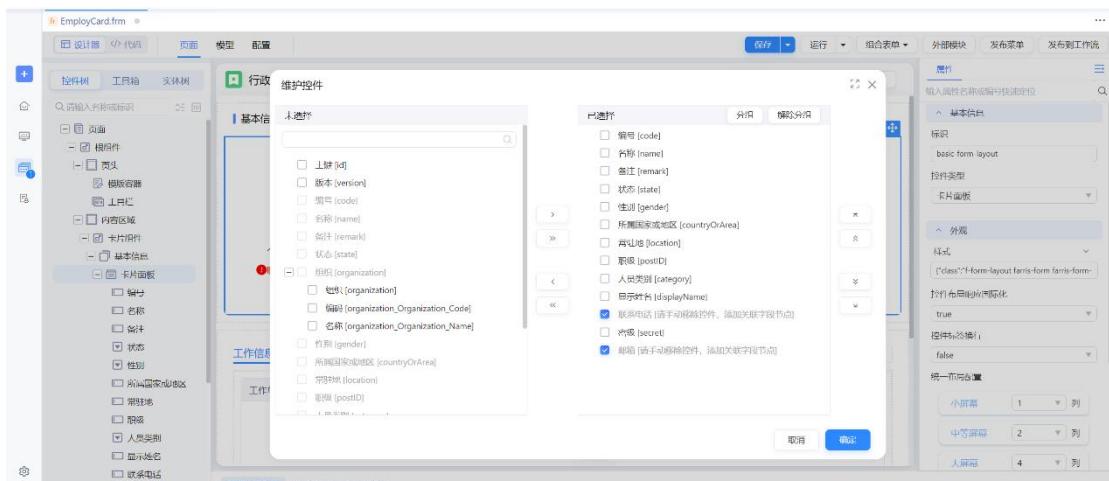
设置字段为必填，枚举类型等

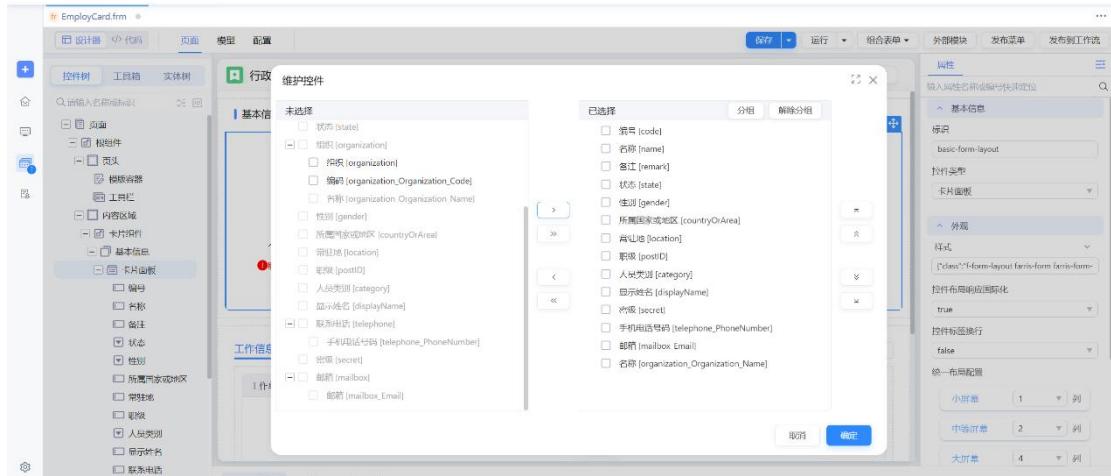


添加关联字段

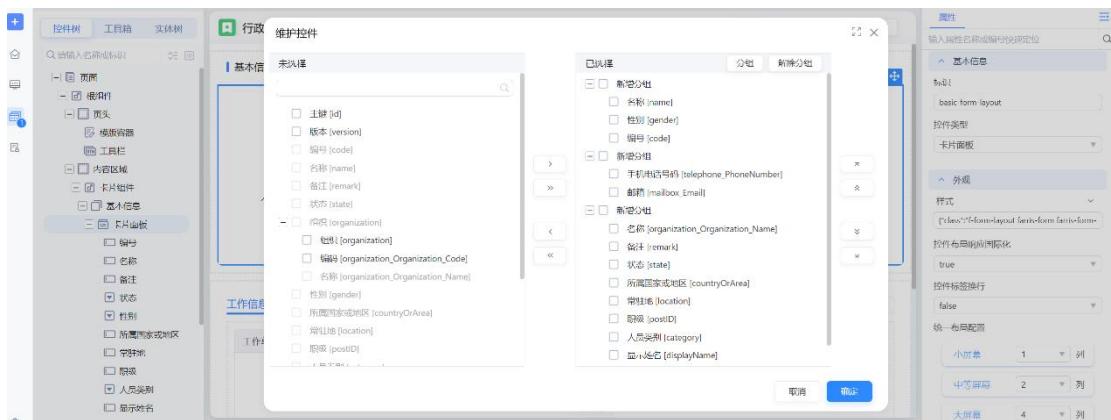


更新表单移除灰色字段，再更新





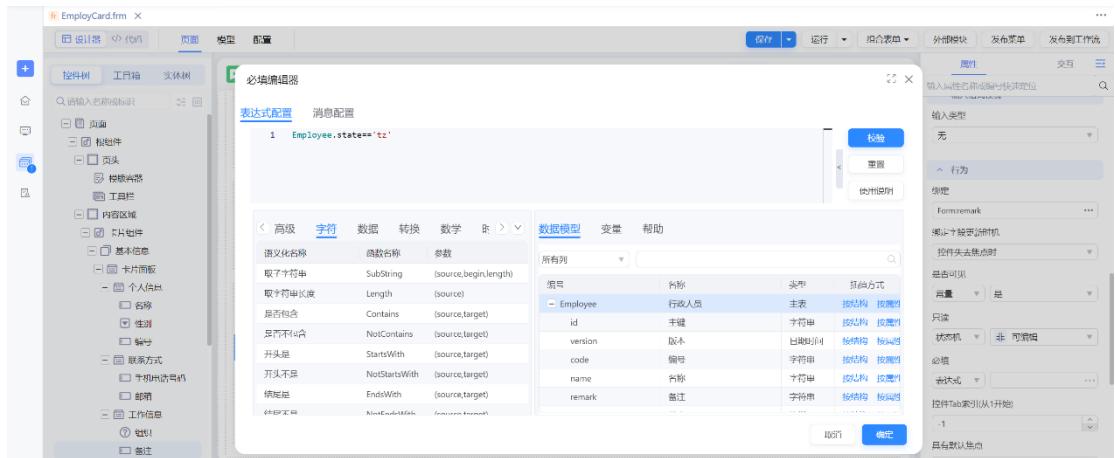
分组完成



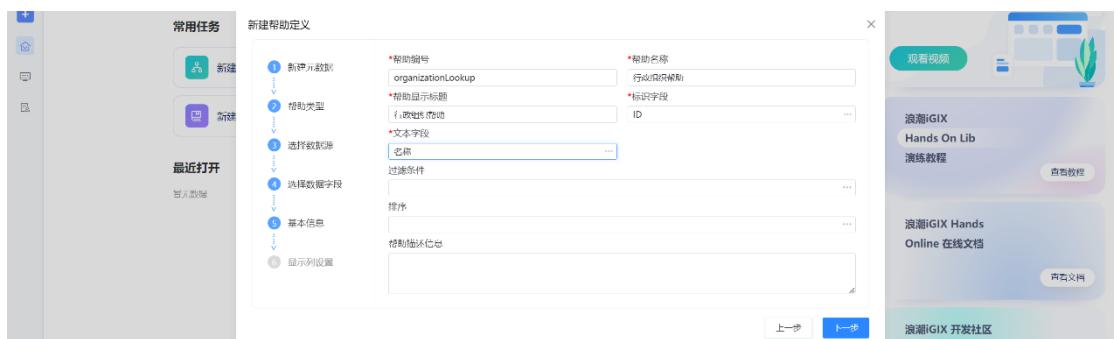
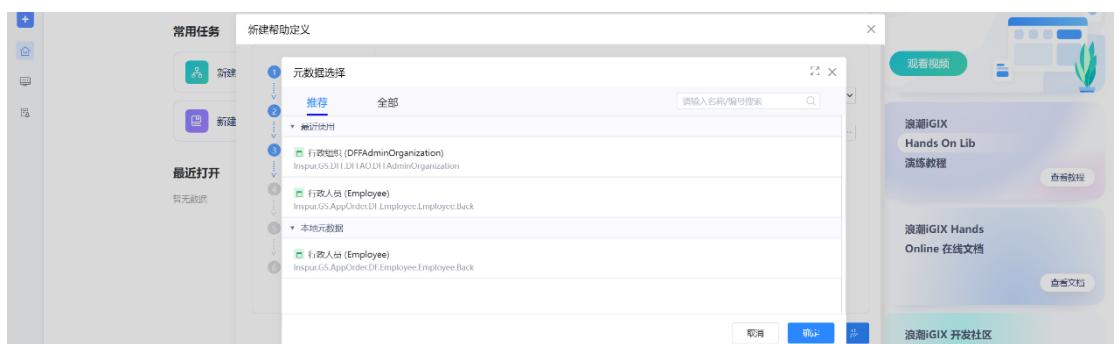
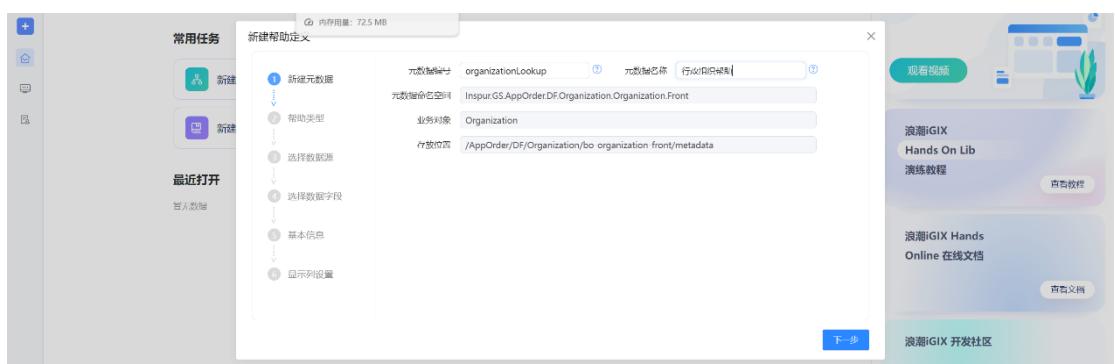
设置布局页面



必填表达式设置

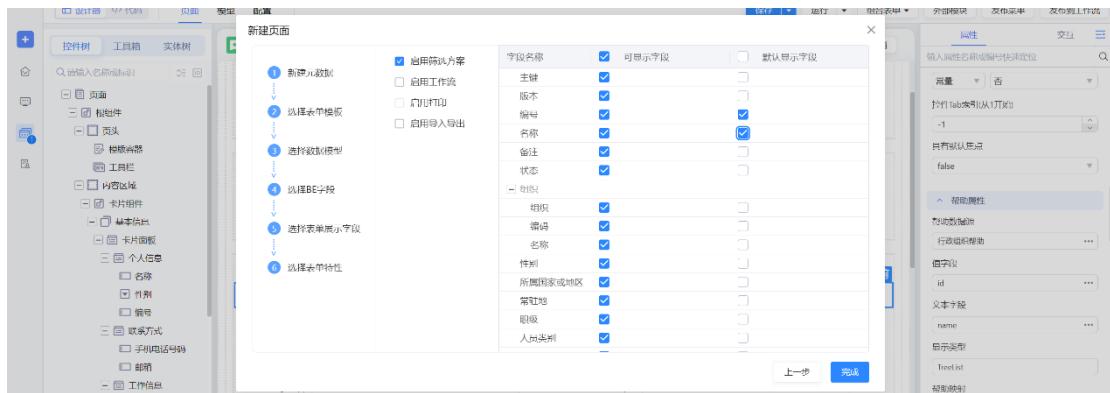


建立新帮助



The screenshot displays four main windows of the Leagoo GIX Low-Code Platform:

- New Help Definition:** A wizard for defining new help definitions. It shows a list of steps (1-6) and a search bar for 'Input Field Name or Path'. A table lists fields like ID[Id], Code[Code], and Name[Name]. On the right, there's a preview of a help card with sections like '观看视频' (Watch Video), '浪潮GIX Hands Online 在线文档' (浪潮GIX Hands Online Document), and '浪潮GIX 开发社区' (Leagoo GIX Development Community).
- Employee Maintenance (映射编辑器 - Mapping Editor):** A window for mapping data fields between source and target entities. It shows a table with rows: '参数字段' (Parameter Field) and '目标字段' (Target Field). For 'id', it maps to 'organization.organization'. For 'name', it maps to 'organization.organization_Name'.
- Configuration Management List Interface (配置管理列表界面 - Configuration Management List Interface):** A window for creating a new page. It shows a step-by-step wizard with the following steps:
 - 1. 新建元数据 (Create Metadata)
 - 2. 选择表单模板 (Select Form Template)
 - 3. 选择数据模型 (Select Data Model)
 - 4. 选择BE字段 (Select BE Fields)
 - 5. 选择表单展示字段 (Select Form Display Fields)
 - 6. 选择表单特性 (Select Form Properties)
 The right panel shows properties for the 'id' field, including '属性' (Properties) like '输入属性名称或别名' (Input property name or alias) and '帮助属性' (Help Properties) like '帮助链接' (Help Link).
- Employee List (员工列表 - Employee List):** Another window for creating a new page. It follows a similar step-by-step wizard to the configuration management interface. The right panel shows properties for the 'id' field, including '属性' (Properties) like '输入属性名称或别名' (Input property name or alias) and '帮助属性' (Help Properties) like '帮助链接' (Help Link).



新增、编辑、查看按钮功能实现，设置超链接

操作列

启用操作列

启用默认操作列

显示操作列

常量 是

操作列宽度 (px)

120

操作列点击编辑

删除当前数据1

待删除数据的标识 (DATA~/#(data-grid-component))

删除后回调方法

请选择

目标组件

请选择

删除成功提示信息

内置列表界面

The screenshot displays the 'Form Creation Wizard' interface for a 'Bill' entity. The process consists of five steps:

- 选择表单模板 (Select Form Template)
- 选择数据模型 (Select Data Model)
- 选择BE字段 (Select BE Fields)
- 选择表单展示字段 (Select Form Display Fields)
- 选择表单特性 (Select Form Properties)

Step 3: 选择BE字段 (Select BE Fields)

字段名称	字段编号	字段标签	字段类型
ID	ID	ID	String
Version	Version	Version	DateTime
状态	BillStatus	BillStatus	Status
流程实例	ProcessInstance	ProcessInstance	Process Instance
报销人员	EmployeeID	EmployeeID	String
编号	Code	EmployeeID_C...	String
名称	Name	EmployeeID_N...	String
所属部门	DomainID	DomainID	String
编号	code	DomainID_code	String
名称	name	DomainID_name	String

Step 4: 选择表单展示字段 (Select Form Display Fields)

字段名称	绑定字段	字段类型	显示
ID	id	字符串	<input type="checkbox"/>
Version	version	日期时间	<input type="checkbox"/>
状态	billStatus_BillState	枚举	<input checked="" type="checkbox"/>
流程实例	processInstance_ProcessInst...	字符串	<input checked="" type="checkbox"/>
报销人员	employeeID	字符串	<input type="checkbox"/>
编号	employeeID_EmployeeID_Co...	字符串	<input type="checkbox"/>
名称	employeeID_EmployeeID_Na...	字符串	<input checked="" type="checkbox"/>

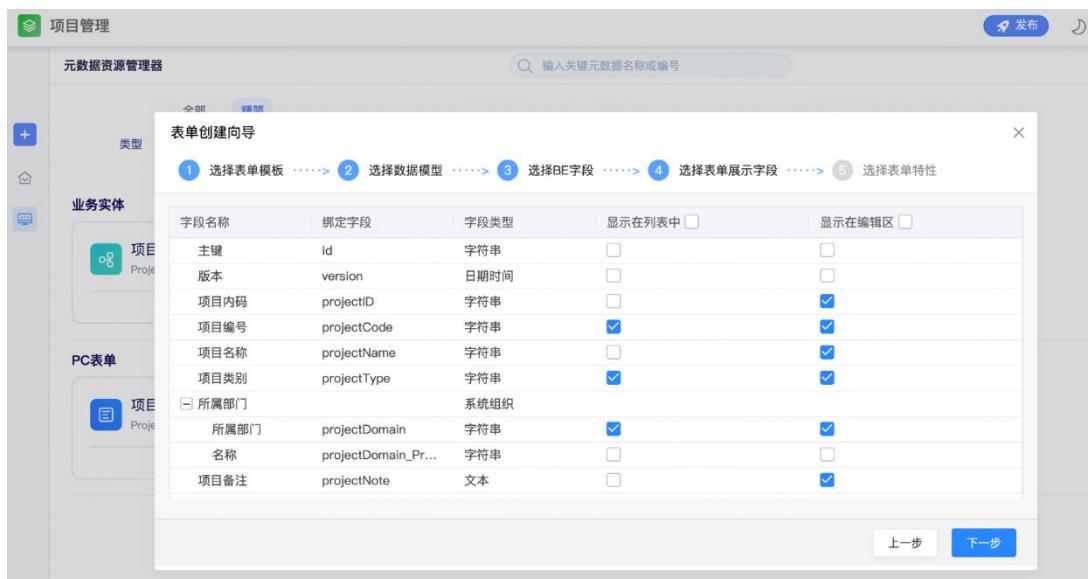
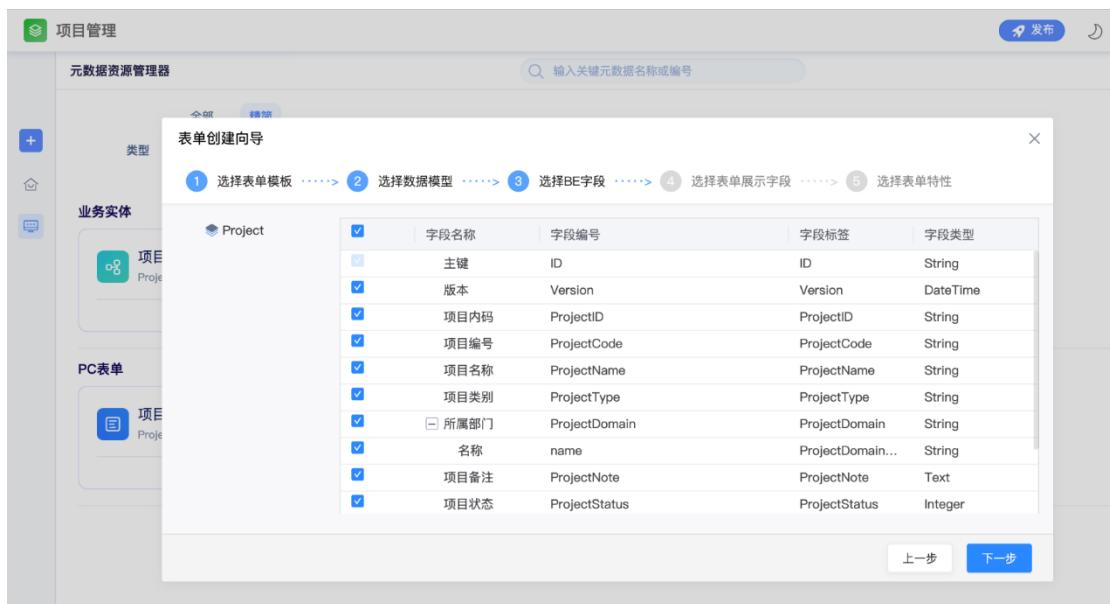
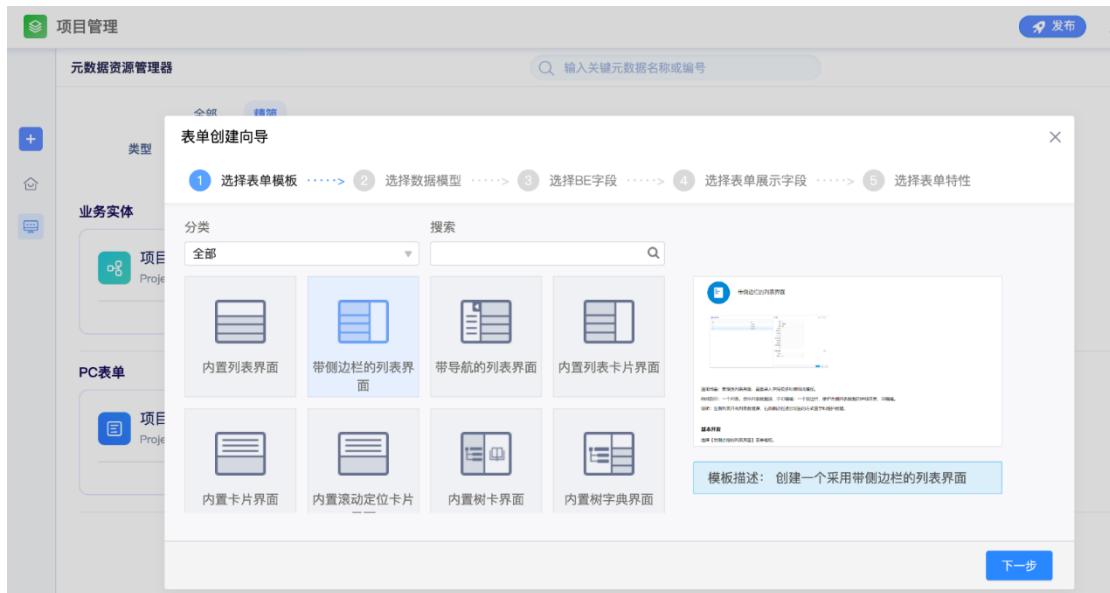
筛选方案字段编辑器

The screenshot shows the 'Filter Scheme Field Editor' interface. It has two main sections:

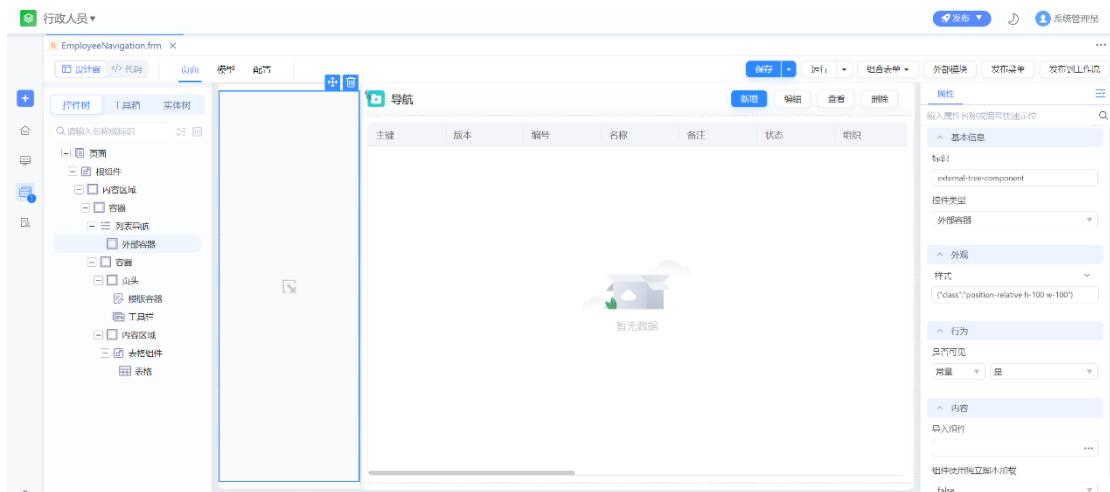
- 未选择 (11 / 14)**: A list of fields with checkboxes:
 - ID [id] (checked)
 - Version [version]
 - 状态 [billStatus]
 - 状态 [billStatus_BillState]
 - 流程实例 [processInstance]
 - 流程实例 [processInstance ProcessInstance]
- 已选择 (1)**: A list of selected fields:
 - ID [id]

Between the two sections are three selection buttons: > (move selected to '已选择'), >> (move all to '已选择'), and < (move selected from '已选择' back to '未选择').

带侧边栏的列表界面



带导航的列表界面



批量编辑的列表页面

The screenshot illustrates the creation of a 'Batch Editing List View' (批量编辑列表界面) for a 'Bill' entity (报销单 BXD).

Step 1: Selecting the Form Template

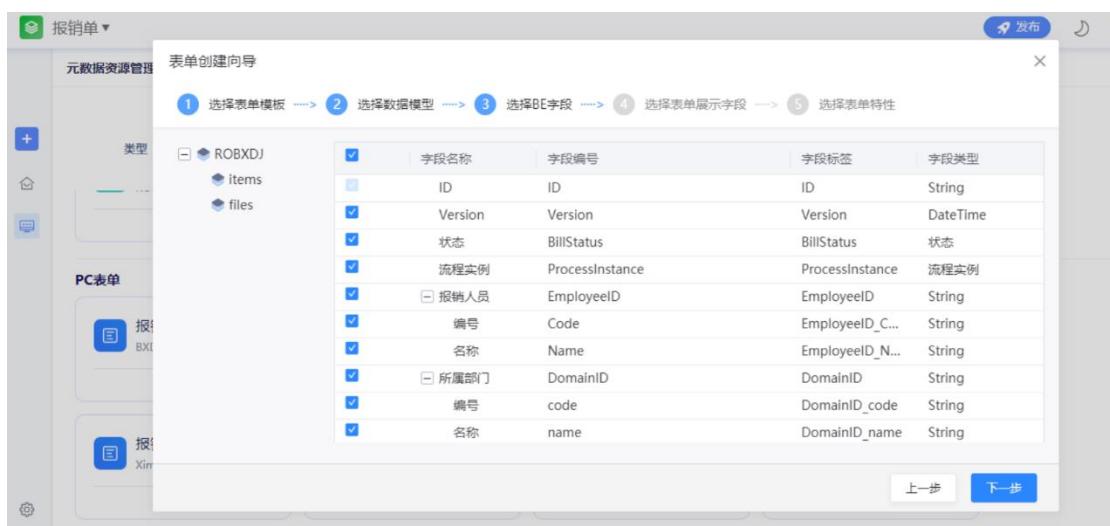
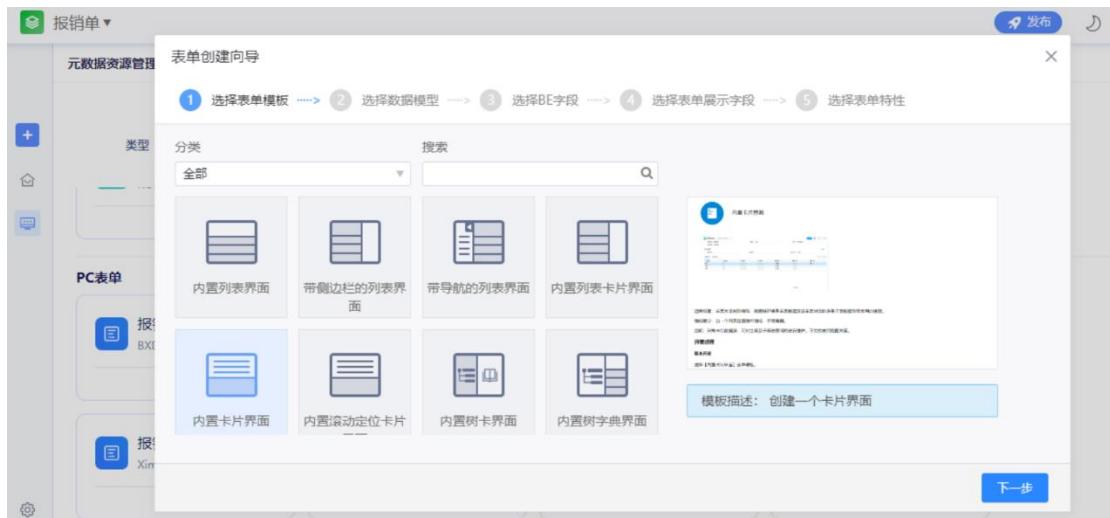
- The 'Form Template' (表单模板) dropdown is set to 'Batch Editing List View' (批量编辑列表界面).
- The 'Template Description' (模板描述) indicates: 'Create a batch editing list interface'.

Step 2: Selecting Data Model

- The 'Data Model' (数据模型) dropdown is set to 'BXD'.
- The table below lists the selected fields and their properties:

字段名称	字段编号	字段标签	字段类型
主键	ID	ID	String
版本	Version	Version	DateTime
报销人员	EmployeeID	EmployeeID	String
所属部门	DomainID	DomainID	String
单据编号	BillCode	BillCode	String
报账金额	TotalSum	TotalSum	Decimal
报销类型	BillType	BillType	String
制单日期	BillDate	BillDate	DateTime
密级ID	SecID	SecID	String
密级	SecLevel	SecLevel	Integer

内置卡片页面



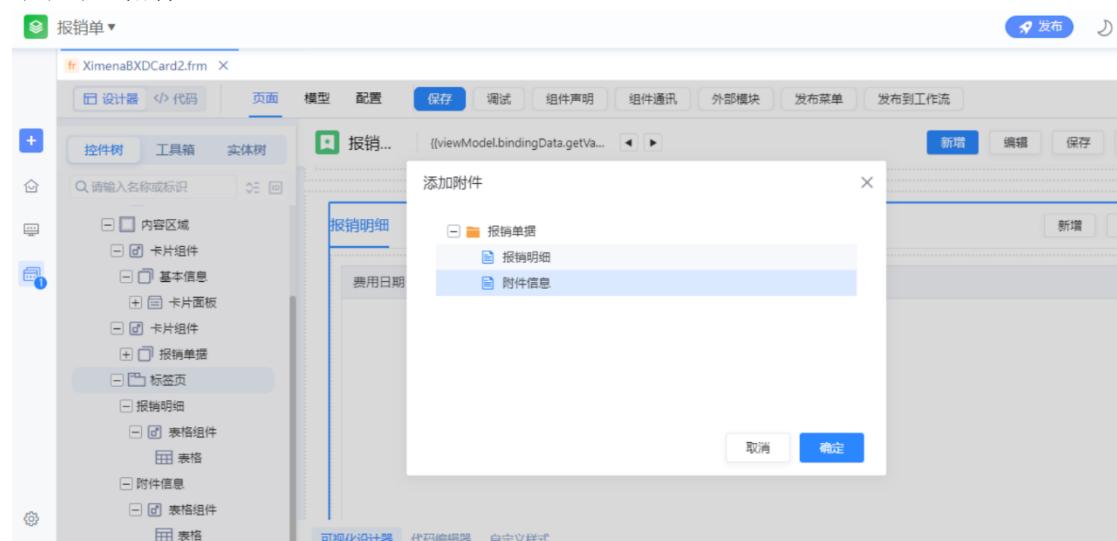
卡片字段分组



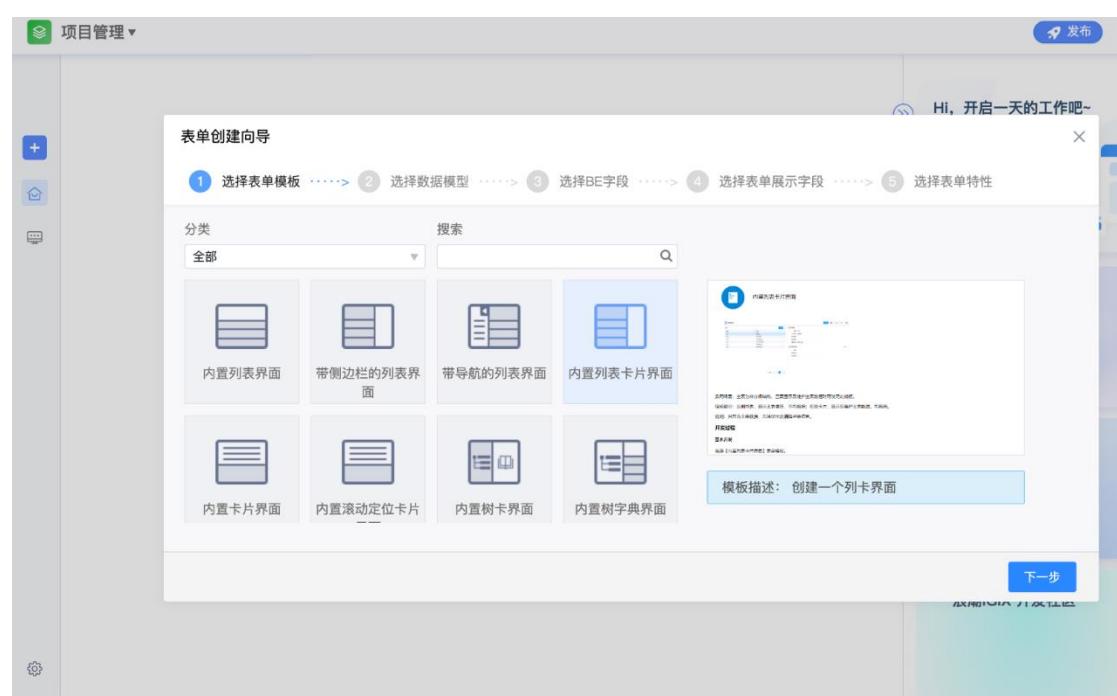
卡片字段拆分组件

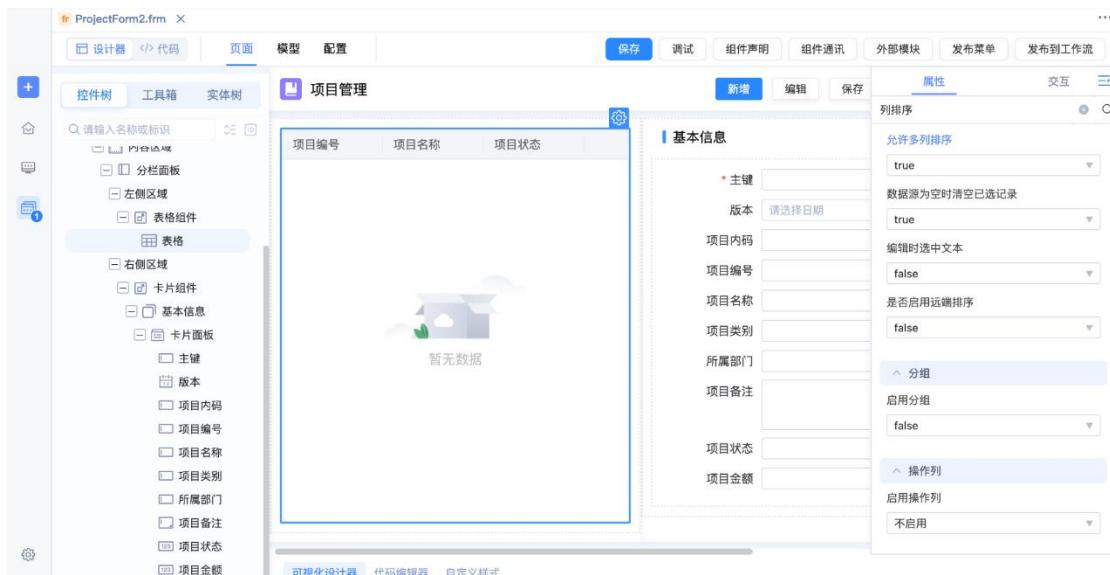


子表添加附件



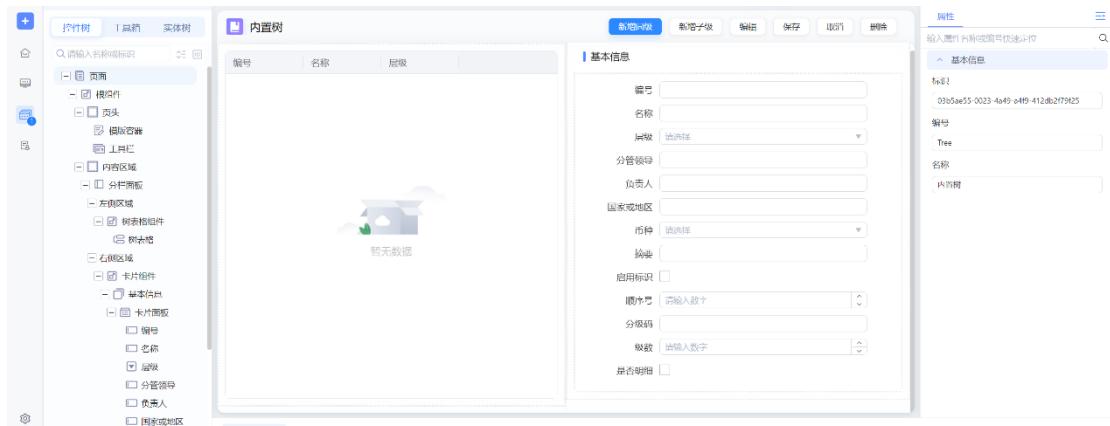
内置卡片列表，启用列排序



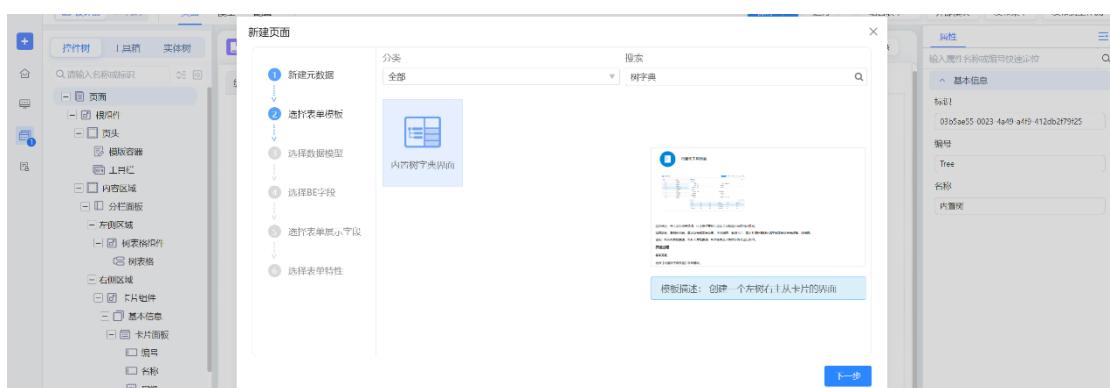


内置树页面

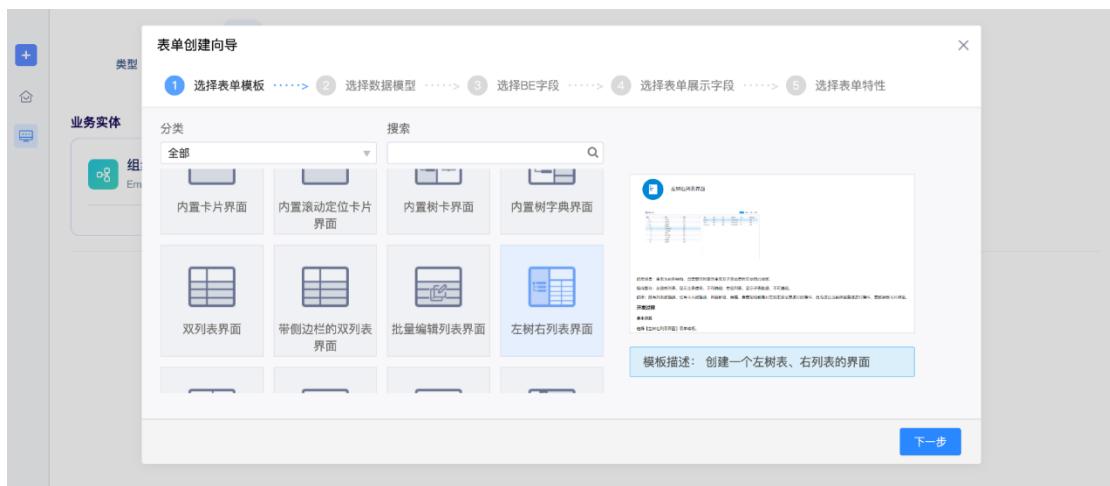


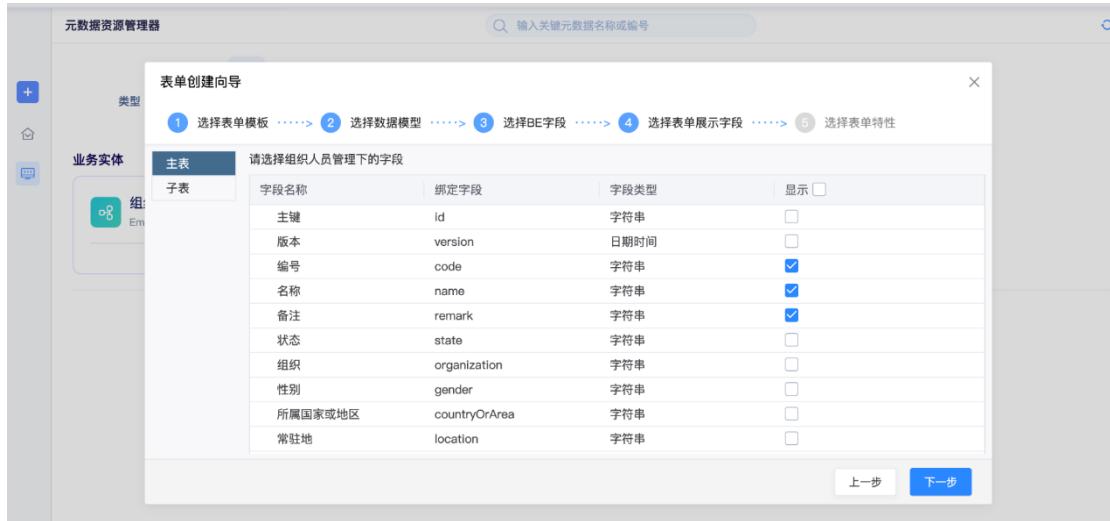


新建内置树字典



创建左树右列表页面





多数据源开发

分类: 全部 搜索:

多数据源列卡界面	向导界面	带侧边栏的向导界面	左导航右卡片的组合界面

模板描述: 创建一个左列表、右主从卡片的界面

字段名称	绑定字段	字段类型	显示在列表中	显示在卡片区
主键	id	字符串	<input type="checkbox"/>	<input type="checkbox"/>
版本	version	日期时间	<input type="checkbox"/>	<input type="checkbox"/>
编号	code	字符串	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
名称	name	字符串	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
备注	remark	字符串	<input type="checkbox"/>	<input checked="" type="checkbox"/>
状态	state	枚举	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
组织	organization	行政组织		
性别	gender	枚举	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
所属国家或地区	countryOrArea	字符串	<input type="checkbox"/>	<input checked="" type="checkbox"/>
常驻地	location	字符串	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

配置表达式:

是否可见	常量	true	计算表达式	...
只读	表达式	{"expr": "DefaultFunction ..."} ...	依赖表达式	...
必填	常量	false	校验表达式	...
			帮助前表达式	...

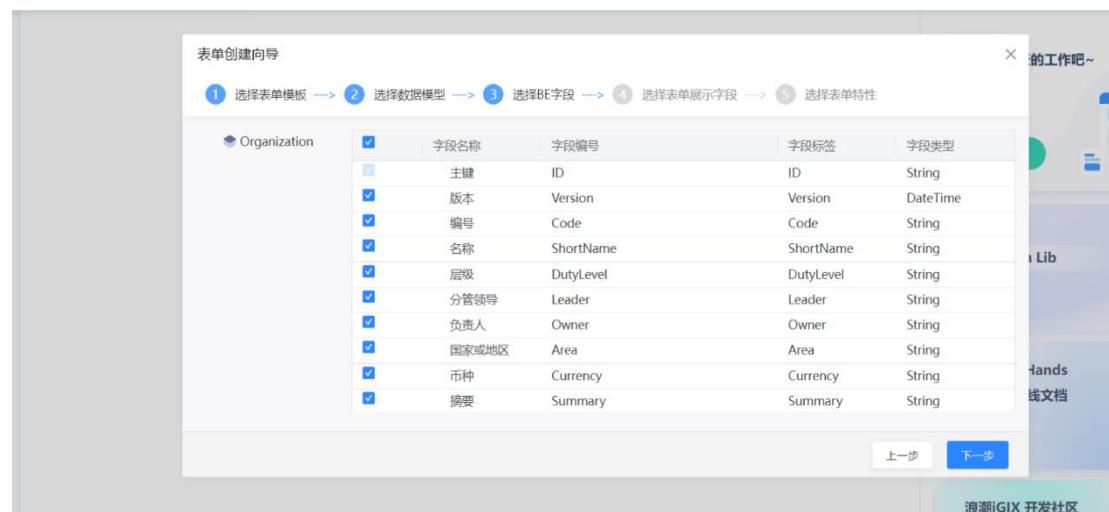
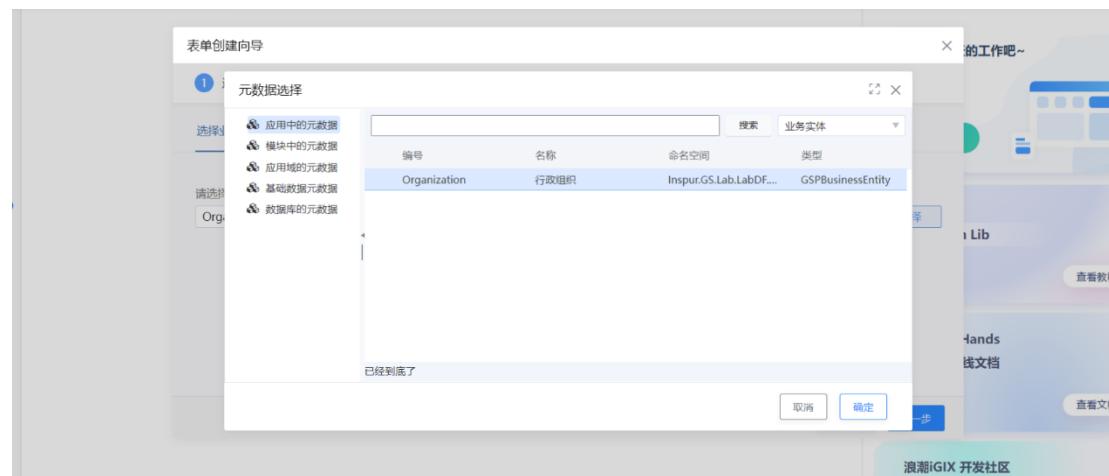
HansOnLab

The screenshot shows a low-code platform interface for managing entity structures. The top window displays the 'Organization.be' entity structure with 13 fields:序号 (ID), 编号 (Version), 名称 (Code), 层级 (ShortName), 层级 (DutyLevel), 分管领导 (Leader), 负责人 (Owner), 国家或地区 (Area), 币种 (Currency), 摘要 (Summary), 启用标识 (Flag), 顺序号 (OrderCode), and 分级码分级信息 (TreeInfo). The bottom window shows the '元数据选择' (Metadata Selection) dialog, which lists various data types under '分级码' (Hierarchical Code). One item is selected: 'PathHierarchyInfo' (编号: PathHierarchyInfo, 名称: 分级码分级信息, 命名空间: Inspur.Gsp.Common..., 类型: UnifiedDataType).

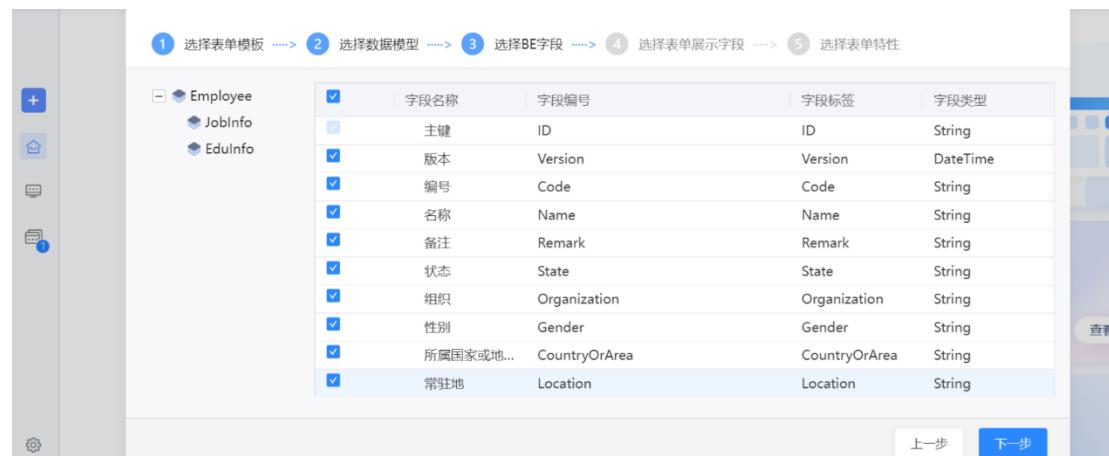
对外接口方法

The screenshot shows the 'Organization.be' entity structure with the '对外接口方法' (External Interface Methods) tab selected. It lists several methods under '分级码分级方式新增同级' (Add Hierarchical Level at Same Level): 'PathHierarchyCreateS...' and 'PathHierarchyCreateC...'. The 'PathHierarchyCreateS...' method is highlighted. A detailed view of this method is shown in a modal dialog, listing its parameters: 'PathHierarchyCreateS...', '分级码分级方式新增同级', 'Inspur.Gsp.Common...', and 'BEMgrComponent'. The dialog also includes a search bar and a '确定' (Confirm) button.

行政组织表单开发



行政人员表单开发

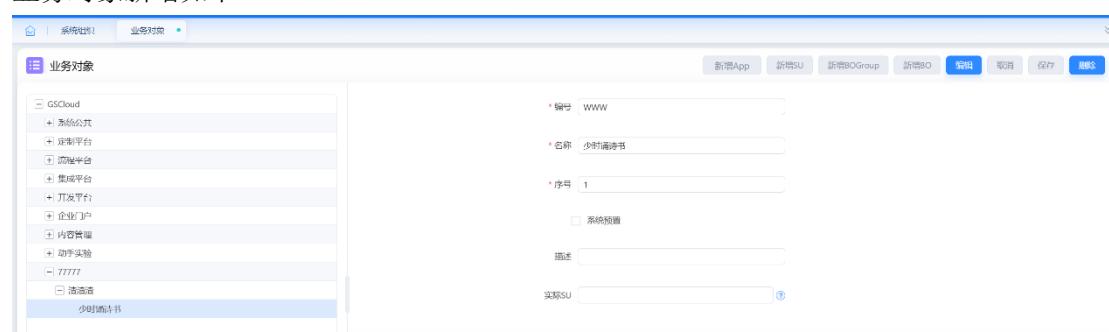




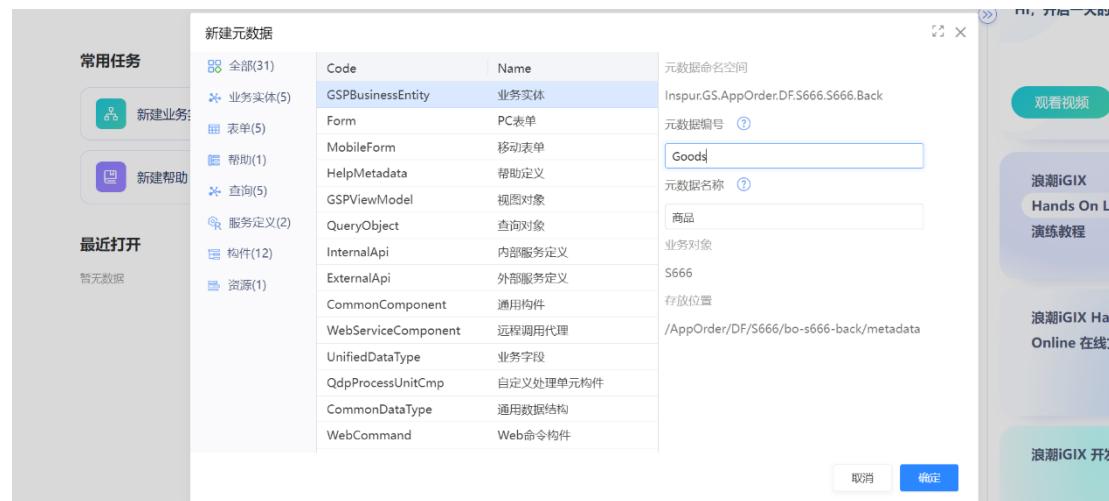
设置各种字段格式等过程略

用户岗位功能组预制过程已经在之前做过，不再赘述

业务对象新增如下



商品字典创建



The screenshot shows two main windows of a low-code development environment:

- Import CDM Fields Window:** A modal dialog titled "导入CDM字段" (Import CDM Fields) is open. It has two steps: 1. 选择CDM文件 ... (Select CDM File) and 2. 字段来源表 (Field Source Table). A file "AppOrder.cdm" is selected in the first step. Buttons include "取消" (Cancel), "下一步" (Next Step), and "保存" (Save).
- Entity Structure Configuration Window:** The main application window shows the "商品" (Goods) entity structure. It includes tabs for "实体结构" (Entity Structure), "方法" (Methods), "事件" (Events), and "设置" (Settings). Under "实体结构", there are buttons for "新增字段" (Add Field), "新增虚拟字段" (Add Virtual Field), "删除字段" (Delete Field), "导入DBO字段" (Import DBO Field), and "导入CDM字段" (Import CDM Field). A table lists fields like ID, Version, GoodsName, etc., with columns for序号 (Index), 编号 (Code), 名称 (Name), 数据类型 (Data Type), 长度 (Length), and 对象类型 (Object Type). On the right, "实体属性" (Entity Properties) and "高级属性" (Advanced Properties) are configured for the "Goods" entity.

商品表单开发

The screenshot shows a "元数据选择" (Metadata Selection) dialog box. It lists various data sources and their details:

	工程名称	编号	名称	类型
bo-goods-back	Goods	商品	GSPBusinessEnti...	

Buttons at the bottom include "取消" (Cancel), "确定" (Confirm), and "一步" (Step One).

商品 表单创建向导

1 选择表单模板 > 2 选择数据模型 > 3 选择BE字段 > 4 选择表单展示字段 > 5 选择表单特性

字段名称	绑定字段	字段类型	显示在列表中 <input type="checkbox"/>	显示在编辑区 <input type="checkbox"/>
主键	id	字符串	<input type="checkbox"/>	<input type="checkbox"/>
版本	version	日期时间	<input type="checkbox"/>	<input type="checkbox"/>
商品名称	goodsName	字符串	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
商品编号	goodsCode	字符串	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
规格型号	specification	字符串	<input type="checkbox"/>	<input checked="" type="checkbox"/>
分类名称	category	字符串	<input type="checkbox"/>	<input checked="" type="checkbox"/>
单价	price	数字	<input type="checkbox"/>	<input checked="" type="checkbox"/>
上架数量	number	数字	<input type="checkbox"/>	<input checked="" type="checkbox"/>
备注	remark	字符串	<input type="checkbox"/>	<input checked="" type="checkbox"/>

上一步 下一步

销售卡片开发

The screenshots illustrate the configuration of a Sales Order entity (SaleOrder) in a low-code platform.

Screenshot 1: Entity Structure (实体结构)

- Entity Properties:**
 - Entity ID: SaleOrder
 - Entity Name: 销售订单
- Events:**
 - Add Event: 新增后事件 (After Create)
 - Query Before Event: 查询前事件 (Query Before)
 - Query After Event: 查询后事件 (Query After)
 - Save Before Event: 保存前事件 (Save Before)
- Fields:**

序号	编号	名称	数据类型	长度	对象类型
1	ID	主键	文本	36	无
2	Version	版本	日期时间	0	无
3	BillStatus	状态	业务字段(状态)	36	枚举
4	ProcessInstance	流程实例	业务字段(流程...)	36	无
5	OrderCode	订单编号	文本	36	无
6	Merchant	商户	文本	36	无
7	OrderTime	下单时间	文本	30	无
8	PayMethod	支付方式	文本	20	无
9	OrderType	订单类型	文本	20	无
10	OrderPerson	下单人	文本	36	无
11	Telephone	联系电话	文本	20	无
12	SendState	发货状态	文本	20	无
13	Remark	备注	文本	100	无
14	TotalPrice	订单金额	浮点数字	10	无
15	TotalDiscounts	折扣优惠	浮点数字	10	无

Screenshot 2: Entity Structure (实体结构) - Sub-object Information

A modal dialog titled "子级对象信息" (Sub-object Information) is displayed, showing the relationship between the Sales Order entity and its sub-object OrderItem. It contains two fields:

- 编号: OrderItem
- 名称: 订单明细

Screenshot 3: Import CDM Fields (导入CDM字段)

This screenshot shows the import process for CDM fields. A modal dialog titled "导入CDM字段" (Import CDM Fields) is open, displaying a list of available fields from a CDM file:

编号	名称
Employee	行政人员
JobInfo	工作信息
AdminOrgan	行政组织
Edulnfo	教育信息
Merchant	商户
Goods	商品
SaleOrder	销售订单
<input checked="" type="checkbox"/> OrderItem	订单明细

The screenshots illustrate the configuration of entity properties in a low-code platform, specifically for the 'SalesOrderDetail' entity.

Screenshot 1: Payment Method Configuration

This screenshot shows the configuration of the 'PaymentMethod' property. The '对象类型' (Object Type) is set to '枚举' (Enumeration). The enum values are:

是否默认值	值	显示值	操作
<input checked="" type="radio"/>	Cash	现金	删除
<input type="radio"/>	Upay	银联	删除
<input type="radio"/>	Wechat	微信	删除
<input type="radio"/>	Alipay	支付宝	删除

Screenshot 2: Delivery Status Configuration

This screenshot shows the configuration of the 'DeliveryStatus' property. The '对象类型' (Object Type) is set to '枚举' (Enumeration). The enum values are:

是否默认值	值	显示值	操作
<input checked="" type="radio"/>	Deliver	发货订单	删除
<input type="radio"/>	Stock	备货订单	删除

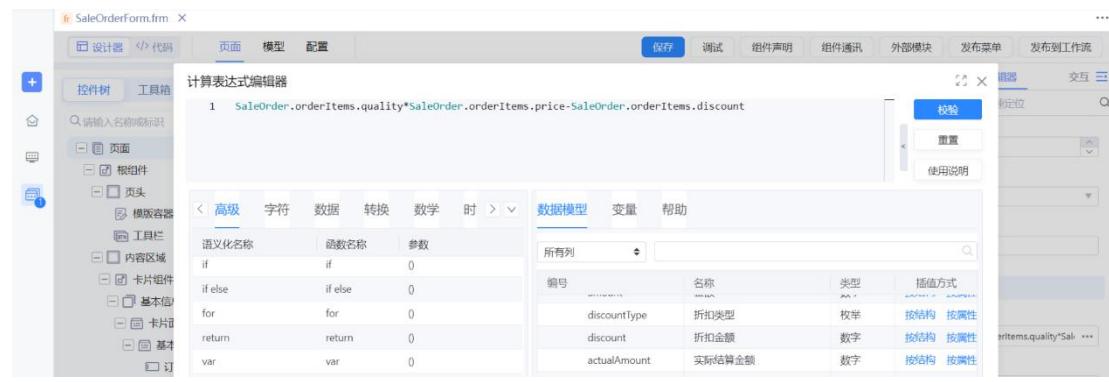
Screenshot 3: Shipping Status Configuration

This screenshot shows the configuration of the 'ShippingStatus' property. The '对象类型' (Object Type) is set to '枚举' (Enumeration). The enum values are:

是否默认值	值	显示值	操作
<input checked="" type="radio"/>	Shipped	已发货	删除
<input type="radio"/>	UnShipped	未发货	删除

The screenshot displays a low-code platform interface for creating a 'Sales Order' form. It consists of three vertically stacked windows:

- Top Window:** '表单创建向导' (Form Creation Wizard) - Step 5: '选择表单特性' (Select Form Properties). This window lists fields from the 'SaleOrder' and 'OrderItem' models, mapping them to specific properties like '字段名称' (Field Name), '字段编号' (Field ID), '字段标签' (Field Label), and '字段类型' (Field Type). Fields include ID, Version, Status, ProcessInstance, OrderCode, Merchant, name, OrderTime, PayMethod, and OrderType.
- Middle Window:** '表单创建向导' (Form Creation Wizard) - Step 5: '选择表单特性' (Select Form Properties). This window shows the same mapping for the 'SaleOrder' model, specifically for the 'OrderItem' relationship.
- Bottom Window:** '销售订单表单' (Sales Order Form) - This is the final form template. It includes sections for '基本信息' (Basic Information) and '订单明细' (Order Details). The '基本信息' section contains fields for Order ID (ORD20), Merchant (折扣店), Date (2023-05-16), Contact (联系人), and Status (发货状态). The '支付信息' (Payment Information) section shows payment method (现金), amount (0.00), discount (0.00), and total payment (0.00). The '订单明细' (Order Details) section is a table with columns: 商品名称 (Product Name), 规格型号 (Specification), 数量 (Quantity), 标准单价 (Standard Price), 金额 (Amount), 折扣类型 (Discount Type), 折扣金额 (Discount Amount), 实际结算金额 (Actual Settlement Amount), and 备注 (Remarks).



销售订单管理列表

字段名称	字段编号	字段标签	字段类型
主键	ID	ID	String
版本	Version	Version	DateTime
状态	BillStatus	BillStatus	状态
流程实例	ProcessInstance	ProcessInstance	流程实例
订单编号	OrderCode	OrderCode	String
商户	Merchant	Merchant	String
名称	name	Merchant_name	String
下单时间	OrderTime	OrderTime	Date
支付方式	PayMethod	PayMethod	String
订单类型	OrderType	OrderType	String



编码规则定义

编码规则定义

编号/名称	序号	编号	名称	所属业务种类	描述	操作
	1	OrderCode	订单编号	销售订单		查看 编辑 删除

左侧树形目录显示了不同业务模块，如系统公共、流程平台、业务公共等，当前选中的是销售订单下的订单管理模块。

新增编码规则

基本信息

唯一标识	667a3ec3-b01e-4bd0-b534-b7598d	编号		名称	
业务种类	销售订单	服务单元		是否连续	否
检查断号格式	否	描述			

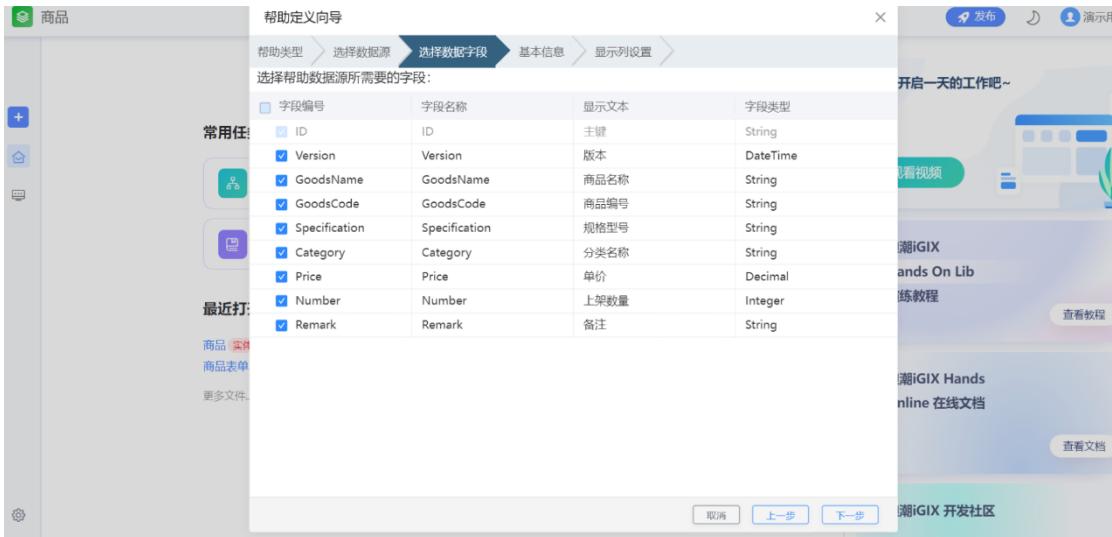
段列表

新增	删除	上移	下移
编号	类型		
No records available.			

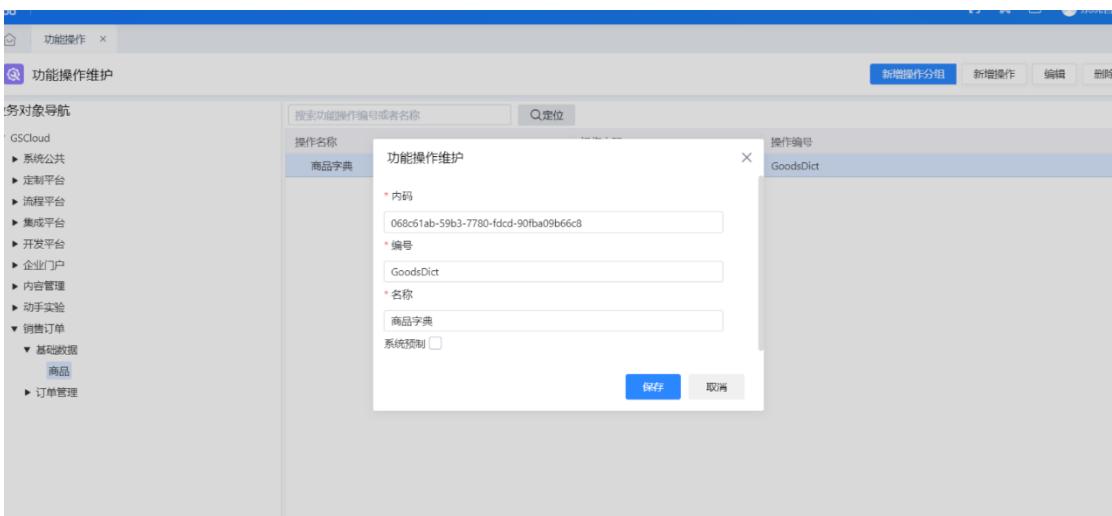
右侧工具栏包含：导入元数据、操作（查看、编辑、删除）、确定、取消。



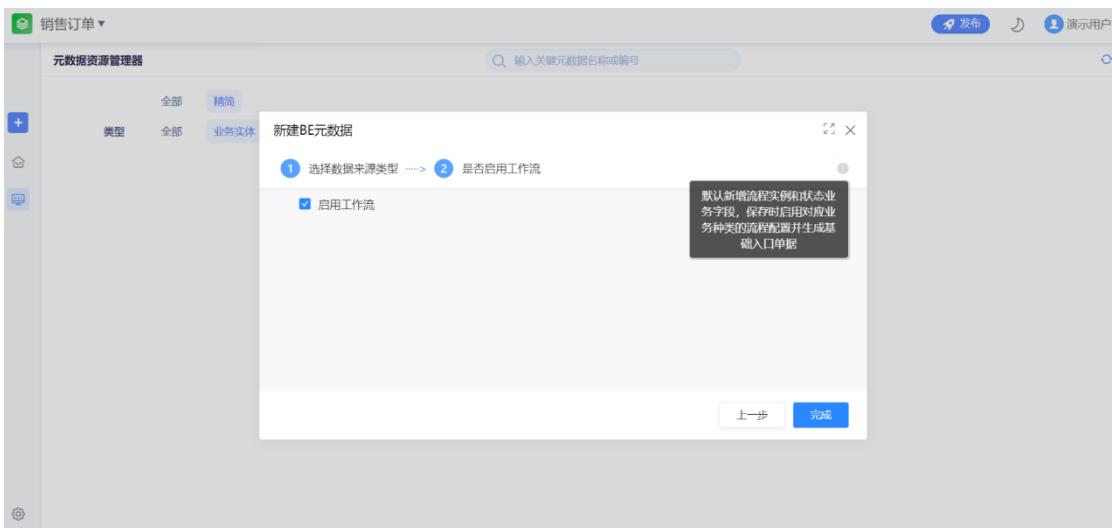
智能帮助开发

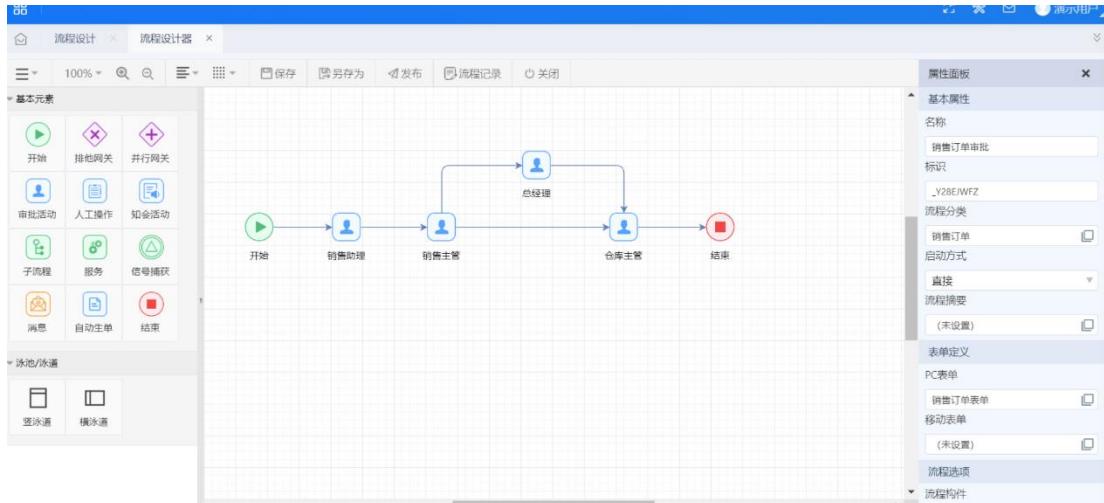


功能菜单发布



工作流配置

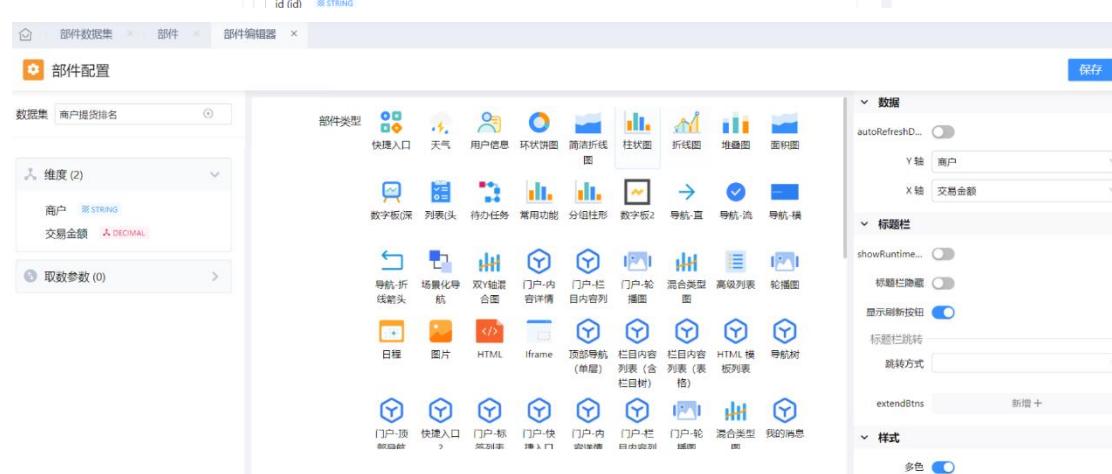
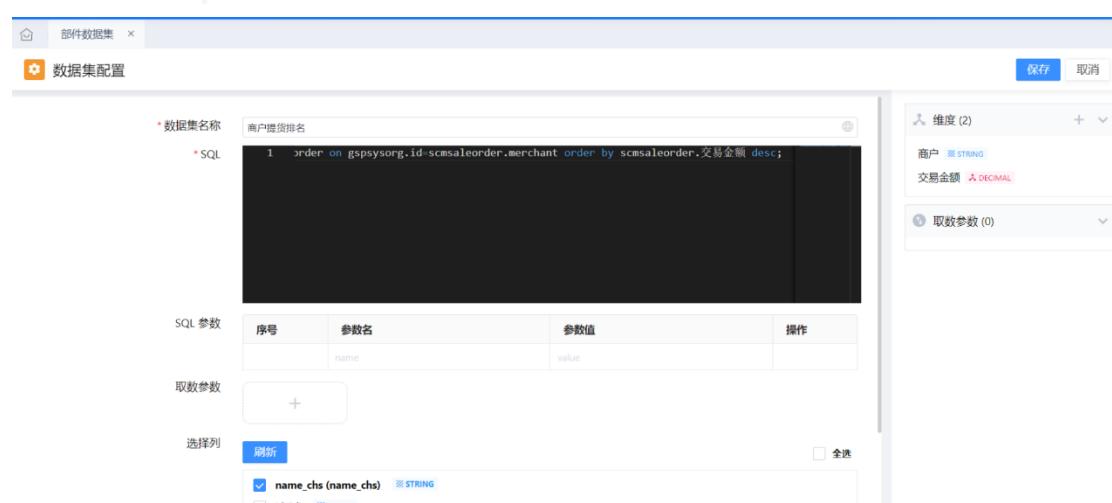
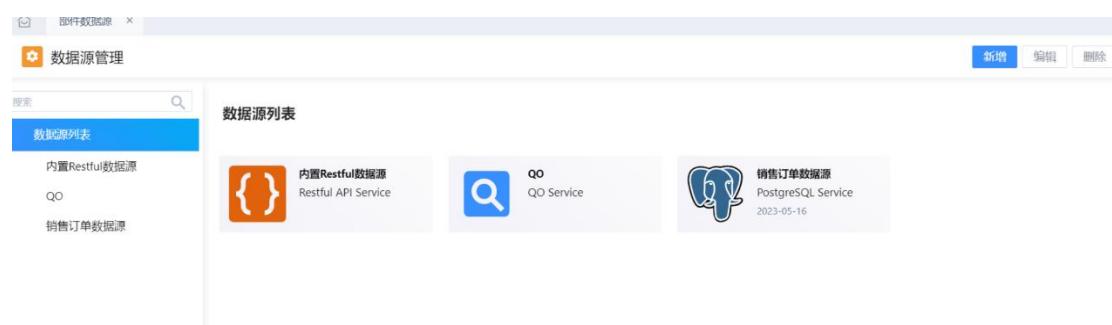
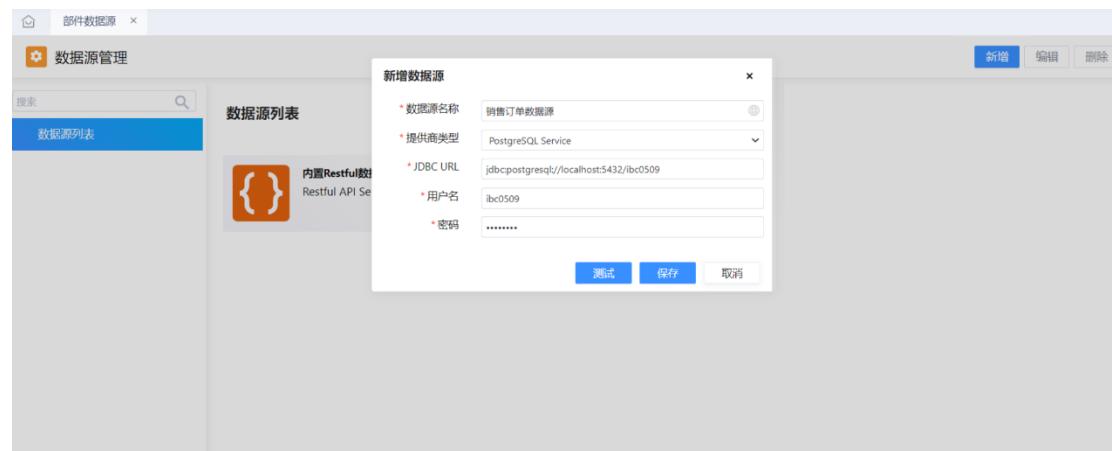




查询配置

The right side shows a preview of the form with two date input fields labeled '下单时间' and '结束时间'.

首页 Dashboard





Farris 模板开发

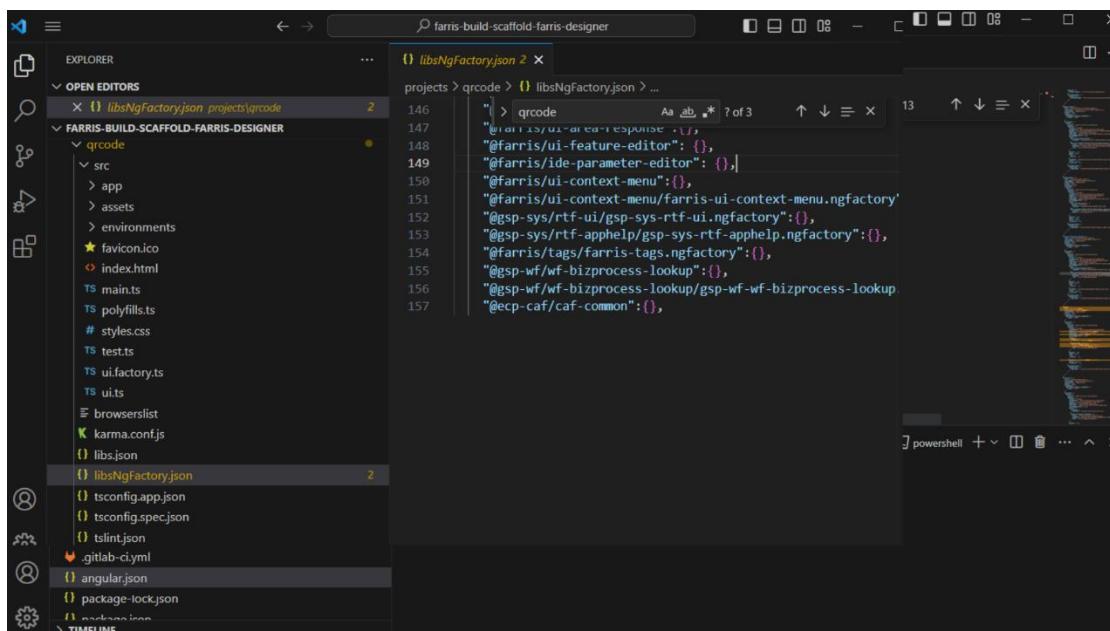
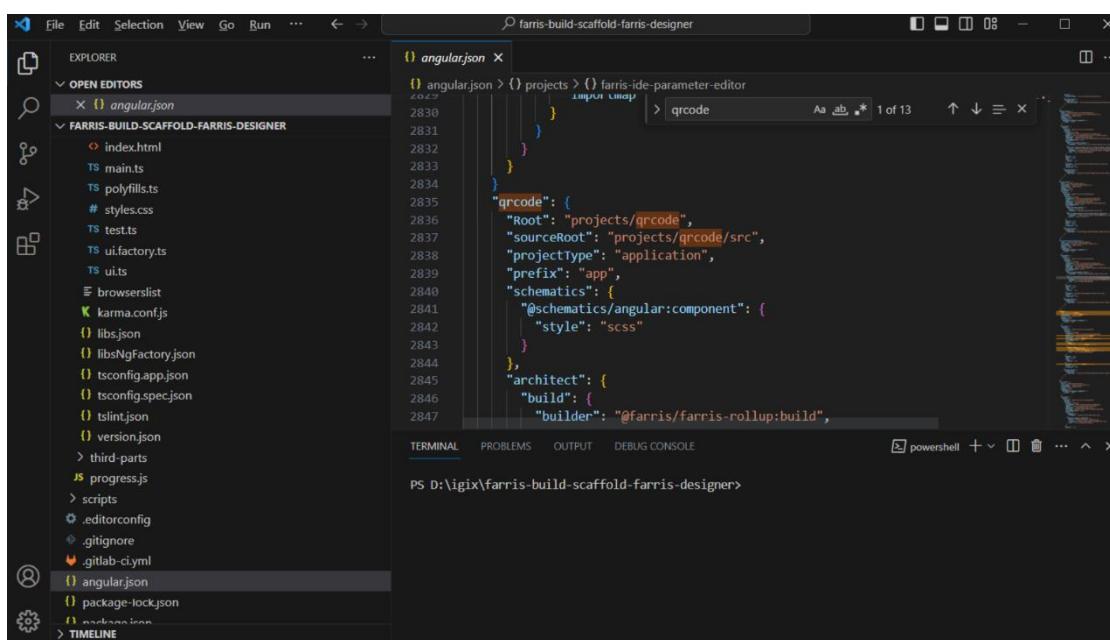
The screenshot displays the Farris template editor. On the left, the '控件树' (Control Tree) shows a tree structure with '外部模块' (External Module) selected. Under it, a '新增' (Add) button is highlighted. The right side shows the configuration for the 'PopoverModule', including fields for '模块名称' (Module Name) and '模块路径' (Module Path). On the far right, there's a detailed configuration panel with tabs like '尺寸' (Size), '显示单元格提示' (Show cell hint), '格式化' (Format), etc. Below these panels is the 'HTML编辑器' (HTML Editor) showing the code for a template named '#contextTmpl'. The code includes logic for 'ngIf' statements based on 'ctxx.rowData['orderType']' values.

```

1 <ng-template #contextTmpl>
2   <div style="padding:20px 10px 26px 20px; font-size: 12px;color: #515151; width:330px">
3     <h4 style="font-weight:600;font-size: 18px;color: rgba(0,0,0,0.8);margin:0 0 14px;">订单类型:</h4>
4     <p style="margin:0 0 12px;"><i class="f-icon f-icon-basic text-success" style="margin-right:10px;"></i><i class="f-icon f-icon-user text-success" style="margin-right:10px;"></i><i class="f-icon f-icon-list-unordered text-success" style="margin-right:10px;"></i><span *ngIf="ctxx.rowData['orderType'] === 'Deliver'">订单类型:发货订单</span>
5     <p style="margin:0 0 12px;"><i class="f-icon f-icon-user text-success" style="margin-right:10px;"></i><i class="f-icon f-icon-list-unordered text-success" style="margin-right:10px;"></i><span *ngIf="ctxx.rowData['orderType'] === 'Stock'">订单类型:备货订单</span>
6   </div>
7 </ng-template>
8 <div class="f-dagrid-cell-content">
9   <span class="mr-5" farrisPopover [placement]="'right'" [triggers]="'hover'" [popover]="#contextTmpl">
10  </span>
11 </div>
12 <div class="f-dagrid-cell-content">
13   <span class="mr-5" farrisPopover [placement]="'right'" [triggers]="'hover'" [popover]="#contextTmpl">
14  </span>

```

Farris 外部组件，修改 qrcode 工程



RPC 远程过程调用服务开发

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

code / src / main / java / com / inspur / gs / scm / orderdf / goods / goods / front / goodsform_frm / voactions / SyntoOrderVOAction.java

```

import com.inspur.edp.bef.api.action.*;
import com.inspur.edp.cefc.entity.entity.IEntityData;
import com.inspur.gs.scm.ordermgr.saleorder.saleorder.front.saleorderlist_frm.OrderRpcService;
import io.iec.edp.caf.commons.utils.SpringBeanUtils;
import io.iec.edp.caf.rpc.client.RpcClassHolder;

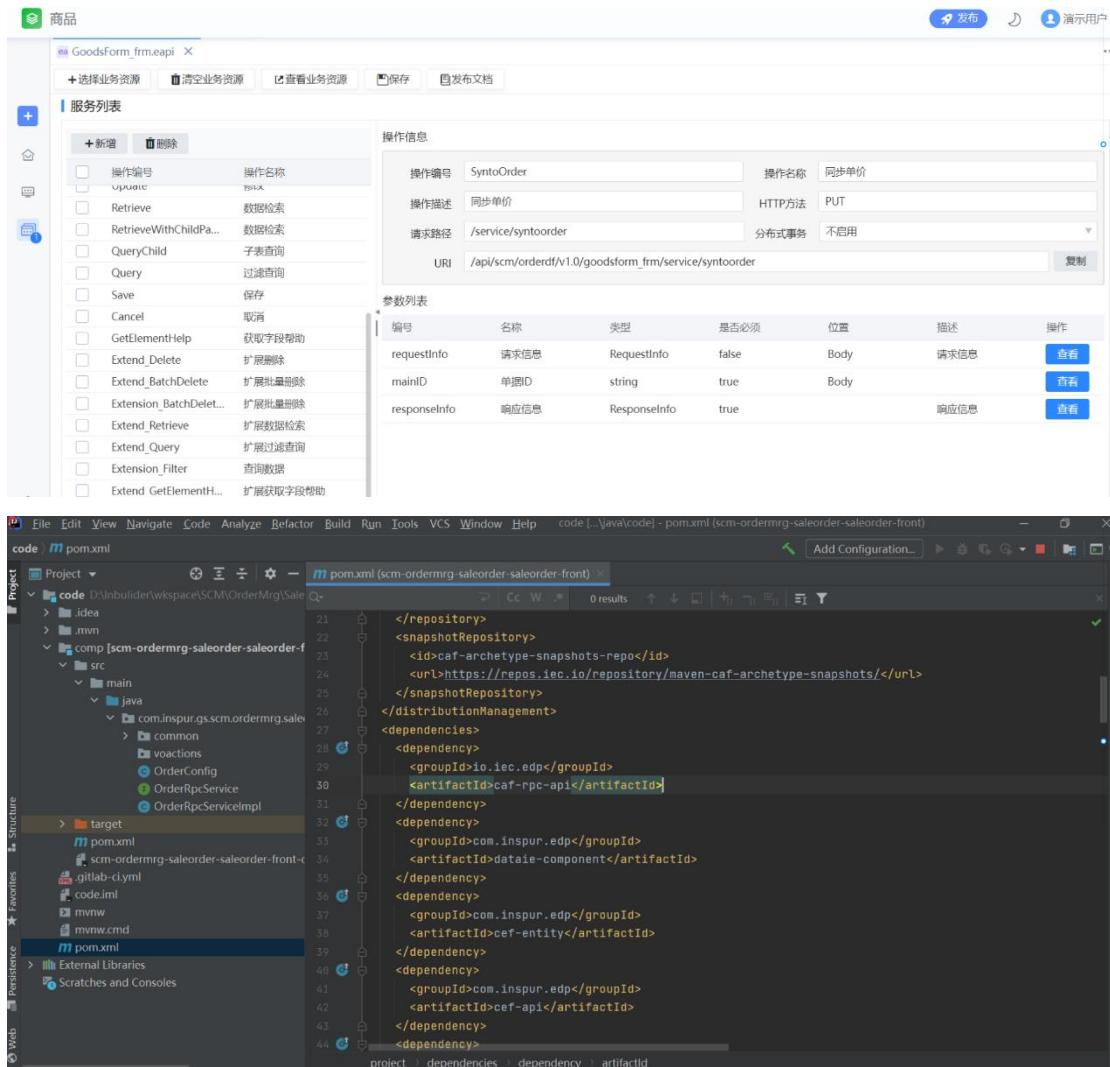
import java.math.BigDecimal;

public class SyntoOrderVOAction extends AbstractFSAAction<String> {

    private RpcClassHolder rpcClassHolder = SpringBeanUtils.getBean(RpcClassHolder.class);
    private String mainID;
    public SyntoOrderVOAction(String mainID) {
        this.mainID = mainID;
    }
    @Override
    public void execute() {
        IStandardLcp meetingLcp = LcpFactoryManagerUtils.createLcp("configId: \"com.inspur.gs.scm.orderdf.goods.goods.front.compo");
        RespectiveRetrieverResult result = meetingLcp.retrieve(mainID);
        IEntityData mainData = result.getData();
        String price = mainData.getValue("Price").toString();
        OrderRpcService orderRpc = rpcClassHolder.getRpcClass(OrderRpcService.class);
        orderRpc.SynGoodPrice(mainID,price);
    }
}

```

操作编号	操作名称
QueryChild	子表查询
Query	过滤查询
Save	保存
Cancel	取消
GetElementHelp	获取字段帮助
Extend_Delete	扩展删除
Extend_BatchDelete	扩展批量删除

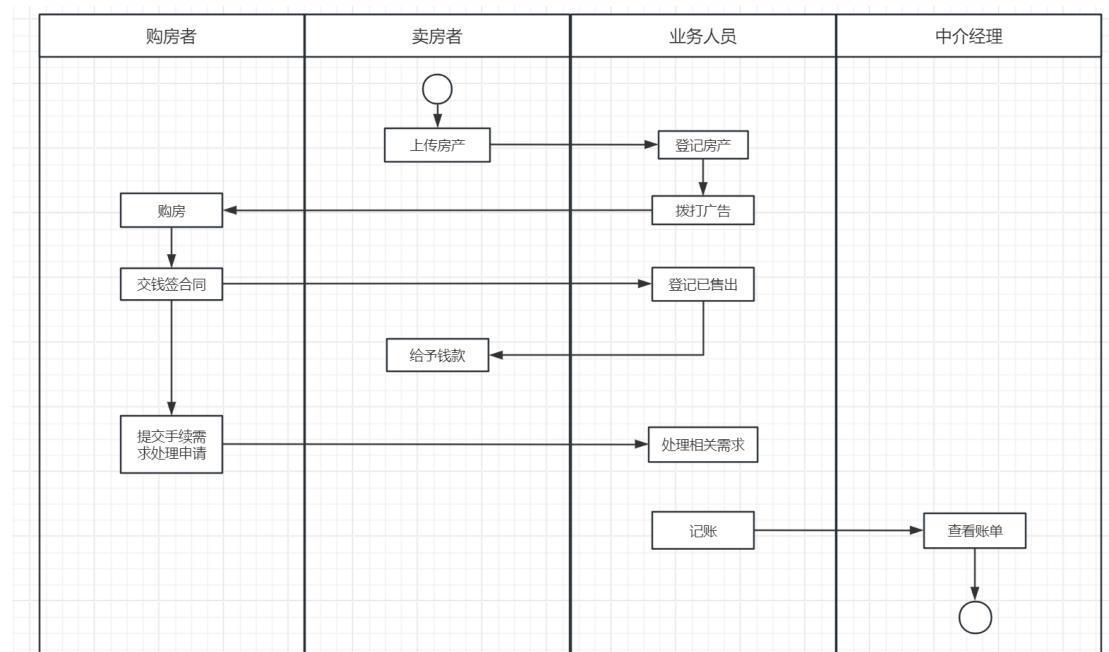
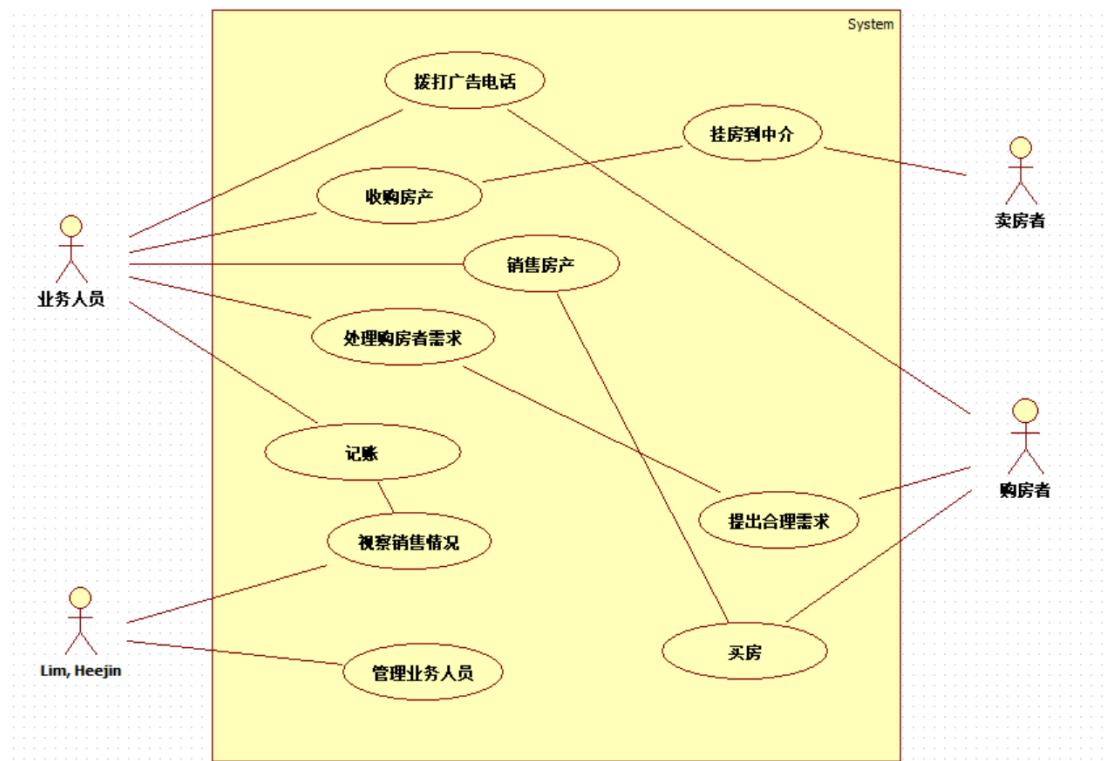


3 利用 inBuilder 开发软件

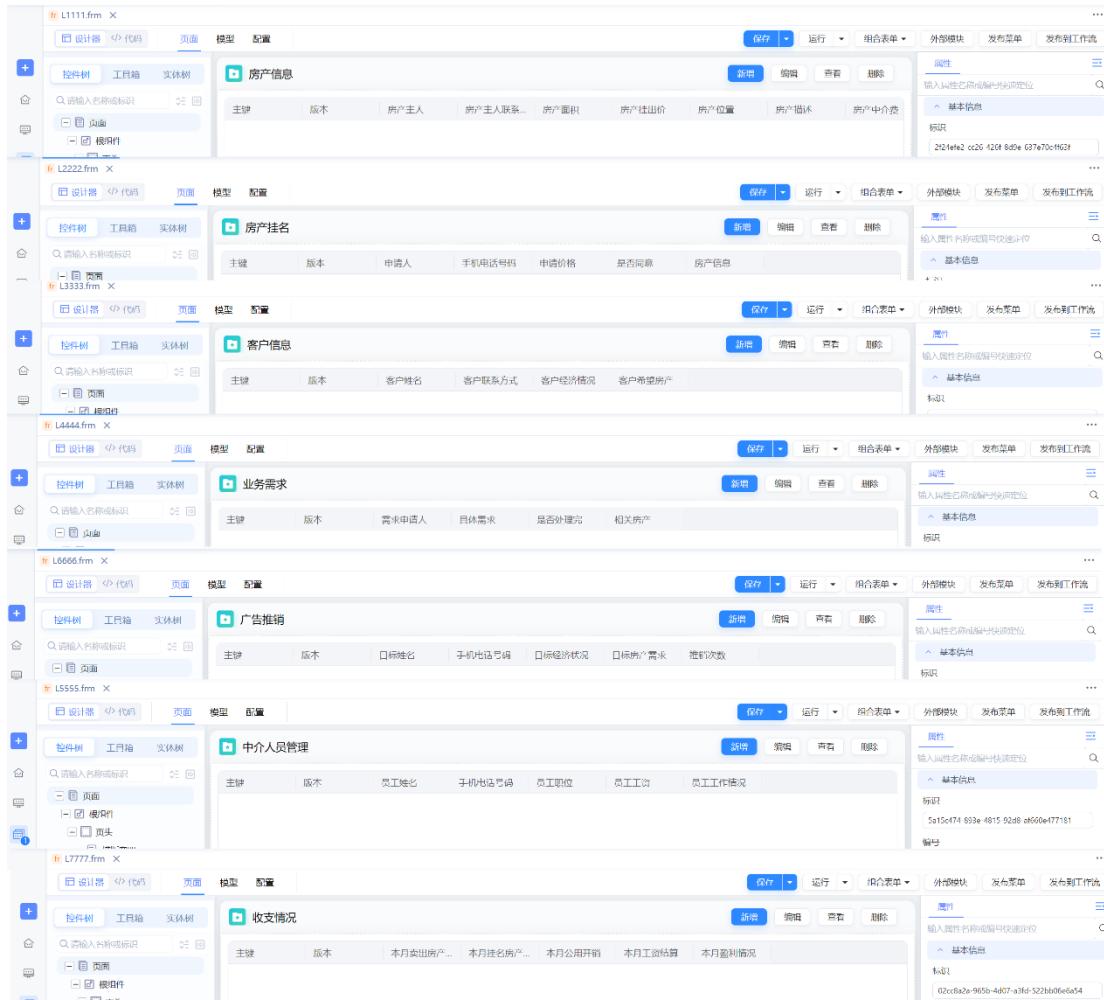
3.1 场景描述和功能设计

本人所选的软件题目是“房产中介系统”，本系统实现以下功能：作为中介业务人员有处理房产买卖，目标客户与潜在客户的处理，广告功能，处理客户的业务需求手续。中介经理要实现视察每月收支情况，管理业务人员调动与业务人员信息。卖房者可以将房产挂名在房产中介。购房者可以在房产中介处购买房产，同时对于期间产生的手续情况以及各种麻烦可以向中介业务人员提出处理业务需求。

该业务场景的用例图和活动图如下：



3.2 界面截图



3.3 Demo 视频

(已上传至仓库) 视频地址: https://v.youku.com/v_show/id_XNjI2MzA3NDMzMg==.html

4 ***熟悉 UBML 项目

PR 地址如下: <https://gitee.com/ubml/ubml-standard/pulls/13>

5 小结

本次实验我熟悉了典型低代码平台的使用和利用低代码平台开发程序，具体掌握了典型低代码平台 inBuilder 的使用，同时能够利用 inBuilder 平台进行软件产品的开发，也熟悉 UBML 开源项目，本次实验让我受益匪浅。