

使用介绍：实现流程审批功能

接上一篇《zPaaS低代码平台使用介绍：文件上传下载服务开发》，本篇主要介绍一下如何使用zPaaS低代码开发平台进行流程审批类功能的快速开发。

1. 开发的功能说明

流程审批功能涉及到工作流引擎，目前低代码平台的工作流引擎集成的camunda，本示例将基于我的休息记录表实现休息记录的管理以及休息申请的会签审批功能，实现的前端功能包括《我的休息记录》以及《申请申请审批》两个功能。

涉及的数据库表为《my_rest_record》，建表语句如下：

```
CREATE TABLE `my_rest_record` (  
  `id` varchar(32) NOT NULL COMMENT '主键',  
  `user_id` varchar(32) NOT NULL COMMENT '用户id',  
  `begin_time` datetime NOT NULL COMMENT '开时时间',  
  `end_time` datetime NOT NULL COMMENT '结束时间',  
  `apply_time` datetime DEFAULT NULL COMMENT '申请时间',  
  `apply_reason` varchar(500) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_0900_ai_ci NOT NULL COMMENT '申请原因',  
  `process_inst_id` varchar(128) CHARACTER SET utf8mb4 COLLATE  
utf8mb4_0900_ai_ci DEFAULT NULL COMMENT '流程实例id',  
  `status` varchar(1) DEFAULT NULL COMMENT '状态',  
  `status_time` datetime DEFAULT NULL COMMENT '状态时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci  
COMMENT='我的休息记录';
```

其中：

- 用户标识user_id字段：取登录用户的用户标识
- 流程实例标识process_inst_id字段：用户记录工作流引擎的流程实例标识，在工作流引擎流程实例的流程参数中同时记录我的休息记录的主键信息{"applyId":""}

2. 前提条件

目前低代码平台未直接提供对数据库以及数据库表的管理功能，因此需要使用其他工具，提前创建一个数据库，并在数据库中创建好对应的表。

前提工作：

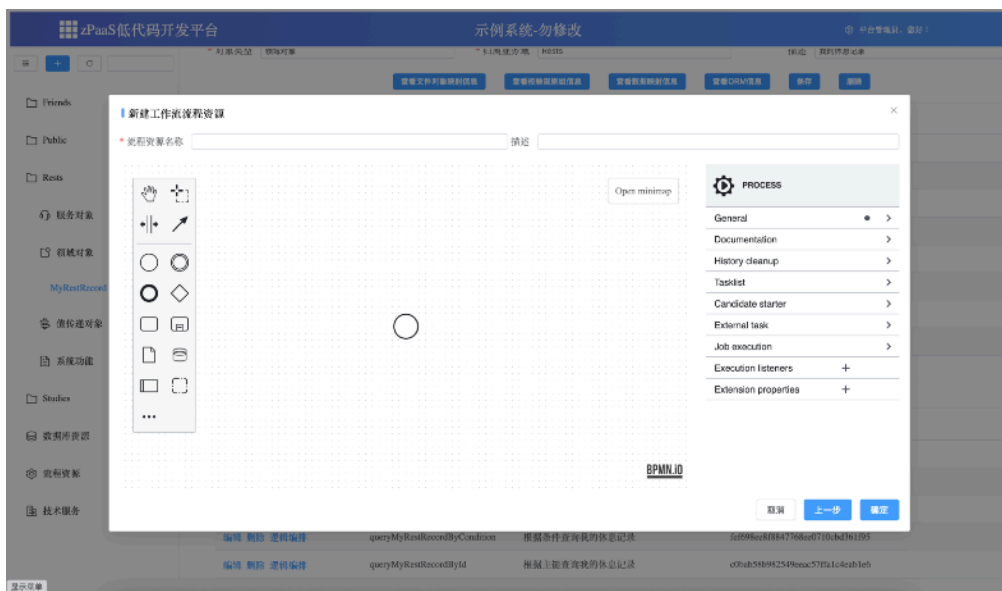
- (1) 部署mysql数据库
- (2) 创建一个database，如demo
- (3) 在demo库中，使用如上的建表语句，创建my_rest_record表

3. 准备工作

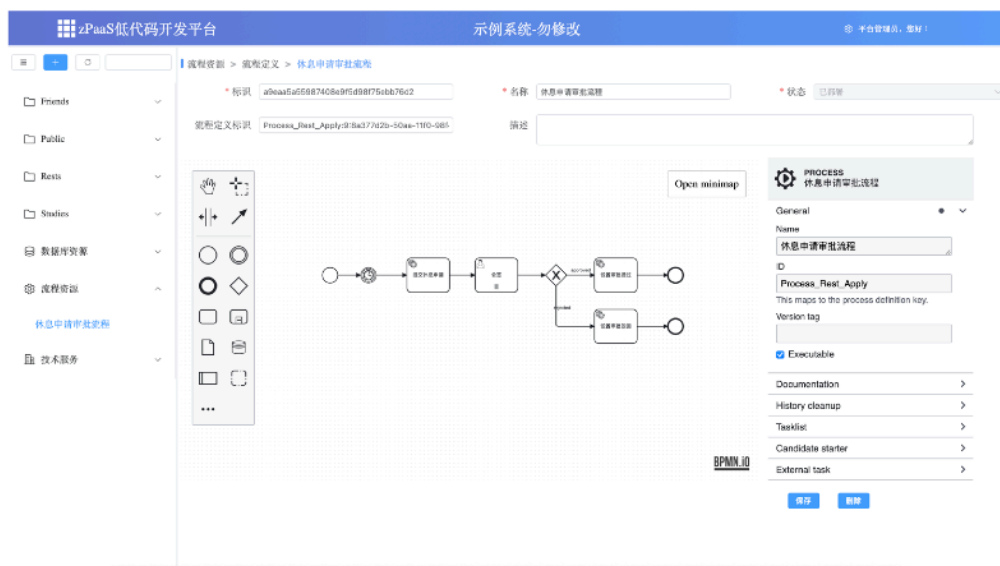
同《zPaaS低代码平台使用介绍：第一个功能开发》的准备工作，其中流程审批相关的功能及服务放到休息域Rests。

4. 流程设计

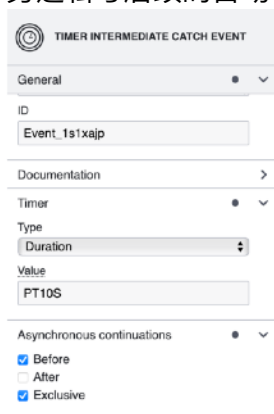
(1) 使用工作台左上角的“+”按钮，创建一个流程资源



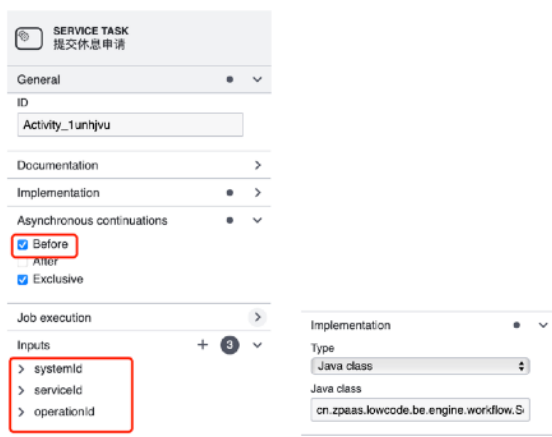
(2) 创建好的流程如下：

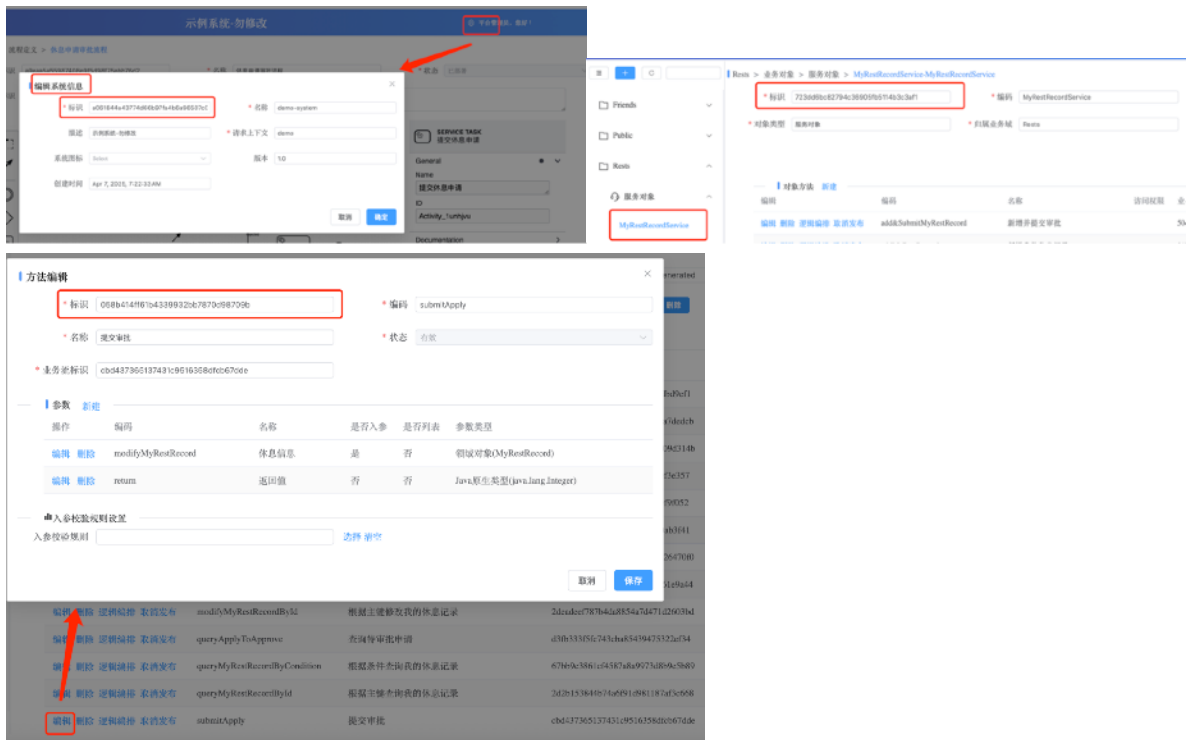


(3) 延时节点，延时10S执行，并使后续的节点执行异步化，主要用来避免流程启动时的业务逻辑与后续的自动节点逻辑执行顺序错乱的问题。



(4) 提交休息申请节点，类型为Service Task，自动节点，主要用于调用后端服务执行相应的逻辑。后端服务通过内置的Java实现类 `cn.zpaas.lowcode.be.engine.workflow.ServiceInvokeDelegate` 以及 `systemId`（业务系统id）、`serviceId`（服务对象id）、`operationId`（方法id）三个固定的输入来定位一个后端的服务方法，三个输入参数的值目前未实现界面化的选择功能，需要通过低代码平台找到相应的值并手工填入。为避免执行顺序的错乱，该节点也设置异步执行。对应后端的方法为 `changeStatus2Approving`。

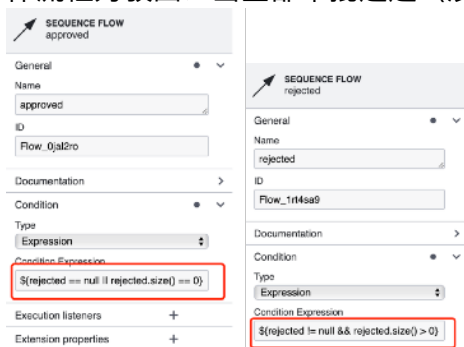




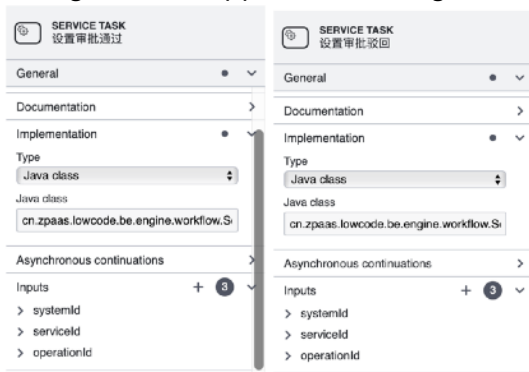
(5) 会签节点，两人会签，必须都审批通过才算通过（ $\{nrOfInstances == nrOfCompletedInstances\}$ ）。涉及审批的人员由流程参数userList传入，示例功能中由前一节点固定两用户id并传入。



(6) 两个分支，rejected是流程参数，由前一个节点设置，只要有一个审批驳回审批时，整体流程为驳回；当全部审批通过（没有驳回时），整体流程才算审批通过。

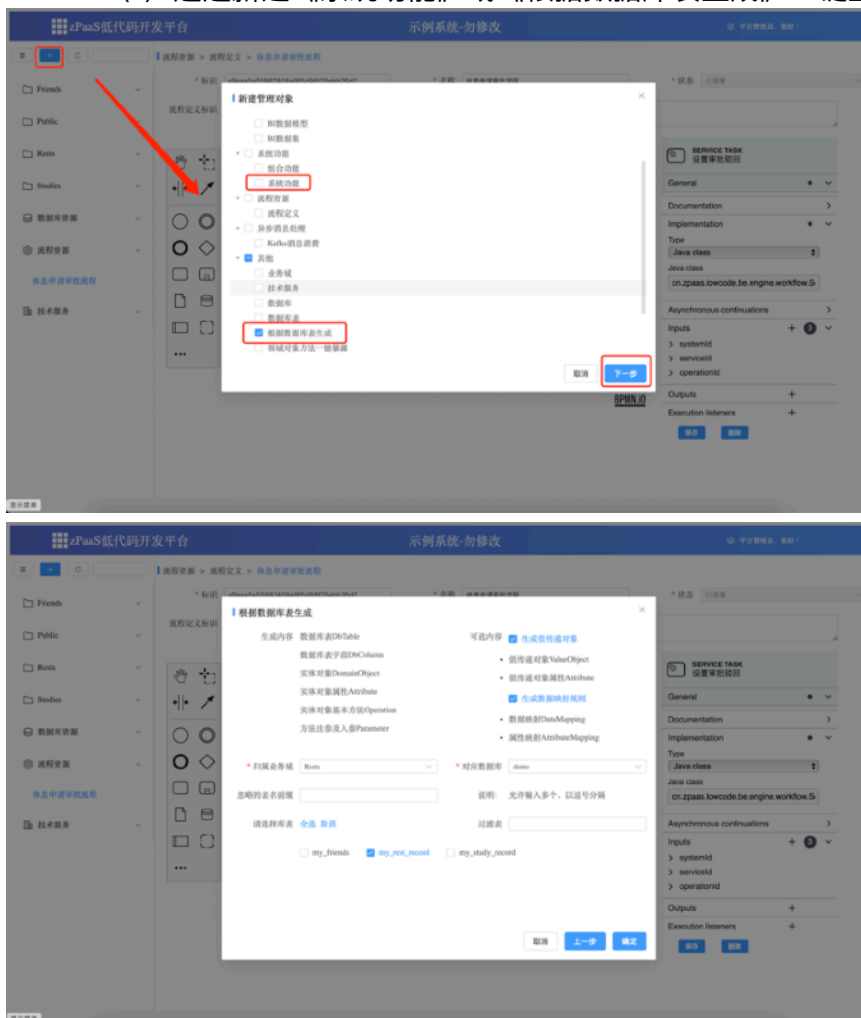


(7) 审批通过与审批驳回节点，调用后端服务实现申请状态的变更。对应后端的方法为changeStatus2Approved和changeStatus2Rejected。



5. 开发后端服务

(1) 通过新建《系统功能》或《根据数据库表生成》一键生成后端所有的基础对象。



(2) 创建服务对象MyRestRecordService

新建管理对象

✕

* 编码

MyRestRecordService

* 名称

MyRestRecordService

* 对象类型

服务对象

* 归属业务域

Rests

描述

取消

上一步

确定

(3) 在MyRestRecordService对象中创建相应的方法如下:

对象名称	标识	编码	名称	返回结果	业务标识
删除 删除 新增发布	addSubmiMyRecorcd		新增并提交文档		50d2478121c049dbf02d41b9ef01
删除 删除 新增发布	addMyRecorcd		新增我的日志记录		21d3990c702d36c448dc8943c1ad6b5
删除 删除 新增发布	approveApply		提交审批申请		693614e07754d50380c7b227396c31d5
删除 删除 新增发布	cancelApply		取消审批申请		6d751e8755c4367b7d72e00f00c357
删除 删除 新增发布	changeStatus2Approved		更新状态为审批通过		0b7c1c1d33d43246a0b45f0e90502
删除 删除 新增发布	changeStatus2Approving		更新状态为审批中		4545900b2c14d8a0b0792ca3a67001
删除 删除 新增发布	changeStatus2Rejected		更新状态为驳回		60a33c707b546023e47f6a2b8c4080
删除 删除 新增发布	deleteMyRecorcdById		根据主键删除我的日志记录		43d0bf13aa040cc2582c230a1e94a
删除 删除 新增发布	modifyMyRecorcdById		根据主键修改我的日志记录		2baeae787b4da854c7d7142d03bd
删除 删除 新增发布	queryApplyToApprove		查询待审批申请		df0b332561743445a30473322c2434
删除 删除 新增发布	queryMyRecorcdByCondition		根据条件查询我的日志记录		670b9c3861c4f03b74a9973d0b6c90
删除 删除 新增发布	queryMyRecorcdById		根据主键查询我的日志记录		2d2b153844b74a6d91d981813b1c668
删除 删除 新增发布	submitApply		提交审批		c9d47365113743149516378d6c874da6

(4) 其中addMyRestRecord、deleteMyRestRecordById、modifyMyRestRecordById、queryMyRestRecordByCondition和queryMyRestRecordById 5个基础的增删改查方法是自动生成的默认方法。其中addMyRestRecord方法是休息记录新增的方法，其业务流增加“对象属性设置”节点，用来设置userId、status和statusTime三个属性的默认值。queryMyRestRecordByCondit方法是休息记录查询方法，其业务流增加“对象属性设置”节点，用于设置默认的查询条件userId，只查询当前用户自己的休息记录。其余三个方法保持生成的状态即可。

业务流编辑

业务流标识: 2760395a70204c4488eca8943a78dc1b | 业务流名称: activityFlowResource管理业务流 | 加入任务: 是 | 业务流类型: 异步业务流 | 对应数据库: demo

业务流

目标对象类型: Select

目标对象: 选择 清空

对象类型: Select 选择 是否列表

对象Key:

属性映射规则: ['fromAttrPath': '', 'toObjectAttr': '']

删除 来源 是否列表 Key

X	过程数据	是	loginInfo
X	固定值	否	0
X	固定值	否	none()

属性映射规则

```

X [{"fromAttrPath": "loginUserId", "toObjectAttr": "userId"}]
X [{"fromAttrPath": "", "toObjectAttr": "status"}]
X [{"fromAttrPath": "", "toObjectAttr": "statusTime"}]

```

(5) 新增提交提交审批方法,

业务流编辑

* 业务流标识

* 业务流名称

* 加入事务

* 业务流类型

* 对应数据库

业务流

条件表达式

子业务流

设置备注

条件对象来源 ☒ 选择 ☐ 返回列表

条件对象 Key 属性

比较操作符 类型

值对象来源 ☒ 选择 ☐ 返回列表

值对象 Key 属性

条件对象				
删除	来源	属性	比较	Key
	X	输入参数	否	modifyMyRecdRecord
	X	输入参数	否	modifyMyRecdRecord
	X	输入参数	否	modifyMyRecdRecord

然后构建参与会签的用户列表，示例程序中写死两个用户id；

节点说明

节点类型

节点处理

目标对象来源

目标对象Key

目标对象属性

目标对象类型

目标对象

设置规则

源对象类型

源对象Key

属性映射规则

删除

来源

是否列表

Key

节点后处理

作为过程数据输出

存储关键字

转换规则

作为领域对象输出

作为业务流结果输出

固定值	5df6ddead0d74ae8bb993e9d2af9eb5e	{{"fromAttrPath":"/","toObjectAttr":["@+"]}}
固定值	6534f4967b3743be86d3f4944f0fe72c	{{"fromAttrPath":"/","toObjectAttr":["@+"]}}

创建审批流程，创建的时候同时设置流程参数applyId和userList；

节点说明

节点类型

节点处理

流程实例来源

流程实例Key

流程操作

流程参数来源

流程参数Key

执行结果类型

执行结果对象

当点要处理

源对象来源

设置规则

源对象类型

源对象Key

属性映射规则

删除

来源

是否列表

Key

节点后处理

作为过程数据输出

存储关键字

转换规则

作为领域对象输出

作为业务流结果输出

输入参数	modifyMyRestRecord	[{"fromAttrPath":"id","toObjectAttr":"applyId"}]
过程数据	userList	[{"fromAttrPath":"/","toObjectAttr":"userList"}]

更新modifyMyRestRecord对象信息；

节点说明

节点类型

节点处理

目标对象来源

目标对象Key

目标对象属性

目标对象类型

目标对象

设置规则

源对象类型

源对象Key

属性映射规则

删除

来源

是否列表

Key

过程数据	processInstId	[{"fromAttrPath":"/","toObjectAttr":"processInstId"}]
固定值	2	[{"fromAttrPath":"/","toObjectAttr":"status"}]
固定值	now()	[{"fromAttrPath":"/","toObjectAttr":"statusTime"}]
固定值	now()	[{"fromAttrPath":"/","toObjectAttr":"applyTime"}]

调用modifyMyRestRecordByld方法保存申请信息；

节点说明

微序入型

节点类型

20-本地方法调用

查看可用数据

节点处理

调用对象类型

服务对象

调用对象

MyRestRecordService

选择

清空

调用对象Key

调用方法

modifyMyRestRecordByld

异步调用

参数实例来源

预处理对象

选择

参数实例Key

执行结果类型

选择

是否列表

执行结果对象

选择

清空

节点预处理

源对象来源

选择

选择

源对象来源

选择

选择

源对象Key

源对象是列表

映射规则

[{"fromAttrPath":"/","toObjectAttr":""}]

添加

操作

来源

源对象Key

是否列表

X

输入参数

modifyMyRestRecord

否

节点后处理

作为过程数据输出

作为领域对象输出

作为业务流结果输出

转换规则

设置

清空

输入参数	modifyMyRestRecord	[{"fromAttrPath":"/","toObjectAttr":"modifyMyRestRecord"}]
------	--------------------	--

(5) 新增add&SubmitMyRestRecord方法，该方法在新增休息记录的时候同时提交审批，生成审批流程。该方法主要调用新增以及提交审批两个方法。

业务流编辑

业务流标识

50d52478121c40b4ad9f024f02d9ef1

业务流名称

add&SubmitMyRestRecord方法的主要业务

加入任务

是

业务流类型

复杂业务流

对应数据源

demo

业务流

节点说明

新建

节点类型

20-本地方法调用

查看可用数据

节点处理

调用对象类型

服务对象

调用对象

MyRestRecordService

选择

清空

调用对象Key

调用方法

addMyRestRecord

异步调用

参数实例来源

预处理对象

选择

参数实例Key

执行结果类型

选择

是否列表

执行结果对象

选择

清空

节点预处理

源对象来源

选择

选择

操作

映射规则

X

[{"fromAttrPath":"/","toObjectAttr":"newMyRestRecord"}]

源对象Key

☐ 源对象是否列表

映射规则

["fromAttrPath":"","toObjectAttr":"",""]

选择

操作

来源

源对象Key

是否列表

X	输入参数	newMyRestRecord	否
---	------	-----------------	---

节点后处理

☐ 作为过程数据输出
☐ 作为领域对象输出
☒ 作为业务逻辑单输出

特殊规则

设置 清空

源对象Key

☐ 源对象是否列表

映射规则

["fromAttrPath":"","toObjectAttr":"",""]

选择

操作

来源

源对象Key

是否列表

X	输入参数	newMyRestRecord	否
---	------	-----------------	---

节点后处理

☐ 作为过程数据输出
☐ 作为领域对象输出
☐ 作为业务逻辑单输出

操作

映射规则

X	["fromAttrPath":"","toObjectAttr":"","modifyMyRestRecord"]
---	--

(6) 新增changeStatus2Approving方法，该方法主要用于 workflowServiceTask 节点调用，用来更新休息记录的状态。

业务流编辑

业务流标识

业务流名称

业务流类型

相对数据库

业务流

节点说明

节点类型

节点处理

调用对象类型

调用对象

调用Key

调用方法

参数实例来源

参数实例Key

执行结果类型

执行结果对象

节点后处理

源对象来源

(7) 新增approveApply方法，该方法主要用于前端的审批操作，支持审批通过和审批驳回。

业务流编辑

业务流标识

业务流名称

业务流类型

相对数据库

业务流

节点说明

节点类型

节点处理

调用对象类型

调用对象

调用Key

调用方法

参数实例来源

参数实例Key

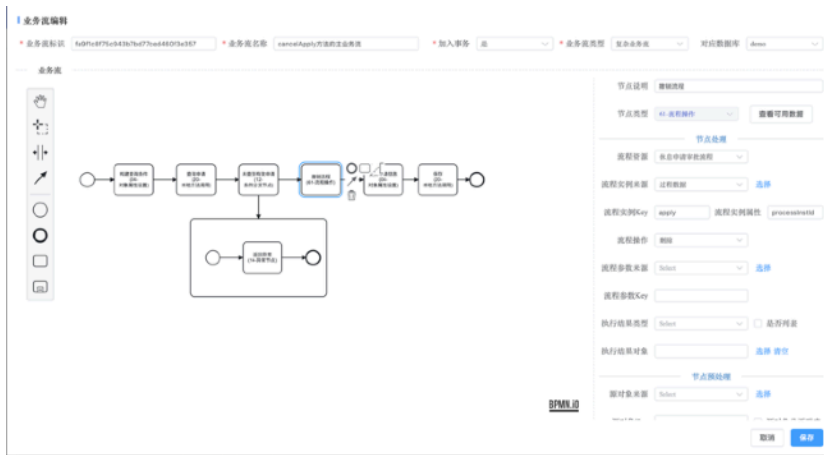
执行结果类型

执行结果对象

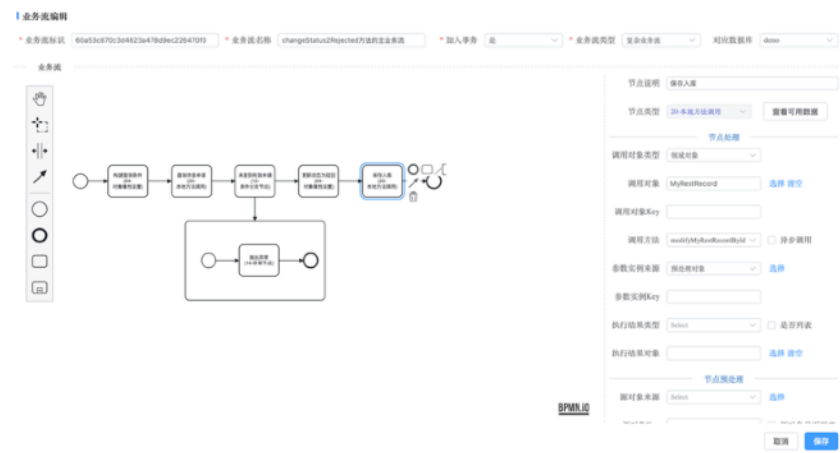
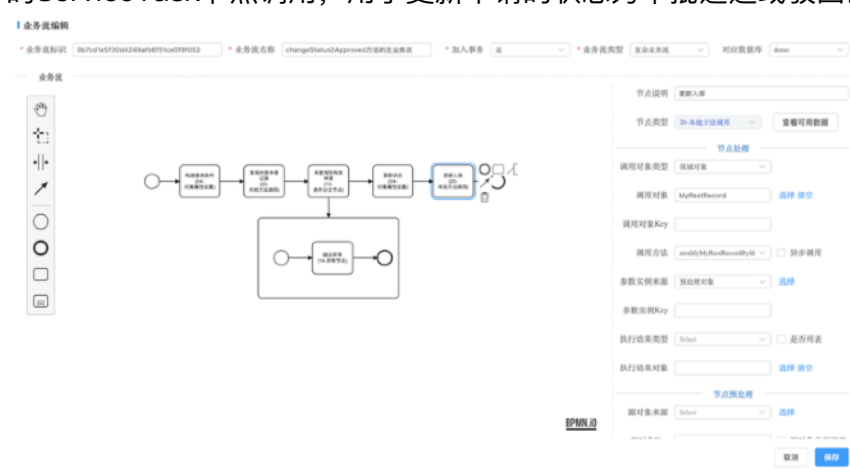
节点后处理

源对象来源

(8) 新增cancelApply方法，该方法主要用于前端撤加申请。



(9) 新增changeStatus2Approved和changeStatus2Rejected方法，该方法主要由工作流的ServiceTask节点调用，用于更新申请的状态为审批通过或驳回。



(10) 新增queryApplyToApprove方法，该方法用于前端查询待审批申请；

(2) 两个功能的绑定对象属性选项基本一致，一个配置好后，可以通过复制功能进行快速同步。特殊配置：申请时间字段支持范围查询、状态字段支持下拉、流程实例Id字段支持弹出查看、用户id字段支持翻译。

```
{
  "default": {
    "funcType": "P",
    "params": [
      {
        "valueType": "A",
        "value": "processInstId",
        "targetAttr": "processInstId"
      }
    ]
  }
}
```

funcType: P为内置的流程查看子功能。

```
[
  {
    "sql": "select real_name from sys_user where id=?",
    "parent": "",
    "datasource": "5195adf9540c4585b413c481bb3fdd6c"
```

```
    }  
  ]  
  
  {  
    "default": {  
      "subFuncId": "c6153ad03ac64a9097fc07ba68932b4c",  
      "params": [  
        {  
          "valueType": "F",  
          "value": "1",  
          "targetAttr": "status"  
        }  
      ],  
      "returnParams": {  
        "labelAttr": "userName",  
        "valueAttr": "id"  
      }  
    }  
  }  
}
```

(3) 休息管理功能

功能区域设计器-查询

可选属性

其他操作

我的信息记录

申请记录: applyRecord

开始时间: beginTime

结束时间: endTime

主键: id

当前主键id: primaryKeyId

结束时间: endTime

主键: name

申请时间

状态

操作

属性编辑

名称: 查询

类型: 查询列表

URL: applyRecordOpen

选择: 清空

归属: Select

目标: 结果

配置

取消

保存

功能区域设计器-结果

可选属性

其他操作

我的信息记录

查询列表: 新增

操作

主键

开始时间

结束时间

状态

申请时间

用户

编辑

新增

删除

1

Go to: 1

Total: 20

属性编辑

属性: name

名称: 状态

类型: 选择框

必填: ☐ 是 ☒ 否

只读: ☐ 是 ☒ 否

行数: 1 列数: 1

宽度: 110

样式

取消

保存

功能区域设计器-新增

可选属性

其他操作

我的信息记录

开始时间

结束时间

申请原因

操作

保存新增

提交新增

属性编辑

名称: 提交新增

类型: 新增

URL: applyRecordRecord

选择: 清空

归属: MyRecord

目标: Select

配置

取消

保存

功能区域设计器-编辑

可编辑性

其他操作

我的休息记录

* 主键

* 开始时间

* 结束时间

* 申请原因

当前实例id

状态

申请时间

状态时间

* 用户

取消

保存

添加字段

删除字段

属性编辑

* 名称

* 类型

URL

选择 清空

归属

目标

配置

取消

保存

(4) 休息申请审批功能

功能区域设计器-查询

可编辑性

其他操作

我的休息记录

主键

申请时间

状态

查询

属性编辑

* 属性

* 名称

* 类型

必填

只读

行数

列数

宽度

样式

取消

保存

功能区域设计器-审批

可编辑性

其他操作

我的休息记录

* 主键

* 开始时间

* 结束时间

* 申请原因

当前实例id

状态

申请时间

状态时间

* 用户

取消

审批通过

驳回

属性编辑

* 名称

* 类型

URL

选择 清空

归属

目标

配置

取消

保存

(5) 最终的效果见示例功能《我的休息记录》和《休息申请审批》

zPaaS低代码开发平台

平台管理员, 您好!

平台管理

系统管理

低代码平台

业务系统授权

业务系统管理

示例功能

我的朋友们

我的学习记录

我的休息管理

我的休息记录

休息申请审批

菜单 > 我的休息记录

申请时间: - 状态: Select

查询

查询列表 新增

操作	主键	开始时间	结束时间	状态	申请时间	用户	流程实例id
编辑 明细 删除	3cd6288a24284866a67f64b12a4d8aca	2025-06-26 00:00:00	2025-06-28 00:00:00	被驳回	2025-06-24 03:29:21	平台管理员	6b4aa8d8-50ab-11f0-98f6
编辑 明细 删除	dc1cb3d7c4bc996342f6d087a7322	2025-06-24 00:00:00	2025-06-25 00:00:00	被驳回	2025-06-24 03:27:26	平台管理员	265f3ac5-50ab-11f0-98f6
编辑 明细 删除	fe03acdd9644e189646d523380d080	2025-06-24 00:00:00	2025-06-25 00:00:00	审批通过	2025-06-24 03:25:55	平台管理员	d0641b25-50ab-11f0-98f6
编辑 明细 删除	d588282ef70941e597a3d0ba2592fcb	2025-06-24 00:00:00	2025-06-25 00:00:00	被驳回	2025-06-24 03:01:51	平台管理员	935a3b64-50a7-11f0-98f6
编辑 明细 删除	3ed53723e0f44092a3ecf99633855a	2025-06-27 00:00:00	2025-06-28 00:00:00	审批通过	2025-06-24 02:47:33	平台管理员	936c80e9-50a5-11f0-98f6
编辑 明细 删除	ec08fddccb2440cb1a95d4587e008d42	2025-06-22 00:00:00	2025-06-23 00:00:00	审批通过	2025-06-24 02:39:12	平台管理员	69c1e9e2-50a4-11f0-98f6
编辑 明细 删除	c186cc89169748b7ba6af0b68574bc58	2025-06-21 00:00:00	2025-06-22 00:00:00	审批通过	2025-06-24 02:36:07	平台管理员	fb4241f7-50a3-11f0-98f6
编辑 明细 删除	0fec30c4a50e4b0a7ffac84a4b8f6	2025-06-25 00:00:00	2025-06-26 00:00:00	审批通过	2025-06-24 02:07:42	平台管理员	0300ab07-50a0-11f0-98f6
编辑 明细 删除	01fe900c55164a4ba11e77f9d721056	2025-06-26 00:00:00	2025-06-27 00:00:00	审批通过	2025-06-23 02:23:11	平台管理员	02572b1e-46d9-11f0-98f6

< 1 > Go to 1 Total 9

编辑

* 主键: 3cd6288a24284866a67f64b12a4d8aca

* 开始时间: 2025-06-26 00:00:00

* 结束时间: 2025-06-28 00:00:00

* 申请原因: 6666

申请时间: 2025-06-24 03:29:21

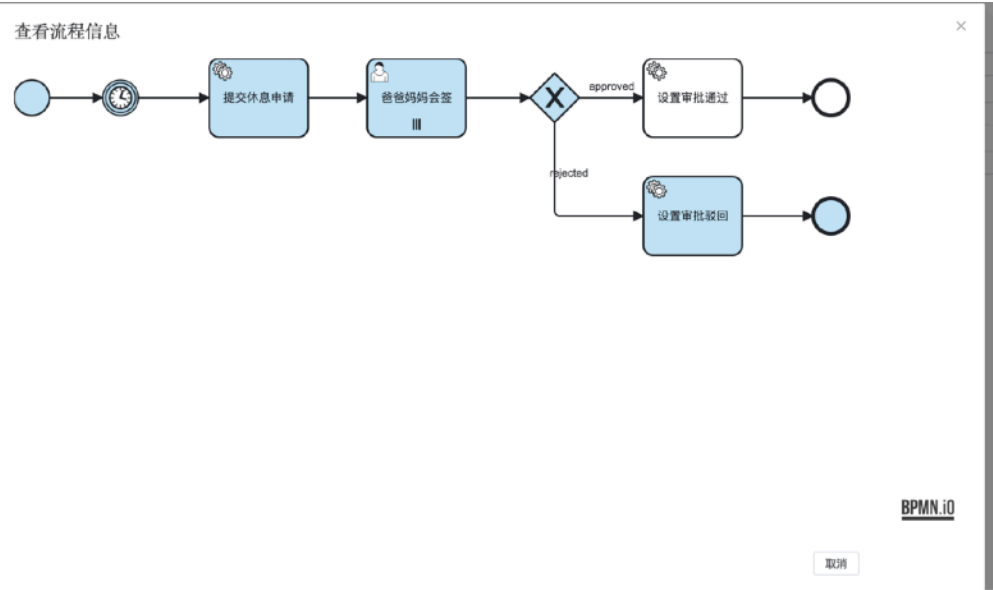
流程实例id: 6b4aa8d8-50ab-11f0-98f6-0242ac110002

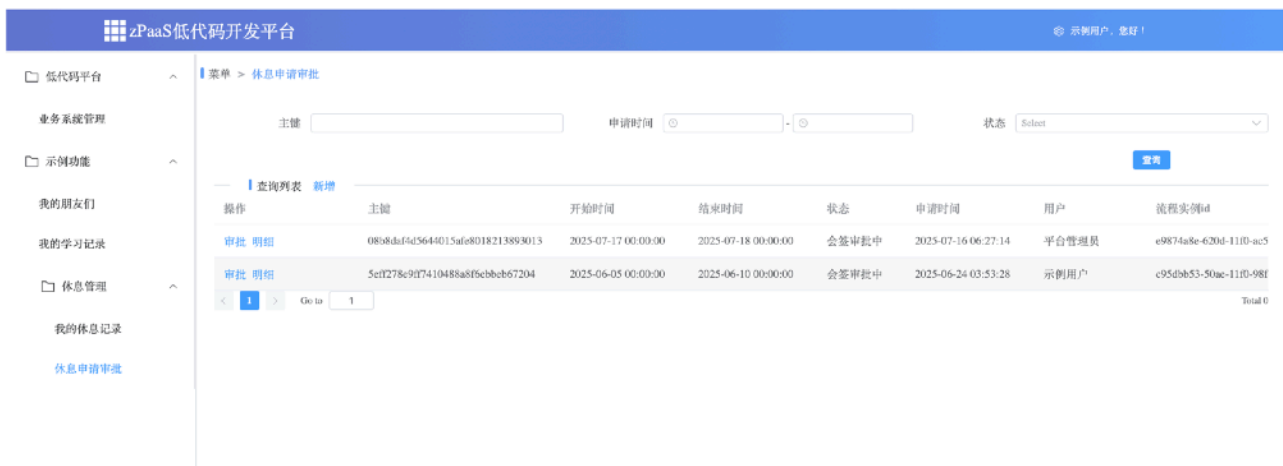
状态: 被驳回

状态时间: 2025-06-24 03:30:02

* 用户: 平台管理员

取消 保存 提交审批 撤回申请





7. 其他

低代码平台的源码已经提交到GitHub：<https://github.com/zjyzju/zPaaS-lowcode>，后续将通过更多的文章，逐步对低代码平台的设计理念/思想、具体的设计、部署以及操作说明进行细化，敬请关注。

另外在一台云服务器上部署了一个演示环境，有兴趣的小伙伴可以通过演示环境进行试用：

访问地址：<https://129.153.118.144:11443>

试用账户：`demo/adl28m2@ma(a!*D2`

`demo1/adl28m2@ma(a!*D2`

注：（1）该演示环境部署在一台免费的云服务器上，且位于国外，在国内访问响应会比较慢

（2）系统管理子系统的菜单框架引用的@micro-zoe/micro-app@1.0.0-rc.24组件，该组件在不同浏览器中存在兼容性问题，苹果的safari浏览器访问一切正常，但是windows的Edge和谷可的Chrome会存在页面刷不出来的情况，需要多等一会儿（有解决方案的可以邮件发给我，多谢！<https://github.com/jd-opensource/micro-app/issues/1559#issuecomment-2784997902>）。

如果有低代码平台以及AI相关的讨论，可以发送邮件到我的邮箱：zjyzju@163.com。