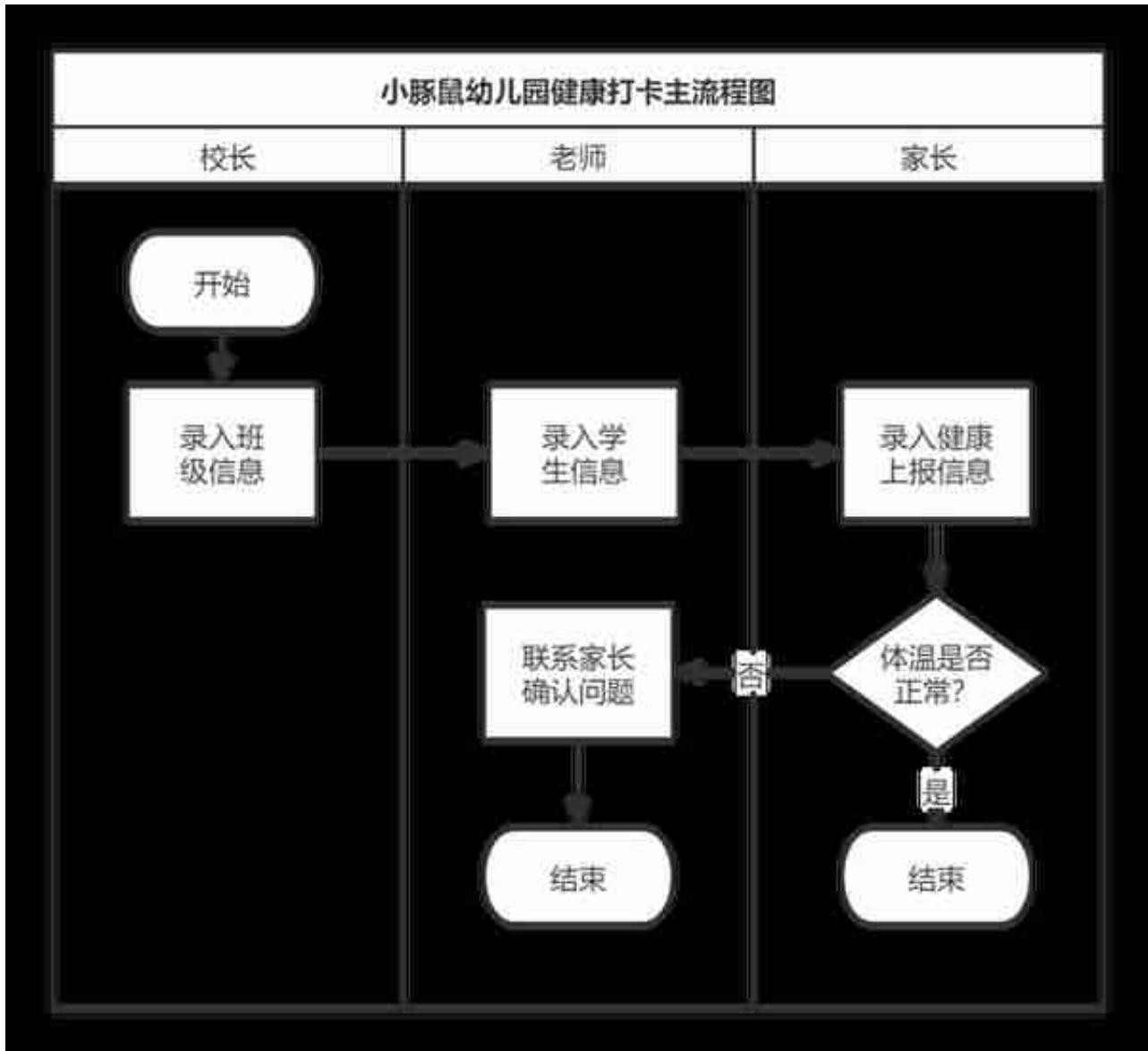


导语：本文基于作者虚构的“幼儿园健康上报系统”，演示如何用低代码平台快速搭建一套应用系统，并介绍演示了国内外的一些知名aPaaS产品，如Mendix、Outsystems、宜搭、明道云，试图探讨低代码平台在产品设计上的核心本质，从而让大家对低代码有一个更直观的理解。



可以想象，班主任家长刷卡打卡，有些基础数据要先维护，对于根据需求，需要维护包括班级信息、学生信息，而打卡动作是针对学生的，因此要对学生关联打卡信息。

另外需求中提到了，如果打卡体温异常，需要有老师跟进，我们考虑生成一个待办任务分配给老师，这条待办任务关联在异常的打卡记录上。对业务有了以上分析和思考，我们可以绘制出业务背后的ER模型图（领域模型），如下图。

页面	校长	老师	家长
班级列表页	√	√	
班级详情页	√	√	
学生列表页	√	√	
学生详情页	√	√	
健康上报列表页	√	√	
健康上报详情页	√	√	√

经过以上分析，虽然细节还不完备，但我们对“幼儿园健康上报系统”的设计思路已经了然于胸，接下来，我们直接进入低代码平台的开发演示环节！

3. 低代码实现——通过工作表定义数据实体

首先，我们来到明道云工作台，点击“添加应用”，创建新应用“小豚鼠幼儿园”（已经在下图中）。



工作表是明道云的概念，所谓工作表，实际上对应着ER建模中的实体，工作表中的相关控件，定义了实体的字段。例如班级表单中定义了自增长的“班级ID”，字符串类型的字段“班级名称”，枚举字段“状态”等。

通过表单来呈现出实体，是一种容易让人理解的设计方式。而实体背后的本质，是所谓的“对象”，以及最终会转移成数据库的表。在有些低代码平台中，则通过对象编辑器来定义管理实体，这是一种灵活性更强，但用起来更复杂的方案，在后文我们还会进一步介绍。

不论是表单编辑器，还是对象编辑器，原理是一样的，管理的都是提炼出的实体。对于非技术人员来讲，表单可能更容易理解接受一些。

在班级表单中，有一个“学生”的控件，是一种关联记录组件。因为班级和学生是一对多关系，每个班级可以拥有多个学生，所以在班级表单中，我们允许看到关联在班级下的所有学生列表，这在B端产品交互中是非常常见的一种设计形态。

实体之间所谓一对多、多对多的关系，体现的是多个表之间的关联，在设计器中可以轻松地定义班级和学生的一对多关系，如下图：



aPaaS平台都会自动完成PC版本和移动版本的适配，并且格式都是可调整的。例如上图呈现的是PC版的班级详情页，下图则是移动版本：



工作表只是定义了实体的具体字段，如何将实体的列表数据呈现出来呢？例如，如何将“班级”列表数据以不同的展现形式呈现出来呢？这就需要视图编辑器了！

在视图编辑器中，可以定义实体对应的多条数据的列表化呈现，包括列表数据默认的筛选条件，默认的展示字段，默认的字段的排序，都可以轻松定制，如上图所示。

在大多数自研的B端产品中，列表页（也就是视图）是最常见的功能页面，而一般情况下，这类页面都是硬编码实现，而非通过类似于视图编辑器这样的前端组件实现。在成熟的软件产品中，已经没有列表页的概念，都会通过视图编辑器来处理，这样就大量的简化编码工作。

如下图，我们针对“班级”表单，定义了四个视图，分别是“全部”、“生效中班级”、“已毕业班级”、“我的”，其中截图呈现的是针对“生效中班级”视图的默认搜索条件配置，可以看到，我们设置了该视图默认查询条件，是“状态”字段为“教学中”的所有班级数据。



下边的红框，定义了ID之间的关联，上边的红框，引入了“班级”表的“名称”字段，以便在“学生”表单和视图中展现。

在其他低代码产品中，针对这类诉求的解决方案不太相同。

严格来讲，表单只是数据对象的外化呈现，根据软件设计MVC的分层理念，数据定义和前端呈现要分层隔离，对象编辑器严格定义了数据实体本身，而如果在表单或视图中需要做多表连接去呈现其他相关表的某些字段，则是可视化层面需要解决的问题。

因此，在很多更复杂一些的低代码产品中，所有可视化的部分，都是基于页面编辑器来完成，和底层数据定义是互相独立的两件事。因为明道云的产品，为了在很大程度上降低用户的学习成本，所以将数据底层的对象编辑器，和展示层的表单编辑器融合在一起了。

现在，我们来解决一个棘手的问题。如果我们希望在学生视图中，呈现出该学生的老师姓名，该如何做到呢？通过学生，可以找到所在班级，但是，负责班级的老师是谁呢？如何定义呢？

一种做法，是针对“班级”表单增加一个字段，可以关联老师的账号，完成老师和班级的关系映射。在明道云，我们采用了另一种取巧的方式，需要由校长，将每一条班级数据的拥有者，修改成具体的老师账号，如下图，图中“王老师”，是一名角色为“老师”的独立用户。



在图中，我们设计了三个流程节点。

1. 第一个触发节点：定义了当“健康上报”表单在新增或更新数据时，如果发现“提问是否正常”字段等于“否”，则往下执行。
2. 第二个节点：当发现体温异常时，创建一条“体温异常跟踪”数据，分配给上报记录学生的老师。
3. 第三个节点：发送应用内消息给校长和老师，提醒处理，效果如下图：



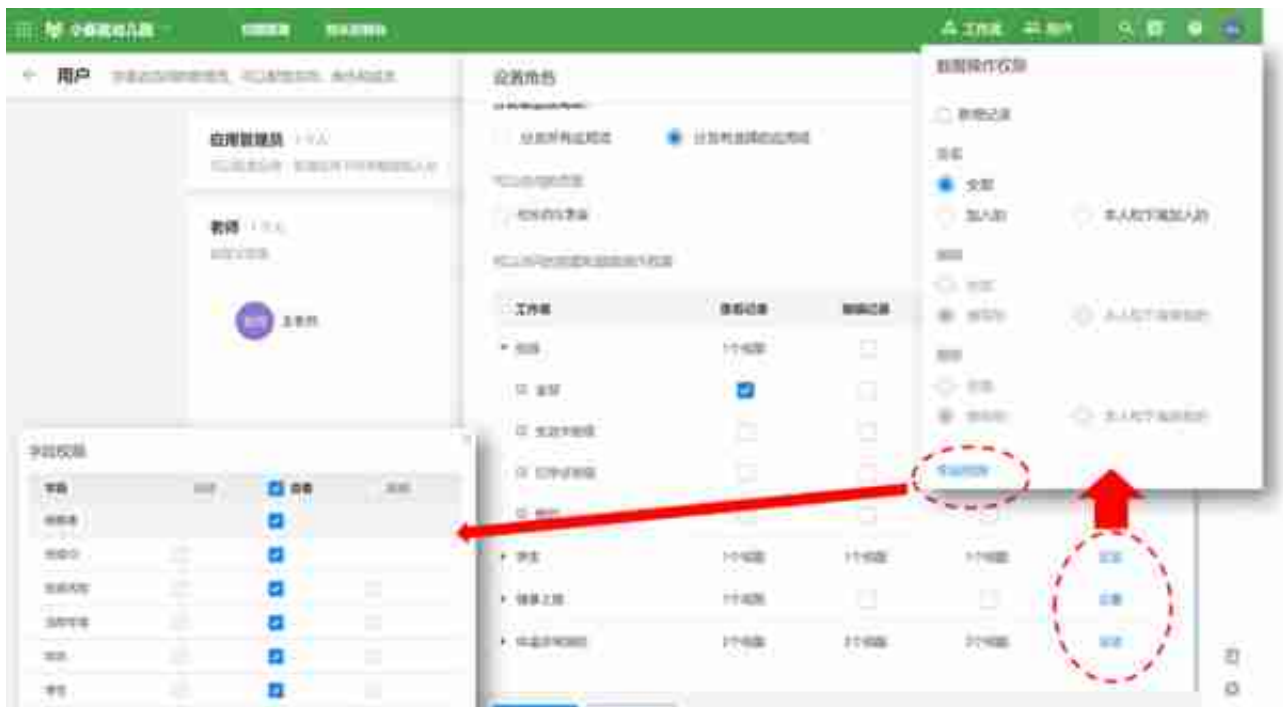
流程编辑器，不是简单地工作流引擎，我们一般理解的工作流引擎，例如审批流，只是针对单一数据对象的多节点处理。而真正复杂的流程编辑器BPM，需要在流程中对不同数据实体进行复杂处理，这也是很多B端业务的核心处理逻辑和过程。

当然明道云的流程编辑器功能很多，如下图，我们不再赘述。



6. 低代码实现——通过报表编辑器定义报表和仪表盘

走到这一步，涉及到业务流程的核心功能和数据表单都开发完毕了，接下来，我们需要给李校长配置她的管理工作台，也就是dashboard。通过类似报表引擎的功能，配置出校长的监控仪表盘，我们将其放在“校长控制台”的一级导航下边，如下图：



可以看到，每个针对每个角色，设计不同表单视图的查看、编辑权限，这是功能权

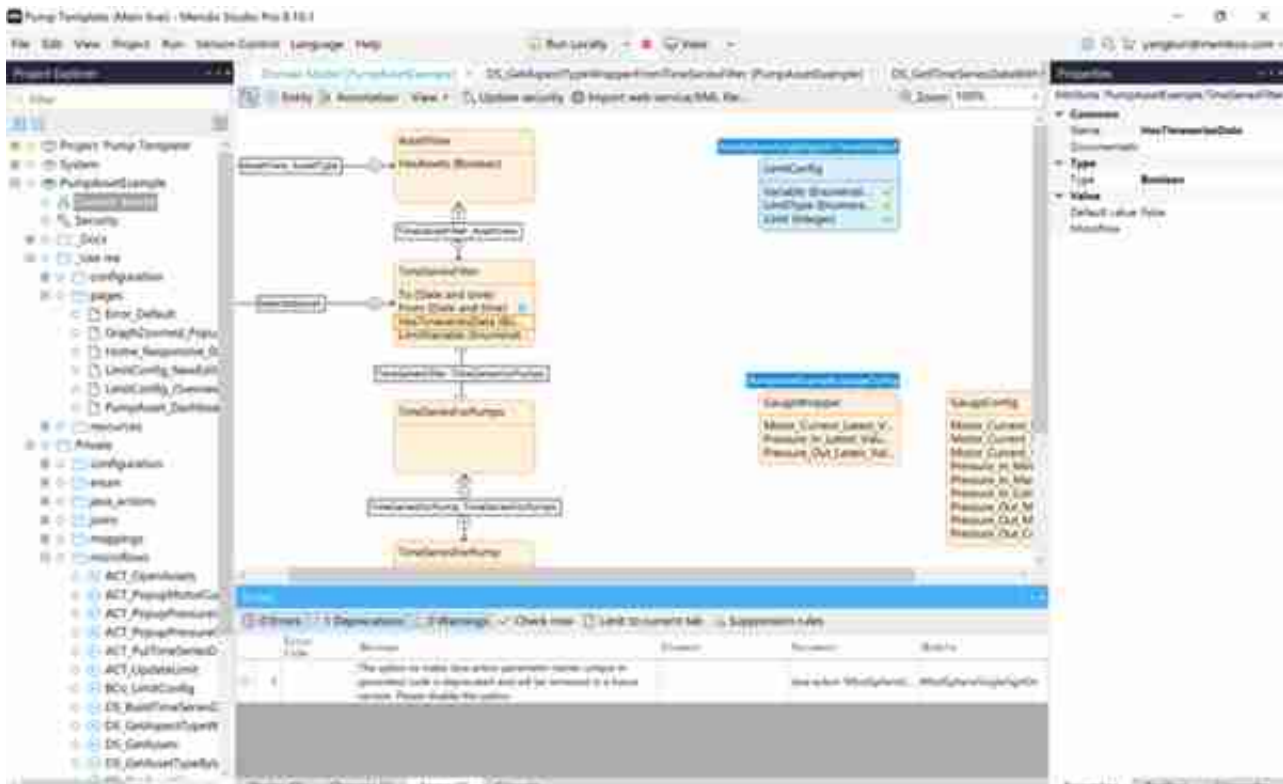
限。

点击每个表单最后的设置按钮，还可以定义角色针对表单的数据权限，包括是否允许处理所有数据，抑或是本人和下属的数据，抑或是本人拥有的数据（记得前面案例中我们对班级的“拥有者”进行了调整么），抑或是本人创建的数据。

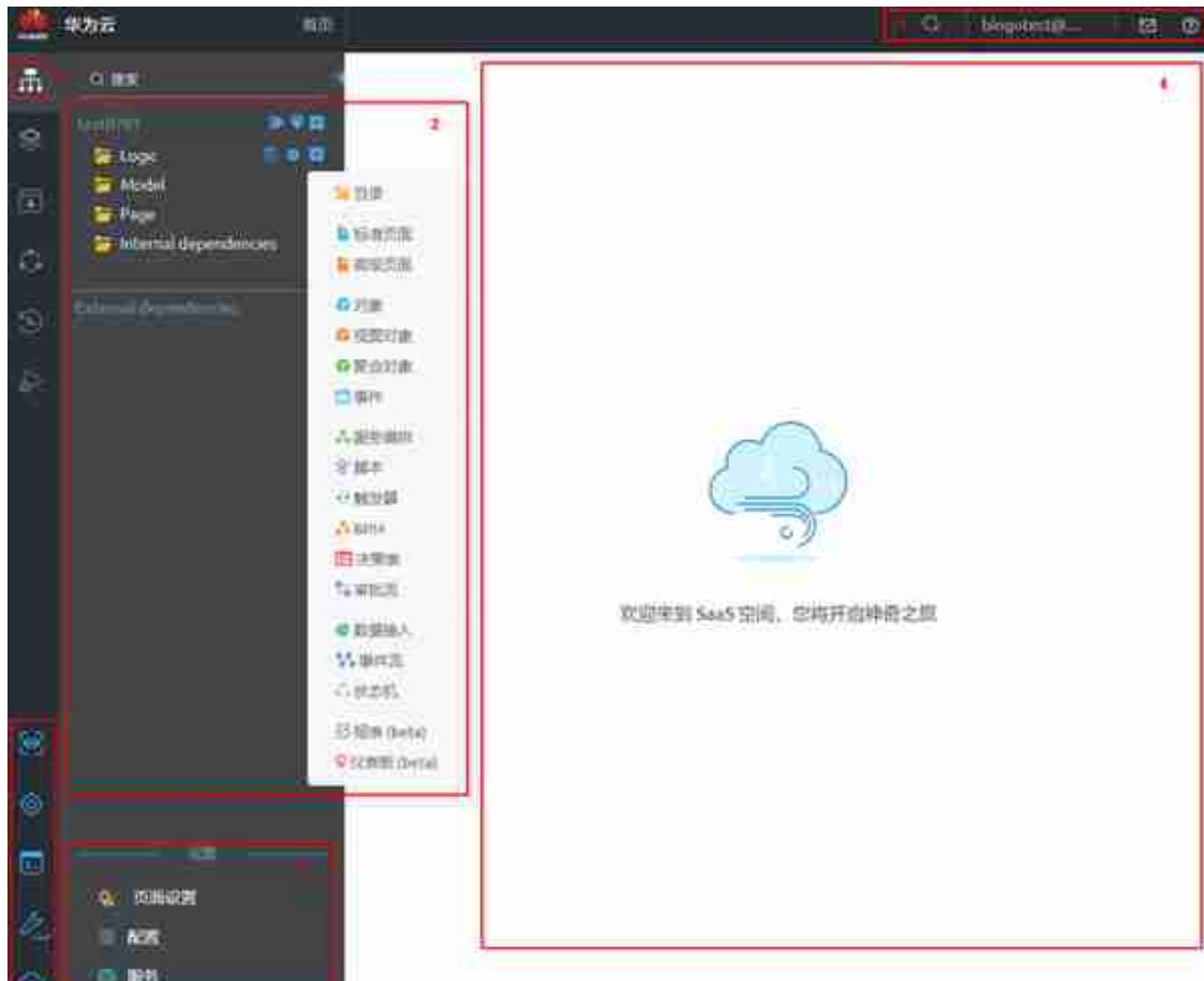
甚至还可以针对具体的字段设置更精细化的权限，如上图左下角窗口所示意。

完成以上配置，我们的低代码平台开发工作就完毕了，明道云的应用系统不需要发布，配置后立即生效。所有用户需要注册明道云账号来使用配置好的系统。配置完成的应用没有独立的应用程序，通过访问明道云官网登陆后使用，移动版需要下载明道云APP，登陆后进行使用。

最后给大家展示下移动版应用的截图，这些都是自动生成的默认设计，没有做过调整。



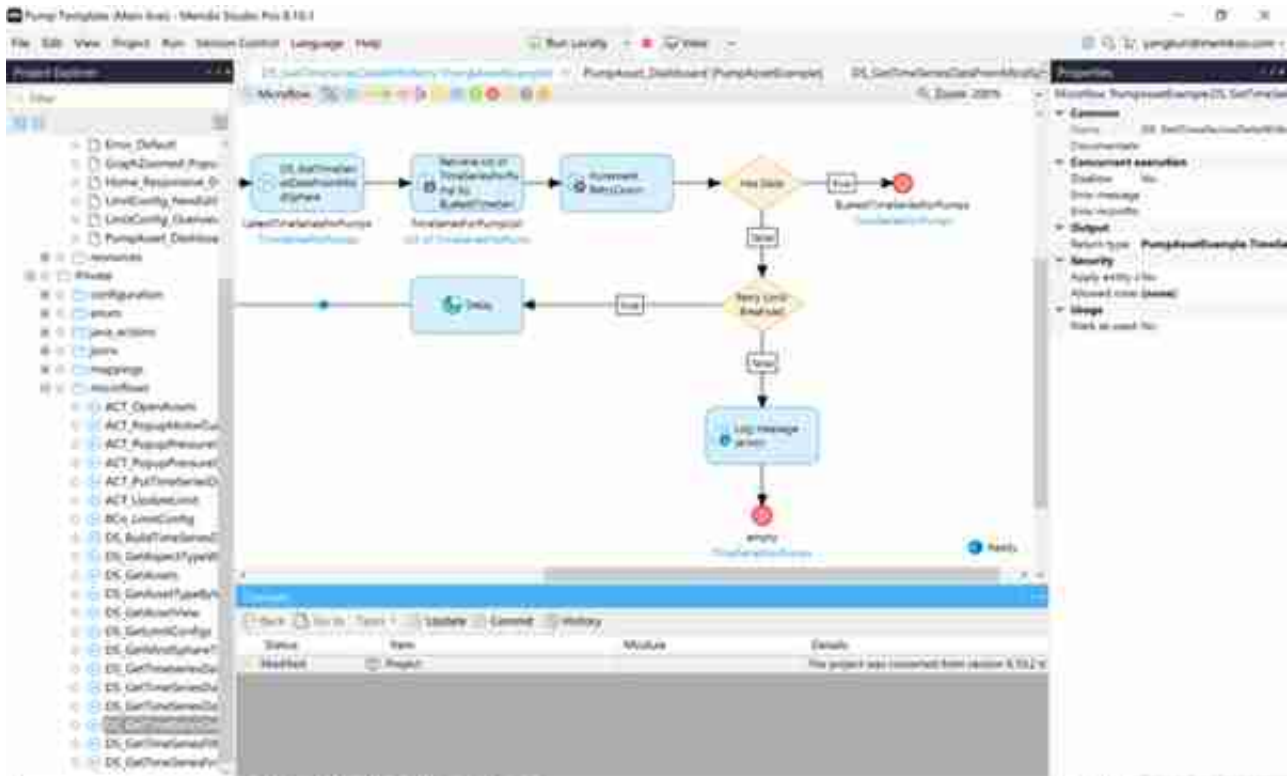
国外另一个知名aPaaS产品outsystems也采用了底层对象驱动的设计，并且也提供了windows版本的客户端，安装后有一个step by step的tutorial，非常惊艳！整套IDE风格的产品化设计，也非常强悍，让人印象深刻！



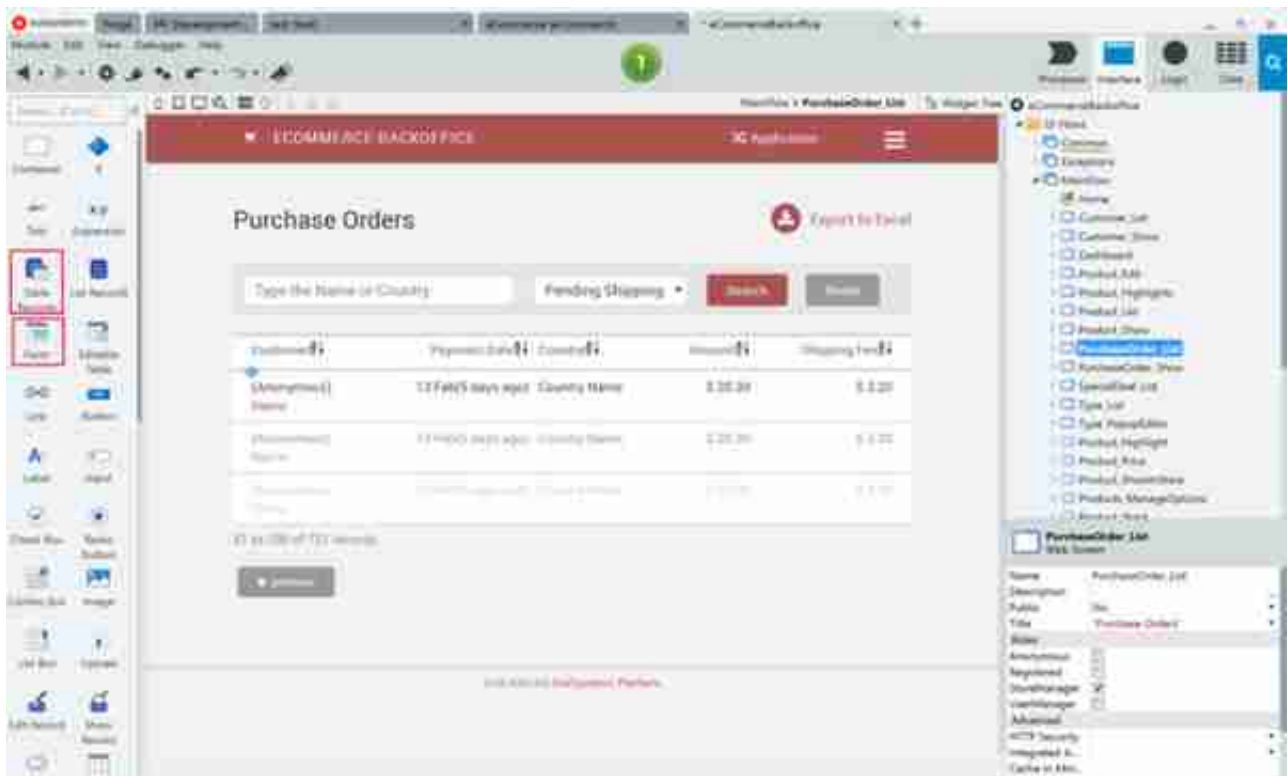
数据模型设计的第二种实现方式，是表单引擎。

对于设计人员来讲，只需要把底层的数据对象，理解成Excel的多张独立的表，每个表通过表单采集数据。用户定义数据模型，只需要将表单中的数据采集控件定义即可。

例如下图，是钉钉的宜搭的表单编辑器，设计思路和明道云的表单编辑器类似。表单编辑器将数据底层设计和可视化呈现打包在一起，对于非技术人员更容易理解，但也会丧失前后端分离的灵活性。

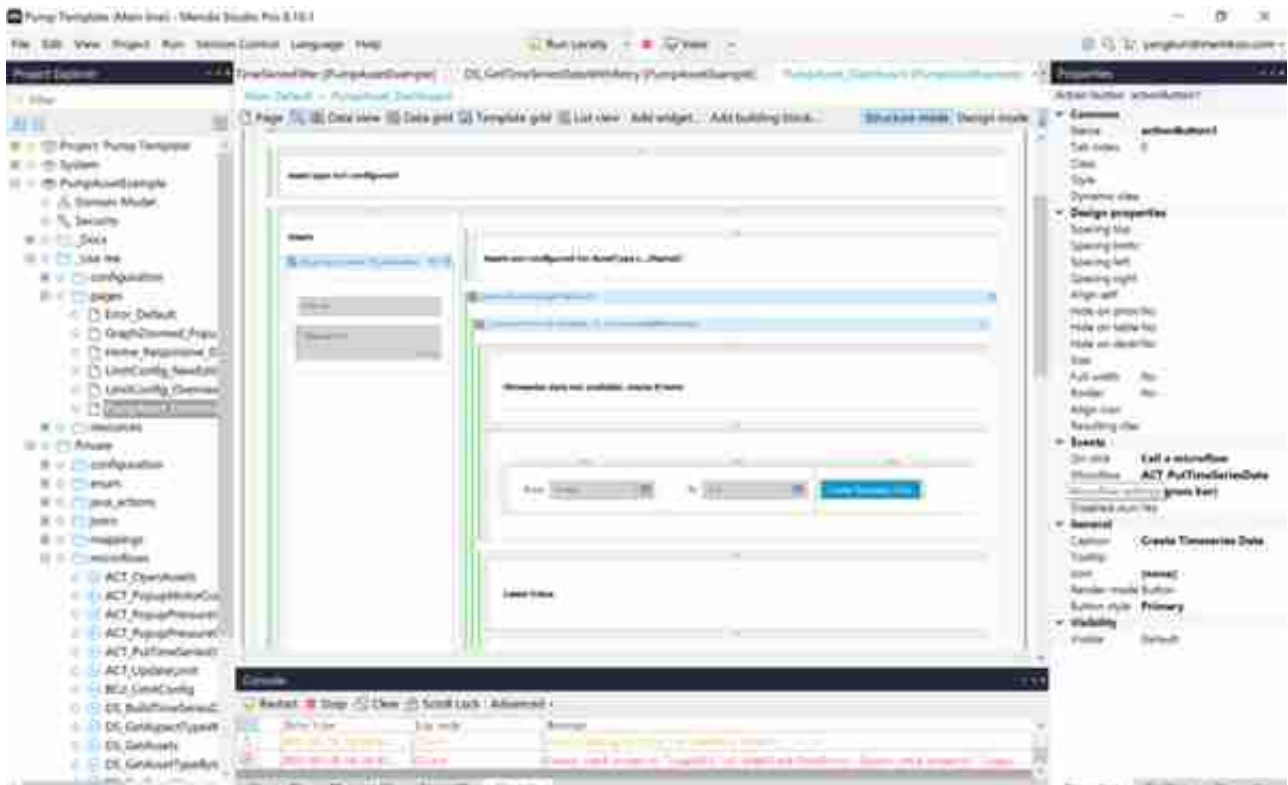


下图展示的是国内产品钉钉宜搭的工作流编辑器，感觉似乎过于简单，只是一个审批流编辑器，也可能是我没找到完整功能的配置界面？



可以看到，这是一套复杂的前端交互组件设计器，包括了类似于数据表集合Table Records的集成控件，也包括了表单控件Form，以及其他各类型控件集合，例如复选框Check Box，单选框Radio Button等等。

在这套编辑器中，操作者可以定义例如列表页、详情页、报表、仪表盘各类型前端页面。再比如Mendix的页面编辑器，也是同样的设计思路，如下图：



低代码平台的报表设计器组件，和传统的报表引擎没有太大区别，都是基于底层的数据，实现前端可视化输出，包括表格输出和图形输出。以上是第一种前端交互设计的产品形态，可以看出，功能强大、灵活，学习成本也比较高。

2) 大大简化了的页面配置器

将不同类型的页面，进行模板化配置，主要分为以下几类。

- 首先：将数据对象和表单相结合，通过定义表单（Form），完成了数据对象的定义，同时也构建出了详情页；
- 其次：通过视图编辑器这类组件，定义了针对数据对象的列表页；
- 最后：通过单独的dashboard配置器，完成类似于报表引擎的定义功能。

当然，低代码产品也会提供集成页面的配置，但功能要比前边提到的功能弱很多。前文已经大量描述了明道云的视图编辑器，不再赘述。

下图是宜搭的页面编辑器，展示了对某个系统首页的编辑。相对明道云，宜搭的页面编辑器更复杂一些，功能也更强大一些。



数据模型设计器、流程设计器和页面设计器，是低代码平台的核心，如果你理解软件设计的MVC分层架构，就很容易理解低代码平台的核心产品功能，以及不同的产品思路。当然不同低代码平台还有更多各具特色的强悍功能，有兴趣的读者可以进一步研究。

四、结语

可以看出，不同的低代码平台，设计思路并不相同。

产品的易用性和产品的灵活性之间存在平衡和取舍。例如，对于数据底层，究竟选择表单驱动的设计，还是领域驱动的设计？这两者区别非常大，后者对于非技术人员，基本不可用，而前者虽然易于学习理解，但功能确实也要弱化很多。

因此，低代码平台要明确目标用户群体，究竟是给ISV或IT团队使用的专业开发辅助工具，还是给非技术人员使用的强化版提效工具？前者更像是IDE的超级插件包，后者更像是Excel + VBA的超强易用版。

对于B端产品经理来讲，体验下类似于明道云这样的低代码产品，对理解软件设计很有益处，不论是表单，还是流程，还是权限管理，所有核心的产品设计问题都会涉及到，并且能够加深理解。

另外，万一业务有个大型需求，研发没排期，你3个小时就用aPaaS配置出来了，年

度CEO特别奖不给你给谁呢！

#专栏作家#

杨堃，公众号：PM杨堃（ID：pmYangKun）。人人都是产品经理专栏作家，《决胜B端》作者，12年互联网研发、产品设计经验，曾任VIPKID产品总监，百度高级产品经理，现为慢酷咨询创始人兼CEO。

本文原创发布于人人都是产品经理，未经许可，禁止转载

题图来自 Unsplash，基于 CC0 协议