蚂蚁科技

智能搭建
使用指南

文档版本：20260204

蚂蚁集团
ANT GROUP

法律声明

蚂蚁集团版权所有©2022，并保留一切权利。

未经蚂蚁集团事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

商标声明

蚂蚁集团 ANT GROUP及其他蚂蚁集团相关的商标均为蚂蚁集团所有。本文档涉及的第三方的注册商标，依法由权利人所有。

免责声明

由于产品版本升级、调整或其他原因，本文档内容有可能变更。蚂蚁集团保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在蚂蚁集团授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过蚂蚁集团授权渠道下载、获取最新版的用户文档。如因文档使用不当造成的直接或间接损失，本公司不承担任何责任。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	**单击设置> 网络> 设置网络类型。**
粗体	表示按键、菜单、页面名称等UI元素。	**在结果确认**页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.产品简介 05
2.基本概念 07
3.页面管理 08
4.页面分组管理 09
5.页面搭建与发布 10
5.1. 页面搭建编辑器 10
5.2. 搭建页面 12
5.3. 发布页面 13
6.组件管理 14
6.1. 创建组件 14
6.2. 管理组件 14
7.接口管理 16
7.1. 创建接口 16
7.2. 校验接口 16
7.3. 管理接口 17
8.开发接入指南 19
8.1. 开发接入流程 19
8.2. 业务组件开发 19
8.3. 接口开发 24

1.产品简介

智能搭建（Mobile Pages Building Service，简称 MBS）是一款互联网应用页面的快速搭建工具，无需研发投入开发，零代码快速搭建页面。MBS 实现了可视化、标准化、组件化的搭建方式，提供丰富的基础组件和多个预置业务组件，同时支持使用自定义的前端组件，以供页面搭建使用，让您一次搭建就可以同时在 App 端内和 PC Web 端使用页面。

依托阿里搭建领域多年的技术积累和淘系前端技术最佳实践，智能搭建提供成熟的开箱即用页面搭建解决方案。

功能特性

智能搭建提供一个统一的页面编辑器，同时支持画布搭建（使用基础组件库）和模块化搭建（使用业务组件库）。

- **画布搭建**

  零代码画布搭建方式，进行页面模块化配置，配置效果可实时预览。支持文字、图片、链接、图片轮播等多种组件的可视化搭建。可对画布上的页面进行分层管理，支持对不同的页面组件在不同页面分层中进行移动。

- **模块化搭建**

  通过添加、配置前端开发的组件来进行页面搭建，适用于有前端开发资源的业务。除基础样式组件外，提供多款预置业务组件，同时支持导入业务自定义的前端组件，供搭建页面时使用。

- **接口管理**

  通过配置接口来实现组件最终展示内容的动态化。

- **页面管理**

  对通过智能搭建创建的页面进行管理，支持新增、查询、修改、删除页面，同时支持对页面进行分组管理。

产品优势

智能搭建具备以下优势：

- **技术门槛低**

  无需研发设计，零代码快速搭建页面活动。将原本属于前端开发人员的工作，转移到非技术人员身上，让更了解用户的业务运营人员能够直接自主产出需要的页面，既减少了前端开发的工作量，又大大提升了运营效率。

- **所见即所得**

  丰富的官方模块生态，可基于编辑器，灵活自由创作。页面搭建过程中，可通过页面预览来查看和验证可视化编辑的效果。输入即所见，所见即所得。

- **一次搭建，多端使用**

  搭建好的页面支持在多种终端应用中使用，包括移动应用（App）、H5、小程序。

应用场景

运营人员通过使用智能搭建服务快速生成和发布页面，提升页面上线效率，加速运营工作的开展。

- **页面高频更新场景**

  借助智能搭建的页面可配置化能力，前端开发人员可在智能搭建平台中导入开发的前端组件，将页面搭建、发布的工作交给产品运营人员，缩短页面迭代周期，提升运营效率。产品运营人员可灵活使用平台提供的各种组件，轻松搭建、发布页面。

- **小型营销活动场景**

  对于小型营销活动，线上只需要少量几个页面，而且内容基本以图文、视频和链接为主。可以通过智能搭建的画布搭建来创建页面，只需设计人员提供营销活动页面设计稿图片，无需前端开发人员参与。

- **公告通知类场景**

对于公告通知类场景，通常只需一个线上页面，内容也基本以图文为主，可以通过智能搭建的画布搭建来完成页面创建。

2.基本概念

本术语表按拼音首字母对术语进行排序。

搭建

页面搭建是从模块到页面的组合过程，包括对页面模块的添加、复制、删除、顺序调整以及数据配置操作。

对象存储

对象存储（Object Storage Service，OSS）是一个分布式的对象存储服务，提供的是一个 Key-Value 对形式的对象存储服务。

页面

页面即智能搭建的产物，一个页面对应一个 URL，对外投放后，用户可以通过手机访问。页面是应用的可视化界面，也是普通用户访问应用时可见的部分。

组件
用于页面搭建的最小物料单位。

3.页面管理

对通过智能搭建创建的页面进行管理，支持页面新增、删除、查询、修改、复制操作。同时，数据概览区域会展示当前应用的页面数据统计信息，包括总页面数、已发布页面数和未发布页面数三项指标。

创建页面

页面是应用的可视化界面，是应用的基本组成部分，在页面中您可以对页面组件进行布局编排，配置组件属性。

页面创建步骤如下：

1. 登录 mPaaS 控制台后，从左侧导航栏进入 **智能搭建 > 页面管理** 页面。
2. 单击 **创建页面**，在页面创建面板上，输入页面名称，选择页面分组后，单击 **确定** 完成创建。如无页面分组可选，可单击 **创建页面分组** 新增页面分组，以供选用。

   新建的页面将展示在页面列表中。

查询页面

在页面列表右上方的搜索框中，输入目标页面名称关键字搜索即可。支持通过选择分组来筛选页面，支持在页面列表右上方的下拉框中选择分组进行页面筛选。

编辑页面

在页面列表中，选择要编辑的页面，单击操作列下的 **编辑**，修改页面信息，包括页面名称和分组信息。

复制页面

为方便您快速创建页面，MBS 支持复制已有页面，然后基于已有页面信息生成新的页面。

在页面列表中，选择要复制的页面，单击操作列下的 **复制**，复制该页面。复制出来的页面，除页面名称外，其它所有数据与原页面保持一致。

删除页面

在页面列表中，选择要删除的页面，单击操作列下的 **删除** 后，确认即可。

> **重要**
>
> 应用页面删除后无法恢复，请谨慎操作。

4.页面分组管理

按照不同的使用场景，对相似或相关页面进行归类、分组，方便对页面进行操作和管理。

创建分组

1. 单击页面列表右上方分组选择框右侧的设置图标，打开页面分组管理面板。
2. 单击 **创建分组** 按钮，输入分组名称后，单击 **确认** 按钮完成分组创建。

   新建的分组将展示在分组管理面板的分组列表中。

编辑分组

在页面分组列表中，选择要编辑的分组，单击操作列下的 **编辑**，修改分组名称。

删除分组

在页面分组列表中，选择要删除的分组，单击操作列下的 **删除** 后，确认即可。

> **说明**
>
> 如果分组下面包含有页面，需要先移除页面，否则无法删除分组。

5.页面搭建与发布

5.1. 页面搭建编辑器

本文对页面搭建编辑器的布局以及支持的组件进行介绍。

页面搭建编辑器由五部分组成： 组件选择区、 基础组件快速选择区、 全局操作区、 组件配置区 及 画布操作区。

组件选择区

组件区提供用于搭建的组件和图标，其中组件包括官方组件（基础组件）和业务组件。

- 官方组件：展示文本、图片、链接、轮播图、视频、音频、二维码、倒计时组件。
- 业务组件：展示四个预置业务组件（资讯信息流、自定义表单、一排三栏、一拖三栏）和用户自定义添加的前端组件。
- 图标：展示页面常用的图标组件。

官方组件

提供八款官方基础组件：

基础组件	说明
文本	用于添加文字。在画布中添加文本后，可随意调整文本框大小。选中文本框后，可在右侧的文本素材配置面板中设置文本样式、字体颜色、字体大小、文本框背景色以及文字跳转链接。
图片	用于添加 JPG、JPEG、PNG、GIF 格式的图片。添加图片后，可在画布上随意调整图片位置和大小。选中图片，可通过右侧的图片素材配置面板更换图片或使图片适应屏幕。图片上传后，默认显示图片的真实大小，可以使用 **适应屏幕** 功能让图片的宽度和画布保持一致。
链接	用于添加链接。添加链接后，可动态调整跳转区域大小，在页面指定区域添加跳转链接。配置链接后，当用户在链接所在区域进行点击时，将跳转到链接指向的地址。
轮播图	用于设置一组或多组序列图的图片，支持播放。可配置各个图片的跳转链接、是否循环播放、是否自动播放、自动播放时间间隔、是否显示提示点（即图片底部的小圆点）。
视频	用于在页面中嵌入视频。添加视频后，可设置视频封面图、视频地址、是否开启静音、是否开启自动播放、是否展示进度条、是否循环播放。
音频	用于在页面中嵌入音频。添加音频后，可设置音频图标、音频地址以及是否自动播放音频。

二维码	用于在页面中嵌入二维码。添加二维码后，可设置二维码的扫码后跳转页面、背景色、前景色以及纠错等级。 二维码具备“纠错功能”，即使部分编码变脏或破损，也可以自动恢复数据。纠错等级的比率是指可以纠错的码字占全部码字的比率。支持四种纠错级别： • 低（7% 的码字可被恢复） • 中（15% 的码字可被恢复） • 四分（25% 的码字可被恢复） • 高（30% 的码字可被恢复）
倒计时	用于在页面中添加倒计时组件。添加该组件后，可配置倒计时起始和结束时间、文案字号、文案颜色、文案对齐颜色、结束文案。

业务组件

展示智能搭建预置的业务组件和用户自定义的前端组件。可参考 创建组件 自定义业务组件。使用自定义业务组件时需要进行接口配置，添加该类组件前，确保已创建相关数据接口，具体参见 创建接口。

智能搭建提供以下四款预置业务组件：

预置业务组件	说明
资讯信息流	信息流列表组件，每行为一个内容，可以展示产品和资讯。支持配置组件标题、跳转链接（承接页地址）、数据源接口及参数、提交表单接口及参数、是否预加载。开启预加载后，会在实际的页面中提前加载组件。
一排三栏	一排三栏的组件，可以展示三个产品项和资讯。支持配置组件标题、跳转链接（承接页地址）、数据源接口及参数、提交表单接口及参数。
一拖三栏	提供 1 + 3 个模块，上面是一块横幅（banner），下面是一排三栏的组件，可以展示产品和资讯。配置项同一排三栏组件。
自定义表单	用于提交用户输入的内容的组件。添加该组件后，可编辑表单结构，在表单中添加各种控件，丰富表单内容。

图标

用于添加图标。添加图标后，可在右侧的配置面板中设置图标样式。

基础组件快速选择区

该区域提供页面常用的 8 个官方基础组件，方便快速添加。

全局操作区

全局操作区提供画布级别的快捷操作。

- 图层管理：置于最上层；置于最下层；向上一层；向下一层
- 操作历史：回退，前进
- 页面信息：展示当前页面的信息，包括页面标题、下线时间、页面地址、二维码。

- 预览：预览生成的页面内容。预览窗口中，可以复制页面连接或在浏览器打开页面，也可以手机扫码预览。
- 发布：发布当前页面到 PC 端。

组件配置区

对所选组件和不同的组件需要配置的内容有所不同，具体参见上述组件说明。

画布操作区

画布操作区域主要进行组件选择、删除（使用快捷键 Backspace）、复制（使用快捷键 Command/Ctrl + C）、拖拽操作。

单击画布空白区域，可配置页面的背景颜色、背景图、画布高度以及是否让画布高度自适应背景图。

单击画布左侧的 **缩放** 菜单，可以针对 PC 端和移动端调整画布宽度。

5.2. 搭建页面

智能搭建提供一个统一的页面编辑器，同时支持画布搭建（仅使用基础组件）和模块搭建（使用基础组件和业务组件）。模块搭建支持在搭建页面时，修改组件的数据源。

页面搭建主要包括添加组件和配置信息两个部分。

前提条件

- 已创建应用页面，具体操作参见 创建页面。
- 如搭建页面时要使用自定义的前端组件，需要先添加自定义组件并完成接口配置，具体操作参见 创建组件 和 创建接口。

操作步骤

页面搭建的操作流程如下：

1. 在页面管理的页面列表中，选择目标页面，单击操作列下的 **页面搭建**，打开页面搭建编辑器。
2. 根据需要，进行画布和页面组件配置。
   - 设置画布：
     - 单击画布空白区域，在右侧的面板中，设置页面的背景颜色、背景图、画布高度、是否让画布高度自适应背景图、是否固定 PC 页面尺寸。
     - 单击编辑器左侧的 **缩放** 菜单，设置针对不同终端的画布编辑区域尺寸。如未设置，默认使用推荐尺寸。
   - 添加页面内容：在页面搭建器左侧的组件和图标选择区，按需选择组件或图标并添加到画布，然后单击画布上的组件或图标，在右侧的配置区域进行相应的素材配置。
     - 官方组件：目前提供 8 款官方基础组件。组件信息参见 官方组件 说明。
     - 业务组件：可使用的业务组件包括四个预置业务组件以及用户自定义的业务组件。组件信息参见 预置业务组件。
       - 使用业务组件时，需要配置数据接口信息，包括用于获取数据的数据源接口和用于存储内容的提交表单接口以及相应的接口参数。
       - 如果使用自定义表单组件，还可对表单结构内容做进一步配置。
     - 图标：包括 31 款常用的页面图标。在画布中添加图标后，设置图标颜色。
3. 在画布中，按需拖拽组件，调整画布布局。
4. 调整及管理图层。
   - 调整图层：右键单击画布上的组件，选择相应的操作菜单项，对图层进行置顶、置底、上移、下移、复制、粘贴、剪切或删除操作。

- 管理图层：单击组件/图标栏右侧的面板折叠按钮（<），在 **图层管理** 栏中，选中目标图层，对图层进行重命名、复制或删除操作。

5. 配置完毕后，单击 **预览** 按钮，在预览窗口查看页面在线预览效果或者通过钉钉、支付宝扫描二维码查看手机端效果。页面搭建的内容会实时自动保存，无需手动保存。

以上便是完整的页面搭建过程。新建的页面将展示在 **页面管理** 页面的列表上方。

后续操作

发布页面

5.3. 发布页面

页面发布指将页面产物（即发布后的代码）存储到 mPaaS 对应的对象存储 OSS 中。目前，通过智能搭建发布的页面不会直接发布到 App。您需要获取发布后生成的手机端预览二维码和页面地址，在客户端进行配置，完成数据投放。

发布操作如下：

1. 在页面列表中，选择要发布的页面，单击 **页面搭建** 打开页面搭建编辑器。
2. 确认页面已配置完毕后，单击页面编辑器右上方的 **预览** 按钮查看预览效果。如果预览效果符合预期，可单击预览窗口右下角的 **发布页面** 按钮或者编辑器界面右上方的 **发布** 按钮进行发布。

   发布成功后，系统会生成页面二维码地址和 URL 地址。

6.组件管理

6.1. 创建组件

智能搭建提供丰富的基础组件和多个预置业务组件供您使用。此外，为满足不同业务场景的需求，智能搭建支持导入用户自定义的前端组件，以便在此基础上进行二次开发，以满足更加多样的业务场景。该功能适用于有自己的前端开发资源的客户。

下面介绍如何在智能搭建平台中添加自定义的前端组件。

前提条件

如要添加自定义前端组件，确保已完成相应组件的开发。

操作步骤

1. 登录 mPaaS 控制台后，从左侧导航栏进入 **智能搭建 > 组件管理** 页面。
2. 单击 **创建组件**，在组件创建面板上，完成组件信息配置。

配置项	说明
组件名称	自定义组件名称，在当前应用下需保持唯一。
所属人	组件的所属人。
组件版本	输入组件版本，不能重复提交同一个版本。
组件类型	目前仅支持创建业务组件。
组件存储地址	提供组件资源文件，有两种方式： ◦ 输入 URL：文件访问地址。确保编译好的组件已上传至可被访问的文件目录中。 ◦ 上传文件：将组件文件上传到 OSS。

3. 配置完毕后，单击 **确定** 完成组件添加。新建的组件将展示在组件列表中。

单击 **查看 schema** 菜单可查看定义组件具体元素的 JSON 数据。

6.2. 管理组件

支持对自定义的组件进行管理，包括编辑组件、切换组件版本、新增组件版本和组件下线。

> **说明**
> 当组件处于下线状态时，无法进行编辑、新增版本和切换版本的操作，需要先对组件进行上线操作。

编辑组件

修改组件信息，操作步骤如下：

1. 登录 mPaaS 控制台后，从左侧导航栏进入 **智能搭建 > 组件管理** 页面。

2. 在组件列表中，选择目标组件。可在列表右上方的搜索框中输入组件名称来查找组件。
3. 单击 **操作** 列下的 **编辑** 菜单，在编辑面板中修改组件信息。仅可修改组件的名称、所属人和组件类型。
4. 编辑完毕后，单击 **确定** 保存修改。

新增组件版本

若要修改组件的资源文件，又避免影响已应用当前组件的页面，可通过新增组件版本的方式来实现。

操作步骤如下：

1. 在组件列表中，选择目标组件，单击 **操作** 列下的 **新增版本** 菜单。
2. 在新增版本面板中，设置新的版本号以及组件存储地址后，单击 **确定** 保存版本信息。

切换组件版本

将组件从当前版本切换到其它版本，操作步骤如下：

1. 在组件列表中，选择目标组件，单击 **操作** 列下的 **切换版本** 菜单。
2. 在版本切换面板中，选择要切换的目标版本后，单击 **确定** 即可。

> **重要**
>
> 版本切换后，组件在页面中不会立即更新为指定版本。重新发布页面后，组件才会切换为指定版本。

下线/上线组件

自定义组件创建后默认为上线状态。若无需使用某个自定义组件，可将其下线，下线的组件将不会出现在页面搭建编辑器的组件库中。如需重新使用已下线的组件，可再次将其上线。

- 下线：在组件列表中，选择目标组件，单击 **操作** 列下的 **下线** 菜单，并确认下线。
- 上线：在组件列表中，选择已下线的组件，单击 **操作** 列下的 **上线** 菜单即可。

查看 schema

在组件列表中，单击 **查看 schema** 菜单可查看定义组件具体元素的 JSON 数据。

7.接口管理

7.1. 创建接口

组件可以通过配置接口来实现最终展示内容的动态化。此处配置的数据源接口，在搭建页面时可直接使用，平台将自动完成调用数据接口查询数据的逻辑。

前提条件

您已完成相关数据源和提交接口的开发。

操作步骤

1. 登录 mPaaS 控制台后，从左侧导航栏进入 **智能搭建 > 接口管理** 页面。
2. 单击 **创建接口**，在接口创建面板上，完成接口配置。

配置信息	说明
接口名称	接口的名称
接口类型	支持两种类型的接口： ◦ 提交接口：用于提交表单的接口。 ◦ 数据源：用于展示组件内容的数据接口。
适用组件	当前接口支持的组件，可设置多个。单击输入框，从已有组件下拉列表中选择适用的组件。
接口 URL	接口地址，需确保接口地址支持跨域访问。
请求方法	支持 GET 和 POST 两种请求类型。 选择请求类型后，需要添加接口请求下包含的参数。

3. 配置完毕后，单击 **确定** 完成接口创建。

新增的接口将展示在接口列表中。接口列表展示了当前应用下的所有接口信息，包括接口适用的组件、适用组件的接口校验情况、接口状态。

7.2. 校验接口

在页面搭建编辑器页面，添加组件后进行组件配置时，需要录入对应的接口以获取和保存数据。接口校验的目的是测试配置的接口是否可用，由前端通过对请求接口的数据和适用业务组件的 Schema 进行适配来完成校验。只有经过校验且对当前组件有效的接口才有用。

因此，在使用组件搭建页面前，需要事先对组件涉及的接口进行有效性验证。

接口校验的操作步骤如下：

1. 登录 mPaaS 控制台后，从左侧导航栏进入 **智能搭建 > 接口管理** 页面。
2. 在接口列表中，选择目标接口，单击操作列下的 **接口校验** 菜单，打开接口校验面板，开始校验过程。

接口校验页面展示了当前接口所有适用组件的接口校验状态，包括校验通过、校验失败、未校验、校验中四种状态。

- 对于校验中的组件，可中断校验。一旦退出页面或中断校验，则校验中的组件将不再运行。
- 对于校验失败的组件，可将鼠标悬浮在状态上方，查看失败原因。在进行相应的调整后，可重新单独对该组件进行校验。

3. 校验完毕后，单击 **接口校验** 面板上的 **关闭校验** 按钮，结束校验。
4. 单击操作列下的 **校验结果** 菜单，查看当前接口各适用组件的校验情况。

7.3. 管理接口

支持对接口进行管理，包括查看接口信息、编辑接口信息、停用接口。

查看接口信息

接口管理 页面上的接口列表展示了当前应用下的所有接口及其信息。可在列表右上方的搜索框中输入接口名称来查找指定接口，也可以通过筛选接口类型和状态来查找接口。

字段	描述
接口名称	接口的名称
接口类型	接口的类型，包括提交接口或数据源两种类型。
适用组件	接口适用的组件。
校验结果	展示接口适用组件的校验进度，分别展示已通过校验、未通过校验和待校验的组件个数。
状态	接口的使用状态，包括正常和停用两种状态。

编辑接口信息

在接口列表中，选择目标接口，单击操作列下的 **编辑** 菜单修改接口信息。

接口的参数和参数值发生变化之后，校验状态会变为待校验，需要再次进行接口校验。

停用接口

在接口列表中，选择目标接口，单击操作列下的 **停用** 菜单停用该接口。一旦停用，接口原来的适用组件将无法使用该接口。

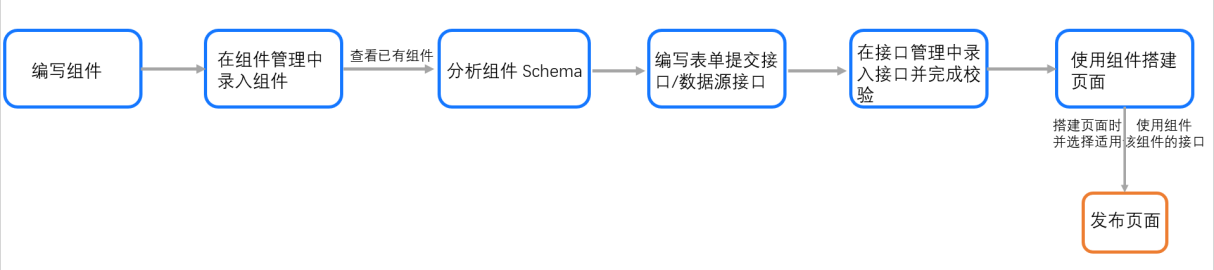
这个停用是在智能搭建中无法再使用，不应该在实际的系统中把接口服务停止掉，除非研发同学能确定的确没有已发布的页面正在使用这个接口。

> **重要**
>
> 这里的停用仅指在智能搭建的页面搭建环节中，组件无法再使用被停用的接口，而非在实际的业务系统中将接口服务停止掉。

8.开发接入指南

8.1. 开发接入流程

使用智能搭建进行模块化页面搭建时，您需要参照以下流程完成组件、接口的研发和接入。



流程步骤说明：

1. 开发人员编写组件，参见 业务组件开发。
2. 在智能搭建控制台的 **组件管理** 模块中，创建组件，将已编写的组件录入到组件库中。
3. 控制台展示已录入的组件，可通过组件列表中的查看 schema 菜单，分析组件 schema。
4. 开发人员编写表单提交接口和数据源接口，参见 接口开发。
5. 在智能搭建控制台的 **接口管理** 模块中，创建接口，录入已编写的接口，并指定该接口适用的组件。然后对接口进行校验，测试接口配置是否可用。
6. 在智能搭建控制台的 **页面管理** 模块中，创建页面，在页面搭建编辑器中，使用上述步骤中开发的组件和接口，进行页面搭建配置。只有通过校验的接口才会在页面搭建编辑器中展示并可以被使用。
7. 完成页面配置后，将页面发布到 PC 端和手机端。

8.2. 业务组件开发

本文针对不同的前端组件开发框架，提供相应的组件开发脚手架供您参考使用，并提供组件开发指导。您可以基于我们提供的脚手架进行开发，也可以使用您自己的组件开发框架完成组件开发。

组件的开发形式多样，开发者可以自由定制插件、扩展等，但必须确保打包上传的文件符合打包规范，具体参见 组件资源上传规范。

业务组件开发脚手架

本文中提供的脚手架（点击相应链接可下载）包含了组件开发的基础代码骨架、组件调试、组件打包等功能。

- 画布 Vue 组件开发脚手架
- 画布 React 组件开发脚手架
- 画布 Rax 组件开发脚手架
- 画布原生 JS 组件开发脚手架

Vue 组件

画布 Vue 组件开发脚手架适用于使用 Vue 技术栈开发的业务组件。

开发步骤：

1. 组件开发

   在 `src/components` 文件夹下开发此组件，其中目录说明为：

```
.
├── index.js 模块文件
├── mock.json 模块可配置的属性的 mock 测试数据
└── schema.json 模块可配置的属性数据
```

您也可以在 `src/components` 下新增其他的目录或文件，根据资源、功能划分进行调整。

2. 组件调试

   运行 `npm start` ，可打开 `http://localhost:8080/` 页面，查看组件是否符合预期。

3. 组件打包

   运行 `npm run build` ，可将组件进行打包，并将 mock.json 和 schema.json 文件拷贝到打包后的文件夹中，将会生成一个 `dist` 文件夹，即组件的打包目录。

4. 组件上传

   通过智能搭建控制台的组件管理模块上传打包好的文件，并填写组件相关信息。

React 组件

画布 React 组件开发脚手架适用于使用 React 技术栈开发的业务组件。

> **重要**
>
> React 16 及以下版本的组件不能与 React 17 及以上版本共存，否则 React 17 的组件会无法响应点击事件。建议全部使用 React 16 及以下版本或 React 17 版本来开发 React 组件。

开发步骤：

1. 组件开发

   在 `src/components` 文件夹下开发此组件，其中目录说明如下：

```
.
├── index.js 模块文件
├── mock.json 模块可配置的属性的 mock 测试数据
└── schema.json 模块可配置的属性数据
```

   您也可以在 `src/components` 下新增其他的目录或文件，根据资源、功能划分进行调整。

2. 组件调试

   运行 `npm start` ，可打开 `http://localhost:3000/` 页面，查看组件是否符合预期。

3. 组件打包

   运行 `npm run build` ，可将组件进行打包，并将 mock.json 和 schema.json 文件拷贝到打包后的文件夹中，将会生成一个 `dist` 文件夹，即组件的打包目录。

4. 组件上传

   通过智能搭建控制台的组件管理模块上传打包好的文件，并填写组件相关信息。

Rax 组件

画布 Rax 组件开发脚手架适用于使用 Rax 技术栈开发的业务组件。

开发步骤：

1. 组件开发

   在 `src` 文件夹下开发此组件，其中目录说明如下：

```
.
├── index.tsx 组件文件
├── index.css 组件样式
├── mock.json 组件可配置的属性的 mock 测试数据
└── schema.json 组件可配置的属性数据
```

您也可以在 `src` 下新增其他的目录或文件，根据资源、功能划分进行调整。

2. 组件调试

   运行 `npm start`，可打开 `http://localhost:3333/` 页面，查看组件是否符合预期。

3. 组件打包

   运行 `npm run build`，可将组件进行打包，并将 mock.json 和 schema.json 文件拷贝到打包后的文件夹中，将会生成一个 `dist` 文件夹，即组件的打包目录。

4. 组件上传

   通过智能搭建控制台的组件管理模块上传打包好的文件，并填写组件相关信息。

原生 JS 组件

画布原生 JS 组件开发脚手架适用于使用原生 JS 开发的业务组件。

开发步骤：

1. 组件开发

   在 `src` 文件夹下开发此组件，其中目录说明如下：

```
.
├── index.html 模块布局
├── index.css 模块样式
├── index.js 模块逻辑
├── mock.json 模块可配置的属性的 mock 测试数据
└── schema.json 模块可配置的属性数据
```

   您也可以在 `src` 下新增其他的目录或文件，根据资源、功能划分进行调整。

2. 组件调试

   运行 `npm start`，可打开 `http://localhost:3000/` 页面，查看组件是否符合预期。

3. 组件打包

   运行 `npm run build`，可将组件进行打包，并将 mock.json 和 schema.json 文件拷贝到打包后的文件夹中，将会生成一个 `dist` 文件夹，即组件的打包目录。

4. 组件上传

   通过智能搭建控制台的组件管理模块上传打包好的文件，并填写组件相关信息。

组件资源上传规范

在完成组件开发后，打包上传组件资源时，需要遵守资源文件上传规范。

bundle.js 文件

- Vue & React 组件：按照 CMD（Common Module Definition，通用模块定义）规范导出 Named Module 组件定义，单文件 Bundle 打包，格式如下：

```
define('moduleName', [], function (require, exports, module) {
  module.exports = function 入口函数
})
```

暴露的入口函数如下：

```
function render({
  element: string | HTMLElement, // id 选择器/Element
  props: {[key: string]: any}, // props 属性
}) {
  const title = props.title || mock.title;
  // 自定义的组件渲染逻辑
}
```

- 原生 JS 组件：打包为可执行的 HTML 文件即可。
- Rax 组件：按照 rax-component 打包规范打包即可。

schema.json 文件

schema.json 用来写业务组件的可配置属性。schema.json 文件的基本格式如下：

```
{
  "title": "Schema",
  "type": "object",
  "properties": {
    ... // 字段属性值
  }
}
```

schema.json 生成方法：

1. 打开 在线表单设计器。
2. 搭建组件的属性配置表单。
3. 复制 schema 到项目的 schema.json 即可。

schema.json 文件中会有一些比较特殊的配置项，请参考下述说明进行配置。

情况 1

针对自定义表单组件，schema.json 会有一个用于打开表单生成器的配置项，此时新增一个 `x-component:"FormGenerator"` 即可，编辑器会特殊解析。

mPaaS 前端需要将表单生成器封装为一个完整的组件，其中定制好需要的表单组件、提交按钮等，使用的组件参见 表单设计器开发指南。

画布侧如果解析到 `x-component:"FormGenerator"` ，就会调用表单生成器。

schema 如下：

```
"xxx": {
  "title": "设计表单",
  "type": "object",
  "x-component": "FormGenerator"  // 此处 FormGenerator 仅作为示例，可以换成其他名字，需要在 mPaaS 和画布前端保持一致
}
```

情况 2

针对数据源下拉列表，schema.json 会有一个用于给该业务组件选择合适的数据源接口的配置项。此时，新增一个下拉列表即可，编辑器会特殊解析。

schema 如下：

```
"xxx": {
  "title": "数据源", // 这里是在编辑器的右侧面板上显示的
  "name": "dataSource",
  "type": "string",
  "x-decorator": "FormItem",
  "x-component": "Select",
    }
```

情况 3

针对数据源的参数表单，schema.json 会有一个配置项是显示该业务组件的数据源接口所需要的参数的，该参数是在创建接口时所填写。此时，新增一个 `x-component:"GetParamsData"` 即可，编辑器会特殊解析。

该面板上的参数和参数值会在调用数据源接口的时候分别作为参数 key 和 value 传递。

schema 如下：

```
"xxx": {
  "type": "array",
  "x-component": "GetParamsData",
    }
```

情况 4

针对提交数据下拉列表，schema.json 会有一个用于给该业务组件选择合适的提交数据接口的配置项。此时，新增一个下拉列表即可，编辑器会特殊解析。

schema 如下：

```
"xxx": {
  "title": "提交接口", // 这里是在编辑器的右侧面板上显示的
  "name": "dataSource",
  "type": "string",
  "x-decorator": "FormItem",
  "x-component": "Select",
    }
```

情况 5

针对提交数据的参数表单，schema.json 会有一个配置项是显示该业务组件的数据源接口所需要的参数的，该参数是在创建接口时所填写。此时，新增一个 `x-component:"SaveParamsData"` 即可，编辑器会特殊解析。

该面板上的参数和参数值会在调用提交数据接口的时候分别作为参数 key 和 value 传递。

schema 如下：

```
"xxx": {
  "type": "array",
  "x-component": "SaveParamsData"
    }
```

情况 6

业务组件中有数组列表。schema.json 中会有一个配置项是用于配置数组列表中的数据的。此时，新增一个 `x-component:"ArrayCards"` 即可。

这样即可在编辑器中修改数组列表中的每一项和排序等。

schema 如下：

```
"xxx": {
      "type": "array",
      "x-component": "ArrayCards",
      "x-component-props": {
        "title": "列表" // 这里的标题为显示在编辑器中的，此时的"列表"仅为示例
      },
      "items": { // 这里的"items"为固定的
        "type": "object",
        "properties": {
          "title": {
            "title": "标题",
            "type": "string",
            "x-component": "Input",
            "required": true
          },
          ... // 字段属性
        }
      }
    }
```

mock.json 文件

mock.json 文件内容格式如下：

```
{
  "title":"标题"，
  ...其他字段
}
```

8.3. 接口开发

您可以参考本文完成组件接口的开发和使用。

整个过程主要分为三步：

1. 获取业务组件数据结构。

   业务组件的数据结构和该组件的 mock.json 文件除去可选参数字段外，其余字段均保持一致。

2. 编写接口。

   - **表单提交接口**

     修改组件后，会调用该接口保存组件的最新数据。

```
{
  apiData:'{ // 组件的数据(jsonString)
    "title":"标题",
    ...
    "params": "{"id":"0"}", // params 代表该提交接口所需要的额外的参数
  }',
  pageId: 134, // 页面id
  componentId: "3bb79d72-52f6-4fe8-b07a-95e3ae193c0c" // 组件在编辑器中生成在画布中的唯一标识
}
```

- **数据源接口**

  获取组件的最新数据。

```
// 参数
{
  pageId: 134,
  componentId:"3bb79d72-52f6-4fe8-b07a-95e3ae193c0c",
  apiData: '{
    "params":"{"id":"0"}", // 额外的参数
  }'
}

//返回值
  apiData: '{ // 组件数据(jsonString)
    "title": "标题",
    ...
    "params": "{"id":"0"}" ,
  }',
  pageId: 134,
  componentId:"3bb79d72-52f6-4fe8-b07a-95e3ae193c0c"
```

3. 录入接口。

   在智能搭建控制台的 **接口管理** 模块中，创建接口，录入已编写的接口，并指定该接口适用的组件。更多内容，参见 创建接口。接口创建完毕后，需要对接口和所适用的组件进行校验。接口校验通过后，您才能在页面搭建编辑器中看到和使用该接口。