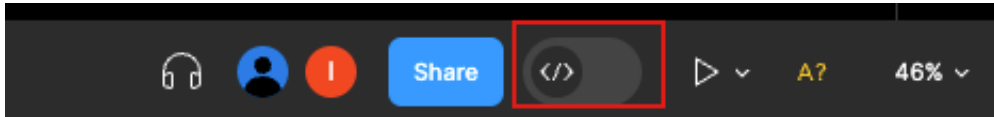


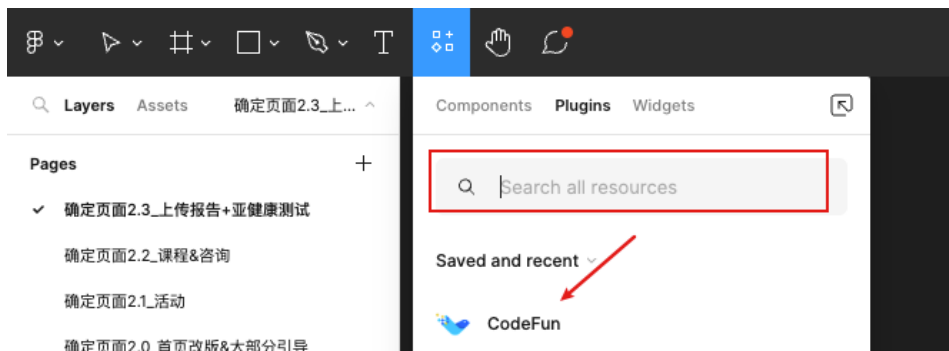
D2C workflow 指导

1、在 figma 中导出设计稿到 codeFun

1、打开 figma （推荐下载桌面版），找到右上角的按钮，切换至设计模式。需要找 伊卡 开通设计模式的权限



2、点击左上角的资源按钮，找到插件 tab, 搜索 codeFun，安装该插件



3、选中需要转代码的画板，选择账号和项目，上传

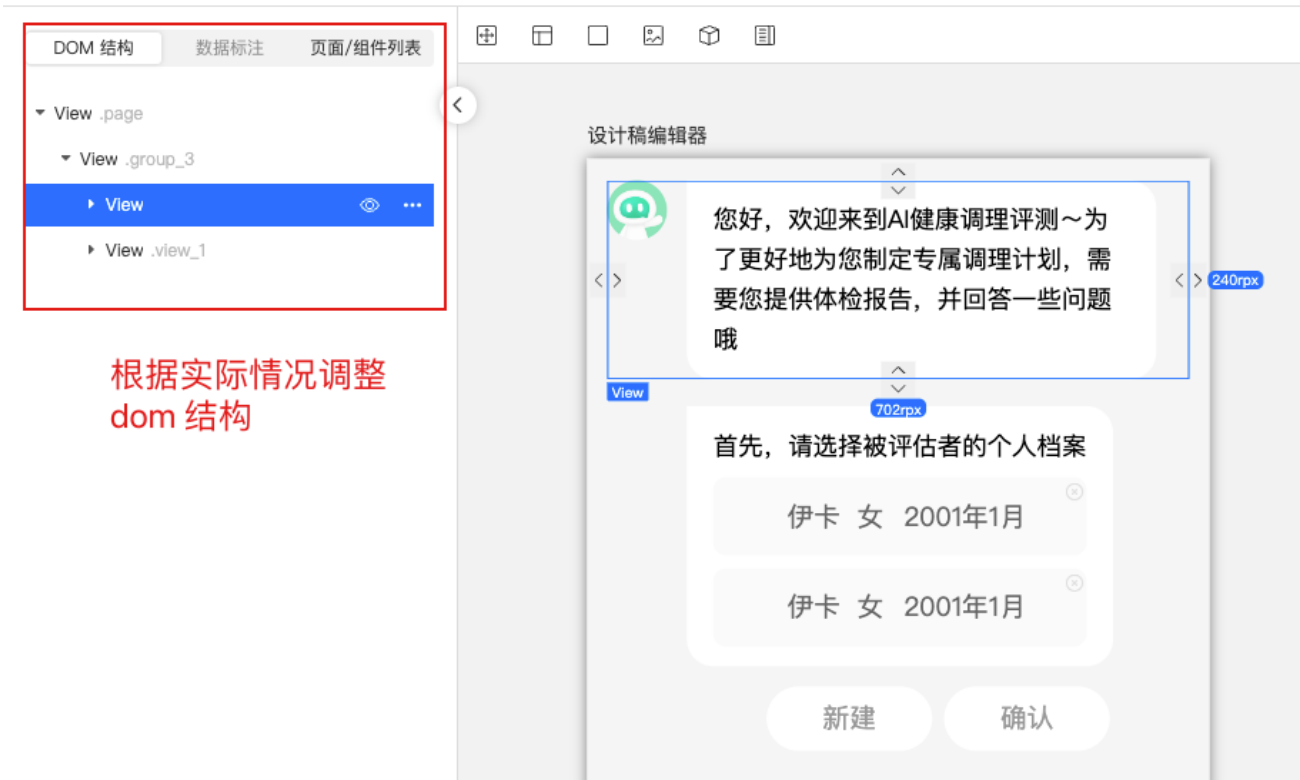


4、上传后，点击查看项目，进入工作空间，选择刚刚导入的页面即可

2、调整 dom 层级结构

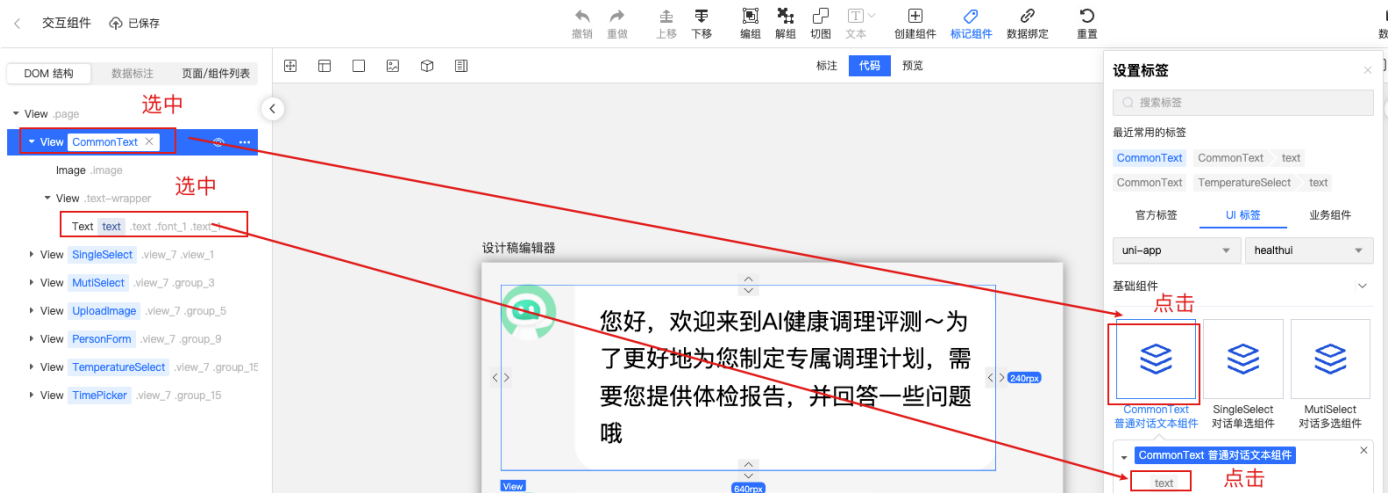
codeFun 的自动识别可能存在结构错误或者部分与视觉稿差异的地方，为了让导出的代码的结构能够更加清晰，便于维护，我们需要在 codeFun 编辑器的左侧 dom 结构面板进行调整。调整的原则如下：

- 一个页面可以由多个小的组件组成，明显属于同一个区域或内容的可以看作是同一个组件，他们应该被移动至同一个 View 下面
- 对于零散的 view 元素，可以点击顶部菜单栏的【编组】按钮将其聚合在一起
- 对于特殊的字体或者零散的图片元素，可以点击顶部的【切图】按钮将其变成图片，到时候导出成图片直接引用即可



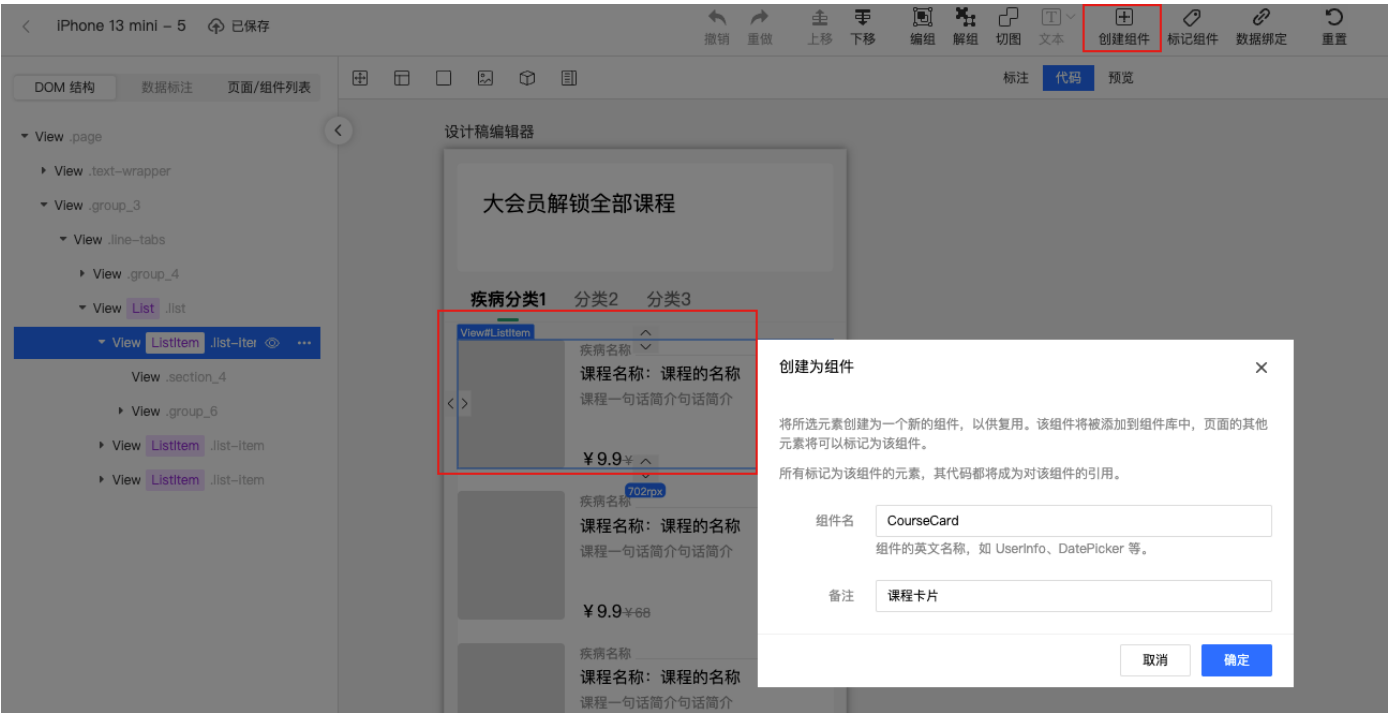
3、如果组件库已经满足需求，直接标记原生组件，uView 组件以及自定义组件

在 codeFun 页面上选中元素，标记组件。在标记组件的【UI 标签】tab下有三类组件，原生组件，uview 组件和自定义组件，根据用到的组件的实际情况进行标记



4、组件库不满足需求，请新增业务组件

以课程卡片组件为例，可以看到其为一个 Item 而且多次被用到，此时则考虑抽离出业务组件



按上图操作创建组件，可以到项目的组件中查看，并且按照2、3步所说的进行编辑。然后下载代码后，直接将其复制到项目的 src/components 目录下即可。然后页面则正常导出下载执行走脚本流程，后面将会讲到。



5、下载代码

修改组件的名字。如图设置好代码平台、css 平台以及像素单位，下载代码



6、执行脚本

下载 health-d2c 项目：<https://gitlab.yunjiglobal.com/bigdata-fe/health-d2c>（找晨火开权限）

将 codeFun 下载的代码放到 health-d2c 项目的 source 下面，然后修改 `config.yaml`

- `target_path` 代表该代码需要放置目标目录，如果该路径以 pages 结尾，则会新建一个页面路径，如果无需新建，则请填写到具体页面路径
- `route_config_file` 代表路由配置文件的路径，一般默认该路径即可

然后，执行 `python3 convert.py`，代码已经转化完毕，并且导入项目当中了。

7、重启项目，查看效果

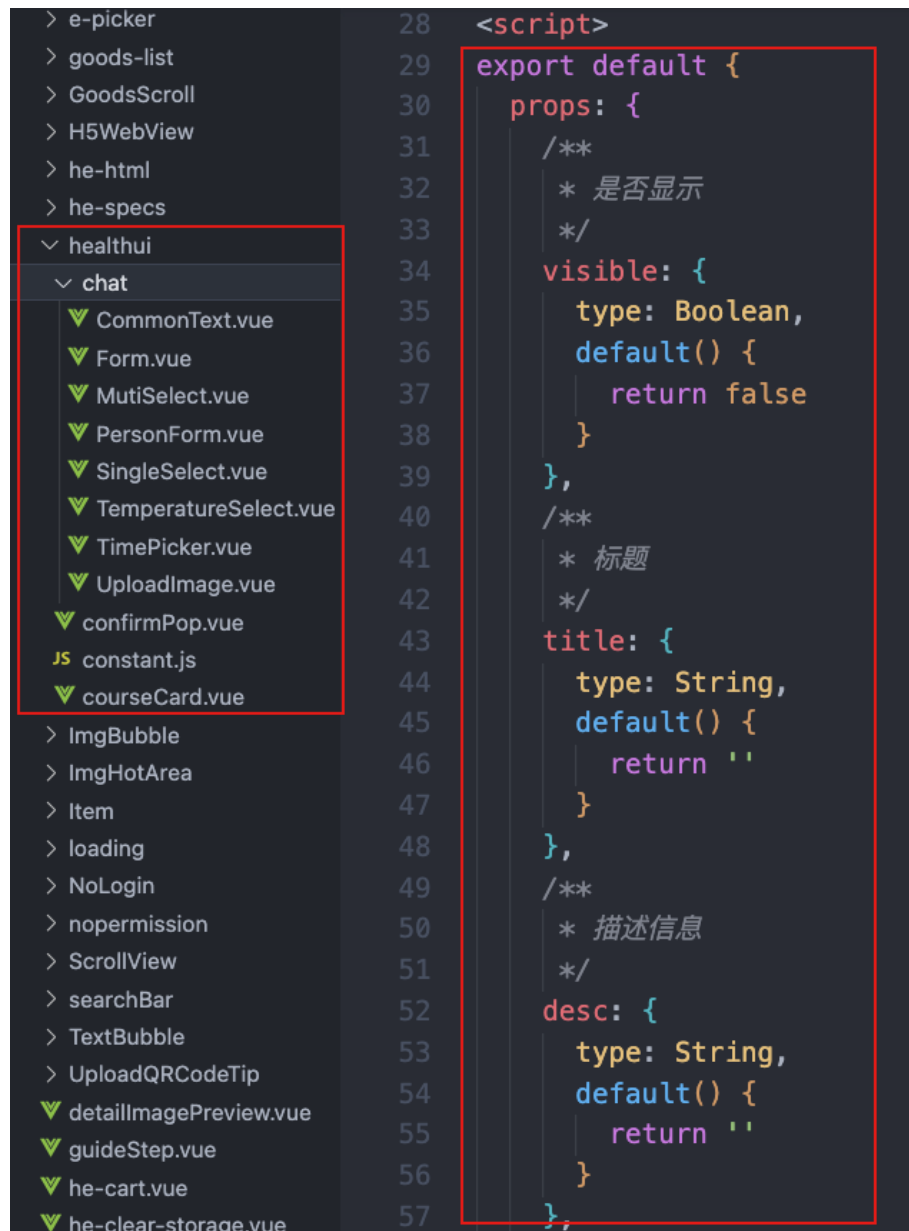
运行 `dev:mp-weixin:mars` 重启项目，在模拟器中添加编译模式，输入页面路径，即可查看。不清楚路径的，可以进入项目目录，看到刚刚自动修改的 `miniJson.json` 的路径新增路径 path

8、添加逻辑代码，完成项目

其他：关于如何新增自定义组件

对于前面第 4 步中提到的新增业务组件。如果该组件被多次使用而且需要传递大量 props，建议将其转化为自定义组件。

- 首先将其移动至，`src/components/healthui` 下，然后规范设置好组件名、props 以及注释



- 然后打开 codeFun，进入 healthui 组件库。然后点击左侧菜单添加刚刚的组件，名称和导入路径一定要配置正确，配置参考如图



添加标签

×

* 名称

confirmPop2

中文名称

弹窗确认组件

智能识别

请选择

依赖导入

本地依赖

导入名称

confirmPop2

是否解构导入

导入路径

@/components/healthui/confirmPop2.vue

* 组件类型

基础组件

组件介绍

文档链接

- 添加好后，请完善组件的属性。组件的名称、类型、默认值请按照代码中的实际情况进行填写，填写后保存即可。

编译模版

1

<confirmPop

2

visible="\$.props.visible"

3

title="\$.props.title"

4

desc="\$.props.desc"

5

confirmText="\$.props.confirmText"

6

cancelText="\$.props.cancelText"

7

confirmBtnStyle="\$.props.confirmBtnStyle"

8

cancelBtnStyle="\$.props.cancelBtnStyle"

9

confirmTextStyle="\$.props.confirmTextStyle"

10

cancelTextStyle="\$.props.cancelTextStyle"

11

>/>

默认插槽

生成模板

属性	排序	类型	名称	属性说明	默认值	属性分类	提取变量	添加属性
		Boolean	visible	是否显示	true	基础属性	<div></div>	删除
		String	title	标题	这是一个测试标题	基础属性	<div></div>	删除
		String	desc	描述信息	这是一段描述信息这是一!	基础属性	<div></div>	删除
		String	confirmText	确认按钮文本	确认	基础属性	<div></div>	删除
		String	cancelText	取消按钮文本	取消	基础属性	<div></div>	删除
		String	confirmBtnStyle	确认按钮样式		基础属性	<div></div>	删除
		String	cancelBtnStyle	取消按钮样式		基础属性	<div></div>	删除
		String	confirmTextStyle	确认按钮文本样式		基础属性	<div></div>	删除
		String	cancelTextStyle	取消按钮文本样式		基础属性	<div></div>	删除

- 使用：按照前面流程中第 3 步进行标记自定义组件即可