

uni-app（一套代码，多端发行）

如果你会使用Vue那咱们继续吧！

什么是uni-app？

uni-app 是一个使用 [Vue.js](#) 开发所有前端应用的框架，开发者编写一套代码，可发布到iOS、Android、H5、以及各种小程序（微信/支付宝/百度/头条/QQ/钉钉）等多个平台。

诞生背景

多端泛滥 iOS、Android、H5、以及各种小程序（微信/支付宝/百度/头条/QQ/钉钉）多套平台，多套文档，加大开发维护成本

uni-app特点

1. 跨平台

一套代码多端发行，而不失优雅（条件编译，保留不同平台独有特色功能方法调用）

```
1. <!--#ifdef MP-WEIXIN-->
2.   <view>只会编译小程序</view>
3. <!--endif-->
4.
5. <!--#ifdef XX-XXX-->
6.   <view>只会编译XXX</view>
7. <!--endif-->
```

2. 通用技术栈，学习成本低

Vue的语法，微信小程序Api，内嵌mpvue可以直接迁移，如果你会Vue可以直接入门

3. 生态丰富

支持npm下载第三方模块，支持小程序自定义组件，SDK，兼容mpvue组件，支持原生混合编码

安装

可以下载两个进行配置，测试微信小程序与支付宝小程序同步实现

编译工具下载[HBuilderX](#)

微信小程序开发[IDE](#)

支付宝小程序 [IDE](#)

推荐使用 HBuilderX 进行开发

跨多端小程序开发

创建目录

1. 1. 打开HBuilderX
2. 2. 左上角创建 项目
3. 1. 选择uni-app项目
4. 2. 下面是提供几个Demo项目, 可以创建学习
5. 3. 项目名称能不能大写, 创建项目

运行项目

1. 找到创建项目uni里面的项目目录
2. 找到 manifest.js文件输入
 - i. 里面有所有平台配置文件
 - a. 需要不通平台测试, 需要分别配置不同平台
 - ii. 选择微信小程序配置
 - a. 配置小程序AppID
 - b. 点击最上面工具栏选择运行
 - c. 选择运行到小程序模拟器
 - d. 选择微信小程序开发工具
 - a. 第一次配置小程序开发者工具, 需要打开服务
3. 打开微信小程序开发左上角小
 1. 3. 安全设置, 将服务端口开启。||工具 -> 设置 -> 安全设置, 将服务端口开启。
4. 选择HBuilderX停止服务, 重新运行自动打开编辑工具

.nvue 是对weex的增强。如果你不开发App, 那么你不需nvue。

发布测试

选择HBuilder最上面工具栏, 点击发布进行打包, 根据命令行提示在点击微信开发者工具发布

注意不要直接修改微信开发工具目录, 以HBuilder开发目录为主

目录介绍

与小程序配置相似

1. `pages` //存放分页目录
2. `static` //存放应用引用静态资源
3. `main.js` //入口文件
4. `manifest.json` //跨平台所有配置项文件
5. `pages.json` //全局配置文件, 类似App.json
6. `uni.scss` //全局scss文件, 在分页任何位置, 注意 `style lang="scss"` 需要设置 `scss`
7. `App.vue` // 应用配置, 用来配置App全局样式以及监听

统一参照uni-app官方文档

初探uni-app

添加分页

1. 鼠标移动pages右键
2. 右键新建页面, 注意检查pages.json文件会自动写入pages项, 文件路径
 - i. pages.json文件 `style` 为分页配置项, 类似小程序 `page.json`
 - ii. 修改 `pages.json`

知识点: uni-app分页配置在style里面写, globalStyle写全部样式配置, 在一个配置文件, 这里是有区别原生小程序开发, 具体参照uni-app文档进行配置 `pages.json`

配置项

配置项

1. `navigationBarBackgroundColor` `HexColor` `// #000000` 导航栏背景颜色
2. `navigationBarTextStyle` `String` `white` //导航栏标题颜色, 仅支持 `black/white`
3. `navigationBarTitleText` `String` //导航栏标题文字内容
4. `navigationStyle` `String` `default` //导航栏样式, 仅支持 `default/custom`。
5. `backgroundColor` `HexColor` `// #ffffff` 下拉窗口的背景色 微信小程序

全局配置

1. `globalStyle`: {
2. `// ... 全局配置`

```
3.   }
```

局部配置 参照 `pages.style`内容

```
1.   > pages数组, 决定谁排在前面
```

修改pages导致文件查找失败, 检查无误, 重启编辑工具

```
1.   pages:[
2.     {
3.       path: "xxx/xxx/xxx",
4.       style: {
5.         //... 局部配置
6.       }
7.     }
8.   ]
```

系统自带底部tabBar栏配置

如果配置tabBar需要保证abBar第一个路由, 配置在第一个pages的第一个

```
1.   "tabBar": {
2.     "color": "#7A7E83",
3.     "selectedColor": "#3cc51f",
4.     "borderStyle": "black",
5.     "backgroundColor": "#ffffff",
6.     "list": [{
7.       "pagePath": "pages/component/index",
8.       "iconPath": "static/image/icon_component.png",
9.       "selectedIconPath": "static/image/icon_component_HL.png",
10.      "text": "组件"
11.    }, {
12.      "pagePath": "pages/API/index",
13.      "iconPath": "static/image/icon_API.png",
14.      "selectedIconPath": "static/image/icon_API_HL.png",
15.      "text": "接口"
16.    }]
17.   }
```

常用基本操作

可以尝试根据Vue使用扩展, 在微信小程序, 出来效果可行

重点, 统一优先Vue操作, uniApi调用

表达式

仅此而已, 有Vue使用相似, 不能使用流程控制语句

1. `<view>{{name}}</view> <!--读取变量-->`
- 2.
3. `<view>{{msg+'Msea'}}</view><!--拼接变量-->`
4. `<view>{{10-9}}</view> <!--运算符-->`
5. `<view>{{0||"NB"}}</view> <!--逻辑运算符-->`
- 6.
7. `<!--数据类型方法-->`
8. `<view>{{'Msea'.indexOf('xxx')!==-1?'太原最靓仔':'no'}}</view>`
9. `<view>{{"我爱北京天安门".substr(0,2)}}</view>`

属性绑定

1. `<image :src=""https://www.w3school.com.cn/i/eg_tulip.jpg""/>`

动态样式

1. `<view :class=""`${'box'}`+' '+12`">{{1+1}}</view>`
2. `<view :class=""{className:true}""</view>`
3. `<view :class=""['box',true?'col2':'']"></view>`
4. `<view class="box1212" :class=""['box',true?'col2':'']"></view>`
5. `<view :style=""{ color: 'red', fontSize: 20 + 'px' }"">666</view>`
6. `<view :style=""[{ color: pink, fontSize: 20 + 'px' }]"">777</view>`

指令

1. `<view class="box" v-show="true">v-show</view>`
- 2.
3. `<view v-if="text==='登陆'">{{'欢迎'+name+'登陆'}}</view>`
4. `<view v-else-if="text==='未登陆'">未登陆</view>`
5. `<view v-else>!!!!</view>`
- 6.
7. `<input type="text" v-model="val">{{val}} <!--数据双向绑定-->`
8. `<!--支持v-model修饰符-->`
9. `<input type="text" v-model.number="val">{{val+11}}`

事件

1. `<!--事件, 阻止事件冒泡-->`
2. `<view @click="test1(true)"> <!--传入参数-->`
3. `<button @click.stop="test2">test</button>`
4. `</view>`

遍历

```
1. <!--数组遍历-->
2. <ul id="example-1">
3.   <li v-for="(item, index) in items">
4.     {{ item.message }}
5.   </li>
6. </ul>
7.
8. <!--对象遍历-->
9. <div v-for="(value, name, index) in object">
10.   {{ index }}. {{ name }}: {{ value }}
11. </div>
```

路由跳转

页面栈最多十层。使用uni.navigateTo频繁切换, 会导致栈溢出, 跳转失败

```
1. // 非tabbar配置的页面我们使用 navigateTo跳转时保留老页面, 一般用于需要返回
2. uni.navigateTo({
3.   url: "../one/one?name=Msea"
4. })
5. // 跳转pages.json>tabbar>配置过的页面, 使用 switchTab
6. uni.switchTab({
7.   url: "../one/one"
8. })
9. // 不保留当前页面, 跳转非配置页面
10. uni.redirectTo({
11.   url: "../one/one"
12. })
```

响应式单位rpx

推荐使用最近公告推荐使用rpx替代upx

基于苹果6 1px =2rpx

```
1. <view class="box1"></view>
2. <view class="box2"></view>
3. <style lang="scss">
4.   .box1{
5.     width:200rpx;
6.     height:200rpx;
7.     background:red;
8.   }
```

```
8.   }
9.   .box2{
10.    width:100px;
11.    height:100px;
12.    background:pink;
13.   }
14. </style>
```

节点操作

微信小程序里面没有 window, document对象, 那如果需要进行dom操作

```
1.  var query=uni.createSelectorQuery();
2.
3.  query.select(".titlerh").boundingClientRect((nodes)=>{
4.    console.log(nodes)
5.  }).exec();
6.
7.  query.select(".box1")
8.    .boundingClientRect()
9.    .selectAll(".box2")
10.   .boundingClientRect()
11.   .exec((res)=>{
12.     console.log(res);
13.   });
```

组件

创建文件夹 components目录统一写法, 鼠标右键创建组件

所有组件与属性名都是小写, 单词之间以连字符  连接。

```
1.  <template>
2.    <uni-test/>
3.  </template>
4.  <script>
5.    import uniTest from "../../components/test.vue"
6.    export default {
7.      data() {
8.        return {};
9.      },
10.     methods:{
```

```
11.   test() {
12.     this.text = "登录";
13.   }
14. },
15.   components: {
16.     uniTest //引入组件
17.   }
18. }
19. </script>
20. <style lang="scss">
21.   ...style
22. </style>
```

生命周期

应用生命周期 App.vue

```
1.  App({
2.    onLaunch (options) {
3.      console.log("小程序初始化");
4.    },
5.    onShow (options)
6.      console.log("小程序显示");
7.    },
8.    onHide () {
9.      console.log("小程序隐藏");
10.   }
11. })
```

分页生命周期 pages

```
1.  Page({
2.    onLoad: function(options) {
3.      // 页面创建时执行
4.      console.log("页面加载");
5.    },
6.    onShow: function() {
7.      console.log("页面进入");
8.    },
9.    onReady: function() {
10.     console.log("页面首次渲染完毕时执行");
11.   },
```



```
12. onHide: function() {
13.   console.log("页面离开");
14. },
15. onPullDownRefresh: function() {
16.   // 触发下拉刷新时执行
17.   console.log("下拉触发");
18.   //enablePullDownRefresh 开启下拉
19. },
20. onReachBottom: function() {
21.   // 页面触底时执行
22.   console.log("下拉到底");
23. },
24. onShareAppMessage: function (e) {
25.   // 页面被用户分享时执行
26.   //通过按钮调用
27.
28.   //点击触发弹窗
29.   <button open-type="share">分享</button>
30.   console.log("用户分享");
31.   return {
32.     title: '妹子图片',
33.     path: '/pages/index/index', //当前页面 path
34.     imageUrl: "/images/1.jpg",
35.     desc: '面向百度开发',
36.     success: (res) => {
37.       console.log("转发成功", res);
38.     },
39.     fail: (res) => {
40.       console.log("转发失败", res);
41.     }
42.   }
43. },
44. onPageScroll: function() {
45.   console.log("页面滚动时执行");
46. },
47. onResize: function() {
48.   console.log("屏幕旋转触发");
49. }
50. })
```

组件生命周期

uni-app components组件支持的生命周期，与vue标准组件的生命周期相同。wx小程序支持生命周期

1. `beforeCreate()` {} // 在实例初始化之后被调用
2. `created()` {} // 在实例创建完成后被立即调用。
3. `beforeMount()` {} // 在挂载开始之前被调用。
4. `mounted()` {} // 挂载到实例上去之后调用。
5. `beforeDestroy()` {} // 实例销毁之前调用。在这一步，实例仍然完全可用。

UI组件

uni-ui

使用方法与微信小程序一致, 推荐使用uni-ui

```
1. <scroll-view
2.   :scroll-x="true"
3.   style="border:1px solid red;white-space:nowrap"
4.   @scroll="scroll">
5.   <view style="
6.     background:red;
7.     width:200px;height:100px;
8.     display:inline-block;"
9.   ></view>
10.  <view style="background:yellow;width:200px;height:100px;display:inline-block;"></view>
11.  <view style="background:pink;width:200px;height:100px;display:inline-block;"></view>
12.  <view style="background:blue;width:200px;height:100px;display:inline-block;"></view>
13. </scroll-view>
```

引入三方UI

例如 Vant Weapp [下载](#)

1. 根目录下创建 wxcomponents
2. 将下载包内部 dist包粘贴到uni-app根目录wxcomponents文件夹下
3. 配置

```
1. {
2.   "path" : "pages/one/one",
3.   "style" : {
4.     "usingComponents":{
5.       "van-button": "/wxcomponents/dist/button/index" //参照wx组件引
6.     }
7.   }
```

```
8. }
```

1. 在App.vue, 引入样式

```
1. <style>
```

```
2. /*每个页面公共css */
```

```
3. @import "/wxcomponents/dist/common/index.wxss";
```

```
4. </style>
```

1. 在one分页内直接使用van-button组件, 不需要引入

2. 注意如果报错, 检查路径如果没有问题, 可以选择重编辑器重新启动微信开发工具

常用API调用

获取用户授权弹窗

```
1. <button open-type="getUserInfo" @getuserinfo="eventName"> 获取头像昵称 </button>
```

```
2. getUserInfo: function(e) {
```

```
3.   app.globalData.userInfo = e.detail.userInfo
```

```
4.   this.setData({
```

```
5.     userInfo: e.detail.userInfo,
```

```
6.     hasUserInfo: true
```

```
7.   })
```

```
8. }
```

返回所有用户授权

```
1. wx.getSetting({success: ()=>{"返回用户所有用户授权"}})
```

照相机接口

因为, 调用照相机获取临时图片格式, 直接上传三方服务器, 是不支持的, 需要微信做解析, 转发

wx.uploadFile 上传微信需要做转发看不到传输的参数

```
1. uni.chooseImage({
```

```
2.   count: 1, //图片张数
```

```
3.   sizeType: ['original', 'compressed'], //原图, 压缩图,
```

```
4.   sourceType: ['album', 'camera'], //本地相册, 拍照
```

```
5.   success :res=> {
```

```
6.     const tempFilePaths = res.tempFilePaths//微信小程序图片临时路径
```

```
7.     this.setData({tempFilePaths});
```

```
8.   }
```

```
9. })
```

```
10. // 预览接口
```

```
11. viewImgs(index) {
```

```
12.   . . . . .
```

```
12. uni.previewImage({
13.   current: this.data.tempFilePaths[index], // 当前显示图片的http链接
14.   urls: this.data.tempFilePaths // 需要预览的图片http链接列表
15. });
16. },
17.
18. //小程序图片上传
19. fileUpload() {
20.   uni.uploadFile({
21.     url: "http://wxs.ixinangou.net/index/index/dofiles",
22.     filePath: this.data.tempFilePaths[0],
23.     name: "file", //上传图片key
24.     formData: {
25.       user: "MSea" //需要额外携带参数
26.     },
27.     header: {
28.       "content-type": "multipart/form-data"
29.     },
30.     success: res => {
31.       console.log("data");
32.     }
33.   });
34. }
```

请求

微信原生请求接口

注意设置, 不校验合法域名, 回忆怎么添加合法域名

```
1. //GET 会自动拼接参数
2. //queryStringParams
3. uni.request({
4.   method: "GET",
5.   url: "https://cnodejs.org/api/v1/topics",
6.   data: {
7.     uname: "Msea"
8.   },
9.   success: (res) => {
10.    console.log(res)
11.  }
12. })
13. //POST 默认参数为payload, 为json
```

```
14. uni.request({
15.   method: "POST",
16.   url: "https://cnodejs.org/api/v1/topics",
17.   data: {
18.     uname: "Msea"
19.   },
20.   success: (res) => {
21.     console.log(res)
22.   }
23. })
24. //POST form-data 数据
25. // 'content-type': 'multipart/form-data' 用于文件上传
26. uni.request({
27.   url: 'https://cnodejs.org/api/v1/topic/5433d5e4e737cbe96dcef312',
28.   data: {fileId: '123'},
29.   method: 'POST',
30.   header: {
31.     'content-type': 'application/x-www-form-urlencoded'
32.   },
33.   success: function(res) {}
34. })
```

地图使用

地图组件

```
1. <map
2.   id="map"
3.   longitude="经度"
4.   latitude="纬度"
5.   scale="缩放级别(14)"
6.   bindcontroltap="点击地图触发FN"
7.   markers="{{添加标记}}"
8.   bindtap="markertap" 点解地图触发
9.   show-location 将地图中心移置当前定位点，此时需设置地图组件 show-location 为true。
10. style="width: 100%; height: 300px;"></map>
```

移动地图中心点

移动端测试有效

```
1.   onShow: function() {
2.     // 地图上定位点
```

```
2. // 地图上上下文对象
3. this.mapCtx = uni.createMapContext('map');
4. },
5.
6. ckMap(e) {
7. //小程序不支持 Object.assign
8. var temp={
9. iconPath: "/assets/img/local.png",
10. id: 0,
11. width: 25,
12. height: 25,
13. ... e.detail
14. }
15. var markers=this.data.markers;
16. markers.push(temp);
17. this.setData({markers}, ()=>{
18. var data={
19. ... e.detail
20. };
21. this.mapCtx.moveToLocation(data)
22. })
23. }
```

定位

```
1. uni.getLocation({
2. type: 'gcj02', //腾讯地图坐标系
3. success:(res)=> {
4. const latitude = res.latitude
5. const longitude = res.longitude
6. }
7. })
```

用户授权弹窗

app.json配置文件

```
1. //app.json
2. {
3. "pages": ["pages/index/index"],
4. "permission": {
5. "scope.userLocation": {
6. "desc": "熊猫创客需要获取您的位置亲" // 高速公路行驶持续后台定位
7. }
8. }
```

```
8.   }  
9.   }
```

获取当前位置并移动地图中心点

```
1.   wx.getLocation({  
2.     type: 'gcj02', //腾讯地图坐标系  
3.     success: (res) => {  
4.       const latitude = res.latitude  
5.       const longitude = res.longitude  
6.       var temp = {  
7.         iconPath: "/assets/img/local.png",  
8.         id: 0,  
9.         width: 25,  
10.        height: 25,  
11.        latitude,  
12.        longitude  
13.      }  
14.      var markers = this.data.markers;  
15.      markers.push(temp);  
16.      this.setData({  
17.        markers  
18.      }, () => {  
19.        var data = {  
20.          latitude,  
21.          longitude  
22.        };  
23.        this.mapCtx.moveToLocation(data)  
24.      })  
25.    }  
26.  })
```

vuex

uni-app已经内置了vuex

1. 在项目的根目录下，创建一个名为store的文件夹

```
1.   import Vue from 'vue'  
2.   import Vuex from 'vuex'  
3.   Vue.use(Vuex)  
4.   const store = new Vuex.Store({  
5.     state: {
```

```
5.   state: {  
6.     num:0  
7.   },  
8.   mutations: {  
9.     add(store) {  
10.    store.num++;  
11.    }  
12.  },  
13.  actions: {}  
14. })  
15. export default store
```

1. main.js入口文件引入

```
1. import store from './store'  
2. Vue.prototype.$store = store;  
3.  
4. const app = new Vue({  
5.   ...App,  
6.   store  
7. })  
8. app.$mount()
```

1. 组件内引入

```
1. <template>  
2.   <view>  
3.     <view>{{num}}</view>  
4.     <button @click="test">test1</button>  
5.   </view>  
6. </template>  
7.  
8. <script>  
9.   import {  
10.    mapState,  
11.    mapMutations  
12.  } from 'vuex';  
13.   export default {  
14.     computed: {  
15.       ...mapState(['num'])  
16.     },  
17.     methods: {  
18.       test() {  
19.         console.log(this.$store.commit("add"))
```



```
20.   }
21.   }
22.   }
23. </script>
```

条件编译

可以安装支付宝小程序进行测试

```
1.   #ifdef %PLATFORM%
2.   //这些代码只在该平台编译
3.   #endif
4.
5.   #ifdef :   if defined  仅在某个平台编译
6.   #ifndef :   if not defined  在除该平台的其他编译
7.   #endif :   end if  结束条件编译
8.   %PLATFORM% 需要编译的平台，上面的MP就是各个小程序的意思
9.
10.  常用 %PLATFORM%
11.  h5 h5平台
12.  MP-WEIXIN 微信小程序
13.  MP-ALIPAY 支付宝小程序
14.  MP-BAIDU 百度小程序
15.  MP-TOUTIAO 头条小程序
```

总结：优先使用Vue用法，只要实现微信开发，支持wx, ui, api, 统一使用uni的Api, 多看文档