

The background is a complex collage. It includes torn pieces of paper with various text fragments such as "LEUC", "HE T", "photos, ge. Let's anymore.", "ns have to h", "outcome of", "olicy.", "FREIHEIT", "ANTIFASCHISTISCH", "ERKAMPFEN!", "RONAL", "ECHTE HA", "f www", ".d", "le Schiff", "n:", "ar 2021", "er P", "P", "ow_you", and ".de". There are also images of a barbed wire fence and a crowd of people.

智联赚赚uni-app和火山小程序 前端实践

创新产品组 - 柳学斌

背景

可能有些同学还不知道uni-app，这里先简单介绍下

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、Web（响应式）、以及各种小程序（微信/支付宝/百度/头条/QQ/快手/钉钉/淘宝）、快应用等多个平台。



跨端开发框架

跨端开发框架深度横评之2020版

- Vue

uni-app 🍊

- React

taro

uni-app 特色

- easycom

符合components/组件名称/组件名称.vue目录结构。就可以不用引用、注册，直接在页面中使用。

- uni_modules

对一组js sdk、组件、页面、uniCloud云函数、公共模块等的封装，用于嵌入到uni-app项目中使用，也支持直接封装为项目模板。

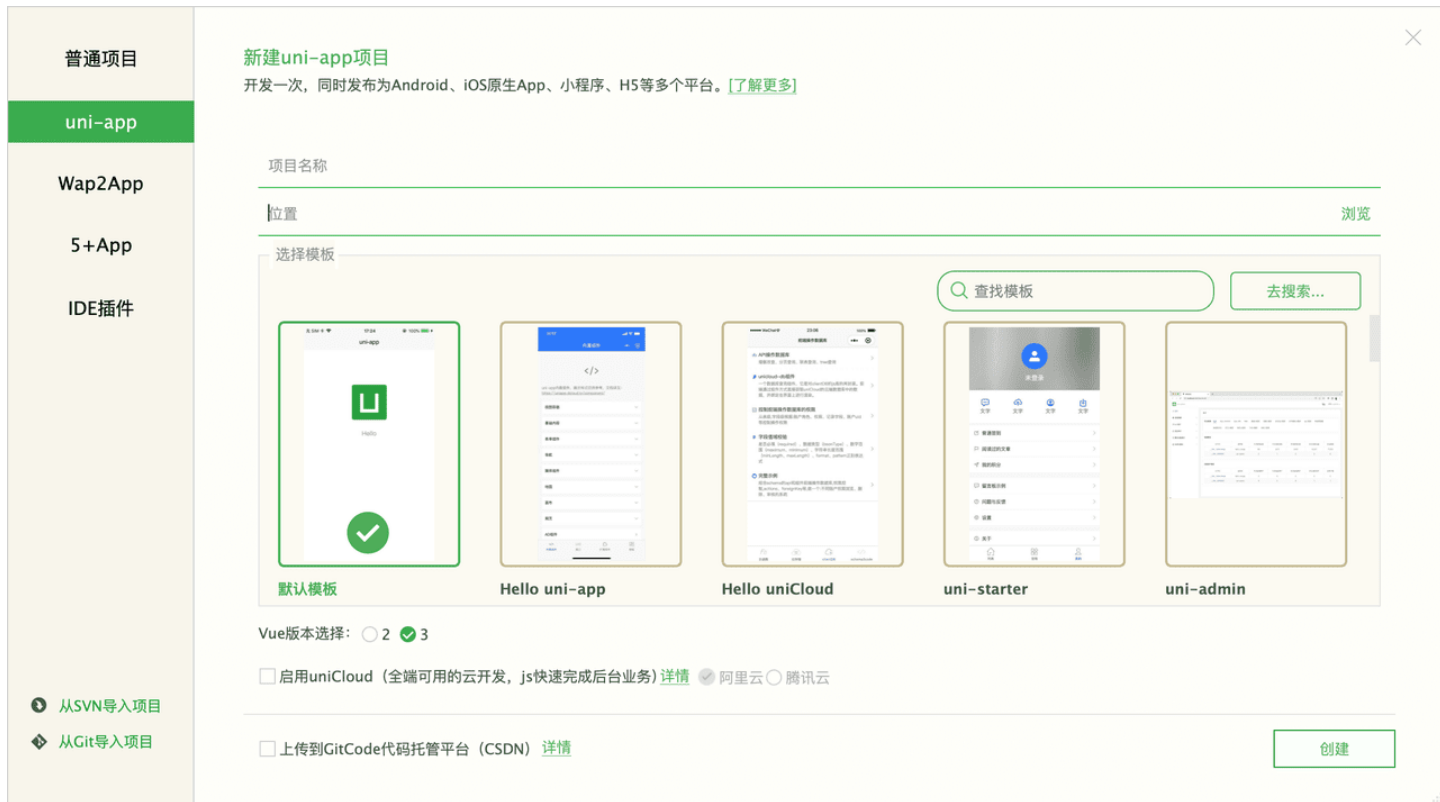
- 条件编译

#ifdef、#ifndef

开始

先从最简方式启动开发环境

HBuilderX 创建工程

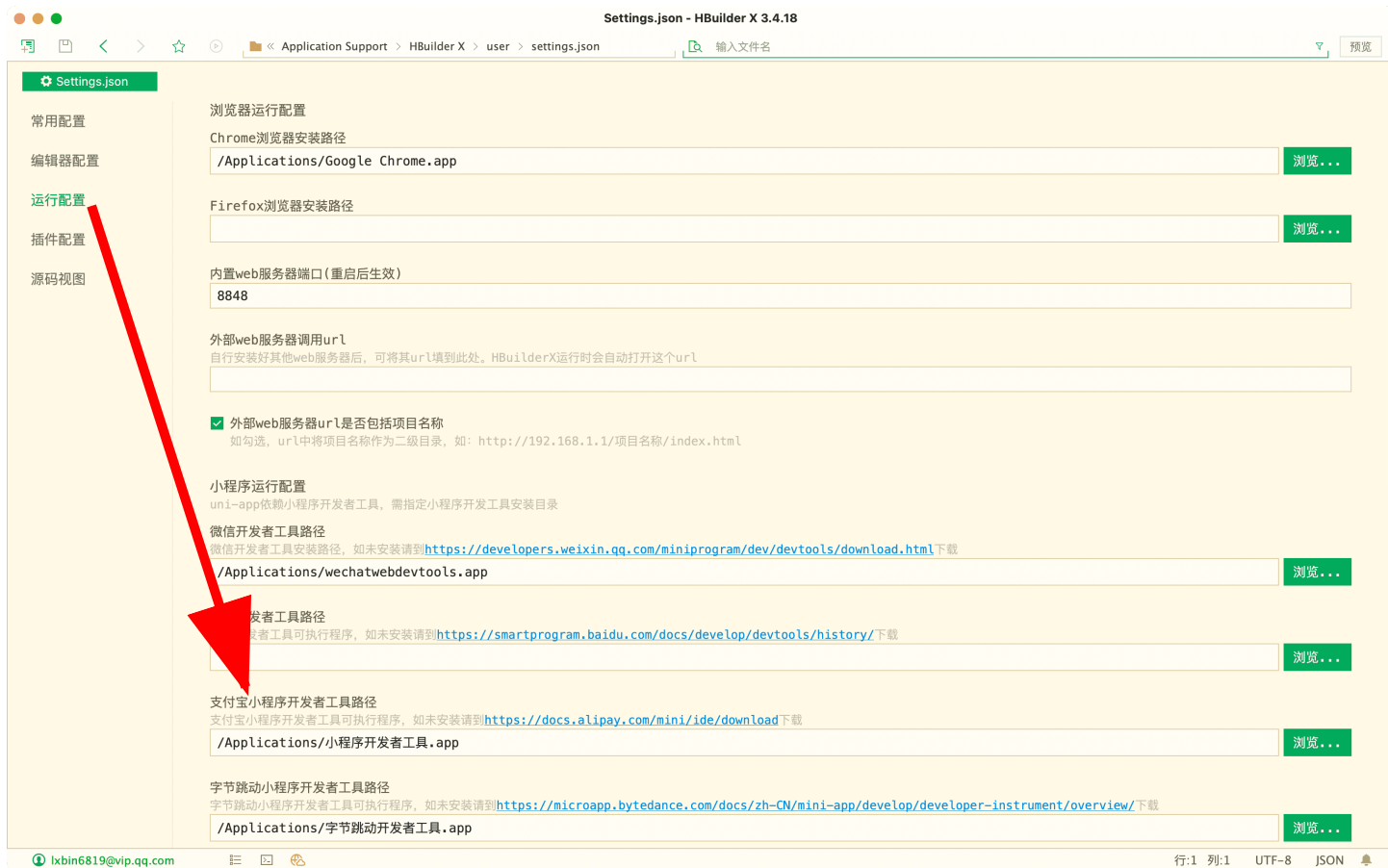


HBuilderX 工程目录

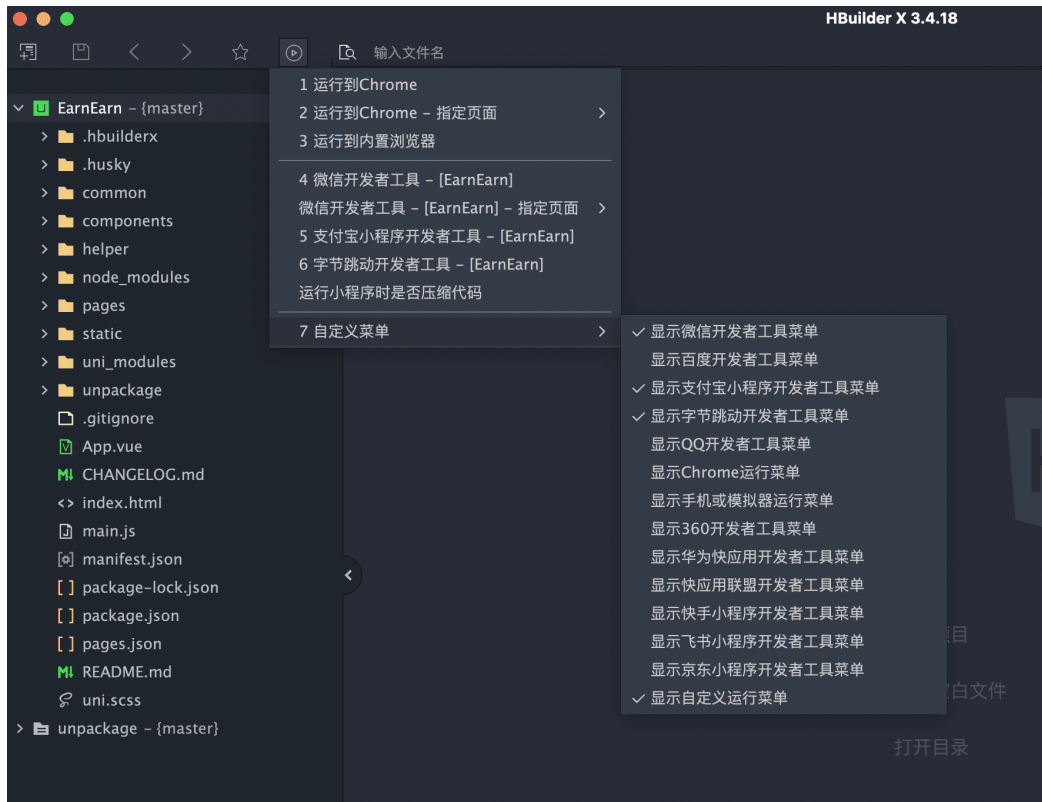
<https://uniapp.dcloud.io/tutorial/project.html#目录结构>

└uniCloud	云空间目录，阿里云为uniCloud-alipay,腾讯云为uniCloud-tcb（详见uniCloud）
└uni_modules	存放[uni_module](/uni_modules)。
└components	符合vue组件规范的uni-app组件目录
└└comp-a.vue	可复用的a组件
└hybrid	App端存放本地html文件的目录，详见
└platforms	存放各平台专用页面的目录，详见
└pages	业务页面文件存放的目录
└└index	
└└└index.vue	index页面
└└list	
└└└list.vue	list页面
└static	存放应用引用的本地静态资源（如图片、视频等）的目录，注意：静态资源只能存放于此
└wxcomponents	存放小程序组件的目录，详见
└nativeplugins	App原生插件 详见
└unpackage	非工程代码，一般存放运行或发行的编译结果
└main.js	Vue初始化入口文件
└App.vue	应用配置，用来配置App全局样式以及监听 应用生命周期
└manifest.json	配置应用名称、appid、logo、版本等打包信息，详见
└pages.json	配置页面路由、导航条、选项卡等页面类信息，详见
└uni.scss	这里是uni-app内置的常用样式变量

HBuilderX 运行前准备 - 配置开发工具路径



启动项目



小程序模拟器配置

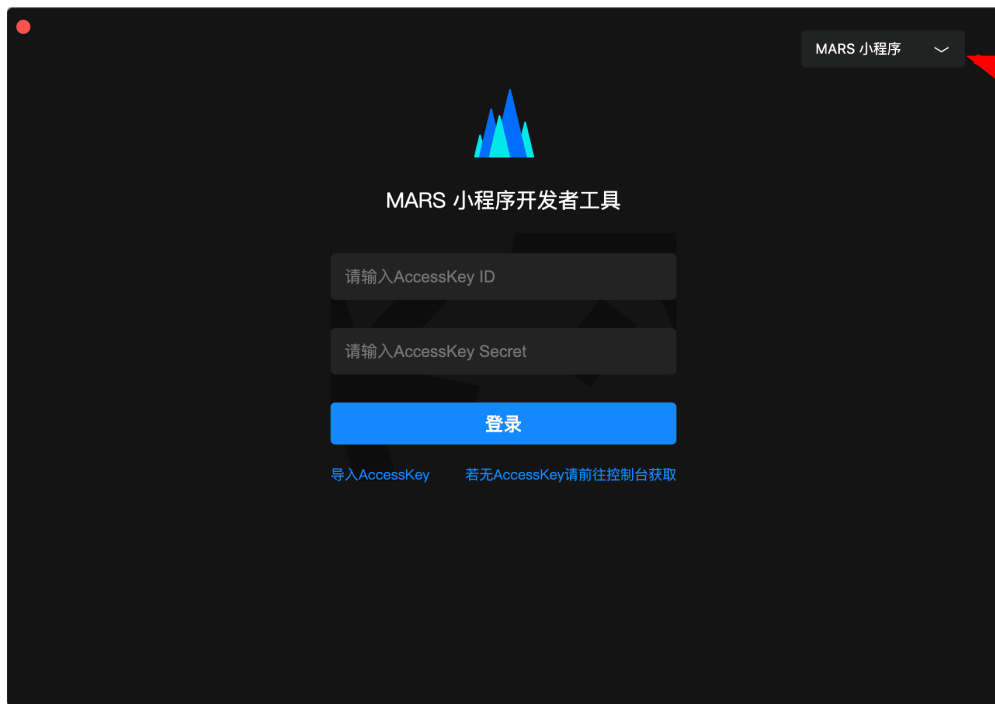
小程序后台配置request域名和业务域名



火山小程序模拟器

和字节小程序客户端不是一个，不要走错了~

火山小程序开发者工具下载地址 →



火山宿主环境 - 封装Promise

宿主APP集成SDK，自定义双向通道

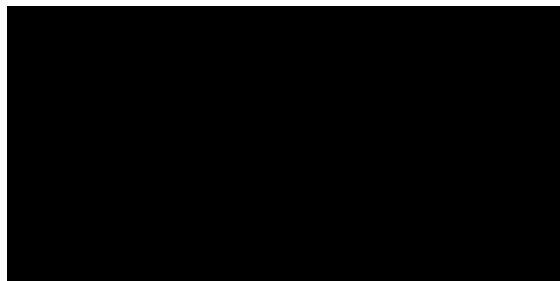
```
export function callHostMethod(method = '', extra = {}) {  
  // #ifndef MP-TOUTIAO  
  throw new Error('请使用智联APP真机调试火山小程序')  
  // #endif  
  
  if(tt.callHostMethodSync) {  
    return tt.callHostMethodSync({ method, extra })  
  }  
  
  return new Promise((resolve, reject) => {  
    tt.callHostMethod({  
      method,  
      extra,  
      success: resolve,  
      fail: reject,  
    })  
  })  
}
```

火山宿主环境 - 注入登录信息

```
// App.vue
export default {
  onShow() {

    // #ifdef MP-TOUTIAO
    callHostMethod('loginInfo')
      .then(({ data } = {}) => {
        const { at, rt } = data || {}
        at && this.$utils.setToken({ at, rt })
      })
      .catch((e) => console.error(`[callHostMethod] -> 设置登录信息`, e))
    // #endif

  },
}
```



火山宿主环境 - 跳转native路由

```
callHostMethod('navigate', {
  url: `/position/native/company`,
  params: {
    companyName: info.companyNumber
  }
})

callHostMethod('navigate', {
  url: `/base/webview/common?urlStr=${url}`,
  params: {
    animated: true, // 是否动画转场, 不传默认为true
  },
})
```



火山小程序 - 外部唤起

```
marsmp://microapp?app_id=ma6ddcc04978a5aefb&bdp_log=%7B%22launch_from%22%3A%22scan%22%7D&scene=12452&version=v2&version,
/container/native/miniProgram?urlStr=marsmp%3A%2F%2Fmicroapp%3Fapp_id%3Dma6ddcc04978a5aefb%26bdp_log%3D%257B%2522launch,
zpm://app-cfg.zhaopin.com?applink=%2Fcontainer%2Fnative%2FminiProgram%3FurlStr%3Dmarsmp%253A%252F%252Fmicroapp%253Fapp_
```


各端小程序 request 网络请求的 Referer

- 微信开放文档 (qq.com)

```
https://servicewechat.com/{appid}/{version}/page-frame.html
```

- 支付宝文档中心 (alipay.com)

```
https://{appid}.hybrid.alipay-eco.com/{appid}/{version}/index.html#pages/index
```

- 火山小程序 (bytedance.com)

```
安卓: https://tmaservice.developer.toutiao.com/?appid=madab9772ccfc01a73&version=1.0.0
```

```
iOS: /?appid=madab9772ccfc01a73&version=1.0.0
```

折腾

HBuilder虽然方便但是很多配置没有公开，做为一个有追求的前端，当然希望尽可能的都在自己的掌控中

cli 创建工程

- Vue2

```
npm install -g @vue/cli@4  
vue create -p dcloudio/uni-preset-vue my-project
```

- Vue3

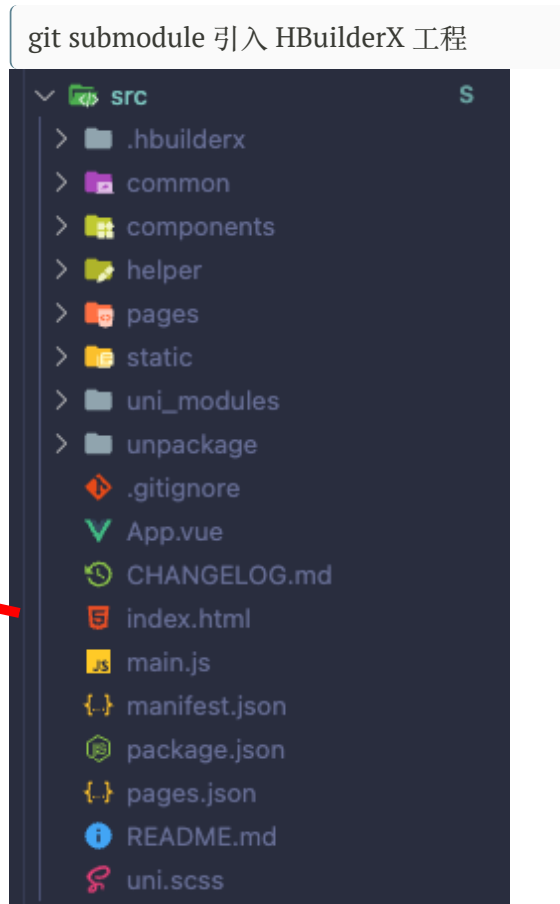
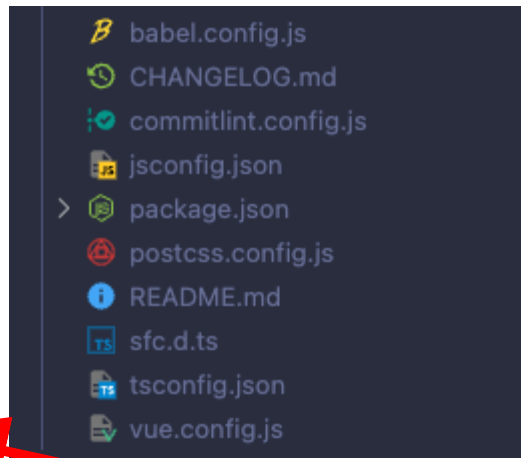
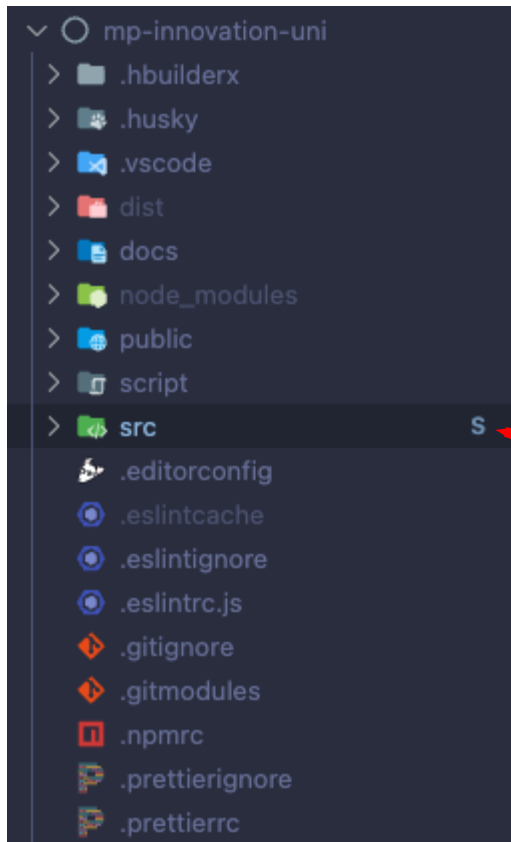
```
// javascript  
npx degit dcloudio/uni-preset-vue#vite my-vue3-project  
  
// typescript  
npx degit dcloudio/uni-preset-vue#vite-ts my-vue3-project
```

cli工程配置

我的前端工程配置

- .editorconfig
- .prettierrc.js
- .eslintrc.js
- commitlint.config.js
- .husky
- lint-staged
- changelog

我的工程目录



cli scripts

<https://gitlab.dev.zhaopin.com/innovation/mp-innovation-uni/-/tree/master/package.json>

```
{
  //...
  "scripts": {
    "serve": "cross-env NODE_ENV=development node_modules/.bin/zx script/dev.mjs",
    "build": "cross-env NODE_ENV=production node_modules/.bin/zx script/build.mjs",
    "-----",
    "build:app-plus": "cross-env NODE_ENV=production UNI_PLATFORM=app-plus vue-cli-service uni-build",
    "build:custom": "cross-env NODE_ENV=production uniapp-cli custom",
    "build:h5": "cross-env NODE_ENV=production UNI_PLATFORM=h5 vue-cli-service uni-build",
    "build:mp-360": "cross-env NODE_ENV=production UNI_PLATFORM=mp-360 vue-cli-service uni-build",
    "build:mp-alipay": "cross-env NODE_ENV=production UNI_PLATFORM=mp-alipay vue-cli-service uni-build",
    "build:mp-baidu": "cross-env NODE_ENV=production UNI_PLATFORM=mp-baidu vue-cli-service uni-build",
    "build:mp-jd": "cross-env NODE_ENV=production UNI_PLATFORM=mp-jd vue-cli-service uni-build",
    "build:mp-kuaishou": "cross-env NODE_ENV=production UNI_PLATFORM=mp-kuaishou vue-cli-service uni-build",
    "build:mp-lark": "cross-env NODE_ENV=production UNI_PLATFORM=mp-lark vue-cli-service uni-build",
    "build:mp-qq": "cross-env NODE_ENV=production UNI_PLATFORM=mp-qq vue-cli-service uni-build",
    "build:mp-toutiao": "cross-env NODE_ENV=production UNI_PLATFORM=mp-toutiao vue-cli-service uni-build",
    "build:mp-weixin": "cross-env NODE_ENV=production UNI_PLATFORM=mp-weixin vue-cli-service uni-build",
    "build:mp-xhs": "cross-env NODE_ENV=production UNI_PLATFORM=mp-xhs vue-cli-service uni-build",
    "build:quickapp-native": "cross-env NODE_ENV=production UNI_PLATFORM=quickapp-native vue-cli-service uni-build",
    "build:quickapp-webview": "cross-env NODE_ENV=production UNI_PLATFORM=quickapp-webview vue-cli-service uni-build",
    "build:quickapp-webview-huawei": "cross-env NODE_ENV=production UNI_PLATFORM=quickapp-webview-huawei vue-cli-service uni-build"
```

自定义运行 - 获取各端小程序的命令行工具

<https://gitlab.dev.zhaopin.com/innovation/mp-innovation-uni/-/tree/master/script>

```
export const cli = {  
  /** [HBuilderX cli命令行工具](https://hx.dcloud.net.cn/cli/README) */  
  hb: `/Applications/HBuilderX.app/Contents/MacOS/cli`,  
  
  /** [Weixin cli命令行工具](https://developers.weixin.qq.com/miniprogram/dev/devtools/cli.html) */  
  weixin: `/Applications/wechatwebdevtools.app/Contents/MacOS/cli`,  
  
  /** [Weixin ci工具](https://developers.weixin.qq.com/miniprogram/dev/devtools/ci.html) */  
  'weixin-ci': `node_modules/.bin/miniprogram-ci`,  
  
  /** [Alipay cli命令行工具](https://opendocs.alipay.com/mini/02q17h) */  
  alipay: `node_modules/.bin/minidev`,  
  
  /** [Mars火山小程序 cli命令行工具](https://www.volcengine.com/docs/6505/116886) */  
  toutiao: `node_modules/.bin/mars-mp`,  
}
```

自定义运行 - 编写运行IDE命令

<https://gitlab.dev.zhaopin.com/innovation/mp-innovation-uni/-/tree/master/script>

```
#!/usr/bin/env zx

import { cli } from 'config.mjs'

const dist = (mp) => path.join(__dirname, `../../dist/dev/mp-${mp}`)

export const open = {
  weixin: () => `${cli.weixin} open --project ${dist('weixin')}`,
  alipay: () => `${cli.alipay} ide --project ${dist('alipay')}`,
  toutiao: () => `${cli.toutiao} open ${dist('toutiao')}`,
}
```


自定义运行 - 编写获取预览二维码命令

<https://gitlab.dev.zhaopin.com/innovation/mp-innovation-uni/-/tree/master/script>

```
#!/usr/bin/env zx

import { cli } from 'config.mjs'

const dist = (mp) => path.join(__dirname, `../../dist/dev/mp-${mp}`)

export const preview = {
  weixin: async () => `${cli.weixin} preview --project ${dist('weixin')}`.nothrow(),
  alipay: async () => {
    if(await fs.readJson(`${os.homedir()}/.minidev/config.json`)?.alipay?.authentication){
      await within(() => `${cli.alipay} login`)
    }
    return await `${cli.alipay} preview --project ${dist('alipay')} --app-id ${appid.alipay}`.nothrow()
  },
  toutiao: async () => {
    await `${cli.toutiao} login-d ${toutiao.ak} ${toutiao.sk}`.quiet()
    return await `${cli.toutiao} preview ${dist('toutiao')} -s`.nothrow()
  },
}
```

自定义运行 - 启动全部端

<https://gitlab.dev.zhaopin.com/innovation/mp-innovation-uni/-/tree/master/script>

```
#!/usr/bin/env zx

import { cli } from 'config.mjs'

const dist = (mp) => path.join(__dirname, `../../dist/dev/mp-${mp}`)

let runMps = [ 'weixin', 'alipay', 'toutiao' ], result = []
Object.keys(runMps).forEach(async (mp) => {
  const p = $`npm run dev:mp-${mp}`.nothrow()
  for await (const chunk of p.stdout) {
    if (`${chunk}`.includes('complete')) {
      result.push(mp)
      if (result.length % runMps.length === 0) {
        if (result.length < runMps.length + 1) await openIde()
        await getPreviewCode()
      }
    }
  }
})
```



. 此操作对电脑性能有一定要求

VSCode

使用VSCode获得与HBuilderX同样的小程序开发体验

VSCode 开发体验优化

Vue 相关

- volar
- vetur
- vue-vscode-snippets

代码美化

- vscode-eslint
- prettier-vscode
- vscode-scss

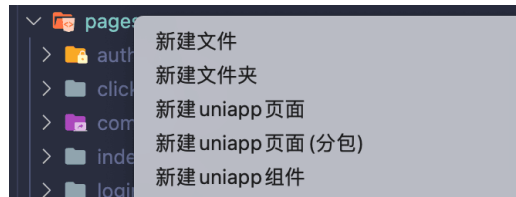
uni-app 代码提示

- `bond.better-comments`

```
function callHostMethod(method = '', extra = {}) {
  // #ifdef MP-TOUTIAO
  throw new Error('请使用智联APP真机调试火山小程序')
  // #endif

  // #ifdef MP-ALIPAY
  <zhiha-customer ref="zhihaCustomer"></zhiha-customer>
  // #endif
}
```

- create-uniapp-view



- `vscode-un-helper`

VScode 插件推荐

添加 .vscode/extensions.json 团队成员间共享配置

```
{
  "recommendations": [
    "Vue.volar",
    "octref.vetur",
    "sdras.vue-vscode-snippets",

    "dbaeumer.vscode-eslint",
    "esbenp.prettier-vscode",
    "mrmlnc.vscode-scss",

    "aaron-bond.better-comments",
    "mrmaoddxaa.create-uniapp-view",
    "modyqyw.vscode-uni-helper",
  ]
}
```

更全面的 uni-app 语法提示

uni-app 项目下的 manifest.json、pages.json 等文件可以包含注释。vscode 里需要改用 jsonc 编辑器打开。

- 补充d.ts

```
npm i @types/uni-app @types/html5plus -D
```

- 安装组件语法提示

```
npm i @dcloudio/uni-helper-json
```

- 导入 HBuilderX 自带的代码块

从 github 下载 uni-app 代码块，放到项目目录下的 .vscode 目录即可拥有和 HBuilderX 一样的代码块。

```
rm -rf .vscode/uni-snippets
git clone https://github.com/zhetengbiji/uniapp-snippets-vscode.git .vscode/uni-snippets
mv .vscode/uni-snippets/* .vscode
rm -rf .vscode/uni-snippets .vscode/.git
```

建议

下面是我们团队在开发过程中的一些经验及建议

打造高性能小程序

UI 组件库

官方《关于uni-app的ui库、ui框架、ui组件》建议 →

UI 推荐

- uView - 仅支持Vue2
- uni-ui
- FirstUI
- tm-vuetify
- ThorUI组件库

插件市场: <https://ext.dcloud.net.cn/>

流畅滚动

- scroll-view
- Page
- mescroll / z-paging

数据使用

- data双向数据 / Object.freeze()冻结数据 / 外部定义变量
- globalData / 挂载在vue实例
- Storage / URL参数
- 分批加载

组件划分

小程序

H5 / UniApp

- 数据交互区域封装成组件
- 避免组件层级过深

静态资源

- local / remote
- 大图 / 压缩雪碧图 / svg / 代码实现

配置全局分享

支付宝

main.js

```
import globalShare from 'helper/globalShare.js'

const OriginPage = Page
Page = function (pageConfig) {
  const onShow = pageConfig['onShow']
  pageConfig['onShow'] = function (...args) {
    globalShare()
    if(typeof onShow === 'function'){
      return onShow.apply(this, args)
    }
  }
  OriginPage(pageConfig)
}
```

微信

App.vue

```
import globalShare from 'helper/globalShare.js'
export default {
  onLaunch() {
    // #ifdef MP-WEIXIN
    wx.onAppRoute(() => globalShare())
    // #endif
  },
  onShareAppMessage() {
    const pages = getCurrentPages() || []
    const page = pages[pages.length - 1] //获取当前页面的对象
    return {
      title: '智联赚赚',
      path: page?.$page?.fullPath,
    }
  },
}
```

```

import _ from 'lodash'
export default function() {
  try {
    const pages = getCurrentPages() || [] //获取加载的页面
    const page = pages[pages.length - 1] //获取当前页面的对象
    if (page) {
      const options = {
        title: '智联赚赚',
        path: (page.$page || {}).fullPath || [
          page.route,
          _.reduce(page.options || {}, (result, value, key) => ([...result, `${key}=${value}`]), []).join('&')
        ].join('?')
      }
      console.log(`%c[路由发生变化] → ${options.path}`, 'background-color: #5ba7da', page)
      const data = page.data || {} // 若想给个别页面做特殊处理 可以给特殊页面加isOverShare为true 就不会重写了
      if (!data.isOverShare) {
        page.onShareAppMessage = () => (options)
        page.onShareTimeline = () => (options)
        uni.showShareMenu(options)
      }
    }
  } catch (e) {
    console.error('[globalShare] ->', e)
  }
}

```

提前规划分包

subPackages

- subPackages 开启分包优化
- preloadRule 配置分包预加载

CSS优化

推荐集成 tailwindcss，减少重复css

你可能不需要 es6 转 es5

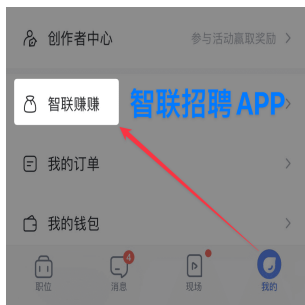
上传小程序时，如果使用了 es6 转 es5、css 对齐的功能，可能会增大代码体积

避免滥用外部js插件

有官方API的就不要额外引用js插件增加项目体积，必要的轮子该造还是要造

官方建议

高效开发技巧 / 性能优化专题



GitLab: mp-innovation-uni →

我的小程序开发笔记，欢迎交流 →



扫码观看PPT