快速搭建一个项目

刘圆圆

1. 准备工作
2. 创建一个版本仓库
3. 从本地拉取github项目仓库
   1. Git clone https://github.com/15301170766/2019-1-8.git
4. 创建项目结构
   1. Yarn global add vue-cli 安装vue脚手架
   2. Vue init webpack \*\*\*搭建vue项目
   3. Yarn install
   4. Yarn run dev
5. 常用组件安装
   1. 可以改成 taobao 的源：（解决yarn拉取资源慢的问题）
   2. yarn config set registry 'https://registry.npm.taobao.org'
   3. less依赖包 npm install --save-dev less less-loader

**配置：**

<style scoped lang='less'>

* 1. 网络请求axios依赖包npm install --save axios
  2. axios低版本浏览器补丁es6-promise依赖包
  3. npm install --save es6-promise
  4. 路由安装npm install --save vue-router
  5. 状态管理安装npm install --save vuex
  6. PC端组件库npm install --save element-ui
  7. 移动端组件库npm install --save mint-ui
  8. 上传服务器ssh2 npm install –save-dev ssh2
  9. cookie封装库 npm install --save js-cookie

1. 测试分支上传修改
   1. 创建分支 git checkout -b 【分支名称】并切换（加-b切换，不加则是只创建）

这相当于执行下面这两条命令：

$ git branch iss53

$ git checkout iss53

* 1. Git branch -a 列出所有本地分支和远程分支。
  2. 分支切换
  3. Git log 显示历史每次版本号 （当前版本的历史版本号）
  4. 现在总结一下：HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，使用命令git reset --hard commit\_id。穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。要重返未来，用git reflog查看命令历史，以便确定要回到未来的哪个版本。
  5. 分支合并
     1. 切换到master
     2. Git merge [分支版本名称]
     3. 等待解决冲突---》合并成功
  6. 可以看到 ======= 隔开的上半部分，是 HEAD（即 master 分支，在运行 merge 命令时所切换到的分支）中的内容，下半部分是在 iss53 分支中的内容。解决冲突的办法无非是二者选其一或者由你亲自整合到一起。比如你可以通过把这段内容替换为下面这样来解决：

1. 引入typescript整合到项目中去
   1. 安装ts
   2. npm install --save-dev typescript npm install --save-dev ts-loader

5.使用 tsc 创建项目需要的 tsconfig.json 文件。 $tsc --init

2、在根目录建tsconfig.json文件

{

   "compilerOptions": {

     "experimentalDecorators": true,

     "emitDecoratorMetadata": true,

     "lib": ["dom","es2016"],

     "target": "es5"

   },

   "include": ["./src/\*\*/\*"]    
}

3、在配置中添加 ts-loader

{

   test: /\.tsx?$/,

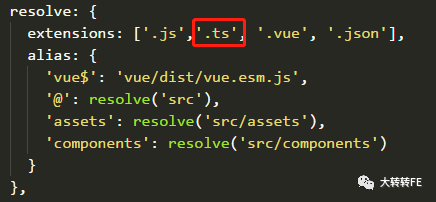
   loader: 'ts-loader',    exclude: /node\_modules/,      options: {

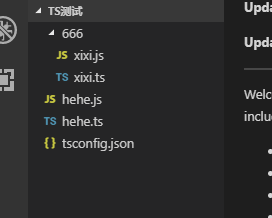
     appendTsSuffixTo: [/\.vue$/],

   }

 }

4、最后把 .ts 后缀添加上就OK了，在webpack.base.conf.js文件下



* 1. <https://www.jb51.net/article/150337.htm>
  2. 当根目录有tsconfig.js时，运行tsc。会直接编译所有ts文件（不适用）
  3. <https://blog.csdn.net/qq_33699981/article/details/79411846>

使用配置entry: {

app: './src/main.ts'

},

* 1. 使用Typescript开发Vue，一切准备就绪。但npm start 时，提示“

ERROR in ./src/main.ts  
Module build failed: TypeError: Cannot read property 'afterCompile' of undefined“错误。

## **解决方法**

将ts-loader从4.0降低到3.1.1解决问题。是由于webpack和ts-loader版本不兼容造成的。

在vue文件中<script lang='ts'>

1. 引用外部公共样式src文件下创建style文件夹，新建public。Less文件

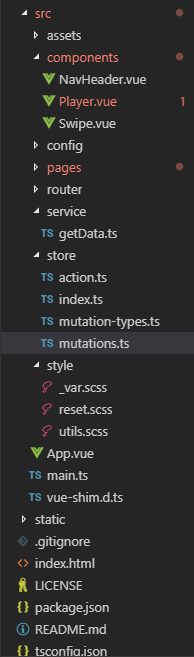
<style scoped lang='less'>

@import '../style/public.less';

* 1. 引用外部公共api
  2. src文件下创建config文件夹，新建api。Ts文件

export const RECOMMEND\_URL = "https://shenzekun-qmusic-api.now.sh/";

import {RECOMMEND\_URL} from '../config/api';

* 1. 
  2. 新建service文件夹ts
  3. 新建store文件夹-ts
  4. assets文件下方照片
  5. dist下方打包文件
     1. 打包yarn run build
     2. 解决方案
     3. 具体步骤如下：
     4. 1、查看package.js文件的scripts命令
     5. 2、打开webpack.dev.conf.js文件，找到publicPath: config.dev.assetsPublicPath,按Ctrl点击，跳转到index.js文件
     6. 3、其中dev是开发环境，build是构建版本，找到build下面的assetsPublicPath: '/',然后修改为assetsPublicPath: './',即“/”前加点。
     7. 4、终端运行 npm run build 即可。
     8. ---------------------
     9. 作者：CHH5431
     10. 来源：CSDN
     11. 原文：https://blog.csdn.net/qq\_41115965/article/details/80796310
     12. 版权声明：本文为博主原创文章，转载请附上博文链接！
     13. 解决照片问题

但是css中的background-image还是报404

此时的问题原因是，使用了相对地址后，在css进行引入的图片路径，其相对的是css文件的路径

此时的修改方法是，修改build文件夹中的utils.js文件，修改如下这一行

