**一 . 配置cli脚手架**

**1.首先用vue-cli创建一个文件夹**

**全局安装npm install -g @vue/cli 创建 vue create 文件夹名**

**.创建方式有两种，一种自动创建默认的配置default（babel，eslint），一种手动创建Manually select features**

**2.选择手动创建可以选择功能特性配置项**

**1:Babel 可以将es6转换为es5的工具**

**2：Router vue router 路由**

**3： CSS Pre-processors css预处理器 用哪种动态语言编写css语法 less，sass，stylus**

**4：Linter / Formatter ： 代码效验和格式化**

**3.选择路由模式，有两种**

**1.hash锚点模式http://协议:端口号/path路径/#/路由路径，特点：简单，兼容性好**

**2.history模式: http://协议:端口号/路由路径 ,特点：url简洁，需要额外的服务器配置**

**4.选择css预处理器，用less**

**5.选择代码格式校验风格和方式，选择 ESLint + Standard config 模式**

**6.选择配置文件的存储位置**

1.In dedicated config files：生成独立的配置文件，推荐，维护方便

**2.In package.json**混到 package.json 文件中，

**7.是否要保存这个文件模板。**

**二.创建好之后的目录结构**

**.browserslistrc 该文件会被 Babel 和 Autoprefix 用来根据浏览器的版本确定需要转译的 JavaScript 特性和 CSS 浏览器前缀**

**.editorconfig 编辑器配置文件，编辑器会根据该文件选择编辑格式**

**.eslintrc.js ESLint配置文件**

**.gitignore Git忽略配置文件**

**babel.config.js Babel转码工具配置文件**

**package-lock.json 包管理工具的锁定文件**

**package.json 包说明文件**

**postcss.config.js postcss配置文件**

**README.md 说明文档**

**node\_modules 第三方包**

**public 公共资源目录**

**src 源码目录**

**│ App.vue 根组件 │ main.js 入口模块 │ router.js 路由模块 │ ├─assets 静态资源目录 │ logo.png │ ├─components 非路由组件目录 │ HelloWorld.vue │ └─views 路由组件目录 About.vue**

**Home.vue**

**3.连接远程仓库**

**方式1：线上仓库拿到本地，远程创建仓库，然后git clone 远程仓库地址 克隆下来，把文件夹移动进去，在提交，如果推送的是非master分支 需要Git push origin 本地分会：远程分支**

**方式2：本地仓库放到线上，本地创建仓库，远程创建一个干净的仓库，我们习惯于把默认的远程仓库地址别名叫做 origin，连接远程仓库，git remote add 别名 远程仓库地址，然后推送**

**git push -u origin 本地分支：远程分支 设置默认推送的地址和分支**

**4.调整目录结构**

**5.引入第三方包element-ui**

**1.首先安装 npm i element-ui**

**2.配置e-ui ，在入口文件引进e-ui，因为是第三方包，可以直接引入，但是e-ui的css需要引入地址**

**import from 'element-ui' ，import 'element-ui/lib/theme-chalk/index.css'**

**然后，用VUE加载e-ui Vue.use(e-ui)**

**6.设计路由**

**首先在路由的js文件中，引入vue，引入路由vue-router，用vue加载路由，实例化路由new Router，把路由表routes:[{path:'',name:'',component:()=>imPort{}}]配置好，name是给这个路由路径起一个名字，跳转链接的时候简单，（）=> import(这里面是单页面组件的地址)然后导出，import defalut**

**其次在入口文件中引入路由，vue实例化中加载路由，在根组件中写入路由的入口**

**7.导入字体图标**

**首先在主页面上引入字体图标的地址<link rel="stylesheet" href="地址">**

**8.登录页样式布局**

**公共的css代码写到styles中，因为选择了less处理css，所以创建一个less页面，**

**让登录页居中，写入的块级元素如果想要居中，首先让他的父级元素宽度设置为100%，**

**app，html，body**

**9.点击登录按钮验证手机号跟验证码，成功跳转到主界面**

**给登录按钮注册点击事件，发送axios到用户认证端口，参数有手机号跟验证码，成功之后跳转到主页面，可以用路由的动态路径 this.$router.push() 里面的参数是一个对象，是路由地址的name名称**