从vue-cli开始搭建一个完整的vue项目

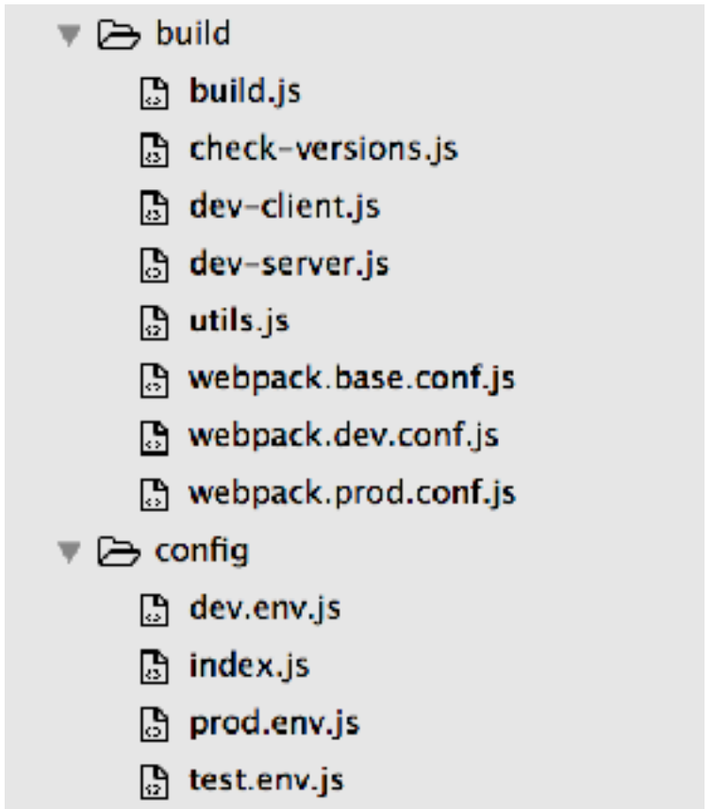
钱凯杰

vuejs官方脚手架项目

- 准备：NPM，ES2015，webpack
- *.vue==>最终应用(webpack)：项目由一堆vue文件构成，在webpack的作用下， 形成最终应用
- vue-loader特征：ES2015，Scoped CSS，CSS Modules,PostCSS, Hot Reload,esLint

项目配置文件结构

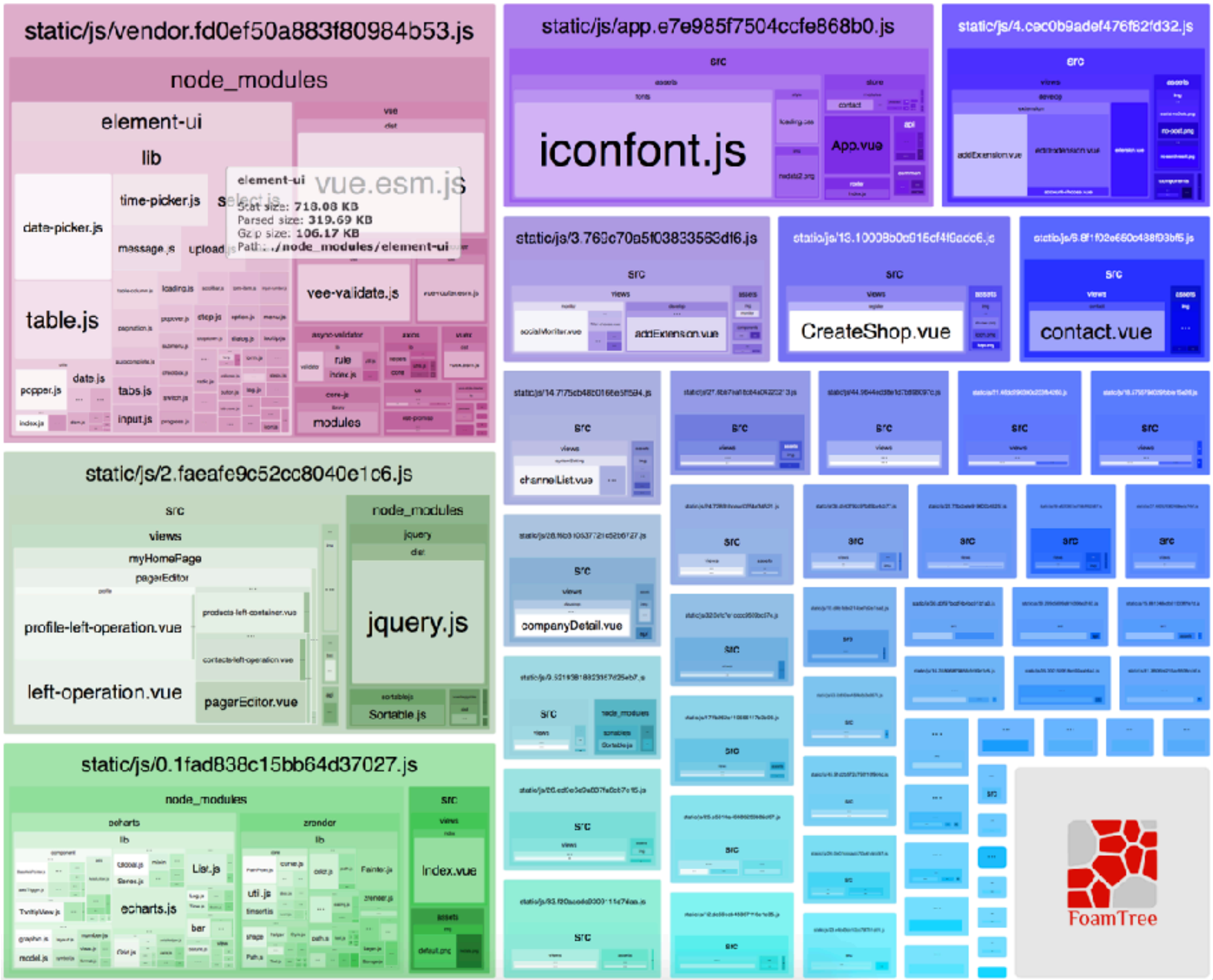
- build - 编译任务的代码
- config - webpack 的配置文件（环境变量，产出路径路径，开发环境接口代理）
- package.json - 脚本入口




```
"scripts": {
  "dev": "node build/dev-server.js",
  "build": "node build/build.js",
  "unit": "karma start test/unit/karma.conf.js --single-run",
  "e2e": "node test/e2e/runner.js",
  "test": "npm run unit && npm run e2e",
  "lint": "eslint --ext .js,.vue src test/unit/specs test/e2e/specs"
},
```

项目几个script

- npm run build【多环境构建】
  - webpack工作方式：把你的项目当做一个整体，通过一个给定的主文件（如：index.js），Webpack将从这个文件开始找到你的项目的所有依赖文件，使用loaders处理它们，最后打包为一个浏览器可识别的JavaScript文件。
- npm run analyz
  - "analyz": "NODE_ENV=production npm_config_report=true npm run build"


static/js/vendor.fd0ef50a883f80984b53.js
node_modules
element-ui
lib
date-picker.js
time-picker.js
table.js
popper.js
date.js
tabs.js
input.js
message.js
upload
vue
vue.esm.js
vee-validate.js
modules
element-ui
Stat size: 718.08 KB
Parsed size: 319.69 KB
Gzip size: 106.17 KB
Path: ./node_modules/element-ui
static/js/app.e7e985f7504ccfe868b0.js
src
iconfont.js
App.vue
static/js/4.cec0b9adef476f82fd32.js
src
static/js/3.769c70a5f03833563df6.js
src
addExtension.vue
static/js/13.1000Bb0e915cf4f6ecc6.js
src
CreateShop.vue
static/js/8.8f1f02e650c438f93bf5.js
src
contact.vue
channelList.vue
companyDetail.vue
static/js/2.faeafe9c52cc8040e1c6.js
src
views
myHomePage
pagerEditor
profile-left-operation.vue
left-operation.vue
pagerEditor.vue
node_modules
jquery
dist
jquery.js
Sortable.js
static/js/0.1fad838c15bb64d37027.js
node_modules
echarts
lib
echarts.js
List.js
zrender
util.js
src
Index.vue
FoamTree

Webpack 一些loader

- 配置(webpack.base.config.js)

```
rules: [
  {
    test: /\.(js|vue)$/,
    loader: 'eslint-loader',
    enforce: 'pre',
    include: [resolve('src'), resolve('test')],
    exclude: [resolve('src/assets/js'),resolve('src/assets/fonts')],
    options: {
      formatter: require('eslint-friendly-formatter')
    }
  },
  {
    test: /\.vue$/,
    loader: 'vue-loader',
    options: vueLoaderConfig
  },
  {
    test: /\.js$/,
    loader: 'babel-loader',
    include: [resolve('src'), resolve('test')]
  },
  {
    test: /\.(png|jpe?g|gif|svg)(\?.*)?$/,
    loader: 'url-loader',
    options: {
      limit: 10000,
      name: utils.assetsPath('img/[name].[hash:7].[ext]')
    }
```

- lint配置（.eslintrc.js）

```
.eslintrc.js
module.exports = {
  root: true,
  parser: 'babel-eslint',
  parserOptions: {
    sourceType: 'module'
  },
  // https://github.com/fer
  extends: 'standard',
  // required to lint *.vue files
  plugins: [
    'html'
  ],
  // add your custom rules here
  'rules': {
    // allow paren-less arrow functions
    'arrow-parens': 0,
    // allow async-await
    'generator-star-spacing': 0,
    // allow debugger during development
    'no-debugger': process.env.NODE_ENV === 'production' ? 2 : 0
  }
}
```

基于vue-cli加的几样东西

- ✔ yarn @done (17-04-05 15:30)
- ✔ 目录结构 @done (17-04-05 15:38)
- ✔ http 模块 ,API层，axios 封装层 @done (17-04-05 17:52)
- ✔ mock数据 @done (17-04-06 11:47)
- ✔ vue-router @done (17-04-05 16:22)
- ✔ 全局css,安装less-loader @done (17-04-06 11:08)
- ✔ 图标字体 @done (17-04-06 11:35)
- ✔ element 按需引入 @done (17-04-06 13:57)
- ✔ vuex @done (17-04-06 16:47) 【做预加载】
- ✔ 路由动态加载，子路由 @done (17-04-06 17:30)

• 几个书写代码注意点

- 1.几个命令：yarn（相当于 npm install），yarn run dev, yarn run build, yarn run e2e.
- 2.样式：每个.vue文件 写样式尽量 scoped标签，避免不必要的样式bug（不同页面之间，同个类名）;全局样式写src/style/base.less里，全局样式变量定义在src/style/variable.less里【复写elementUI 样式，须去掉scoped标签】
- 3.目录结构：项目对应的所有单页都要放在src/views 文件夹下，每个模块的层级结构必须清晰，有父子关系的，子模块要放在cpnts(components的简写)目录下【若层级复杂，可递归创建cpnts目录】
- 4.请求&mock数据： 一个模块一个API文件（可照着 src/api/contactAPI.js 写）；可以将mock数据放在/static/mock文件夹下，这时候只需要将/src/common/config.js 下 mock赋值为true 即可
- 5. ESLint（Standard）：项目配置了 ESLint（Standard）,写代码的时候注意书写规范 比如 a = 1 (等号左右都有空格)
- 6.iconfont字体：由于在App.vue全局引入了iconfont，所以只需要在页面写DOM即可调用到字体，如
- 7.vuex的使用可参照src/store/contactDemo文件夹和src/store/contactDemoComplex文件夹。若模块较为简单，用前者的方式把action，mutations,getters等都放在一个文件夹，若模块比较复杂，则建议用后者的方式.
- 8.饿了吗UI插件按需引入组件：项目用到新组件时候，需要在src/common/element-import.js 引入对应的组件模块
- 9.字符串处理尽量用 过滤器 ， 全局过滤器参照 src/common/filters.js

像素级开发插件

- PerfectPixel

推荐一款效率神器

- **Alfred(mac OS), Wox(Windows)**
  - 开应用,找文件，打开文件夹,计算器
  - 打开网站 （淘宝，知乎）
  - 剪切板历史
  - workflow(翻译,sublime,dash,sourceTree，股票)
  - 自定义的workflow