项目说明

项目名	说明
jnpf-web	主项目前端
jnpf-web-datascreen	大屏前端
<pre>jnpf-web-datareport</pre>	报表前端
<pre>jnpf-web-tenant</pre>	多租户前端

环境要求

基础要求

因环境不同,在某些情况下无法顺利安装前端依赖,需要安装 Node.js 最新LTS版本及 git 本文档将以 最新LTS版本 进行演示

- Git
- Node.js
- Yarn

推荐IDE

- Visual Studio Code
- WebStorm

Visual Studio Code推荐插件

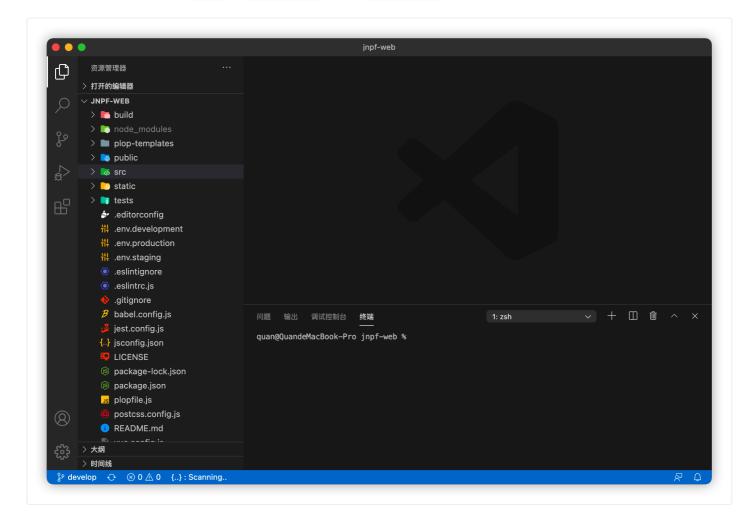
- Vetur
- Auto Close Tag
- Auto Rename Tag
- JavaScript(ES6) code snippets
- Beautify

来自 CODING 1/11

主项目前端

• 方式1: 打开 Visual Studio Code 编辑器,把 jnpf-web 项目拖到编辑器主界面即可

• 方式2: 依次选择菜单 文件 -> 打开文件夹 , 选中 jnpf-web 前端工程目录

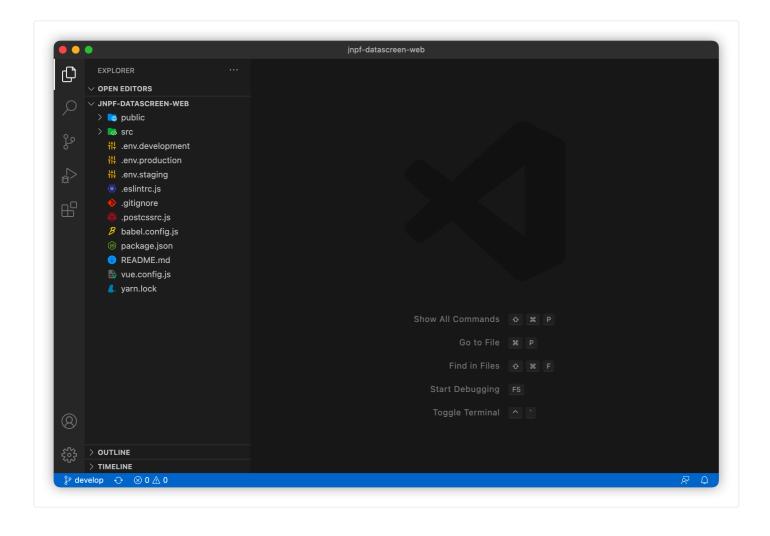


大屏前端

• 方式1: 打开 Visual Studio Code 编辑器,把 jnpf-web-datascreen 项目拖到编辑器主界面即可

• 方式2: 依次选择菜单 文件 -> 打开文件夹 ,选中 jnpf-web-datascreen 前端工程目录

来自 CODING 2/11

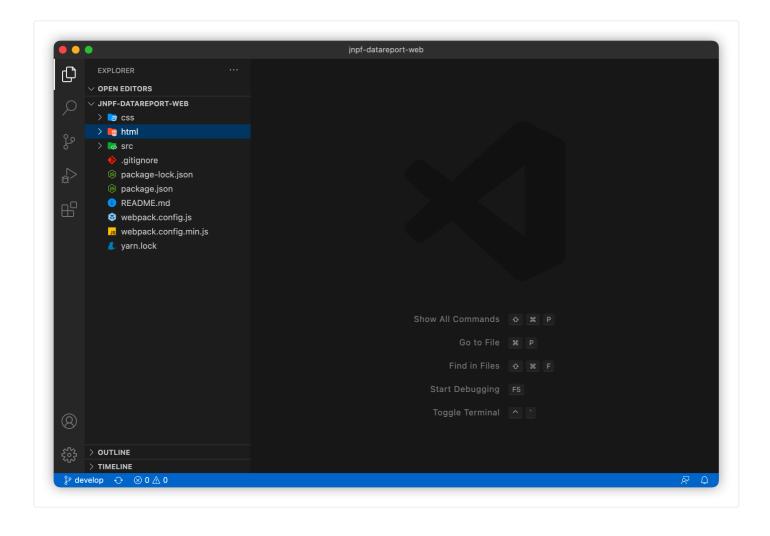


报表前端

• 方式1: 打开 Visual Studio Code 编辑器,把 jnpf-web-datareport 项目拖到编辑器主界面即可

• 方式2: 依次选择菜单 文件 -> 打开文件夹 ,选中 jnpf-web-datareport 前端工程目录

来自 CODING 3/11

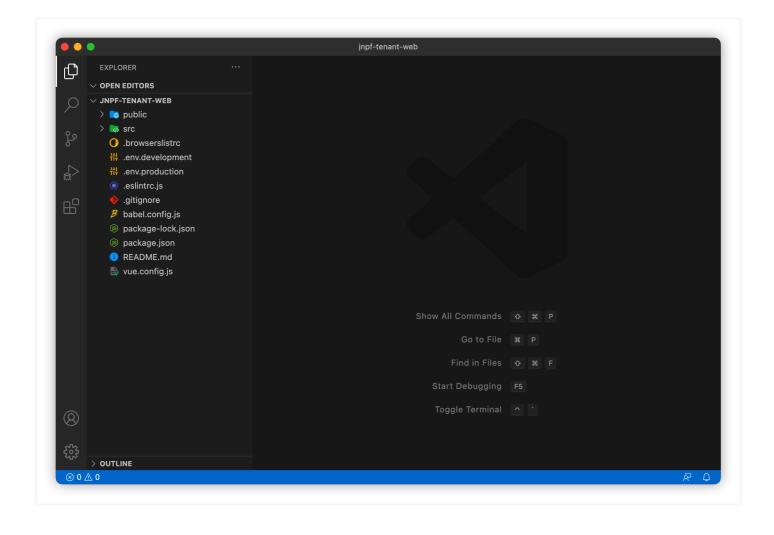


多租户前端

• 方式1: 打开 Visual Studio Code 编辑器,把 jnpf-web-tenant 项目拖到编辑器主界面即可

• 方式2: 依次选择菜单 文件 -> 打开文件夹 , 选中 jnpf-web-tenant 前端工程目录

来自 CODING 4/11



项目运行

主项目前端

1) 打开 src/utils/define.js ,修改开发环境接口

```
// 开发环境接口配置
const APIURl = 'http://192.168.0.25:30000'

module.exports = {
    APIURl: APIURl,
    timeout: process.env.NODE_ENV === 'development' ? 10000 : 1000000,
    WebSocketUrl: process.env.NODE_ENV === 'development' ? APIURl.replace('http', 'ws')
) + '/websocket' : process.env.VUE_APP_BASE_WSS,
    comUploadUrl: process.env.VUE_APP_BASE_API + '/api/file/Uploader',
    comUrl: process.env.VUE_APP_BASE_API,
    // 大屏应用前端路径
    dataV: process.env.NODE_ENV === 'development' ? 'http://localhost:8100/DataV' : pr
ocess.env.VUE_APP_BASE_API + '/DataV',
    // 数据报表
    reportServer: process.env.NODE_ENV === 'development' ? 'http://localhost:9000' : p
```

来自 CODING 5/11

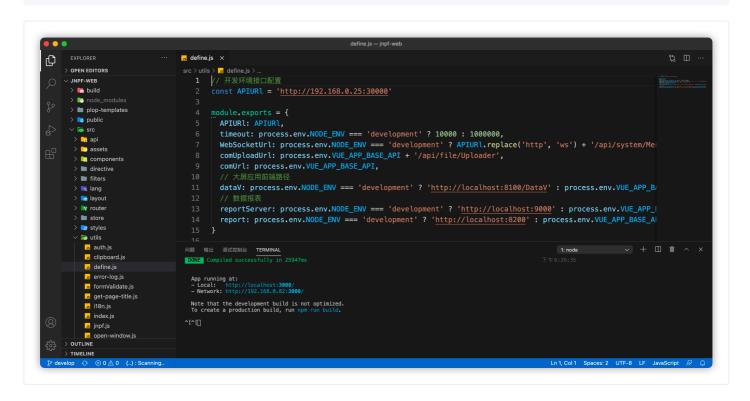
```
rocess.env.VUE_APP_BASE_API + '/ReportServer',
  report: process.env.NODE_ENV === 'development' ? 'http://localhost:8200' : proces
s.env.VUE_APP_BASE_API + '/Report'
}
```

2) 依次选择菜单 终端 -> 新终端 ,执行以下命令,安装项目所需依赖

```
npm i
```

3) 安装完依赖后,执行以下命令来运行您的项目,

npm run dev



3)运行成功后,会自动打开项目界面,如下图所示

来自 CODING 6/11



大屏前端

1) 打开 src/utils/define.js ,修改开发环境接口

```
// 开发环境接口配置
const APIURl = 'http://192.168.0.25:30000'

module.exports = {
   apiURI: APIURl,
   comUploadUrl: process.env.VUE_APP_BASE_API + '/api/visualdev/DataScreen/Images/',
   comUrl: process.env.VUE_APP_BASE_API,
   staticPath: process.env.NODE_ENV === 'development' ? '' : '/dataV'
}
```

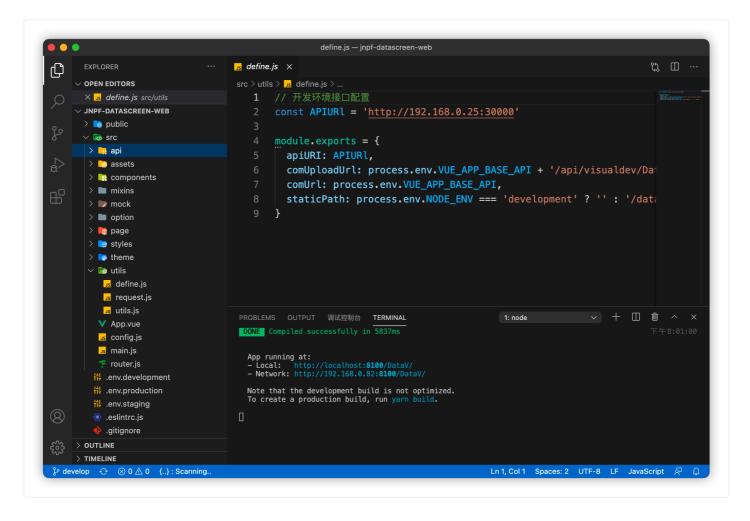
2) 依次选择菜单 终端 -> 新终端 ,执行以下命令,安装项目所需依赖

```
yarn
```

3) 安装完依赖后,执行以下命令来运行您的项目

```
yarn dev
```

来自 CODING 7/11



3)运行成功后,通过主项目前端界面打开,【在线开发】-【大屏设计】-【新建大屏】中查看(建议先运行后端)

报表前端

1) 安装 http-server

```
npm install --global http-server
```

2) 进入 html 目录,并修改 index.html 和 preview.html 的配置

index.html文件

```
<script type="text/javascript">
    // 报表接口
    window._server = "http://localhost:9000";
    // 报表前端
    window._contextPath = "http://localhost:8200";
    // 主项目接口地址
    window._mainServer = "http://127.0.0.1:7772";
    </script>
```

preview.html

来自 CODING 8/11

```
<script type="text/javascript">
    // 报表接口
    window._server = "http://localhost:9000";
    window.formElements = [];
</script>
```

3) 依次选择菜单 终端 -> 新终端 , 执行以下命令

```
http-server -a localhost -p 8200
```

4)运行成功后,通过主项目前端界面打开,【在线开发】-【报表设计】-【新建报表】中查看(建议先运行后端)

多租户前端

1) 打开 src/utils/define.js ,修改开发环境接口

```
// 开发环境接口配置
module.exports = {
    APIURl: 'http://192.168.0.100:7100'
}
```

2) 依次选择菜单 终端 -> 新终端 ,执行以下命令,安装项目所需依赖

```
yarn
```

3) 安装完依赖后,执行以下命令来运行您的项目

```
yarn dev
```

来自 CODING 9/11

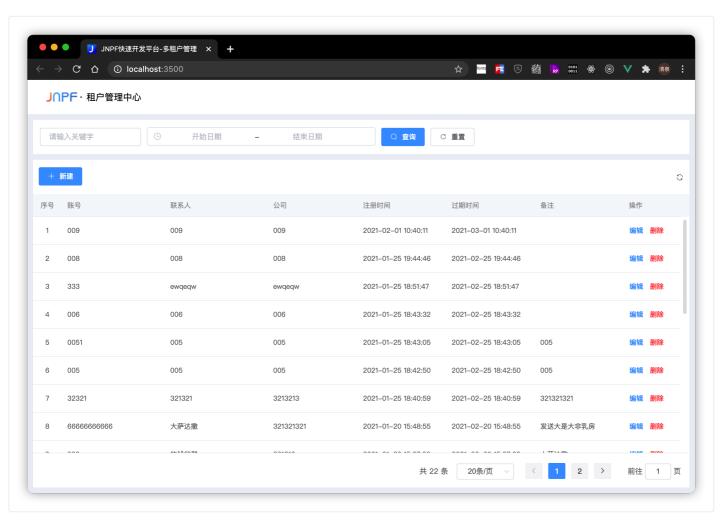
```
define.js — jnpf-tenant-web
                                              Js define.js X
      > OPEN EDITORS
                                               src > utils > _{JS} define.js > [9] < unknown > /> // APIURI
                                                       // 开发环境接口配置
      ∨ JNPF-TENANT-WEB
        > node_modules
        > 🧓 public
                                                       module.exports = {
        ∨ 🔯 src
                                                          APIURl: 'http://192.168.0.21:5600'
         > 📴 api
         > 🍺 assets
         > 📑 components
         > 📭 router
         > 🥽 styles
         ∨ 📠 utils
             us define.js
             us request.js
                                               PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

→ + □ 

□ ^ ×

                                                                                                      1: node
             us scroll-to.js
                                               DONE Compiled successfully in 2337ms
          > 튢 views
           App.vue
                                                 App running at:
    - Local: http://localhost:3500/
    Network: http://192.168.0.82:3500/
           us main.js
          browserslistro
          # .env.development
                                                 Note that the development build is not optimized. To create a production build, run yarn build.
          # .env.production
          eslintrc.is
             .gitignore
          B babel.config.js
          package-lock.json
(8)
           package.json
          i README.md
      > OUTLINE
                                                                                                  Ln 4, Col 36 Spaces: 2 UTF-8 LF JavaScript 🔊
⊗ 0 ≜ 0 {..} : Scanning..
```

3)运行成功后,会自动打开项目界面,如下图所示



来自 CODING 10/11

来自 CODING 11/11