



链滴

# Go1.16 embed 和 Vue

作者: [Allenxuxu](#)

原文链接: <https://ld246.com/article/1618851384184>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## vue

相关代码: <https://github.com/Allenxuxu/ginvue>

### 先全局安装下 vue cli 并创建一个 demo 项目

```
npm install -g @vue/cli
vue create web
```

然后我们进入 web 目录, 修改生成的 package.json 文件调整一下 build 生成的静态文件目录。

#### --dest 是指定输出的目录

\*\*--no-clean 是让他不要每次覆盖我们的目录, 因为后面我们会放一个 go 文件到那个目录。 \*\*

```
"build": "vue-cli-service build --no-clean --dest ../static",
```

再新增一个 vue.config.js 文件来修改下, 这里将 production 的 publicPath 修改成带一个前缀 /ui/ , 这里主要就是为了后面我们的go 代码路由设置方便, 所有的前端静态文件请求都带上 /ui 前缀, 和端 API 接口带 /api 前缀区分。

```
module.exports = {
  publicPath: process.env.NODE_ENV === 'production'
    ? '/ui/'
    : '/'
}
```

最后我们再 web 目录运行 **npm run build**, 会生成一个 static 目录 (也就是我们修改的 package.j on 里指定的目录), 里面会存放生成的静态文件。

```
├── css
│   └── app.fb0c6e1c.css
├── favicon.ico
├── img
│   └── logo.82b9c7a5.png
├── index.html
├── js
│   ├── app.cdde1042.js
│   ├── app.cdde1042.js.map
│   ├── app.e656f618.js
│   ├── app.e656f618.js.map
│   ├── chunk-vendors.ff672a17.js
│   └── chunk-vendors.ff672a17.js.map
└──
```

## go

我们再 static 目录里增加一个 go 文件, 这里使用 1.16 的 embed 来嵌入当前目录的静态文件:

```
package static
└──
```

```
import "embed"
[]
//go:embed index.html favicon.ico css img js
var Static embed.FS
[]
```

最后看一下 main.go，主要就是这行 `r.StaticFS("/ui", http.FS(static.Static))`。

```
package main
[]
import (
    "fmt"
    "net/http"
    "time"
[]
    "github.com/Allenxuxu/ginvue/static"
    "github.com/gin-gonic/gin"
    "github.com/pkg/browser"
)
[]
func main() {
    r := gin.Default()
[]
    r.StaticFS("/ui", http.FS(static.Static))
[]
    r.GET("/ping", func(c *gin.Context) {
        c.JSON(200, gin.H{
            "message": "pong",
        })
    })
[]
    go func() {
        time.Sleep(time.Second)
        err := browser.OpenURL("http://127.0.0.1:8080/ui")
        if err != nil {
            fmt.Println(err)
        }
        fmt.Println("Open: http://127.0.0.1:8080/ui")
    }()
[]
    r.Run() // listen and serve on 0.0.0.0:8080 (for windows "localhost:8080")
}
[]
```

这里就可以直接 `go run main.go` 了，不依赖前端静态文件，直接在浏览器打开 <http://127.0.0.1:8080/ui> 即可。

为了验证，可以 `build` 生成到其他目录，然后运行。

```
go build -o /tmp/demo main.go
```

相关代码：<https://github.com/Allenxuxu/ginvue>