

Please visit website: <http://cxyroad.com>

框架和创建新项目的步骤。

安装 Wails

...

```
go install github.com/wailsapp/wails/v2/cmd/wails@latest
```

...

验证安装结果：

...

```
wails version
```

...

也可以通过 `wails doctor` 来检查是否所有必要的依赖都已正确安装。

...

```
# Wails
```

```
# ...
```

```
# System
```

```
# ...
```

```
# Dependencies
```

```
# ...
```

```
# Diagnosis
```

```
# ...
```

```
SUCCESS Your system is ready for Wails development!
```

...

我的本地开发版本：

```
| # | Version |  
| --- | --- |
```

```
| Wails | v2.6.0 |
| Go | v1.19.1 |
| Node.js | v16.19.0 |
| npm | v8.19.3 |
```

创建新项目

使用 [Wails CLI](<http://cxyroad.com/> "https://wails.io/docs/reference/cli") 创建一个名为 `go-run-wechat-demo` 的新项目：

```
...
```

```
wails init -n go-run-wechat-demo -t react-ts
```

```
...
```

项目结构

![项目结构](<https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/b1a287fe2df44e1799907f5f67f2ebb9~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=797&h=617&s=121311&e=png&b=1c1c1c>)

- * **`main.go`** 和 **`app.go`**：Go 应用程序，处理业务逻辑、数据管理和与前端的通信。
- * **`frontend`**：包含前端的所有代码，使用 React、Vue 或你选择的任何其他框架，负责用户界面和与用户的交互。
- * **`go.mod`** 和 **`go.sum`**：Go 的模块依赖文件。
- * **`wails.json`**：Wails 项目的配置文件，定义了如何构建和打包应用。
- * **`build`**：用于存放构建后的应用程序和相关资源。

项目开发：仿登录界面

```
-----
```

进入开发模式

进入项目根目录，输入并执行 `wails dev` 命令，首次执行会安装前后端依赖，执行成功后可以看到默认应用页面。

![默认应用页面](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/ec17601853ae4c17a4dda04d951e43bc~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1017&h=790&s=94412&e=png&b=1b2635)

并且可以在浏览器调试页面：

```
...  
To develop in the browser and call your bound Go methods from  
Javascript, navigate to: http://localhost:34115
```

任何代码修改也都能够热更新：

```
...  
1:42:21 PM [vite] hmr update /src/App.tsx
```

```
### 修改代码
```

```
#### 窗口样式和布局
```

为了模仿登录界面，在 `main.go` 文件中，通过 Wails 框架的配置选项修改了应用程序窗口的尺寸 `Width` & `Height`、背景色 `BackgroundColour` 和标题 `Title`。

```
m lv-k3u1fbpfcp-jj-  
mark:3024:0:0:0:q75.awebp#?w=280&h=433&s=22674&e=png&b=fcfcf)
```

底部图标：

```
![底部图标](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/b1afc4120eef4c2fa66c82eceb865dd4~tplv-k3u1fbpfcp-jj-  
mark:3024:0:0:0:q75.awebp#?w=200&h=209&s=18062&e=png&b=f4f4f4  
)
```

打包应用

在项目根目录，运行 `wails build` 即可打包当前环境下的应用程序。但是在开发模式下，已经有了一些缓存文件，可以配合 `-clean` 来清理 `build/bin` 目录：

```
...  
wails build -clean
```

打包 macOS App：

```
...  
wails build -platform=darwin/amd64
```

打包 Windows 程序：

```
...  
wails build -platform=windows/amd64
```

![打包](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/e33ec0137a384b58b79123eb8ca9624a~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=569&h=128&s=17551&e=png&b=fafafa)

使用 [create-dmg](http://cxyroad.com/ "https://github.com/create-dmg/create-dmg") 为 macOS 创建 `.dmg` 文件：

```
...  
create-dmg WeChat.dmg WeChat.app
```

...

![macOS](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/175417ebdd684cfbad2f99f76737adfc~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=300&h=240&s=25627&e=png&b=fbfbfb)

以上文件可以进入 Releases 页面查看：

[github.com/mazeyqian/g...](http://cxyroad.com/"https://github.com/mazeyqian/go-run-wechat-demo/releases/tag/v1.0.0")

![Releases](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/176aaa43b7f5458ab6f61b601fb9d6bc~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=600&h=150&s=22250&e=png&b=fcfcfc)

总结

--

Wails 框架提供了一种简洁而强大的方式，让开发者能够利用 Go 的性能优势和 Web 前端的灵活性，从而能够使用更高效、更轻量级的方法来构建跨平台的桌面应用。

****版权声明****

本博客所有的原创文章，作者皆保留版权。转载必须包含本声明，保持本文完整，并以超链接形式注明作者[后除](http://cxyroad.com/"https://github.com/mazeyqian")和本文原始地址：
： [blog.mazey.net/4499.html](http://cxyroad.com/"https://blog.mazey.net/4499.html")

(完)

原文链接: <https://juejin.cn/post/7354075693947404351>