

部署指南

本项目是纯前端应用，推荐使用 Vercel 进行部署。

一、部署前准备

1. 初始化 Git 仓库

```
# 在项目根目录执行
git init
git add .
git commit -m "Initial commit"
```

2. 创建 GitHub 仓库

1. 登录 [GitHub](#)
2. 点击右上角 `+` `→` `New repository`
3. 填写仓库名称（如 `prompt-manager`）
4. 选择 `Public` 或 `Private`
5. 点击 `Create repository`

3. 推送代码到 GitHub

```
# 将 <your-username> 替换为你的 GitHub 用户名
# 将 <repo-name> 替换为你的仓库名
git remote add origin https://github.com/<your-username>/<repo-name>.git
git branch -M main
git push -u origin main
```

二、Vercel 部署（推荐）

1. 注册/登录 Vercel

1. 访问 [Vercel](#)
2. 点击 `Sign Up` 或 `Log In`
3. 选择 `Continue with GitHub` 使用 GitHub 账号登录

2. 导入项目

- 1. 登录后点击 `Add New...` → `Project`
- 2. 在 `Import Git Repository` 中找到你的仓库
- 3. 点击 `Import`

3. 配置项目

Vercel 会自动检测到这是 Vite 项目，默认配置通常无需修改：

配置项	值
Framework Preset	Vite
Build Command	<code>npm run build</code>
Output Directory	<code>dist</code>
Install Command	<code>npm install</code>

4. 部署

- 1. 点击 `Deploy` 按钮
- 2. 等待构建完成（通常 1-2 分钟）
- 3. 部署成功后获得访问链接：`https://<project-name>.vercel.app`

5. 后续更新

每次将代码 push 到 GitHub，Vercel 会自动重新部署：

```
git add .
git commit -m "Update something"
git push
```

三、自定义域名（可选）

1. 在 Vercel 添加域名

- 1. 进入项目 Dashboard
- 2. 点击 `Settings` → `Domains`

3. 输入你的域名（如 `prompt.example.com`）

4. 点击 `Add`

2. 配置 DNS

根据 Vercel 提示，在你的域名服务商处添加 DNS 记录：

方式一：CNAME 记录（推荐子域名）

类型：CNAME

名称：prompt（或你想要的子域名）

值：cname.vercel-dns.com

方式二：A 记录（根域名）

类型：A

名称：@

值：76.76.21.21

DNS 生效后（通常几分钟到 24 小时），即可通过自定义域名访问。

四、备选部署方案

GitHub Pages

1. 修改 `vite.config.ts`：

```
export default defineConfig({
  base: '/<repo-name>', // 替换为你的仓库名
  plugins: [react(), tailwindcss()],
})
```

2. 安装部署工具：

```
npm install -D gh-pages
```

3. 在 `package.json` 添加脚本：

```
{
  "scripts": {
    "deploy": "npm run build && gh-pages -d dist"
  }
}
```

4. 执行部署：

```
npm run deploy
```

5. 在 GitHub 仓库的 `Settings` → `Pages` 中设置 Source 为 `gh-pages` 分支

Netlify

1. 访问 [Netlify](#) 并用 GitHub 登录
2. 点击 `Add new site` → `Import an existing project`
3. 选择 GitHub 仓库
4. 构建设置保持默认，点击 `Deploy site`

自有服务器（Nginx）

1. 构建项目：

```
npm run build
```

2. 将 `dist` 目录上传到服务器

3. Nginx 配置示例：

```
server {  
    listen 80;  
    server_name your-domain.com;  
    root /var/www/prompt-manager/dist;  
    index index.html;  
  
    location / {  
        try_files $uri $uri/ /index.html;  
    }  
  
    # 启用 gzip 压缩  
    gzip on;  
    gzip_types text/plain text/css application/json application/javascript text/xml  
    application/xml;  
}
```

五、常见问题

Q: 部署后页面空白?

检查浏览器控制台是否有资源加载错误，可能是 `base` 路径配置问题。

Q: 刷新页面 404?

单页应用需要服务器配置将所有路由指向 `index.html`。Vercel/Netlify 已自动处理，自有服务器需配置 Nginx 的 `try_files`。

Q: 数据会丢失吗?

本应用使用 `localStorage` 存储数据，数据保存在用户浏览器本地：

- 同一浏览器、同一域名下数据持久保存
- 清除浏览器数据会导致数据丢失
- 不同设备/浏览器数据不同步

建议用户定期使用导出功能备份数据。

六、技术栈

技术	版本
React	18.x
TypeScript	5.x
Vite	6.x
Tailwind CSS	4.x
Framer Motion	11.x

如有问题，请提交 Issue 或联系开发者。