

# 如何將智能知識庫 Agent 部署到 Railway

這份指南將協助您將 V14 版本的單一檔案 React 應用程式，轉換為可部署的標準專案，並上傳至 Railway。

## 第一步：建立本地專案結構

請在您的電腦上建立一個新資料夾(例如 knowledge-base-agent)，並依照以下結構建立檔案：

```
knowledge-base-agent/
├── public/
├── src/
│   ├── App.jsx    <-- 這是原本的 KnowledgeBaseAgentV14.jsx (內容要複製進來)
│   ├── main.jsx   <-- 入口點 (下方有程式碼)
│   └── index.css   <-- Tailwind 設定 (下方有程式碼)
├── index.html     <-- 網頁入口 (下方有程式碼)
├── package.json   <-- 專案設定 (下方有程式碼)
├── vite.config.js <-- 建置設定 (下方有程式碼)
├── tailwind.config.js <-- 樣式設定 (下方有程式碼)
└── postcss.config.js <-- CSS 處理設定 (下方有程式碼)
```

## 第二步：複製設定檔內容

請依序建立以下檔案，並複製貼上對應的程式碼。

### 1. package.json

定義專案依賴，Railway 會讀取這個檔案來安裝環境。

```
{
  "name": "knowledge-base-agent",
  "private": true,
  "version": "1.0.0",
  "type": "module",
  "scripts": {
    "dev": "vite",
    "build": "vite build",
    "preview": "vite preview"
  },
  "dependencies": {
    "react": "^18.2.0",
    "react-dom": "^18.2.0",
```

```

    "lucide-react": "^0.263.1"
  },
  "devDependencies": {
    "@types/react": "^18.2.15",
    "@types/react-dom": "^18.2.7",
    "@vitejs/plugin-react": "^4.0.3",
    "autoprefixer": "^10.4.14",
    "postcss": "^8.4.27",
    "tailwindcss": "^3.3.3",
    "vite": "^4.4.5"
  },
  "engines": {
    "node": "18.x"
  }
}

```

## 2. vite.config.js

建置工具設定。

```

import { defineConfig } from 'vite'
import react from '@vitejs/plugin-react'

// [https://vitejs.dev/config/](https://vitejs.dev/config/)
export default defineConfig({
  plugins: [react()],
  server: {
    host: true
  }
})

```

## 3. tailwind.config.js

樣式設定, 確保 Agent 的美觀 UI 能正常顯示。

```

/** @type {import('tailwindcss').Config} */
export default {
  content: [
    "./index.html",
    "./src/**/*.{js,ts,jsx,tsx}",
  ],
  theme: {

```

```
    extend: {},  
  },  
  plugins: [],  
}
```

## 4. postcss.config.js

```
export default {  
  plugins: {  
    tailwindcss: {},  
    autoprefixer: {},  
  },  
}
```

## 5. index.html

放在專案根目錄。

```
<!doctype html>  
<html lang="en">  
  <head>  
    <meta charset="UTF-8" />  
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />  
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />  
    <title>智能知識庫 Agent V14</title>  
  </head>  
  <body>  
    <div id="root"></div>  
    <script type="module" src="/src/main.jsx"></script>  
  </body>  
</html>
```

## 6. src/index.css

引入 Tailwind CSS。

```
@tailwind base;  
@tailwind components;  
@tailwind utilities;
```

```
body {  
  margin: 0;
```

```
padding: 0;
background-color: #f2f6fa;
}
```

## 7. src/main.jsx

React 的入口點。

```
import React from 'react'
import ReactDOM from 'react-dom/client'
import App from './App.jsx'
import './index.css'
```

```
ReactDOM.createRoot(document.getElementById('root')).render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
)
```

## 8. src/App.jsx (關鍵步驟)

1. 在 src 資料夾內建立 App.jsx。
2. 將剛剛對話視窗中 **KnowledgeBaseAgentV14.jsx** 的所有程式碼完整複製。
3. 貼上到 src/App.jsx 中。
4. 重要修改：找到程式碼最後一行的 `export default function KnowledgeBaseAgentV14() { ... }`，建議將函數名稱改為 `App`，或者保持原樣（因為是 default export，import 端改名即可，上述 main.jsx 已經處理了）。只要確保檔案內容是完整的即可。

## 第三步：推送到 GitHub

1. 在 GitHub 上建立一個新的 Repository（例如命名為 kb-agent）。
2. 在本地專案資料夾開啟終端機 (Terminal)，執行以下指令：

```
# 初始化 git
git init
```

```
# 加入所有檔案
git add .
```

```
# 提交
git commit -m "Initial commit V14 Agent"
```

```
# 連接到您的 GitHub Repo (請將 URL 換成您自己的)
```

```
git remote add origin  
[https://github.com/YOUR_USERNAME/kb-agent.git](https://github.com/YOUR_USERNAME/kb-agent.git)
```

# 推送

```
git push -u origin main
```

## 第四步：在 Railway 上部署

1. 前往 [Railway Dashboard](#) 並登入。
2. 點擊 "New Project"。
3. 選擇 "Deploy from GitHub repo"。
4. 選擇您剛剛建立的 kb-agent Repository。
5. 點擊 "Deploy Now"。

Railway 會自動識別這是一個 Vite 專案，並執行 npm install 和 npm run build。

### 設定公開網址

1. 部署完成後，點擊該專案卡片。
2. 前往 "Settings" -> "Networking"。
3. 點擊 "Generate Domain"。
4. 您現在會獲得一個 xxx.up.railway.app 的網址，點擊即可開始使用您的智能知識庫 Agent！

## 常見問題

Q: 為什麼圖片無法顯示？

A: 此 Agent 目前是純前端模擬，所有資料都存在記憶體中（重新整理網頁後資料會重置）。如果您希望資料能永久保存，未來需要對接後端資料庫（如 Firebase 或 Supabase）。

Q: 部署失敗怎麼辦？

A: 請檢查 Railway 的 "Build Logs"。最常見的原因是 package.json 遺失或 src 檔案結構不正確。確保 vite.config.js 存在於根目錄。