

Quick Start Guide

Verificación del Proyecto

El proyecto ha sido configurado y verificado correctamente. Todos los archivos de TypeScript se compilan sin errores.

✓ Verificado

- ✓ Estructura de carpetas creada
- ✓ Dependencias instaladas (390 paquetes)
- ✓ TypeScript configurado correctamente para main, renderer y preload
- ✓ Build exitoso (`npm run build`)
- ✓ Electron configurado con seguridad estricta
- ✓ React + Vite configurado
- ✓ Preload script con APIs IPC tipadas
- ✓ Git inicializado con commit inicial

🚀 Cómo Ejecutar

Para ejecutar la aplicación en modo desarrollo:

```
cd /home/ubuntu/mindmapper
npm run dev
```

Esto iniciará:

1. El servidor de desarrollo de Vite en `http://localhost:5173`
2. La aplicación Electron que cargará el contenido de Vite
3. Las DevTools se abrirán automáticamente

📦 Cómo Empaquetar

Para crear un ejecutable:

```
# Para la plataforma actual
npm run package

# 0 para plataformas específicas
npm run package:win      # Windows
npm run package:mac      # macOS
npm run package:linux    # Linux
```

Los ejecutables se guardarán en la carpeta `build/`.

🔍 Verificar Build

Para verificar que todo compila correctamente:

```
npm run build
```

Este comando:





1. Compila TypeScript del renderer con `tsc`
2. Bundlea React con Vite
3. Compila TypeScript del main process
4. Compila TypeScript del preload script

Estructura del Proyecto

```
mindmapper/
├── src/
│   ├── main/
│   │   ├── main.ts           # Proceso principal de Electron
│   │   └── preload/
│   │       ├── preload.ts    # Bridge seguro IPC
│   │       └── renderer/
│   │           ├── components/ # Componentes React (vacío por ahora)
│   │           ├── store/
│   │           │   ├── mindMapStore.ts # Store de Zustand
│   │           │   ├── styles/
│   │           │   │   ├── App.css     # Estilos del App
│   │           │   │   └── index.css    # Estilos globales
│   │           │   └── types/
│   │           │       ├── electron.d.ts # Tipos para Electron API
│   │           │       ├── App.tsx      # Componente principal
│   │           │       ├── main.tsx     # Entry point de React
│   │           │       └── index.html    # HTML template
│   │           └── dist/                # Build output
│   ├── node_modules/                  # Dependencias (390 paquetes)
│   ├── package.json                   # Configuración del proyecto
│   ├── tsconfig.*.json                # Configuraciones TypeScript
│   ├── vite.config.ts                 # Configuración de Vite
│   └── README.md                      # Documentación principal
```

Seguridad

La aplicación implementa todas las mejores prácticas de seguridad de Electron:

-  `contextIsolation: true` - Aisla el contexto del preload
-  `sandbox: true` - Habilita el sandbox de Chrome
-  `nodeIntegration: false` - Deshabilita Node.js en el renderer
-  Preload script tipado - APIs seguras expuestas mediante IPC

Notas

- El proyecto usa **Electron 28**, **React 18**, **TypeScript 5** y **Vite 5**
- Todas las comunicaciones entre procesos usan IPC handlers seguros
- Los tipos están completamente definidos en `electron.d.ts`
- El estado se maneja con Zustand (store minimalista por ahora)
- Los iconos vienen de Lucide React
- Dagre está instalado para futuros algoritmos de layout

Próximos Pasos (Fase 2)

El scaffold está completo. Las siguientes fases incluirán:

1. **Canvas de Mind Map:** Implementar el área de trabajo con zoom y pan
2. **Sistema de Nodos:** Crear, editar y eliminar nodos

3. **Conexiones:** Conectar nodos con líneas/flechas
4. **Persistencia:** Guardar y cargar archivos .mindmap
5. **Layout Automático:** Usar Dagre para organizar nodos
6. **Temas:** Modo claro/oscuro
7. **Atajos de teclado:** Mejoras de UX
8. **Export:** Exportar a PNG, SVG, PDF

? Troubleshooting

Si `npm run dev` no funciona:

1. Asegúrate de que el puerto 5173 esté libre
2. Verifica que Node.js sea versión 18+: `node --version`
3. Reinstala dependencias: `rm -rf node_modules && npm install`
4. Verifica que el build funciona: `npm run build`

Si Electron no abre:

1. En Linux, puede requerir permisos adicionales
2. Asegúrate de tener un entorno gráfico disponible
3. Verifica los logs en la consola



Recursos

- [Electron Documentation](https://www.electronjs.org/docs) (https://www.electronjs.org/docs)
- [React Documentation](https://react.dev) (https://react.dev)
- [Vite Documentation](https://vitejs.dev) (https://vitejs.dev)
- [Zustand Documentation](https://github.com/pmndrs/zustand) (https://github.com/pmndrs/zustand)
- [Electron Security Guide](https://www.electronjs.org/docs/latest/tutorial/security) (https://www.electronjs.org/docs/latest/tutorial/security)