https://ui.shadcn.com Docs Components Blocks Charts Themes Colors Search... Docs > components.json **Getting Started** components.json Introduction Installation Configuration for your project. components.json The components.json file holds configuration for your project. **Theming** Dark mode We use it to understand how your project is set up and how to CLI generate components customized for your project. Monorepo New Note: The components.json file is optional and only required if you're Next.js 15 + React 19 using the CLI to add components to your project. If you're using the copy Typography and paste method, you don't need this file. Open in vO Figma You can create a components.json file in your project by running the following command: Changelog Installation pnpm npm yarn bun Next.js pnpm dlx shadcn@latest init Vite Remix See the CLI section for more information. Astro Laravel \$schema Gatsby Manual You can see the JSON Schema for components.json here. Components components.json Accordion Alert "\$schema": "https://ui.shadcn.com/schema.json" Alert Dialog Aspect Ratio Avatar style Badge Breadcrumb The style for your components. This cannot be changed after **Button** initialization. Calendar Card components.json Carousel Chart "style": "default" / "new-york" Preview Code Style: New York > Create project Deploy your new project in one-click. Name Name of your project Framework Select Deploy Cancel tailwind Configuration to help the CLI understand how Tailwind CSS is set up in your project. See the installation section for how to set up Tailwind CSS. tailwind.config Path to where your tailwind.config.js file is located. components.json "tailwind": { "config": "tailwind.config.js" / "tailwind.config.ts" tailwind.css Path to the CSS file that imports Tailwind CSS into your project. components.json "tailwind": { "css": "styles/global.css" tailwind.baseColor This is used to generate the default color palette for your components. This cannot be changed after initialization. components.json "tailwind": { "baseColor": "gray" / "neutral" / "slate" / "stone" / "zinc" tailwind.cssVariables You can choose between using CSS variables or Tailwind CSS utility classes for theming. To use utility classes for theming set tailwind.cssVariables to false. For CSS variables, set tailwind.cssVariables to true. components.json 凸 "tailwind": { "cssVariables": `true` / `false` For more information, see the theming docs. This cannot be changed after initialization. To switch between CSS variables and utility classes, you'll have to delete and re-install your components. tailwind.prefix The prefix to use for your Tailwind CSS utility classes. Components will be added with this prefix. components.json "tailwind": { "prefix": "tw-" rsc Whether or not to enable support for React Server Components. The CLI automatically adds a use client directive to client components when set to true. components.json "rsc": 'true' / 'false' tsx Choose between TypeScript or JavaScript components. Setting this option to false allows components to be added as JavaScript with the .jsx file extension. components.json "tsx": 'true' / 'false' aliases The CLI uses these values and the paths config from your tsconfig.json or jsconfig.json file to place generated components in the correct location. Path aliases have to be set up in your tsconfig.json or jsconfig.json file. Important: If you're using the src directory, make sure it is included under paths in your tsconfig.json or jsconfig.json file. aliases.utils Import alias for your utility functions. components.json "aliases": { "utils": "@/lib/utils" aliases.components Import alias for your components. components.json "aliases": { "components": "@/components" aliases.ui Import alias for ui components. The CLI will use the aliases.ui value to determine where to place your ui components. Use this config if you want to customize the installation directory for your ui components. components.json "aliases": { "ui": "@/app/ui" aliases.lib Import alias for lib functions such as format-date or generate-id. components.json { "aliases": { "lib": "@/lib" aliases.hooks Import alias for hooks such as use-media-query or use-toast. components.json "aliases": { "hooks": "@/hooks" < Installation Theming > Built by shaden. The source code is available on G Created by JUST ONE PAGE PDF