**Metadados de Controle de Versão**

.git/\*\*/\*

Metadados internos do Git e histórico (HEAD, refs, logs, objetos empacotados etc.). Esses arquivos mantêm o controle de versão do projeto.

.gitignore

Especifica arquivos e diretórios que o Git deve ignorar (por exemplo, artefatos de build).

**Documentação do Projeto**

README.md

README em português descrevendo o editor de desenho em braille, passos de instalação e status do projeto.

CONTRIBUTING.md

Diretrizes breves de contribuição: criar branches, abrir pull requests, manter compatibilidade de código e testar antes de submeter.

**Arquivos de Configuração**

components.json

Configuração para componentes do shadcn/ui, definindo aliases, ajustes do Tailwind e opções de estilo.

package.json & package-lock.json

Definições de pacotes Node, listando dependências e scripts para desenvolvimento, build, lint etc.

tsconfig.json, tsconfig.app.json, tsconfig.node.json

Configurações do TypeScript. O tsconfig.app define modo do bundler e opções JSX, enquanto o tsconfig.node é voltado para arquivos de configuração do Vite.

tailwind.config.ts

Setup do Tailwind CSS com paleta de cores e configuração de plugins para o design system.

postcss.config.js

Plugins do PostCSS (Tailwind e autoprefixer).

eslint.config.js

Configuração do ESLint usando TypeScript ESLint e plugin de React Hooks.

vite.config.ts

Setup de build do Vite com plugin React e alias `@/` apontando para a pasta src.

**Recursos Estáticos**

index.html

Ponto de entrada HTML contendo meta tags, fontes e a div root onde o app React monta.

public/logo-og.png, public/logo-puncao.png

Logos da aplicação usadas para compartilhamento em redes sociais e no cabeçalho do site.

**Código-fonte (src/)**

src/main.tsx

Script de entrada que monta a aplicação React.

src/index.css

Estilos Tailwind CSS e variáveis CSS customizadas definindo paleta de cores e fontes.

**Componente Raiz do App**

src/App.tsx

Configura rotas com React Router, cliente React Query, providers de tooltip e toast.

**Páginas**

src/pages/Index.tsx

Exibe o componente BrailleEditor na rota raiz.

src/pages/BrailleEditor.tsx

Página principal do editor. Gerencia estado de ferramentas, histórico de desfazer/refazer, atalhos de teclado, seleção, hooks de desenho e modais.

src/pages/NotFound.tsx

Página 404 simples que registra caminhos inválidos no console.

**Tipos e Utilitários**

src/types/braille.ts

Interfaces TypeScript definindo BrailleCell, BrailleGrid, enums de ferramentas e tipos auxiliares.

src/lib/brailleMappings.ts

Mapeamento de letras para padrões de pontos em braille e funções auxiliares para conversão entre texto e braille.

src/lib/utils.ts

Função utilitária cn para combinar nomes de classe via clsx e tailwind-merge.

**Hooks**

src/hooks/use-mobile.tsx

Detecta tamanho de viewport móvel.

src/hooks/use-toast.tsx

Sistema customizado de notificações toast, construído sobre Radix UI.

src/hooks/useDrawing.ts

Lida com desenho e apagamento de pontos braille no canvas. Rastreia ações do mouse e atualiza células da grade.

src/hooks/useKeyboardShortcuts.ts

Escuta eventos de teclado para desfazer/refazer, copiar/colar, alternar visualização, mover seleção etc.

src/hooks/useSelection.ts

Gerencia seleção de células, operações de clipboard (copiar, recortar, colar) e exclusão.

src/hooks/useShapes.ts

Auxiliar de desenho para formas (linhas, retângulos, círculos, triângulos) e flood-fill.

src/hooks/useTextInsertion.ts

Insere texto na grade convertendo caracteres em padrões de braille.

src/hooks/useTextOverlay.ts

Mantém elementos de sobreposição de texto arrastáveis que podem ser editados ou excluídos.

**Componentes de UI**

src/components/ui/button.tsx, input.tsx

Componentes primitivos de botão e input estilizados com Tailwind e class-variance-authority.

src/components/ui/tooltip.tsx

Wrapper para componentes Radix Tooltip.

src/components/ui/toast.tsx, toaster.tsx

Componentes de toast e provider para exibição de notificações.

src/components/ui/sonner.tsx

Integração com a biblioteca “sonner” de toasts.

src/components/ui/sidebar.tsx

Implementação extensa de sidebar (colapsável, responsiva) usando Radix Dialog e hooks customizados.

**Componentes do Editor**

src/components/Canvas/BrailleGrid.tsx

Componente de renderização do canvas. Desenha a grade de braille, letras ou pontos, e trata eventos de mouse para desenho e seleção.

src/components/Canvas/DrawingArea.tsx

Envolve o BrailleGrid com controles de zoom, resolução e sobreposições de texto.

src/components/CellEditor/CellEditor.tsx

Modal para editar manualmente uma única célula de braille (letra e posições dos pontos).

src/components/TextOverlay/TextElement.tsx

Componente de sobreposição de texto arrastável que pode renderizar texto em braille, ser editado ou excluído.

src/components/Controls/CopyLetters.tsx

Copia letras da grade para o clipboard usando sistema de toast para feedback.

src/components/Controls/ResolutionControls.tsx

Dropdown e inputs para alterar dimensões da grade.

src/components/Controls/ZoomControls.tsx

Botões e menu select para ajustar nível de zoom.

src/components/HelpModal/HelpModal.tsx

Modal exibindo atalhos de teclado e instruções.

src/components/Sidebar/AppSidebar.tsx

Toolbar com botões de ferramentas agrupados (desfazer/refazer, ferramentas de desenho, ações de clipboard, ajuda). Usa primitivos de sidebar UI.

src/components/Sidebar/Sidebar.tsx

Implementação alternativa de sidebar usando componentes ToolButton.

src/components/Sidebar/ToolButton.tsx

Botão reutilizável para ferramentas na sidebar.

**Configuração de Build**

vite.config.ts

(ver seção de configuração acima) – define host do dev server e alias.

tailwind.config.ts

(ver seção de configuração acima) – define tokens de design e uso de plugins.

tsconfig.\*.json

Configurações do TypeScript.

Resumo

No total, esses arquivos formam um app web React/TypeScript para desenhar caracteres em braille em uma grade, com ferramentas de formas, sobreposições de texto, seleção e clipboard, sidebar UI e notificações toast. Os arquivos de configuração definem TypeScript, Vite, Tailwind, ESLint e shadcn UI. A pasta public armazena imagens, enquanto index.html e src/main.tsx iniciam a aplicação. O diretório .git contém o histórico de controle de versão. esses são todos os arquivos do repositório.