Anpassningar för Flashcards-App

# UI & komponenter (shadcn/ui)

- Tailwind-konfig: säkerställ att shadcn-paths är med i tailwind.config.ts.  
- Dark mode: använd shadcn's tokens och dark-toggle.  
- Form-hantering: react-hook-form + zod för validering.  
- Toast/feedback: använd shadcn toast-komponent.  
- Dialog/Drawer: redigering av kort i modaler.  
- Tillgänglighet: tangentbordsstöd för flip, respekt för prefers-reduced-motion.

# Data, validering & progress

- Zod-schema med tydliga fel vid import.  
- Stabila card-id:n (hash på fråga + kategori).  
- Progress-mappning vid uppdateringar.  
- IndexedDB-migreringar vid schemaVersion.  
- Lagra dueAt i ISO (UTC).

# Import/Export & källor

- UI-uppladdning med filstorleksgräns och typkontroll.  
- URL-import med ETag/If-None-Match.  
- Export: kort eller kort + progress.  
- Sanering av HTML (dompurify) om markdown tillåts.

# PWA & offline

- vite-plugin-pwa: cache statiska assets men exkludera runtime/config.json från hård cache.  
- Offline-fallback: tydlig UI-state om fjärr-URL inte kan hämtas.

# Docker & server

- docker-entrypoint.sh genererar runtime/config.json.  
- SPA-routing i Nginx: try\_files $uri /index.html;.  
- Hantera CORS för externa lekar.  
- Säkerhetsheaders: CSP, X-Content-Type-Options.

# Prestanda & UX-polish

- Virtuellt listande (react-virtual) för många kort.  
- Debounce autosave (300–500 ms).  
- Globala hotkeys (flip, rätt/fel, nästa).  
- Loading states: skeletons/spinners från shadcn.  
- Enkel lokal statistik (rätt/fel per kategori).

# Test & kvalitet

- Zod-tests för korrupt JSON.  
- SR-logik (bucket/SM-2) enhetstester.  
- UI-tester: flip, tangenter, importfel, editor autosave.  
- ESLint/Prettier + husky för pre-commit checks.

# React Router

- Installera react-router-dom.  
- Skapa sidor: Home, Train, Edit, Manage.  
- Använd <BrowserRouter> i main.tsx.  
- Strukturera routes för enkel navigation:  
 / = hem  
 /train/:deckId = övningsläge  
 /edit/:deckId = redigering  
 /manage = hantera lekar