

■ Front-End Styling Guidelines (React + Bootstrap + CSS)

1. General Principles

- Bootstrap utilities first → use them whenever possible for spacing, alignment, colors, font sizes, etc.
- Custom CSS for design system → fonts, brand colors, shadows, gradients, and unique styles.
- React components for repeating patterns → don't duplicate CSS for section headers or cards; abstract into reusable components.
- Keep CSS minimal → only for things Bootstrap doesn't cover well.

2. Typography

- Font family & base font size → set in global CSS (body { font-family: ...; font-size: 16px; }).
- Responsive text → use Bootstrap fs-* utilities (fs-6 fs-md-5 fs-lg-4).
- Font weight & text styles → use Bootstrap (fw-bold, fw-semibold, text-uppercase, text-muted).
- Custom sizes (like 15px) → write in CSS under a design token (e.g., .text-small { font-size: 15px; }).

3. Colors

- Use Bootstrap's text and background utilities (text-white, bg-dark) for simple cases.
- For brand-specific colors, define CSS variables in :root.
- Apply via CSS classes or inline styles, not hardcoded hex values in components.

4. Layout & Spacing

- Bootstrap utilities for spacing → px-3 py-2 mb-4 gap-2.
- Flexbox/Grid → use Bootstrap (d-flex flex-column justify-content-between align-items-center).
- Only use CSS for complex layouts (nested grids, custom breakpoints).

5. Sidebar Example

- CSS handles structure.
- Bootstrap handles spacing, alignment, text color.

6. Components vs. CSS

- If a style is used in 3+ places, make a React component (not a CSS class).
- Use CSS classes only for low-level styling (fonts, colors, variables).

7. Don'ts ■

- Don't use !important unless absolutely necessary (use utilities instead).

- Don't duplicate Bootstrap classes in your CSS (e.g., don't write `.text-white { color: white; }`).
- Don't hardcode inline styles for brand colors; use variables or utility classes.

8. File Structure Suggestion

- `/src`
- `/components`
- `Sidebar.jsx`
- `SidebarSection.jsx`
- `/styles`
- `variables.css` # brand colors, font tokens
- `global.css` # body, baseline typography
- `sidebar.css` # structural sidebar styles only

Summary

- Bootstrap = utilities for spacing, responsiveness, typography, alignment.
- CSS = design tokens + baseline + brand-specific overrides.
- React components = repeating UI patterns.