

You are the design + front-end team. Analyze current project to find the existing logo assets and integrate the official CyberSathi (Flask screen app) logo into the app header, navbar, favicon, mobile nav, login page, and meta tags.

NOTE: I could not access the uploaded ZIP asset from this environment. If `assets/logo.svg` or `static/logo.png` exists, use that. If not, create a placeholder `logo.svg` using the color palette used in the screenshots: deep purple (`#5B38FF`) accent, teal/green (`#2FA47A`) for "Sathi", and use simple "CyberSathi" wordmark + rounded flask-like emblem. Replace later with the real asset.

TASKS:

1. **Search & Identify**:

- Search repo for any `logo.*` files and list their paths. If multiple variants exist, prefer `logo.svg` (vector) for retina quality.

2. **Asset placement**:

- Place primary logo at `frontend/public/assets/logo.svg` and fallback at `frontend/public/assets/logo.png`.
- Place favicon `frontend/public/favicon.ico` and `favicon-32x32.png`, `favicon-16x16.png`. Add `manifest.json` entry if not present.

3. **Navbar integration**:

- Replace current text header with an accessible logo component:
 - ``
 - ``
 - `CyberSathi`
 - Use `srcset` for retina: `logo@2x.png 2x`
- Ensure responsive behavior: logo reduces to circular avatar on small screens (mobile).

4. **Login & Profile**:

- Use square variant `logo-square.svg` (40x40) in login header and user dropdown.

5. **Favicon & meta**:

- Update `index.html` head with favicon links and `<meta property="og:image" content="/assets/logo.png" />`.

6. **Styling**:

- Add CSS/Tailwind utility classes:
 - `h-10 md:h-12` for desktop, `h-8` for small devices.
 - Add `filter drop-shadow` for subtle elevation.

7. **Accessibility**:

- Provide `alt` text for each image.
- Ensure `aria-label` on logo link for screen readers.

8. **Image optimization**:

- Provide an SVG optimized file (remove metadata). If only PNG exists, generate `logo.webps` and include `` fallback.

9. **Implementation details & files to change**:

- `frontend/src/components/Header.jsx` or `Navbar.jsx` — replace current title with `` component.
- Add `frontend/src/components/Logo.jsx` reusable component that accepts `size` prop.
- Update `public/index.html` head to include favicons.

10. ACCEPTANCE CRITERIA:

- Logo renders in navbar, mobile nav, login page, and user dropdown.
- Favicon is visible in the browser tab.
- Images are responsive and accessible; retina support implemented.
- All paths are referenced relative and fallback to PNG if SVG fails.

EXAMPLE React `Logo.jsx` component (to implement):

```
``jsx
export default function Logo({size = 'md'}) {
  const heights = {sm: 'h-6', md: 'h-10', lg: 'h-12'}
  return (
    <a href="/" aria-label="CyberSathi — Cybercrime Helpline 1930" className="flex items-center gap-3">
      
      <span className="hidden md:inline font-semibold text-xl text-gray-800">CyberSathi</span>
    </a>
  )
}
```