PII Detection/Redaction: Simple Deployment Plan

What I am trying to do

We want to stop PII from leaking anywhere. So I am putting the checks at a few practical places:

- **Backend middleware (Express)** – main filter. Works on both request and response.
- **MCP tool wrapper** – when we call external tools, we clean their output before showing or saving.
- **Frontend proxy (Next.js)** – small pre-check so obviously bad payloads don't even hit backend.
- **Logs/SSE** – scrub lines before streaming to UI.
- **Database hooks + nightly batch** – mask before write and clean old data daily.

Why this layout

- **Low effort** – mostly middleware/wrapper, no big refactor.
- **Latency is okay** – 2–5 ms per API call, < 1 ms per log line.
- **Scales naturally** – same as our app replicas and tool call volume.
- **Config driven** – one policy file with regex + actions (detect/mask/block).

Rollout plan

1. Run in *detect-only* for a week, watch counts on dashboard.
2. Turn on *mask* for phone/email/UPI on `/api/analyze*`.
3. Expand to all endpoints. Block Aadhaar/Passport in requests.
4. Enable DB hooks and first batch job. Share weekly report.

Architecture (clean diagram)