Aurora OS

0. Product Name & One-Liner

(Working name; swap later.)

Name: Aurora Research OS
One-liner:

> AI-assisted, template-locked “Research-in-a-Box” that turns a clinician’s idea into a coherent, standards-aligned **baseline draft package** and live eCRF workspace—so good ideas don’t die before Ethics.

---

1. First Principles (Everyone Agrees)

P1. Clinician-first.
Start from how clinicians naturally talk/think. No stats jargon gatekeeping. The system translates clinical reasoning into structured research.

P2. Baseline, not black box.
We generate **strong drafts**, not decrees. Everything is editable. PI + IEC remain responsible.

P3. Constrained, not “anything goes”.
v1 only supports well-understood designs:

- Prospective / retrospective cohort
- Simple registries
- Simple 2-arm RCTs
- Diagnostic accuracy studies

Out-of-scope ideas are explicitly called out as such.

P4. Transparent & auditable.
Every assumption, formula, and AI decision traceable. No magic.

P5. Ethics & safety by design.
We enforce presence of core ethical elements and block obviously unsafe / nonsensical setups.

P6. Plug-in humans, don’t replace them.
Easy share flows for statisticians, IEC members, research offices to review and improve the

draft.

This is the lens through which all specs below are set.

2. Core Value Proposition

For clinicians:
Turn, “We should study this,” into a **concrete starting package in hours**, without leaving your clinical comfort zone.

For IECs & biostatisticians:
Receive coherent, structured proposals with transparent assumptions instead of chaotic Word docs.

For institutions:
Standardized, auditable study workflows that lower friction and raise baseline quality.

3. In-Scope vs Out-of-Scope (Hard Lines)

In v1 (must work well):

- Supported study types:
  - Cohort (prospective/retrospective)
  - Disease/quality registry
  - Simple 2-arm randomized trial
  - Diagnostic accuracy
- Adult, non–high-risk scenarios as default.
- Drafting:
  - Protocol
  - Sample size justification
  - SAP
  - CRF / eCRF schema
  - PIS/ICF
  - IEC cover note
  - Compliance checklist
  - Registry mapping sheet (e.g., CTRI-style fields)
- Live:
  - Study workspace
  - eCRFs + roles
  - Audit trail

    - Safe exports

Out of v1:

- Adaptive/Bayesian / complex multi-arm trials.
- High-risk first-in-human / gene therapy style stuff.
- Automatic CTRI/DCGI/IRB submission.
- Any claim of legal / regulatory approval.

UI must clearly say when an idea is **not supported** by v1; no fake confidence.

---

4. Functional Spec (Bible Version)

4.1 FR-0: Clinician Comfort Zone Intake

- Intake is **plain language**:
    - “What problem are you seeing?”
    - “In which patients?”
    - “What do you want to measure/change?”
    - “Over what timeframe?”
    - “What’s realistic in your setting?”
- System converts this to structured JSON (`StudyDefinition`):
    - Condition, population, setting
    - Candidate primary outcome
    - Candidate design from supported set
- Always show:
    - A human-readable “Study Story” alongside structured fields for confirmation.

4.2 FR-1: Design Selection & Validation

From `StudyDefinition`, system:

1. Suggests **one or more supported design patterns**.
2. For chosen pattern, proposes:
    - Primary/secondary objectives
    - Endpoint definitions
    - Inclusion/exclusion criteria
    - Follow-up schedule
    - Core variable list

Constraints:

- All options come from curated **Template Libraries**.
- User can edit but within validation:
  - Every primary endpoint must map to:
    - A variable in the CRF
    - A planned analysis in the SAP

If broken → blocking warnings until resolved/acknowledged.

4.3 FR-2: Deterministic Sample Size Engine

- Implemented purely in backend code (no LLM math).
- Supports:
  - Single-arm precision
  - Two-arm comparisons (means/proportions)
  - Basic survival (log-rank)
  - Diagnostic accuracy (Se/Sp, etc.)

Behaviour:

- Ask user for assumptions or offer **clearly labeled suggestions**.
- Output:
  - N per group/overall
  - Scenario table
  - Auto-generated explanation text sourced from stored inputs.
- Store:
  - Formula choice
  - Parameters
  - Version

LLM may explain; **cannot** alter the numbers.

4.4 FR-3: Document Factory (Baseline Draft Package)

On command ("Generate Baseline Package"), system produces **drafts**:

1. **Protocol**
   - Standard headings (Background, Rationale, Objectives, Methods, Safety, Data Mgmt, etc.).

    - Populated from structured spec + template snippets.
    - Mandatory sections **not deletable** (can be edited, not removed).
2. **SAP**
    - Derived from:
        - Design type
        - Endpoint types
        - Sample size rationale
    - Outlines:
        - Analysis populations
        - Primary/secondary analysis methods
        - Handling of missing data (from templates).
3. **CRF Pack**
    - Auto-build from variable list:
        - Screening, Baseline, Procedure/Exposure, Follow-ups, Outcomes.
    - Exports:
        - Printable PDFs
        - JSON schema for eCRFs.
4. **PIS/ICF**
    - Templates containing all essential elements.
    - Auto-filled with study-specific info.
    - Ready for localization.
5. **IEC Cover Note**
    - Short study summary, design overview, risk/benefit.
6. **Compliance Checklist**
    - Presence (not legal sufficiency) of:
        - Objectives, endpoints, methods
        - Consent, risks, confidentiality, data protection, compensation, etc.
    - Traffic-light style.
7. **Registry Mapping Sheet**
    - Key study fields mapped to public registry-style fields.
    - Flags values PI must add manually (IEC number, DCGI status, etc.).

Mandatory:
Every generated doc has:

- Footer / cover text:
    - “Draft generated by Aurora Research OS based on user inputs.”
    - “Requires review and approval by Principal Investigator and Ethics Committee.”
- No “approved/ certified/ guaranteed” language.

4.5 FR-4: Compliance Gate (“No Blind Launch”)

Before a study can be “Locked & Launched”:

- System runs compliance & consistency checks:
  - Objectives ↔ Outcomes ↔ SAP aligned?
  - PIS/ICF essential clauses present?
  - Study type supported?
- Critical issues:
  - Must be resolved OR explicitly acknowledged via checkbox by PI.
- Only then:
  - Study Version can be locked.
  - Live workspace is generated.

This “gate” is non-negotiable.

4.6 FR-5: Risk & Sanity Guardrails

- If user proposes clearly unethical/unsafe/unsound structure:
  - E.g., high-risk intervention, no safety oversight, no consent.
- System must:
  - Flag with strong warning.
  - Stop auto-normalizing.
  - Suggest modifications or “Expert review required”.
- Logs these events in audit trail.

4.7 FR-6: Study Workspace & eCRFs

When locked:

- System creates **Study Workspace**:
  - Configured eCRFs from CRF schema.
  - Roles:
    - Owner (PI)
    - Editor (Co-I)
    - Data Entry
    - Monitor / Read-only
- Features:
  - Add / edit participant records.

  - Track status: screened / enrolled / completed.
  - Basic dashboard:
    - Enrollment
    - Follow-up
    - Data completeness
- Export:
  - CSV + codebook
  - JSON schema
  - Audit report PDF (config + changes).

Any protocol amendment:

- Creates new StudyVersion.
- Diffs logged (IEC amendment-ready).

4.8 FR-7: Collaboration & Handoff

- “Share for Review” link:
  - Grants read/comment access to:
    - Statisticians
    - IEC/research cell
  - Comments are tracked; no silent overwrites.
- Final control:
  - PI confirms final content.
  - All changes versioned.

This operationalizes “baseline they can build off”.

---

5. Non-Functional Spec

- **Security**
  - HTTPS, TLS
  - Encryption at rest
  - Role-based access
  - Study-level isolation
- **Data Residency**
  - Region selection (e.g., India region for Indian clients).
- **Auditability**
  - Immutable event log:

        - Who changed what, when.
        - Which AI/template version used.
- **Performance**
    - Baseline package generation typically < 60s.
    - eCRF interactions fast (< 300ms typical).
- **Reliability**
    - 99.5% uptime target for core app.
- **No PHI in model training**
    - Production runs on non-retaining LLM endpoints or self-hosted models for PHI zones.

---

6. AI / ML Architecture (Guardrailed)

Components:

1. **Idea Parser**
    - LLM → structured PICO/spec JSON.
    - Validated by schema.
2. **Design Assistant**
    - Runs against **whitelisted templates**.
    - Can only pick/edit inside allowed schema.
3. **Document Composer**
    - Uses structured spec + snippet libraries.
    - No freeform regulatory inventions.
4. **Compliance Checker**
    - LLM + rules:
        - Only flags gaps/inconsistencies.
        - Does NOT claim compliance.

Guardrails:

- Low temperature for structural outputs.
- JSON schema enforcement.
- Deterministic stats outside LLM.
- Every AI step recorded in `AuditEvent`.

---

7. System Architecture (High-Level)

- **Frontend:**
  - Next.js / React
  - Tailwind + component library (e.g., shadcn-style)
- **Backend:**
  - Typed framework (NestJS / similar)
  - Services:
    - Auth / Org
    - Study & Templates
    - Orchestrator (AI pipeline)
    - Docs (DOCX/PDF)
    - CRF & Data
    - Audit Log
- **Data:**
  - Postgres for core entities
  - S3-compatible storage for docs
  - Redis for cache/queues

---

8. Canonical Data Entities

- `User`
- `Organization`
- `Study`
- `StudyVersion`
- `Template` (designs, CRFs, PIS/ICF blocks)
- `Document` (type, version, URL)
- `CRFSchema`
- `ParticipantRecord`
- `AuditEvent` (actor, action, entity, timestamp, payload)

This is the minimum model; extend carefully.

---

9. Golden Path (Demo & Real Use)

1. Clinician: types natural-language idea.
2. System: maps to supported design; clinician confirms.
3. System: guides through endpoints, follow-up, core variables.
4. Sample size: runs deterministic calc, shows explanation.

5. “Generate Baseline Package”:
    - Protocol, SAP, CRFs, PIS/ICF, IEC note, checklist, mapping sheet.
6. Compliance gate: clinician reviews, fixes/acknowledges.
7. “Lock & Launch”:
    - Study workspace + eCRFs go live.
8. Same day:
    - Enter first patient.
    - Export full draft pack + audit summary → send to IEC.

That’s your promise. That’s what everyone on the (real or virtual) team is building.