

Specifications – Psychological Supervision Application

1. Overview

A web application (mobile friendly) designed for psychologists to:

- Manage patient records
- Structure clinical cases
- Interact with a clinical AI assistant
- Take notes (typed or handwritten with OCR recognition)
- Generate anonymized reports

2. User Flow (MVP)

2.1 Patient Management

- Create a patient record:
 - Name or ID/pseudonym
 - Age
 - Reason for consultation
 - Consents
 - Anonymization option on/off

2.2 Patient Instructions

- Structured free text:
 - Case history (anamnesis)
 - Diagnostic hypotheses (CBT, EMDR...)
 - Therapeutic goals
 - Cultural/legal constraints
- Secure storage and consultation

2.3 Interaction with the Clinical Bot

- Chat area per patient
- Suggest formulations, intervention ideas, session plans
- Clinical risk alerts

2.4 Note-taking

- During or after the session
- Standard text mode
- Handwriting mode via stylus/mouse:
 - Drawing area (“net” style)

- Handwriting → text conversion via OCR
- Save both raw handwriting and text transcription
- Automatic linking to the session and patient

2.5 Structured Notes and Trackers

- Evaluation scales
- Goal tracking
- Patient history

2.6 Export and Summary

- Generate anonymized summaries
- PDF export
- Export history

3. Constraints & Security

- Secure authentication (NextAuth, JWT)
- Encrypted database (patients, notes)
- Regular backups
- GDPR compliance

4. User Stories (extracts)

Auth & organizations

- US1.1: Sign up with email/password + 2FA (AC: account created, email verification, OTP enabled)
- US1.2: Create organization + invite members (AC: roles Owner/Admin/Member)

Patient management

- US2.1: Create pseudonymized patient (AC: no identifying data, audit log)
- US2.2: List/search/archive (AC: status filters, anonymization)

Instructions

- US3.1: Write instructions with versioning (AC: diff between versions, restore possible)

AI Chat

- US4.1: Contextual chat per case file (AC: access limited to current patient's data, quick actions, legal disclaimer)

- US4.2: Per-case history, pin a response

Notes & scores

- US5.1: Create session note (AC: full fields, quick templates)
- US5.2: Enter scores (auto-calculation + warning)

PDF exports

- US6.1: Generate summary (AC: anonymized, encrypted storage, download)

Traceability

- US7.1: Safety checklist (side panel, risk checklist)
- US7.2: Complete audit log

5. Data Model (simplified)

Main tables:

- User
- Patient
- CaseFile
- Instructions
- SessionNote
- ChatThread
- Message
- Attachment
- AuditLog

Keys and indexes: FK, index on org_id/patient_id/casefile_id, soft-delete if needed.

6. REST API (examples)

- POST /auth/register
- GET /patients?query=...
- POST /casefiles/{id}/instructions
- POST /casefiles/{id}/threads
- POST /casefiles/{id}/notes
- POST /casefiles/{id}/exports

7. AI Prompts (modes)

System: Clinical supervision assistant, no diagnosis/prescription, risk alerting

Examples:

- CBT formulation in 6 columns + 3 alternative hypotheses
- 45-minute session plan (timed)
- At-home exercises adapted to context
- Targeted Socratic questions
- Risk analysis (suicidal ideation, violence...)
- EMDR preparation + targets

8. GDPR & Security

- Pseudonymization by default
- EU-based hosting
- Encryption at rest/in transit
- DPIA, processing register
- No storage of prompts/responses in plain text
- Auto-masking of sensitive data in exports
- Organization-level purge/deletion + DCP export

9. NFR (Non Functional Requirements)

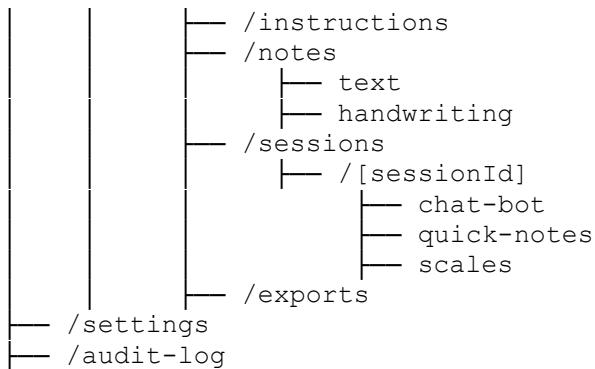
- TTFB < 400 ms (HTML)
- Key actions \leq 3 clicks from patient page
- Availability 99.5%
- Backups: RPO 24h, RTO 4h
- 80% test coverage for critical lines

10. Risks & Mitigation

- **HDS compliance:** MVP with strict pseudonymization, HDS switch later
- **AI costs:** batching + caching + EU models
- **Adoption:** freemium (3 case files), fast onboarding
- **Prompt leakage:** input/output filtering, redaction

11. UI Tree – Interfaces & Navigation

```
/ (Dashboard)
  └── /login
  └── /patients
      └── /[id] (patient record)
```



12. Recommended Tech Stack

Frontend

- Next.js 14 (App Router)
- React 18
- Tailwind CSS
- Shadcn/UI (UI components)
- React Query (TanStack)
- Konva.js / Canvas API (handwriting)
- date-fns

Backend (Next.js API)

- Prisma 5 (ORM)
- PostgreSQL
- NextAuth.js (authentication)
- bcryptjs (password hashing)
- Zod (validation)
- pdf-lib / PDFKit (PDF)
- Azure Computer Vision or Google Cloud Vision (OCR)
- OpenAI / Claude API (AI)

DevOps & Tools

- Vercel (hosting)
- Railway / Supabase / Render (PostgreSQL)
- GitHub Actions (CI/CD)
- ESLint + Prettier (code quality)
- Docker (optional)
- Postman / Bruno (API testing)