Thought for 17s
Nice — here's a practical, ready-to-run **step-by-step execution roadmap** to take *CareLink* from idea → MVP → pilot → production. I'll keep it actionable: sprints, owners, deliverables, privacy controls, costs, and KPIs. You can start building from Day 1.

Roadmap overview (high level)

- Goal: ship an **MVP (demo)** in **~12 weeks** (6 × 2-week sprints) that proves device → app → cloud → doctor → delivery flow.
- Pilot: onboard ~50 users + 5 local doctors + 2 pharmacies in one city for 4 weeks.
- Scale: stabilize, run security/compliance audits, expand partners and region.

Team & roles (recommended)

- Product / PM (1) — owns roadmap, backlog, pilot ops
- Mobile dev (1) — React Native (patient app)
- Backend dev (1) — FastAPI / Django + infra + APIs
- ML engineer (0.5–1) — anomaly models (initially rules + light ML)
- Frontend dev (1) — React doctor dashboard & admin
- DevOps (0.5) — CI/CD, monitoring, infra, backups
- QA / Tester (0.5) — manual + automated tests
- Legal / Compliance consultant (contracted) — privacy & med regs

(If you're building solo, you can wear several hats — cut timeline and costs, but expect longer delivery.)

Tech stack (MVP)

- Mobile: React Native
- Backend: Python + FastAPI
- DB: PostgreSQL (TimescaleDB optional)
- Realtime: Redis / WebSockets / Firebase Messaging
- ML: TensorFlow / PyTorch (start with simple rule-based + lightweight model)
- Hosting: AWS (or Render/Railway for cheap demo)
- Maps: Google Maps / Mapbox
- Notifications: Firebase Cloud Messaging (push)
- Video/Calls: Twilio / Jitsi (optional later)

- Third-party integrations: Fitbit / Google Fit / Apple HealthKit (MVP: support 1 or CSV upload fallback)
- Keys/secrets: AWS KMS or managed secret store

12-week sprint plan (detailed)

Sprint 0 — Prep & discovery (1 week, optional)

Deliverables: requirement backlog, success metrics, contact list of pilot doctors/pharmacies, data privacy checklist, high-level architecture diagram.

Sprint 1 (Weeks 1–2) — Core infra + auth + data model

- Tasks:
    - Setup repo, CI/CD, staging environment.
    - Create DB schema: users, patients, devices, vitals (time-series), alerts, doctors, prescriptions, deliveries.
    - User auth & role-based auth (JWT + refresh).
    - Basic mobile app skeleton: login, device connect UI (mocked).
- Deliverable: deployed backend + mobile skeleton on staging, DB ready.

Sprint 2 (Weeks 3–4) — Wearable integration + ingestion pipeline

- Tasks:
    - Integrate 1 wearable API (Fitbit or Google Fit). Provide fallback CSV upload & manual entry.
    - Build ingestion API that accepts batched vitals.
    - Implement preprocessing (validation, smoothing).
    - Implement initial rule-based anomaly detector (thresholds: HR, SpO_2, temp).
- Deliverable: wearable → mobile → backend pipeline working; sample data stored.

Sprint 3 (Weeks 5–6) — Alerts & doctor flow (core feature)

- Tasks:
    - Implement alert rules + severity scoring.
    - Doctor discovery matching (nearest available doctors; simple geofence).
    - Doctor dashboard (web): list of alerts, patient snapshot, accept/reject action.
    - Push notifications to doctors and patients (FCM).
- Deliverable: end-to-end alert flow: device triggers alert → doctor receives → doctor accepts.

Sprint 4 (Weeks 7–8) — Prescription & delivery MVP

- Tasks:
    - Doctor can create digital prescription in dashboard.
    - Integrate with one local pharmacy (or mock) to create order.
    - Implement dispatch workflow: create delivery request, track status (mock or integrate Rapido/Ola API later).
    - Patient sees order + delivery tracking.
- Deliverable: prescription → pharmacy → delivery tracking (mocked or real for pilot).

Sprint 5 (Weeks 9–10) — UX polish, logging, monitoring & privacy features

- Tasks:
    - Implement consent screens & data-sharing controls in patient app.
    - Encrypt data at rest / use KMS keys.
    - Audit logs for all data access.
    - Add analytics dashboard (response times, alerts count).
    - Basic unit & integration tests.
- Deliverable: privacy flows live, basic monitoring & metrics.

Sprint 6 (Weeks 11–12) — Pilot prep & launch

- Tasks:
    - Fix bugs, harden error handling, finalize onboarding flows.
    - Create pilot onboarding kit (device setup guide, consent form).
    - Run small internal test with 10 users; iterate.
    - Deploy pilot to 50 users + on-call support for doctors.
- Deliverable: pilot launched, tracking KPIs, bug log and timeline for iteration.

Privacy & compliance — concrete checklist

Implement these during Sprints 1–5 (do not postpone):

1. **Minimum data collection** — collect only what you need for alerting (no extra PII in vitals).
2. **Explicit consent** — timestamped consent screen; allow revoke.
3. **Separation & pseudonymization** — store PII in separate encrypted table; use patient_id in analytics.
4. **Encryption** — TLS everywhere + AES-256 at rest; use managed KMS (AWS/GCP).
5. **RBAC** — scope doctor access to only accepted patient cases.

6. **Audit logs** — immutable logs for every access (who/when/what).
7. **Breach response plan** — documented steps and contact list.
8. **Data retention policy** — keep only required timeframe; allow data export/deletion.
9. **Legal review** — one compliance/legal audit before public launch (consult local counsel re: medical device law and India DPDP).
10. **Pen test & vulnerability scan** — before going beyond pilot.

Minimum viable feature set (must-haves for MVP)

- Device data ingestion (at least one wearable or CSV)
- Rule-based anomaly detection + alert
- Doctor notification with basic verify/accept flow
- Digital prescription creation
- Pharmacy order creation (mock acceptable)
- Consent & basic encryption
- Pilot onboarding & monitoring

Cost estimate (MVP) — practical ranges (India, INR)

- Hardware (test devices x5): ₹20k – ₹40k
- Dev (if contracting): ₹1.5L – ₹3.5L (small team) or near zero if you self-develop
- Cloud & APIs (3 months): ₹15k – ₹50k
- Legal/compliance (consultation): ₹20k – ₹80k
- Misc (twilio/fcm/maps): ₹10k – ₹40k
  MVP total (lean, if you self-build): ~₹80k – ₹3L
  MVP total (contracted team): ~₹3L – ₹6L

Pilot plan (4 weeks)

- Recruit 50 test users (offer device + free monitoring).
- Onboard 5 verified local doctors (pay small stipend per accepted case).
- Onboard 2 pharmacies / 3 delivery riders.
- Monitor KPIs daily: alerts, doctor response time, deliveries completed, false positives.

- Run daily standups and weekly retrospective; implement fixes quickly.

KPIs & success criteria

- Alert detection precision > 75% (initial), target 90% after iteration.
- Avg doctor response time < 10 min (pilot target).
- Medicine delivery success rate > 80% within 60 minutes (pilot).
- User satisfaction (NPS) > 6/10 after pilot.
- No critical privacy incidents.

Risk mitigation (concrete)

- **Wrong alert / false positive:** Always require doctor verification before dispatching in-person or ambulance. Use escalation levels.
- **Legal exposure:** Clear T&C + emergency disclaimer + doctor verification. Contract doctors and pharmacies with insurance clauses.
- **Security:** Use managed cloud services and rotate keys; monthly vulnerability scans; keep minimal PII.

Deliverables I can prepare next (pick one)

- a) Sprint-by-sprint Trello / GitHub Projects board (user stories + acceptance criteria)
- b) API spec (OpenAPI) & DB schema for MVP
- c) Example mobile screens & UX flows (consent, alert, tracking)
- d) Simple architecture diagram (png/SVG)
- e) Starter repo skeleton (FastAPI + React Native scaffolding)

Tell me which one you want first and I’ll create it (I can generate the board or API spec next).