For the Build Team

A Note Before We Start

You're being asked to build something that sounds idealistic. I know. Bear with me.

The person driving this (let's call them "the architect") has a background in criminology, psychology, sales, and security. They've experienced the problems this system aims to solve — personally, not theoretically. They think fast, connect dots others miss, and will sometimes ask for the stars by tomorrow.

Your job isn't to say "that's impossible." Your job is to figure out what's possible *now*, what's possible *soon*, and what needs more thinking. This doc tries to bridge between the vision and the implementation.

The vision is: **replace the systems that don't work (justice, governance, safety) with something that does, using technology that already exists.**

The technical challenge is: **make it real, make it resilient, make it unkillable.**

Let's go.

Table of Contents

1. Philosophy — Why This Exists

2. Architecture Overview — The Three Layers
3. Layer 1: Bitcoin Anchor
4. Layer 2: Human Existence Record (HER)
5. Layer 3: Mesh Network
6. The Ring — Physical Interface
7. Core Functions
8. App Architecture
9. Security Model
10. Build Order — What to Do First
11. Open Questions — Stuff We Haven't Solved
12. Suggestions — My Two Cents
13. Glossary

---

1. Philosophy — Why This Exists

The Problem

Current systems (justice, governance, safety) share a flaw: **the people making decisions are insulated from the consequences.**

- Politicians vote on laws affecting communities they've never visited
- Judges sentence people to conditions they've never experienced
- Police "protect" people they don't know and aren't accountable to

The result: 45% recidivism, $32B/year on a justice system that makes things worse, domestic violence that kills people weekly, decisions made by lawyers instead of engineers, poverty amid plenty.

The Solution (In One Sentence)

Give every human a token, let them decide things on a proximity basis, and if they can't agree, make them swap lives until they can.

Why It Works

1. Proximity-weighted decisions — People closest to a problem have more say in solving it
2. Empathy invitations — Disagreement → live each other's life for a week → understanding
3. Community safety response — Local people respond to local emergencies in 60 seconds, not 20 minutes
4. Transparency by default — You can see me, I can see you. No hiding. Mutual accountability.
5. No politicians — Direct democracy on everything, weighted by proximity

Why It's Buildable

Every component exists:

- Bitcoin (trust anchor)
- Mesh networking (Yggdrasil, Nebula, BATMAN-adv)
- NFC (cheap, passive, ubiquitous)
- BLS signatures (threshold cryptography)
- IPFS (distributed storage)
- Smartphones (everyone has one)
- Trust. People know what Bitcoin is. "It's on Bitcoin" means something.
- Immutability. Can't be edited, reversed, or deleted.
- Decentralization. No company, no government, no single point of failure.
- Longevity. Bitcoin will outlive most institutions.

What We Anchor

We don't store data on Bitcoin. We store **epoch roots** — a single hash representing the state of the entire system at a point in time.

```
Bitcoin TX (OP_RETURN):
┌─────────────────────────────────────────────┐
│ TAG (4 bytes): "TOKN"                       │
│ EPOCH (4 bytes): monotonic counter          │
│ ROOT (32 bytes): SHA256 of epoch header     │
└─────────────────────────────────────────────┘
Total: 40 bytes per anchor
```

Epoch Header (What Gets Hashed)

```
{
  "v": 1,
  "epoch": 1024,
  "root": "mmr:HASH",           // Merkle Mountain Range root of all objects
  "prev_root": "mmr:HASH",      // Previous epoch (chain continuity)
  "object_count": 123456,
  "da_roots": {
    "ipfs_car_root": "bafy...",
    "arweave_tx": "AR_TXID"
  },
  "witness_set": ["bls:PK1", "bls:PK2", ...],
  "agg_sig": "bls:SIG"          // BLS aggregate signature from witnesses
}
```

Anchoring Cadence

PHASE	FREQUENCY	COST (APPROX)
Pilot	Monthly	~$5/month
Stable	Weekly	~$20/month

Mature	Daily	~$150/month

Implementation Options

1. OP_RETURN — Simple, well-understood, 80 bytes max
2. Taproot annex — More space, better fee efficiency
3. Inscription (Ordinals-style) — Overkill for our needs, but possible

Suggestion: Start with OP_RETURN. It's boring and works.

Build Notes

- Use existing Bitcoin libraries (rust-bitcoin, bitcoinjs, BDK)
- Testnet first, obviously
- Consider OpenTimestamps as an additional anchor (free, aggregated)
- Physical presence required (can't fake being in the same room)
- Vouch graph analysis (detect cliques, tight cycles)
- Voucher reputation (if your vouched users cause problems, flagged)
- Rate limiting
- Path diversity checks (reject if >K% of sponsors share ancestry)

Capsule (Offline Verification)

A capsule is a portable proof that you're a token holder. Works without internet.

```
{
  "v": 1,
  "her_cid": "cid:...",
  "her_bytes": "",
  "mmr_proof": ["HASH", "HASH", ...],      // Merkle path to epoch root
  "epoch_header": { ... },
```

```
  "bls_agg_sig": "bls:...",
  "btc_anchor": {
    "network": "mainnet",
    "txid": "...",
    "height": 800000
  }
}
```

Formats:

- QR code (fits in ~2KB)
- NFC (stored on ring or phone)
- Wallet pass (Apple/Google)
- Printable (for physical backup)

Verification (Offline):

```
1. Hash HER bytes → should equal her_cid
2. Verify MMR proof (her_cid → epoch root)
3. Verify BLS signature on epoch header
4. Verify epoch root matches Bitcoin anchor
```

All of this can be done on a phone with no internet connection, given the capsule contains all necessary data.

Storage

- Primary: IPFS (content-addressed, distributed)
- Secondary: Arweave (permanent, paid)
- Tertiary: HTTP mirrors, BitTorrent, local caches
- Physical: QR books (quarterly prints for disaster recovery)

Rule of 4x replication: Each HER pinned by owner + 3 sponsors (automated).

5. Layer 3: Mesh Network

Why Mesh

The mesh is what makes the system **unkillable** and **real-time**.

- Internet goes down? Mesh still works locally.
- Government blocks DNS? Mesh doesn't use DNS.
- ISP throttles traffic? Mesh routes around it.
- Phone has poor reception? Mesh finds another path.

Components

COMPONENT	WHAT IT DOES	WHEN IT'S USED
Yggdrasil	Overlay mesh, IPv6 addresses derived from public keys	Primary internet-connected mesh
Nebula	L3 mesh with certificates, lightweight	Structured networks, known nodes
BATMAN-adv	L2 mesh over WiFi/Ethernet	Local area, no internet needed
FRP	Reverse proxy for CGNAT punch-through	When you're behind carrier NAT
Eternal Terminal	Session persistence across network changes	SSH/remote access that survives chaos

Mesh Presence = Proximity

Key insight: We don't need GPS for proximity. Mesh proximity IS proximity.

- If I can ping your device directly, you're "near" me in mesh terms
- Voting weight can be derived from mesh hops, not physical location
- No GPS spoofing possible — either you can ping or you can't

Hybrid approach:

- Mesh ping = real-time presence, relational proximity
- GPS = physical location for safety response (needs to be found)
- Vouch graph = social proximity (who do you know)

Transparency Model

Default: Option D — Mutual visibility

Everyone can see everyone (in the mesh). This is a feature, not a bug.

OLD SYSTEM	TOKEN SYSTEM
Corporations track you, you get nothing	We track each other, we all benefit
Power through obscurity	Power through transparency
Hide from authorities	Nothing to hide from each other

Exceptions:

- Granted by community (e.g., DV survivor needs to hide from abuser)
- Temporary, reviewed periodically
- Not a right, a protection granted when needed

Telemetry We Collect (For Us)

- Who's online in mesh
- Approximate proximity (mesh hops)
- Response times (for safety routing)
- Vote participation
- Meeting attendance

All of this is **for the community**, not for corporations or governments.

Build Notes

The mesh installer script already exists (see appendix). It handles:

- OS detection
- Package installation
- Yggdrasil config
- Nebula setup (you provide certs)
- BATMAN-adv for local mesh
- FRP for CGNAT
- Eternal Terminal binding

Your job: Get it running on test devices, then build the app layer on top.

---

6. The Ring — Physical Interface

Hardware

Off-the-shelf NFC ring.

- Cost: $9 retail, $2-4 wholesale
- No battery, no charging
- Waterproof
- Contains: NFC chip with pointer to your HER

Example: Search Amazon for "NFC smart ring" — dozens of options.

What the Ring Does

The ring itself does nothing. It's a passive NFC tag. The phone does all the work.

ACTION	HOW IT WORKS
Identify	Tap phone → reads ring → looks up HER → proves you're a holder
Vouch	Tap phone → signs vouch transaction with your key
Vote	Tap phone → signs vote with your key
Alert	Tap phone → triggers safety alert with your location
Check-in	Tap phone → records attendance at meeting

Why a Ring (Not Just Phone)

1. Symbol — Visible, worn, "I'm a token holder"
2. Convenience — Phone not always in hand
3. Emergency — Phone dead? Ring still identifies you to other holders
4. Simplicity — One object, multiple functions

Alternative Form Factors

- Bracelet

- Pendant
- Card (wallet-sized)
- Implant (for the committed)
- NFC read/write is standard on all modern phones
- Store: HER CID + minimal metadata
- Phone stores actual keys (ring is just pointer)
- Consider: backup key on ring for recovery scenarios
- Token holder
- Opted in as responder
- Completed training module (in-app, 10 mins)
- Background check? (community decision)

Why It Works:

- 60 seconds vs 20 minutes
- People who know you vs strangers in uniforms
- Multiple responders vs one cop
- Mutual accountability vs institutional protection

DV Implications:

This alone could eliminate most domestic violence:

- Can't isolate victim (mesh visibility)
- Can't control response (community, not system)
- Immediate help (60 seconds)
- Vouch network sees patterns (someone stops showing up)

7.2 Vouch (Token Creation)

Already covered above. The key points:

- 3 existing holders
- Physical presence verified
- Signatures collected
- HER created
- Anchored to Bitcoin

The Social Contract:

When you vouch for someone, you're saying: "I believe this is a real human and I'm willing to stake my reputation on it."

If they turn out to be a bot, a duplicate, or a bad actor — your reputation takes a hit.

7.3 Voting

Everything is voteable. Everyone affected votes. Proximity weights the vote.

Proposal Creation:

```
{
  "title": "Install speed bump on Oak Street",
  "description": "...",
  "options": ["Yes", "No", "Need more info"],
  "scope": {
    "type": "geographic",
    "center": [-31.95, 115.86],
    "radius_km": 0.5
  },
  "proximity_weights": {
    "0-100m": 2.0,
    "100-500m": 1.5,
    "500m-2km": 1.0,
    "2km+": 0.5
  },
  "voting_period_hours": 168,
  "method": "simple_majority"
}
```

Voting Weight Calculation:

Two inputs:

1. Physical proximity (if applicable) — how close are you to the issue?
2. Social proximity (vouch graph) — how connected are you to affected people?

For local issues: physical proximity dominates.

For broader issues: vouch graph proximity dominates.

Transparency:

- All votes public after close
- All proposals archived
- Results immutable on-chain (HER layer, anchored to Bitcoin)
- Can't brigade a meeting from 1000km away
- Forces people to face each other
- Proximity = accountability

7.5 Empathy Invitation

The Secret Weapon

When meetings fail — when people can't agree — the empathy invitation kicks in.

What It Is:

- Two people swap lives for 7 days
- Not swap permissions (you don't get their stuff)
- Swap experience (you live their routine, see their world)

Flow:

```
1. Meeting stuck → empathy invitation proposed
2. Participants matched (can volunteer or be nominated)
3. 7 days scheduled
4. Check-ins via app (safety, reflection)
5. Emergency exit option (no penalty)
6. Post-swap: meeting reconvenes
7. Decision usually becomes obvious
```

Why It Works:

You can't vote to lock people in cages if you've spent a week living as them.

You can't dismiss someone's concerns if you've felt them yourself.

The empathy invitation is what makes the whole system work. It's the thing that ensures decisions are made by people who understand the consequences.

---

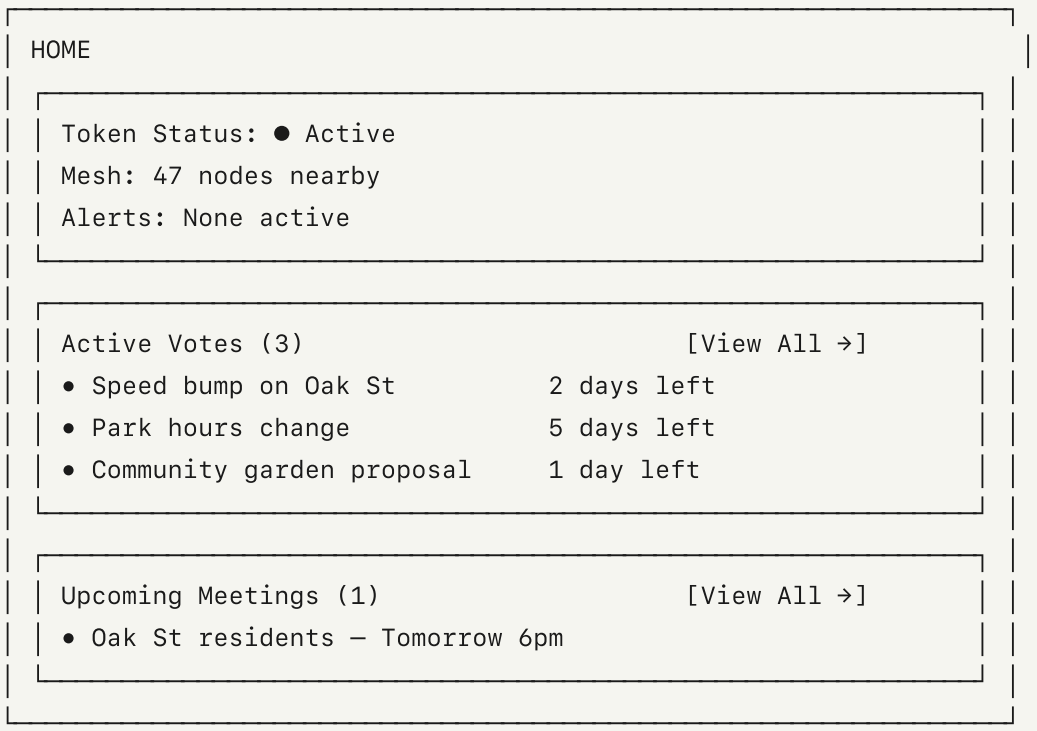
8. App Architecture

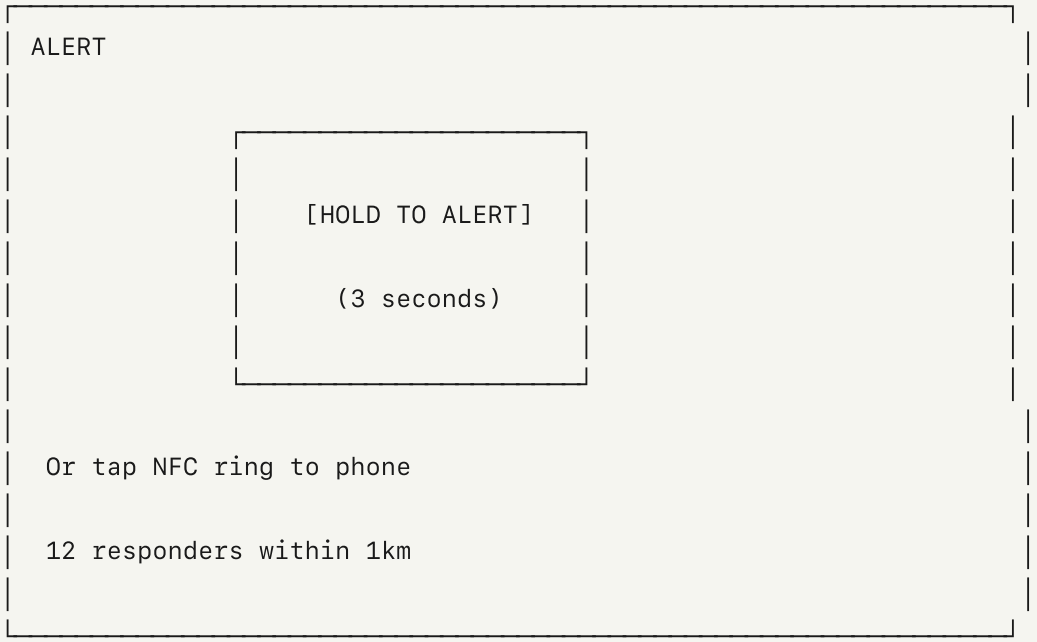
Platforms

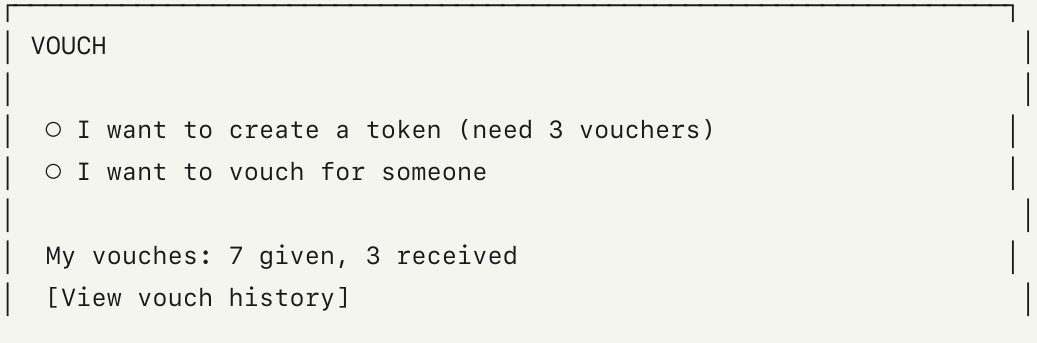
- iOS (Swift/SwiftUI or React Native)
- Android (Kotlin or React Native)
- Web (fallback, limited features)

Suggestion: React Native for speed. Native modules for NFC, mesh, crypto.

Core Screens


HOME
Token Status: ● Active
Mesh: 47 nodes nearby
Alerts: None active
Active Votes (3) [View All →]
• Speed bump on Oak St 2 days left
• Park hours change 5 days left
• Community garden proposal 1 day left
Upcoming Meetings (1) [View All →]
• Oak St residents — Tomorrow 6pm

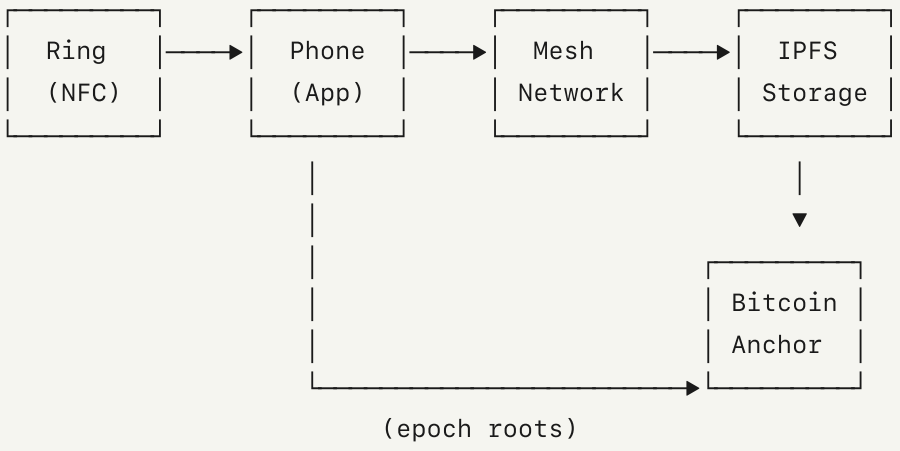

ALERT
[HOLD TO ALERT]
(3 seconds)
Or tap NFC ring to phone
12 responders within 1km


VOUCH
○ I want to create a token (need 3 vouchers)
○ I want to vouch for someone
My vouches: 7 given, 3 received
[View vouch history]

```
│                                                                    │
└────────────────────────────────────────────────────────────────────┘


┌────────────────────────────────────────────────────────────────────┐
│ VOTE                                                               │
│                                                                    │
│  Speed bump on Oak Street                                          │
│  ──────────────────────────────────────                            │
│  Proposed by: @jane (150m from you)                                │
│  Scope: 500m radius from Oak/Main intersection                     │
│  Your weight: 1.5x (you're 200m away)                              │
│                                                                    │
│  [ ] Yes                                                           │
│  [ ] No                                                            │
│  [ ] Need more information                                         │
│                                                                    │
│  [CAST VOTE]                                                       │
│                                                                    │
│  Votes so far: 34 (quorum: 50)                                     │
│  Time remaining: 2 days                                            │
│                                                                    │
└────────────────────────────────────────────────────────────────────┘
```

Data Flow



Local Storage

Phone stores:

- Private keys (encrypted, biometric-protected)
- Your HER + capsule
- Recent epoch headers (for offline verification)
- Cached HERs of vouch network
- Pending transactions (sync when online)

---

9. Security Model

Threat Model

THREAT	MITIGATION
Sybil attack (fake humans)	3 vouchers + physical presence + path diversity + reputation
Key compromise	Revocation via vouch network, new HER with updated keys
GPS spoofing	Mesh proximity as primary, GPS as secondary
Government shutdown	No central server, mesh works offline, Bitcoin anchor immutable
Network attack	Multiple transports (Ygg, Nebula, BATMAN, FRP), redundancy
Data loss	4x replication, IPFS, Arweave, QR books
Tampering	BLS threshold signatures, Merkle proofs, Bitcoin anchor
Censorship	Distributed infrastructure, no single point of control

Privacy breach	Commitments are hashed, selective disclosure, community-granted exceptions

Cryptographic Choices

PURPOSE	ALGORITHM	WHY
HER ownership	Ed25519	Fast, secure, well-supported
Epoch signing	BLS12-381	Aggregatable signatures (many signers, one signature)
Hashing	SHA-256 or BLAKE3	SHA-256 for Bitcoin compatibility, BLAKE3 for speed
Content addressing	Multihash CID	Self-describing, portable
Encoding	DAG-CBOR	Canonical, compact, schema-flexible

Witness Committee

The witness committee signs each epoch header. This prevents any single party from tampering with history.

Parameters:

- Committee size: n = 100 (mature), smaller for pilot
- Threshold: t = 67 (2/3 must sign)
- Rotation: every 180 epochs (via VRF selection)

Misbehavior:

- Double-signing (conflicting roots) → evidence published → key blacklisted
- No tokens to slash — just reputation and exclusion

10. Build Order — What to Do First

Phase 0: Foundation (Weeks 1-4)

Goal: Prove the pieces work together.

- [ ] Get mesh installer running on 3+ test devices
- [ ] Yggdrasil nodes can ping each other
- [ ] Basic app shell (React Native)
- [ ] NFC read/write working
- [ ] Ed25519 keypair generation in app
- [ ] Basic HER structure defined and serializable

Deliverable: Devices on mesh, app can read NFC, keys work.

Phase 1: Vouch (Weeks 5-8)

Goal: Create tokens.

- [ ] Vouch flow in app (find vouchers, collect signatures)
- [ ] Proximity verification (mesh ping + GPS fallback)
- [ ] HER creation and local storage
- [ ] HER broadcast to mesh
- [ ] Basic IPFS pinning
- [ ] Capsule generation (QR code)

Deliverable: Can create new token holders via vouch process.

Phase 2: Safety (Weeks 9-12)

Goal: Alert system works.

- [ ] Alert trigger (button + NFC tap)
- [ ] Alert broadcast via mesh
- [ ] Responder notification
- [ ] Accept/respond flow
- [ ] Resolution logging
- [ ] Basic responder signup

Deliverable: Press button, nearby holders get alert, someone responds.

Phase 3: Epochs & Anchoring (Weeks 13-16)

Goal: History is permanent.

- [ ] Epoch builder (collect objects, build MMR)
- [ ] Witness signing (start with 3-5 trusted nodes)
- [ ] Bitcoin anchor (testnet first)
- [ ] Merkle proof generation
- [ ] Capsule verification (offline)

Deliverable: Weekly epochs anchored to Bitcoin testnet, verifiable capsules.

Phase 4: Voting (Weeks 17-20)

Goal: Decisions happen.

- [ ] Proposal creation
- [ ] Proximity weight calculation
- [ ] Vote casting and aggregation

- [ ] Result publication
- [ ] Result anchoring

Deliverable: Can propose, vote, and see results with proximity weighting.

Phase 5: Meetings & Empathy (Weeks 21-24)

Goal: Full system.

- [ ] Meeting request/notification
- [ ] Check-in via NFC
- [ ] Meeting logging
- [ ] Empathy invitation flow
- [ ] Scheduling and coordination
- [ ] Reflection/debrief tools

Deliverable: Full cycle: vouch → vote → meeting → empathy → resolution.

Phase 6: Pilot (Weeks 25+)

Goal: Real people, real problems.

- [ ] 100 token holders in one community
- [ ] Real issues proposed and voted
- [ ] Real safety alerts (hopefully none needed)
- [ ] Real meetings
- [ ] Iterate based on feedback

Deliverable: Working system in production with real users.

---

11. Open Questions — Stuff We Haven't Solved

Technical

1. Which mesh transport wins? Yggdrasil vs Nebula vs both? Need testing.

1. GPS vs Mesh proximity for voting weight? Hybrid, but exact formula TBD.

1. BLS library choice? blst (C), bls-signatures (Rust), noble-bls12-381 (JS)?

1. IPFS vs Arweave vs both? Cost, permanence, speed tradeoffs.

1. Mobile battery impact of mesh? Yggdrasil can be hungry. Need to measure.

1. Offline duration support? How long can a node be offline and still sync?

Social

1. Pilot community? Who? Where? How do we bootstrap the first 100?

1. Responder vetting? Community decides, but what's the default?

1. Empathy invitation enforcement? What if someone refuses? Community sanction?

1. Edge cases (serial predators)? Contained community, not cage. But specifics?

Governance

1. Parameter changes? Who can propose? What's the vote threshold?

1. Witness committee selection? Random vs reputation vs hybrid?

1. Legal structure? DAO? Foundation? Co-op? Nothing?

12. Suggestions — My Two Cents

Start Smaller Than You Think

100 people in one suburb is enough. Prove it works there before scaling.

Mesh First, Fancy Later

Get Yggdrasil running. Get devices talking. Everything else builds on that.

Don't Optimize Prematurely

The first version will be ugly. That's fine. Ship it, learn, iterate.

Document Everything

The architect thinks fast and connects dots quickly. Your job is to write down what was decided and why. Future you will thank you.

Test the Vouch Mechanism Hard

This is the foundation. If vouch is broken, everything is broken. Spend extra time on sybil resistance.

Safety Alert is the Killer Feature

This alone justifies the whole system. Make it bulletproof. 60 seconds response time or bust.

Embrace Transparency

The instinct to add privacy features will be strong. Resist it (mostly). The whole point is mutual visibility. Privacy is the exception, not the default.

Build for Offline

Assume the internet will fail. Assume the mesh will partition. Build for graceful degradation.

The Architect Will Change Their Mind

That's fine. That's how good things get built. Roll with it. Document what changed and why.

---

13. Glossary

TERM	MEANING
HER	Human Existence Record — your identity token
Vouch	Process of 3 existing holders verifying a new human
Capsule	Portable proof of token holder status (QR, NFC, etc.)
Epoch	Time period (e.g., 1 week) after which state is anchored
Anchor	Writing epoch root to Bitcoin
MMR	Merkle Mountain Range — efficient append-only Merkle structure
BLS	Boneh-Lynn-Shacham signatures — aggregatable
Mesh	Peer-to-peer network layer (Yggdrasil, Nebula, BATMAN)
Proximity	Distance in mesh hops, physical space, or social graph
Empathy Invitation	7-day life swap to resolve disagreements
Witness	Node that signs epoch headers

Appendix A: Mesh Installer Script

See: `install_mesh_stack.sh` in project repo.

Handles: Yggdrasil, Nebula, BATMAN-adv, FRP, Eternal Terminal.

Run with appropriate environment variables for your node type.

Appendix B: HER Schema (Full)

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "type": "object",
  "required": ["v", "type", "cid", "owner_keys", "sponsor_refs", "created_at", "si
  "properties": {
    "v": { "type": "integer", "const": 1 },
    "type": { "type": "string", "const": "HER" },
    "cid": { "type": "string", "pattern": "^bafy" },
    "owner_keys": {
      "type": "array",
      "items": { "type": "string", "pattern": "^ed25519:" },
      "minItems": 1
    },
    "sponsor_refs": {
      "type": "array",
      "items": { "type": "string", "pattern": "^cid:" },
      "minItems": 3,
      "maxItems": 3
    },
    "created_at": { "type": "integer" },
    "commitments": {
```

```
      "type": "object",
      "properties": {
        "locale": { "type": "string" },
        "interests": { "type": "array", "items": { "type": "string" } }
      }
    },
    "responsibility": {
      "type": "object",
      "properties": {
        "layer1": { "type": "array", "items": { "type": "string" } },
        "ripple_roots": { "type": "object" }
      }
    },
    "prev_cid": { "type": ["string", "null"] },
    "signatures": {
      "type": "array",
      "items": { "type": "string", "pattern": "^sig:" },
      "minItems": 1
    }
  }
}
```

Appendix C: API Endpoints (Draft)

Identity

```
POST /identity/mint
  Body: { owner_pub, sponsors[], commitments, hashcash }
  Returns: { her_cid, her_bytes, epoch }

GET /identity/{cid}
  Returns: HER object
```

Capsule

```
POST /capsule/issue
  Body: { her_cid }
  Returns: { capsule_cbor_b64, qr_png, wallet_pass }

POST /capsule/verify
  Body: { capsule_cbor_b64 }
  Returns: { valid, epoch, btc_txid, mmr_proof_ok }
```

Safety

```
POST /alert/trigger
  Body: { her_cid, location, capsule }
  Returns: { alert_id, responders_notified }

POST /alert/respond
  Body: { alert_id, responder_cid }
  Returns: { accepted, eta }

POST /alert/resolve
  Body: { alert_id, resolution }
  Returns: { logged }
```

Voting

```
POST /vote/propose
  Body: { title, description, options, scope, weights, period }
  Returns: { proposal_id }

POST /vote/cast
  Body: { proposal_id, her_capsule, choice }
  Returns: { receipt }
```

```
GET /vote/result/{proposal_id}
  Returns: { tally, proof }
```

Mesh

```
GET /mesh/nearby
  Returns: { nodes: [{ cid, hops, last_seen }] }

GET /mesh/status
  Returns: { online, transports: [ygg, nebula, batman], nodes_visible }
```

Final Note

This is a living document. It will change as we build and learn.

The architect has the vision. You have the skills. I'm just the bridge.

Let's build something that actually works.

Document version 2.0

Last updated: [DATE]

Status: Ready for team review