nzinga

NZINGA NETWORK · PROOF OF CONCEPT

Proof of concept plan: product, data and business.

How Nzinga collects consented voice data in African languages, pays the people who record it, turns it into datasets and models, and licenses it, with the decisions that need to be made before the pilot.

Draft 0.1, 5 November 2026. Builds on `SPEC.md` (product and technical specification, draft 0.1) and on what the code already enforces in `packages/core`. Everything marked **Decision** needs the owner's call; everything marked **Proposal** is a recommendation that can change. Nothing here states a market price or a legal conclusion as fact.

0. The Proof of Concept in One Paragraph

Within twelve weeks, show that Nzinga can (1) collect consented, paid voice recordings in one African language from about 100 invited members, (2) clean, review and structure them into a versioned dataset with a consent record per item, (3) measurably improve a speech model with that dataset, and (4) license it to one design-partner buyer under a pilot licence, with members paid through one real payout rail. If all four happen with the consent and privacy rules below intact, the model works end to end.

Success criteria (Proposal)

GOAL	MEASURE	TARGET FOR THE PILOT
Collection works	Accepted hours, one language	Set with the design partner (for example 50 to 100 hours)
Members stay	Members recording in week 4	At least half of those who started

GOAL	MEASURE	TARGET FOR THE PILOT
Consent holds	Recordings with a valid consent record per speaker	100%
Quality holds	Share of submissions accepted after review	Tracked; target agreed after week 2
Data is useful	Word error rate of the fine-tuned model vs the base model, on a held-out test set from speakers not in training	A measured drop, reported whatever it is
Money flows	Members paid within the promised window	100% of accepted, non-fraud contributions
Someone pays	Signed pilot licence with one buyer	1

1. Who Our Customers Are

Nzinga is a three-sided system. Each side needs a different promise.

SIDE	WHO	WHAT THEY GET	WHAT THEY GIVE
Members (supply)	Adults who speak African languages, invited by someone they know	Pay per accepted minute, a share of licence income, useful tools (notes, transcripts), control over every recording	Recordings, reviews, invites
Data buyers (demand)	AI labs and speech-technology companies building African-language models; telecoms, banks and fintechs building voice services; research groups; public-interest projects	Clean, consented, documented speech data in languages that are hard to get, with a licence that holds up to due diligence	Licence fees, custom collection budgets
Organisations (teams)	Companies whose employees and customers speak African languages: contact centres, banks, field-service and health programmes	A way to build speech tools for their own languages, using their own teams, with consent handled properly	Seats, collection projects, licence fees
Nzinga itself (internal)	The Nzinga tokenizer and models	Training data under the "Nzinga's models" choice	Better models, which lift the value of everything else

Proposal for the pilot: one design-partner buyer (a speech or language AI company that already needs the pilot language) and, if possible, one organisation pilot (a team of employees contributing under section 12). The buyer co-designs the dataset spec and the licence before collection starts, so the first dataset has a customer on day one.

Decision: which language first (pick by buyer demand and member reach; the tokenizer work suggests Swahili, Hausa or Yoruba) and which design partner.

---

2. What We Collect

Proposal: voice only for the pilot. Voice notes against prompts (built today) and, once the native build exists, member-to-member calls with one track per speaker (`SPEC.md` 3.3). No photos, no video, no free text from contacts. Photos and video multiply the consent and privacy problem (bystanders, faces, locations) and are not needed to prove the model; see section 6 for how they would work later.

Per recording we keep:

FIELD	SOURCE	NOTES
Audio	Phone microphone	Opus on the phone; converted to FLAC 16 kHz mono for datasets
Speaker	Member id	Replaced by a pseudonymous speaker id in every dataset (section 5)
Language, dialect	Member profile + prompt	Required to contribute; checked by language identification
Prompt	Prompt library	Prompts are written to avoid personal information
Duration, device class, recording conditions	Phone	Coarse only (no device ids)
Consent records	Consent flow	One per speaker, append-only (built: `packages/core/src/consent.ts`)
Profile bands (optional)	Member, with the reason shown	Region (broad), age band, gender. Never ethnicity, health, religion, politics

---

3. Where and How We Store It

Today: audio stays on the phone. The API (Cloudflare Workers, `apps/api`) stores members, invites, recording metadata, consent records and the ledger in a single Durable Object snapshot. That is right for a prototype and wrong for real members.

Proposal for the pilot

DATA	STORE	PROTECTION
Raw audio uploads	S3-compatible object storage (Cloudflare R2 is the shortest path from today's stack) in a `raw/` area	Encrypted at rest; per-member data keys wrapped by a master key (envelope encryption), so deleting a member's key makes their raw audio unreadable
Accounts, invites, recordings metadata, consent records, ledger	A relational database (Cloudflare D1 or Postgres) replacing the Durable Object snapshot	Consent records and ledger entries are append-only tables; no updates or deletes, only new rows
Cleaned and reviewed audio	`clean/` area, separate keys	Only the pipeline writes here
Released datasets	Private Hugging Face dataset repos (as `SPEC.md` 6 says), one per release, plus a copy in our storage	Pseudonymous; licence and consent snapshot attached
Contacts	Not stored	Hashes are compared and discarded (built: `packages/core/src/contacts.ts`)

Data lifecycle (Proposal): raw audio is kept until review plus a short appeal window, then only the cleaned version is kept. "Only me" recordings stay on the phone and are never uploaded unless the member turns on backup.

Sovereignty path: the pilot can run on global cloud regions, but the plan should name the route to African hosting for raw and cleaned audio (an African cloud region or a local partner) before public launch. `SOVEREIGNTY.md` in the tokenizer repo already tracks "nothing is hosted in Africa yet" as a gap; this closes it for the voice library.

Decision: pilot storage region, and whether raw audio may leave the continent at all during the pilot.

4. How We Structure It

Each dataset release is a versioned, documented package.

```
nzinga-voice-<lang>-v0.1/
  audio/<speaker_pid>/<item_id>.flac
  metadata.parquet          one row per item
  speakers.parquet          one row per pseudonymous speaker (bands only)
  consent.parquet           consent snapshot per item at release time (choice, text version
  splits/train.txt dev.txt test.txt   split by speaker, never by item
  DATASHEET.md              how it was collected, who is in it, what it may be used for, kn
  LICENSE.md                the licence this release is issued under
  CHANGELOG.md              items added and withdrawn since the last release
```

`metadata.parquet` columns (Proposal): `item_id, speaker_pid, language (ISO 639-3), dialect, prompt_id, prompt_text, duration_s, sample_rate, transcript, transcript_status (draft|reviewed), review_score, snr_db, consent_choice, consent_text_version, release, licence_scope`.

Rules: test sets use speakers who appear in no training split; every item carries the consent choice that was valid on the release date; a withdrawn item appears in the next release's changelog as removed.

5. How We Anonymise It (and What We Cannot Promise)

A voice can identify a person. Speech data can be pseudonymised and stripped of personal details, but not made fully anonymous. We say so to members and buyers, and design around it.

MEASURE	WHAT IT DOES
Pseudonymous speaker ids	Members are `spk_…` ids in datasets. A different id per buyer (Proposal), so two buyers cannot join their copies
No direct identifiers	No names, phone numbers, device ids, precise location or contact data in any dataset

MEASURE	WHAT IT DOES
Prompts that avoid personal details	Prompts ask about recipes, markets, weather, stories, not about the speaker's address, health or finances
Redaction	Transcripts are scanned for names, numbers and places; matching audio spans are silenced and transcript spans masked before release
Bands, not values	Age band, broad region; rare combinations are merged or dropped so no small group can be singled out
Licence limits	Licensees may not identify speakers, clone voices or build speaker-recognition systems from the data (section 9)
Open release	"Everyone" recordings get the strictest redaction, and members are told plainly that open data cannot be recalled

Decision: whether voice cloning or text-to-speech training is allowed at all. Proposal: not under any of the four current choices; if wanted later, it becomes a separate, explicit opt-in with its own pay.

---

6. Consent: Both Sides, Every Time

Built today (`packages/core/src/consent.ts`): every person whose voice is in a recording has their own record; four choices from narrowest to widest ("Only me", "Nzinga's models", "Nzinga and approved partners", "Everyone"); records are append-only; each records the exact consent text by version and SHA-256; a whole conversation may be used only if every speaker's latest choice allows it; withdrawal adds a record and stops all future use.

Calls (Proposal, needs the native build):

1. Recording is off by default. Either person can ask to record; both must accept on their own phone before any audio is kept.
2. Each phone records only its own member's microphone. A person who declines has no track, and the conversation as a whole is never used.
3. A tone and an on-screen banner run for the whole recorded part; either person can stop it, and stopping ends recording for both.
4. After the call, each person chooses who can use their own track. The narrower choice governs the conversation.
5. Calls to people who are not members are never recorded.

Photos and video (future, not in the pilot): every identifiable person must be a member who consents in the app; faces of non-members and bystanders are blurred on the phone before upload; no children; no location metadata. Until those controls exist, the app does not collect images.

Other rules (Proposal): adults only (age declared at setup, checked if fraud signals appear); re-consent when the consent text changes (the version and hash make this enforceable); withdrawal reaches licensees by contract (they delete withdrawn items within a set period after each changelog); members can export and delete their data from the app (export exists; account deletion needs an API route).

---

7. How We Clean It

```
upload → technical checks → language check → duplicate and fraud checks → draft transcript
       → native-speaker review → redaction → segmentation → score → accept or reject → pay
```

STAGE	TOOLING (FROM `SPEC.MD` 6 WHERE NAMED)	REJECTS
Technical checks	ffmpeg; signal-to-noise, clipping, silence ratio	Too short, silent, clipped, noisy
Language check	Language identification model	Wrong language or dialect
Duplicates and fraud	Audio fingerprinting; speaker consistency per member; synthetic or replayed audio detection	Re-submissions, someone else's voice, generated audio
Draft transcript	Best open speech model per language (Omnilingual ASR, Whisper variants)	(none; drafts help reviewers)
Review	Paid native-speaker reviewers from trusted members, in Argilla or Label Studio	Off-prompt, unclear, personal information, consent problems
Redaction	Named-entity detection on transcripts, mapped to audio spans	(masks, does not reject)
Score and accept	Combined score and reviewer decision	Below threshold

Reviewers are paid per item reviewed. Every rejection gives the member a reason (the app already shows a reason line). Payment happens only after acceptance, so fraud does not cost money.

8. How We Use It to Train Models

USE	DATA	CONSENT NEEDED	STATUS
Tokenizer	Reviewed transcripts (text)	"Nzinga's models" or wider	The tokenizer exists; transcripts would add spoken-language text
Speech recognition	Audio + reviewed transcripts	"Nzinga's models" or wider	Pilot: fine-tune an open model (Whisper or Omnilingual ASR) and report word error rate before and after
Language model	Transcripts, with the Nzinga tokenizer	"Nzinga's models" or wider	The experiment in the tokenizer repo is built and waiting on compute billing
Licensed training by partners	Released datasets	"Nzinga and approved partners" or wider	Pilot licence with the design partner

Withdrawal and models: each model version records which dataset release trained it. Withdrawn items are excluded from the next release and the next training run; already-trained models cannot forget them, and the app says so. **Evaluation:** held-out test speakers from our own data, plus public sets where they exist (FLORES+ for text, Belebele for reading comprehension, as in the tokenizer repo).

9. How We Price It and License It

Licence products (Proposal):

PRODUCT	WHAT THE BUYER GETS	PRICING BASIS
Dataset licence	A released dataset, a language at a time	Per accepted hour, tiered by transcription level and language scarcity; research and commercial rates

PRODUCT	WHAT THE BUYER GETS	PRICING BASIS
Collection project	A custom campaign (language, dialects, topics, speakers) run on the network	Project fee covering member pay, review and a margin; buyer gets a licence to the result
Subscription	Each new release of a language for a year	Annual fee
Model access (later)	Nzinga speech or language models by API	Usage-based

Licence terms (Proposal): non-exclusive always (the community keeps the right to license again); no speaker identification, voice cloning or surveillance uses; no resale of raw data; delete withdrawn items after each changelog; attribution to the Nzinga Network; audit right for consent records.

Price levels: Decision. Benchmark against what buyers pay for comparable transcribed speech in low-resource languages, through the design-partner conversation, before publishing anything. The site and app keep saying "rate set at launch" until then.

Where licence money goes (Decision; Proposal shown):

SHARE	GOES TO
Member pool	Paid to members as revenue-share units, in proportion to their accepted seconds in the licensed release (the ledger already tracks units)
Community fund	Language-community projects, governed under the CARE principles (`SPEC.md` 4)
Operations	Review, storage, payouts, compliance
Nzinga	Margin for the network to last

10. How We Pay Members

Built today: an append-only ledger in integer minor units per currency (contributions, referral bonuses, adjustments, payouts) and revenue-share units tracked separately; development rates are placeholders. The app keeps payout methods and requests on the phone only (`apps/mobile/lib/payouts.ts`).

Proposal for the pilot:

PART	CHOICE
When money is earned	At acceptance, never at upload
Rate	Set per language so that a member's effective hourly pay, including rejected attempts, is fair for the place they live; reviewed after week 2
Rail	One payout aggregator that covers mobile money and bank transfer in the pilot market; the provider interface in `SPEC.md` 3.6 keeps it swappable
Minimum payout	As in the ledger policy (placeholder today)
Identity	Phone number verified at sign-in; name on the payout account must match the profile name; extra checks only above a threshold
Fraud holds	Payouts paused for an invite tree under investigation (`packages/core/src/invites.ts` already penalises inviters)
Records	Every payout is a ledger entry with the provider reference; statements are already downloadable in the app
Funding	Pilot pay is funded from the pilot budget before licence income exists; this is a cost line in the budget

Decision: the pilot market and payout provider; who holds the funds; tax and reporting obligations for paying members in that market (legal review).

11. Invites and the Social Graph

Built today (`packages/core/src/invites.ts`): invite-only; three invites to start; one more per accepted hour; at most ten unused; referral pay to the direct inviter only (never up a chain, so it cannot become a multi-level scheme); fraud by an invitee removes three invites from the inviter. Codes are easy to read aloud.

How the graph is used (Proposal): fraud detection (clusters of bad accounts share an inviter), trust (reviewers are drawn from members with long good records), and conversation partners for calls. **How it is never used:** it is not sold, not shown to other members beyond "friends on Nzinga", and not included in any dataset.

Communities (Proposal): a group that joins together (a village association, a church choir, a university department) can hold a shared agreement and share in the community fund.

12. Organisations and Teams ("Nzinga for Teams")

Companies can use Nzinga with their employees in two ways.

MODE	WHAT IT IS	DATA GOES TO
Contribute	The organisation runs a collection campaign with its staff (for example a contact centre gathering customer-service speech in its languages)	The Nzinga library, under each person's own choice; the organisation gets a licence to the result
Private	The organisation uses the app to build data only for itself	A private organisation dataset, not the library

Rules that protect employees (Proposal): consent stays individual (an employer cannot consent for anyone); taking part is voluntary and must not affect anyone's job; pay goes to the person, not the employer, unless the person is recording during paid work time under an agreed policy; the employer never sees an individual's recordings unless that person shares them; employees can leave and withdraw like any member.

What organisations get (Proposal): an admin console (seats, team invite codes, campaign progress, aggregate quality), single sign-on later, a data processing agreement, and the same datasheet and consent snapshot as any buyer. **Pricing basis:** per seat or per campaign, plus the licence. **Decision:** whether Teams is in the pilot or the next phase.

13. How the App Works With the Phone

CAPABILITY	TODAY	PILOT
Microphone	Expo Audio recording, voice notes	Same, plus background-safe upload queue that waits for Wi-Fi if the member chooses

CAPABILITY	TODAY	PILOT
Contacts	Optional; hashed numbers, matches only	Same
Calls	Not built	Native build (not Expo Go): CallKit on iOS, Telecom/ConnectionService on Android, WebRTC through a self-hostable server, one track per phone (`SPEC.md` 3.3)
Notifications	Settings only	Push for review results and payouts
Storage on the phone	Recordings in the app's private storage	Encrypt files at rest with a key in the phone's secure store; "Only me" recordings never leave the phone unless backed up
Data and battery	Not measured	Opus at low bitrates, uploads on Wi-Fi by default, a monthly data budget the member can see
Distribution	Expo Go preview, web demo, Android APK	Android first for the pilot (`SPEC.md` 7), TestFlight for iOS testers

14. Compliance

Jurisdiction-neutral by design (`SPEC.md` 2): one strict common standard, with country modules for payouts and legal text. Before any external member records (Proposal):

1. Legal review in the pilot country: data protection and cross-border transfer, call-recording consent, payments and money transmission (through the provider), tax on member pay, contributor status (members are not employees), and referral rules.
2. A data protection impact assessment and a record of processing.
3. A plain-language privacy notice in the app (the Privacy screen and site summary are drafts, not legal text).
4. A licence agreement and data processing agreement template for buyers.
5. Retention schedule, breach response plan, and a named data protection contact.
6. An independent security review before public launch (`SPEC.md` 9).

15. Sovereignty

Sovereignty is measured with the **Sovereignty Scorecard** (`SOVEREIGNTY.md` in the tokenizer repo, generated by `scripts/sovereignty.py`): five layers, each indicator with a Today value and a Target, kept separate rather than combined into one score. The pilot extends it to the voice network with the same format (Proposal until adopted):

LAYER	INDICATOR (NETWORK)	TODAY	TARGET
Control of the data	Recordings with a per-speaker consent record	All (enforced in `packages/core/src/consent.ts`)	All
	Licences that are non-exclusive	No licences yet	All
	Members able to withdraw any recording	Built	Always
Infrastructure	Raw and cleaned audio hosted in Africa	Not applicable: audio stays on the phone	Hosted in Africa before public launch
Governance	Named body governing the library and its licences	None	Trust, cooperative or foundation (**Decision**)
People and benefit	Share of licence income to members and a community fund	Decision pending	A written split, published
	Native-speaker reviewers paid by the network	0	Every pilot language

How pilot decisions move the scorecard: storage region (section 3) moves Infrastructure; licence terms (section 9) and the income split move Control of the data and People and benefit; the governance body moves Governance; reviewer hiring (section 7) moves People. Each decision in section 18 names the rows it changes.

16. AI Infrastructure and Integrations

NEED	PILOT CHOICE	ALREADY IN USE
API	Cloudflare Workers	Yes (`apps/api`)

NEED	PILOT CHOICE	ALREADY IN USE
Database	D1 or Postgres	No (Durable Object snapshot today)
Object storage	R2 (S3-compatible)	No
Review	Argilla or Label Studio	No
Speech models	Whisper variants, Omnilingual ASR	No
Training compute	Hugging Face Jobs or Modal (scripts exist in the tokenizer repo; billing is the blocker)	Scripts only
Dataset hosting	Private Hugging Face repos	Tokenizer release only
Sign-in codes	An SMS provider per market	Development sender only
Payouts	One aggregator	No
Calls	LiveKit (self-hostable)	No
App delivery	Expo / EAS (Update, Build)	Yes

17. Pilot Plan (Twelve Weeks, Proposal)

WEEKS	BUILD	PROVE
0 to 2	Database and object storage; encrypted upload from the app; consent records server-side; account deletion; legal review starts; design partner and dataset spec agreed	A recording goes from phone to storage with its consent record
2 to 6	Cleaning pipeline v1 (technical, language, duplicate checks, draft transcripts); reviewer tool; payout provider integrated in test mode	20 internal testers record and get reviewed
6 to 10	Invite 100 members in the pilot language; prompts campaign; real payouts	Accepted hours, retention, acceptance rate, payout times
10 to 12	Dataset v0.1 with datasheet; fine-tune a speech model; pilot licence signed	Word error rate change; first licence

Budget drivers (Decision): member pay (hours × rate), reviewer pay, payout fees, storage, compute for one fine-tune, legal review, security review.

---

18. Decisions for the Owner, in Order

1. The pilot language and the design-partner buyer (moves Equal performance).
2. The pilot market, payout provider, and whether raw audio may leave the continent during the pilot (moves Infrastructure).
3. Member rate per accepted minute and the split of licence income (member pool, community fund, operations, Nzinga) (moves People and benefit).
4. Licence terms: non-exclusive (proposed), banned uses, and whether voice cloning is ever allowed (moves Control of the data).
5. Whether Nzinga for Teams is in the pilot.
6. Governance body for the library and the community fund (moves Governance).
7. When calls (native build) enter scope, given they are the heaviest consent and engineering work.

Nzinga Network · HQ Protocol · Draft for discussion · Generated 05 November 2026 from docs/POC-PLAN.md