

Personal Brand Website — Engineering Spec

BlitzMetrics \$1/Day Platform — built on WordPress, engineered for Googleability

Goal: turn raw proof (videos, podcasts, reviews, talks) into structured, machine-readable authority that ranks, earns a Knowledge Panel, and drives leads. WordPress is the rendering + editorial layer; our services/plugin-ins do ingestion, structuring, schema, and distribution.

0) Design Principles

- **Entity-first:** everything ties to a Person and/or Organization.
 - **Opinionated structure** over visual templates.
 - **Source of truth** lives in WP (posts + taxonomies + postmeta) with immutable IDs for assets.
 - **Automate the boring:** ingestion, transcripts, schema, sitemaps, internal links, social posts.
 - **Measure or it didn't happen:** GA4, GSC, GBP, YouTube Analytics pulled into a unified KPI layer.
 - **SEO-safe by default:** canonical rules, noindex traps guarded, schema linted.
-

1) System Architecture (high-level)

Core: Managed WordPress (PHP 8.2+, MariaDB/Aurora MySQL), NGINX/HTTP3, PHP-FPM.

Custom plugin suite (monorepo): `blitz-pb-engine` (main), with modules: - `ingest/` connectors (YouTube, Apple/Spotify, Instagram, LinkedIn, GBP, Google Sheets) - `transcribe/` (Whisper or Amazon Transcribe adapter) - `repurpose/` (video→article pipeline, snippet generator) - `schema/` (JSON-LD builders, validator) - `linker/` (internal linking graph) - `sitemaps/` (video + news + standard) - `kpis/` (GA4, GSC, GBP, YT pulls + dashboards) - `api/` (REST API + WP-CLI) - `jobs/` (queues + retries)

Infra: - Storage: AWS S3 for media + transcripts, CloudFront CDN; image transforms via `imgproxy` or WP Image Editor. - Queue/cron: WP-Cron + server cron; optional SQS for bursty ingestion. - Cache: FastCGI + Redis object cache; page CDN cache (CloudFront/Cloudflare). - Logs/obs: CloudWatch + WP debug logs, HTTP access logs; optional OpenTelemetry.

Optional headless: WPGraphQL → Next.js/Remix front-end. Default is server-rendered theme hardened for CWV.

2) WordPress Data Model

Custom Post Types (CPTs) - `person` — the human brand entity (1 primary required) - `organization` — company/brand entity - `asset_video` — YouTube/native video - `asset_podcast` — Apple/Spotify episode - `asset_article` — long-form post auto-generated from video/podcast - `testimonial` —

review/UGC (text or video) - `offer` — productized service, package, price anchor - `topic_hub` — Topic Wheel nodes (5–7 recommended) - `event_speaking` — talks, panels, media appearances

Taxonomies - `topic` (hierarchical) — matches `topic_hub` pages (term $\approx$ entity topic) - `persona` — ICP tags (e.g., “Wisconsin homeowner”, “Roofing GM”) - `geo` — city/region/state (used for GBP + local SEO) - `platform` — source systems (YouTube, IG, LinkedIn, Apple, Spotify)

Key Postmeta (selected) - `entity.person_id` / `entity.org_id` — foreign keys - `source.url`, `source.platform_id`, `source.published_at` - `transcript.s3_uri`, `transcript.word_count` - `repurpose.status` (queued|draft|needs_review|published) - `schema.type` (VideoObject, PodcastEpisode, Review, FAQ, Event) - `kpi.impressions`, `kpi.ctr`, `kpi.entity_coverage`

Relationship conventions - Asset $\leftrightarrow$ Topic: via `topic` taxonomy (+ score 0–1 relevance) - Asset $\leftrightarrow$ Offer: postmeta `offer_ids` (array) - Testimonial $\leftrightarrow$ Offer (required); Testimonial $\leftrightarrow$ Persona (optional) - Person $\leftrightarrow$ Organization: `primary_org_id`

Rule: No orphan assets — every `asset_*` must map to ≥ 1 `topic` OR `offer`.

3) Ingestion Connectors

YouTube - Method: YouTube Data API v3 pull by Channel ID; webhook via PubSubHubbub if enabled. - Fields: `videoId`, `title`, `description`, `tags`, `publishedAt`, `thumbnails`, `duration`. - Mapping: create `asset_video`; set `source.*`; attach `platform:youtube`. - Thumbs: fetch maxresdefault; upload to S3; store `thumb.s3_uri`.

Podcasts (Apple/Spotify) - Method: RSS polling + Spotify API enrichment. - Fields: `guid`, `title`, `summary`, `audio URL`, `pubDate`, `duration`. - Mapping: `asset_podcast` with audio enclosures $\rightarrow$ S3 proxy (optional).

Instagram/LinkedIn - Method: Graph API (user auth) for media + captions; LinkedIn shares via REST. - Mapping: choice: import as `asset_video` (if video) or `asset_article` (short form). Canonical remains source network; we `rel=canonical` if mirrored.

GBP (Google Business Profile) - Method: GBP API for reviews + posts. - Mapping: reviews $\rightarrow$ `testimonial` (with rating, reviewer initials), GBP posts $\rightarrow$ `asset_article` (short updates), geo tags attached.

Google Sheets bulk upload - Method: Sheet with columns [Title, URL, Type, Topic, Offer, Date, Notes]. - Mapping: creates CPTs; idempotent by URL hash.

Validation & Idempotency - Source URL hash unique index $\rightarrow$ prevents dupes. - Retries with exponential backoff; poison queue for persistent failures.

4) Transcription & Enrichment Pipeline

- **Transcribe:** Whisper (local) or Amazon Transcribe (batch) → SRT/VTT + JSON.
- **Chunking:** semantic chunking (500–1000 tokens) with timestamps.
- **Summaries:** TL;DR, key quotes, entities (People, Orgs, Places, Products) → stored as postmeta.
- **Highlights extraction:** candidate H2s; FAQ pairs (Q/A) scored.
- **Compliance:** optional profanity filter and PII redaction.

Storage - Transcripts in S3 (`transcripts/{postID}.json`), short excerpt in postmeta.

5) Repurposing: Video → Article Generator

State machine (`repurpose.status`) 1. `queued`: new asset eligible 2. `draft`: AI draft created 3. `needs_review`: editor pass required 4. `published`: converted to `asset_article` + indexed

Article blueprint - H1 from value proposition; byline = Person - Intro (benefit-led), body sections from highlights - Pull-quotes and callouts auto-placed every ~300–400 words - FAQ (2–5 Q/A) only if confidence $\geq$ threshold - CTAs: primary offer, city + persona variants

Internal links insertion - First mention of each Topic → link to `topic_hub` - If Offer mentioned → link to `offer` - If a testimonial exists for that Offer → sidebar card injection

6) Schema (JSON-LD) Generation

Global: Person, Organization on home/about; Website + WebSite/Schema SearchAction.

Per type - `asset_video` → `VideoObject` (+ `UploadDate`, `Duration`, `ThumbnailUrl`, `Transcript`) with `about` (Topics) and `mentions` (Entities). - `asset_podcast` → `PodcastEpisode` with `isPartOf` `PodcastSeries`. - `asset_article` → `Article` / `BlogPosting` with `Author` (Person), `about` (Topics), `mentions`. - `testimonial` → `Review` with `itemReviewed` (Offer/Organization), `reviewRating`. - `event_speaking` → `Event` (location, startDate, organizer). - `offer` → `Service` / `Product` (price, areaServed, offers).

Example

```
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@type": "VideoObject",
  "name": "How we doubled review velocity in 30 days",
  "description": "Dennis Yu explains the review flywheel for local services.",
  "thumbnailUrl": ["https://cdn.example.com/thumbs/abc123.jpg"],
```

```

"uploadDate": "2025-08-01",
"duration": "PT4M20S",
"transcript": "...",
"about": [{"@type": "Thing", "name": "Local SEO"}],
"mentions": [{"@type": "Organization", "name": "Infinity Home Services"}],
"author": {"@type": "Person", "name": "Dennis Yu"}
}
</script>

```

Validation - Build + lint server-side. Block publish if fatal schema errors.

7) Internal Linking & Navigation Rules

- **Topic hubs:** every hub lists top 5 assets by `topic_score * recency`.
- **Contextual links:** first occurrence per section; max 1 link per 150 words to avoid spam.
- **Breadcrumbs:** Entity → Topic → Asset.
- **Canonicalization:** if source is external (YouTube/LinkedIn), our article is canonical unless 80% text overlap with original; then `rel=canonical` to source.
- **Sitemaps:** segmented — `/sitemap-assets.xml`, `/sitemap-topic-hubs.xml`, `/video-sitemap.xml`.

Scoring formula example:

```
score = (0.6 * topic_relevance) + (0.3 * freshness_decay) + (0.1 * engagement)
```

8) Performance & CWV

- SSR theme with minimal JS; defer non-critical.
- Critical CSS inlined per template; font-display: swap.
- Lazy-load images/video; responsive srcsets.
- Redis object cache; full-page cache via CDN.
- Target: **LCP < 2.0s**, **INP < 200ms**, **CLS < 0.05** on 4G mid-tier.

9) Analytics & KPIs Instrumentation

Data sources: GA4 (events), GSC (queries/CTR), GBP (impressions/calls), YouTube (views/watch time), server logs.

Collected KPIs - Branded search impressions (GSC) - Branded CTR (GSC) - Entity coverage % (assets with Topics, Offers, schema OK) - Review velocity (GBP, per month) - Lead events (GA4: `generate_lead`, `submit_form`, tel: clicks)

Storage & surfacing - Pull nightly via cron into WP `kpis` module tables; cache last 30/90 days. - Admin dashboard cards + sparkline.

10) Editorial Workflow

Roles: **Creator, Editor, Publisher, Engineer.**

- Ingestion auto-creates `asset_*` in `queued`.
- Editors see queue with confidence scores + suggested titles.
- WP-CLI commands:

```
wp blitz ingest youtube --channel=UCxxxxx
wp blitz transcribe run --post=123
wp blitz repurpose build --post=123
wp blitz publish --post=123
```

- Webhooks out: `asset.published`, `testimonial.created`, `kpi.updated` (CloudEvents format).

CloudEvents example

```
{
  "specversion": "1.0",
  "type": "blitz.asset.published",
  "source": "/wp/asset/456",
  "id": "evt_01H...",
  "time": "2025-08-27T18:10:00Z",
  "datacontenttype": "application/json",
  "data": {"post_id":456, "type":"asset_article", "topics":["Local SEO"],
  "url":"https://..."}
}
```

11) REST API (selected)

Base: `/wp-json/blitz/v1`

- `POST /ingest` — body: `{url, type}` to enqueue a single asset
- `POST /repurpose/{post_id}` — build article draft
- `POST /publish/{post_id}` — publish with validations
- `GET /kpis` — range filters, aggregates
- `GET /entities` — list Persons/Orgs with linkage

Auth: Application Passwords or JWT; rate-limit per IP/token.

12) Security, Privacy, Compliance

- Principle of least privilege for API tokens (YT/GBP/Meta/LinkedIn).
 - Secrets via environment vars + AWS Secrets Manager.
 - Reviewer initials only for GBP reviews (no doxxing); allow redaction.
 - Backups: DB (hourly PITR), uploads/S3 (versioned), configs in Git.
 - WAF (CloudFront/Cloudflare); brute-force login protection; automatic core/plugin updates in staging → canary → prod.
-

13) DevOps & Environments

- Envs: `dev` (feature branches), `staging`, `prod`.
 - CI: lint PHP (PHPStan), unit tests (Pest/PHPUnit), ESLint for admin JS, JSON schema tests.
 - CD: zero-downtime deploy; DB migrations via WP-CLI task.
 - Observability: error budgets for Core Web Vitals; alerts in Slack.
-

14) Testing Strategy

- Unit: schema builders, link rules, scoring fns.
 - Integration: ingestion → article publish happy path.
 - Regression: SEO checks (noindex, canonicals, sitemap diffs).
 - Lighthouse CI on critical templates.
-

15) Known Limitations / Edge Cases

- LinkedIn API is rate- and scope-restricted; some items require manual import.
- Instagram permissions may expire; refresh tokens cron.
- Podcast RSS inconsistencies (GUID reuse) → use URL hash fallback.
- If creators delete source videos, preserve canonical and 410 accordingly.

16) Backlog / Next Up (prioritized)

1. **GBP Post Scheduler** from inside WP with UTM tagging.
 2. **Auto-FAQ** generator with confidence thresholds + human-in-the-loop.
 3. **Entity Graph UI**: visualize Person/Org/Topic/Asset relationships.
 4. **OpenSearch** integration for on-site semantic search.
 5. **Tuned Whisper** with diarization + speaker labels.
 6. **Partner Mentions Miner**: discover likely podcasts/press to pitch.
 7. **Programmatic Local Pages**: geo × offer grid with canonical rules.
 8. **Video Chapters** → **How-To/Clip pages** with structured data.
 9. **GSC anomalies detector** (alerts on CTR/impressions drops).
 10. **Multi-tenant hardening** for agency roll-outs (namespaces, caps).
-

17) Appendix

Minimal plugin set - Advanced Custom Fields Pro (or internal field API) - WPGraphQL (optional, if headless)
- Redis Object Cache - Disable Unneeded Schema (to avoid dupes with our JSON-LD)

DB Hints - Indices on `postmeta(meta_key, meta_value(32))` for `source.url_hash` - Separate `wp_blitz_kpis` table (post_id, metric, day, value) PK(post_id, metric, day)

Robots/Sitemaps - Block thin tax pages; allow Topic hubs - Video sitemap for `asset_video` + `asset_article` with embedded video

TL;DR for Engineers

We're using WordPress as a robust CMS and permissioned UI, but the core win is our **structured entity/topic model**, **automated ingestion→repurpose pipelines**, and **first-class schema + internal linking** tuned for Google's entity understanding. This is not a theme; it's an opinionated content OS for personal brands.