BulletinScanner – 6-Month Founders First Principles & Operating Blueprint

Time horizon: Months 0–6
Goal: Prove that real subs will pay for BulletinScanner because it reliably catches scope changes they would have missed.

---

1. Six-Month Win Condition

In 6 months, this is a win if:

- We have 3–5 paying design partners across 1–2 trades (e.g., electrical + mechanical).
- Those partners upload real jobs (past or active) into BulletinScanner.
- On those jobs, our pipeline consistently surfaces meaningful changes they would have missed and they say so in their own words.
- We are at ~90% accuracy on a tight, agreed set of "critical fields" per trade (not everything in the docs).
- There is a clear path from today's "concierge audits" to a repeatable self-serve flow ("upload a job → see changes").

If those conditions are true, we continue and invest harder. If not, we stop or pivot.

---

2. Strategic Focus for the First 6 Months

2.1 What We Are Doing

1. Wedge into 1–2 trades, not "construction"
   - Start with Electrical and Mechanical/HVAC subs.
   - Everything we build must help these two ICPs catch scope changes and protect margin.
2. Product = "Previous Job Audits" (for now)
   - Offer: "Give us one job where you got burned. We'll reconstruct what changed and what you could've claimed."
   - Use these audits to:
     - Learn schemas and validation rules.
     - Prove value concretely in dollars.
     - Build trust and case studies.

3. Trade-partner co-design of schemas
   - Use 3–5 design partners to define:
     - The 20–30 fields per trade that truly matter (capacity, loads, hazard class, counts, etc.).
     - Validation rules (units, plausible ranges, code-informed constraints).
   - Bullet: we optimize for those fields, not everything.
4. Build the accuracy pipeline, not a big app
   - Azure Document Intelligence + custom models for these trades.
   - Python pipeline for:
     - Parsing → extraction → validation → diff → change events.
     - Confidence thresholds + flagging low-confidence items.
   - A simple, consumable audit report (could be a PDF/HTML/Google Doc) that a PM can understand.
5. Lay the groundwork for self-serve
   - Even if we run jobs by hand, we design the flow as:
     - Create project → upload base set + bulletins → see structured change events.
   - Instrument usage and time-to-value from day one.

2.2 What We Are NOT Doing (First 6 Months)

- No GC/owner dashboards or multi-stakeholder workflow.
- No supporting more than 2 trades in depth.
- No "this might also work for residential / architects / random PDFs."
- No heavy custom integrations (deep Procore, ERPs, etc.) beyond simple PDF input.
- No long enterprise pilots with committees and security reviews.
- No fundraising roadshow before:
  - We have real usage,
  - Real partners, and
  - A believable accuracy + ROI story.

---

3. GTM Founder – First Principles & Focus

3.1 Core Responsibilities (Months 0–6)

1. Own ICP and customer access
   - Clarify and refine ICP:
     - Trade: electrical + mechanical subs.

- Size: $5–100M revenue, multiple concurrent commercial projects.
- Psychographic: "feel nickel-and-dimed by scope creep," leadership obsessed with gross margin and risk.
  - Generate and manage pipeline of:
    - 30–50 conversations with qualified subs.
    - 3–5 committed design partners.
2. Run the discovery & value loop
   - Weekly:
     - Talk to PMs, estimators, operations leaders.
     - Get real jobs and doc sets.
     - Ask about where they lost money and how they currently catch changes.
   - Convert conversations into:
     - Schema requirements (what fields matter).
     - Story language (how they describe the pain and value).
3. Drive "previous job audits" as a product
   - Pitch: "Hand us one job where you got burned; we'll give you a structured breakdown of changes you missed."
   - Work with technical founder to:
     - Scope the audit.
     - Set expectations on turnaround.
     - Present findings in a simple, high-impact format (slides, report, call).
4. Close and collect money (even if small)
   - Charge something where possible, even nominal:
     - E.g., $2–5K per audit, or a small monthly pilot fee.
   - Goal is not short-term revenue, but proof people will pay and a clearer sense of pricing power.
5. Shape product from the field back
   - Translate messy field stories → clear product requirements:
     - Which fields must we capture?
     - How should change events be grouped and displayed?
     - What "feels trustworthy" vs. "feels like AI hand-waving" to a PM?

3.2 What GTM Should NOT Focus On (Now)

- Building complex marketing funnels or fancy website instead of talking to subs.
- Over-optimizing brand, decks, and messaging before audits prove value.
- Chasing every inbound or adjacent segment ("we could also do legal / manufacturing").

- Getting stuck as de facto CSM for non-ICP customers.
- Raising a big round on a purely conceptual deck.

---

4. Technical Founder – First Principles & Focus

4.1 Core Responsibilities (Months 0–6)

1. Design and implement trade-specific schemas
   - For Electrical:
     - Entities: Panel, Circuit, EquipmentLoad, Feeder, Voltage/Phase, etc.
     - Fields: amps, kVA, counts, voltage, phases, safety/emergency tags.
     - Relationships: which loads roll up to which panel; which changes imply cross-trade impact.
   - For Mechanical/HVAC:
     - Entities: MechanicalEquipment (CFM, tons, MBH, voltage), Zone/VAV, ControlPoint.
     - Fields: capacities, counts, areas, design temps, etc.
2. Build the extraction + validation pipeline
   - Use Azure Document Intelligence (custom models) for:
     - Layout, tables, schedules, key-value.
   - Wrap it in a Python service that:
     - Normalizes units and formatting.
     - Applies engineering checks (plausible ranges, code-informed rules).
     - Computes meaningful diffs between versions.
     - Calculates confidence and routes low-confidence items to review.
3. Optimize for accuracy, not completeness
   - Select the 20–30 key fields per trade that drive cost/risk.
   - For those fields:
     - Drive coverage and precision as high as possible.
     - Target ~90–95% precision on fields we auto-accept, and explicit flagging for the rest.
   - Resist requests to "extract everything" if it harms stability.
4. Deliver a simple, trustworthy output
   - Start with:
     - CSV/JSON + a generated human-readable report (e.g., a simple HTML or Markdown → PDF).
   - Make it easy for the GTM founder + customers to:

- Scan a list of change events.
- See exactly where in the document the change came from.
- Understand business impact in plain language.

5. Instrument reliability and feedback
   - Log:
     - Extraction confidence.
     - Validation pass/fail.
     - Corrections from humans.
   - Use this to prioritize:
     - Where to add examples.
     - Where to tighten rules or adjust prompts.

4.2 What Technical Should NOT Focus On (Now)

- Over-architecting for massive scale before we have 3–5 design partners.
- Building a full multi-tenant, role-based, integration-heavy web app.
- Supporting more than 2 trades deeply in the first 6 months.
- Building broad "AI features" not anchored in a partner's use case.
- Premature optimization for infra cost or latency over accuracy and reliability.

---

5. How We Work Together (Team Principles)

5.1 Shared Truths

1. Customer reality > our opinions
   - If subs say, "this isn't actually useful," we change the product.
   - If they say, "this one thing is gold," we double down on that, even if it's not what we expected.
2. Accuracy and trust > feature count
   - A few fields that are always right beat many that are sometimes wrong.
   - We are in a domain where wrong outputs destroy trust instantly.
3. Speed + iteration over perfection
   - Small, weekly increments:
     - One new validation rule,
     - One better report layout,
     - One more pilot customer.

5.2 Division of Responsibility

GTM Founder Owns:

- ICP definition and refinement.

- Prospecting, outreach, and closing design partners.
- Running discovery and demo calls.
- Narrative, messaging, and pricing experiments.
- Translating field feedback into prioritized product asks.

Technical Founder Owns:

- Architecture and tools selection (Azure, Python stack, infra).
- Schemas, extraction models, validation rules.
- Accuracy, robustness, and performance of the pipeline.
- Dev process and code quality.
- Data handling, security basics, and reliability.

Shared:

- Which trades to prioritize and when to expand.
- Who qualifies as a good/bad design partner.
- When to tighten focus vs. explore new angles.
- Exit criteria at the 6-month mark.

5.3 Operating Rhythm

Weekly (non-negotiable):

- 30–60 min Founders Sync:
  - What did we ship?
  - What did we learn from customers?
  - What's blocking us?
  - Top 3 priorities for next week (per founder).
- 1–2 joint customer sessions:
  - Review an audit together or walk through screenshots.
  - GTM leads the conversation, Technical listens for requirements.

Monthly:

- Review progress vs. 6-month win condition:
  - # design partners.
  - # audits completed.
  - Accuracy on key fields.
  - Any revenue or strong intent signals.

5.4 How to Avoid Common Co-Founder Traps

- Over-indexing on one side:

- If we go weeks without talking to customers, GTM rings the alarm.
- If we sign deals we can't deliver, Technical rings the alarm.
- Hidden resentment over workload:
    - Once a month, each founder answers:
        - "What's energizing me?"
        - "What's draining me?"
    - Adjust roles and expectations accordingly.
- Scope creep via "just one more feature":
    - Every feature idea must answer:
        - Does this help us land or retain our 3–5 design partners?
        - Does this improve accuracy/speed for our two main trades?
    - If not, it goes into the "later" bucket.

---

6. Concrete 6-Month Milestones (Draft)

You can tune these, but as a starting point:

Month 1–2

- Talk to 15–25 subs in electrical + mechanical.
- Secure 3 design partners willing to share past jobs.
- Implement v0 of:
    - Schemas for electrical + mechanical.
    - Azure + Python pipeline on at least one real job.
- Deliver 1–2 audits, even manually assisted.

Month 3–4

- Expand to 3–5 design partners.
- Deliver 5–10 audits total.
- Achieve 80–85%+ accuracy on defined critical fields for at least one trade.
- Stand up a simple "upload job → get report" interface (can still be semi-manual behind the scenes).

Month 5–6

- Push accuracy toward 90–95% on key fields for at least one trade.
- Have 3–5 partners actively using the product and at least a couple paying.
- Have clear testimonials / case studies:
    - "BulletinScanner found X changes worth $Y that we would've missed."
- Decide:

- Double down + start building more self-serve and acquisition loops, or
- Pivot/stop based on data.

---

7. Final First-Principles Check

For the first 6 months, almost everything can be reduced to three questions:

1. Does this help us secure, serve, or learn from our 3–5 design partners in electrical/mechanical?
2. Does this increase the accuracy, speed, or clarity of our change detection for those partners?
3. Does this make it easier for a new sub to upload a job and see value fast?

If the answer isn't clearly "yes" to at least one of those, it's probably not a 6-month priority.

Copy this into Google Docs as your "Founders Operating Blueprint", tweak the milestones/numbers to your taste, and use it as the thing you both revisit every week.