

24-Hour Low-Code Challenge Brief — Article Drafting MVP

Objective

Build a small, lovable MVP that turns an **interview transcript + supporting sources** into a **story-driven draft article** with a **human-in-the-loop (HITL)** review. You have **24 hours** to design, implement, and demo.

Who this is for

Graduates applying for **Software Engineering Intern – AI-Assisted Development** (Sri Lanka, hybrid). We're testing product thinking, AI-assisted dev skills, and execution under time constraints.

Scope (What to Build — minimally)

Deliver an end-to-end path that lets an editor: 1. **Create a project** and **paste/upload a transcript** (any AI-related product interview you choose, public sources only).

2. **Attach at least one supporting source** (e.g., a brochure PDF or a reputable web article/YouTube link).

3. **Extract key points automatically** → show a review list the editor can **approve/edit/reorder**.

4. **Set a story direction** (tone/angle/length) via a simple control or prompt field.

5. **Generate an outline or full draft** that references the approved points.

6. Provide **basic source mapping** (hover/click to see which source supports a paragraph/claim).

7. Run a **minimal Quote Checker**: for any quoted text in the draft, show the matching source snippet (exact or clearly indicated with ellipses if non-contiguous).

8. **Export** the draft as **Markdown** (bonus: include a small provenance JSON).

Keep it focused. A polished Happy Path beats a sprawling, half-working app.

Constraints & Levers

- **Platform**: Build primarily with **Lovable.dev** (low-code). You may also use Cursor/Replit/bolt.new locally as you wish.
- **Data**: Use **publicly accessible** transcripts and sources. Do **not** upload private data.
- **AI usage**: Any LLM/service is allowed if you supply your own keys. If unavailable, you may simulate parts but clearly note it in the README.
- **Design**: Clean, minimal editorial UX. Prioritize clarity over flash.
- **Timebox**: 24 hours from brief receipt to submission (Sri Lanka time, UTC+5:30).

Creative freedom

You own the micro-interactions, data model, and prompt design. We intentionally leave details open—surprise us with thoughtful choices.

Deliverables (What to Submit)

1. **Lovable link** to your live app (viewer access).
2. **GitHub repo** with source, minimal instructions, and a short **README** covering:
3. Problem framing & assumptions
4. Architecture diagram or description
5. How you handle key-point approval, source mapping, and quote checks
6. Trade-offs & what you'd do next with another day
7. **3-minute demo video** (unlisted link OK) or a concise walkthrough GIF/GIFV.
8. **Sample inputs** you used (links to your interview + sources).

Optional extras (nice to have):

- Light tests for your quote-matching logic.
 - A provenance JSON in the export (mapping paragraph/quote → source).
-

Evaluation Rubric (100 pts)

- **Core Flow Works (Happy Path)** — 30 pts
Ingest → Approve points → Direction → Outline/Draft → Export.
- **Quote Integrity & Source Mapping** — 20 pts
Quotes trace back to sources (exact or ellipses); claims have visible provenance.
- **UX Clarity & HITL Controls** — 15 pts
Can an editor understand and steer the story without reading code?
- **Product Thinking & Creativity** — 10 pts
Sensible defaults, tidy scope, thoughtful prompts, small delights.
- **Code Quality & Repo Hygiene** — 15 pts
Clear structure, readable code, brief comments, a helpful README.
- **Stability & Deployment** — 5 pts
Works reliably in the browser; error states don't trap the user.
- **Presentation & Communication** — 5 pts
Crisp demo; decisions and trade-offs explained.

Tiebreakers: Handles a longer transcript ($\geq 10k$ chars) without choking; sensible performance.

Ground Rules

- Respect copyright: quote sparingly and attribute.
 - Keep PII out. Use public materials only.
 - No security theatre: if you use API keys, do not commit them.
 - Be honest about mocks/simulations.
-

Timeline & Process

- **Kickoff:** You receive this brief.
 - **Build window:** 24 hours.
 - **Submission:** Post your Lovable link + GitHub repo + demo video + input links.
 - **Shortlist Interviews:** 10–15 minute follow-up where we'll give you a fresh, short transcript snippet to run through your app live.
-

FAQ (short)

- **Do I need a backend?** Not required—client-only is fine if you can demonstrate the flow.
 - **How fancy must the Quote Checker be?** Minimal viable: detect quotes in the draft and show the matching source snippet (case/whitespace normalized). Handle multi-span quotes with ellipses.
 - **Do I need full YouTube transcription?** Not necessary—pick interviews with transcripts available or use articles as your main sources.
 - **Can I pre-seed prompts or templates?** Yes—document them in your README.
-

Stretch Ideas (choose zero or one)

- A tiny **Direction Preset** system (e.g., Founder story vs. Launch deep-dive).
- A basic **versioning** dropdown (Latest + 1 checkpoint).
- A hover “**Sources (n)**” popover per paragraph.

Good luck—build something small, clear, and lovable. We value judgment and finish over buzzword bingo.