Critical Technical Landmines & Demo Risk Analysis

This document outlines the key technical, architectural, and demo-related risks for the Ally Lite hackathon project, along with practical mitigation strategies. The goal is to maximize demo reliability, credibility, and judge impact.

1. Lighthouse Performance Will Kill Your Demo

Key risks:

● Running Lighthouse in real-time can take 30–90 seconds per audit.

● Cloud Functions introduce cold start delays (2–5 seconds minimum).

● On-demand scans during a live demo often result in long, silent waits.

What kills you: Judge enters a URL → spinner → awkward silence → loss of attention.

Mitigation: Pre-run Lighthouse reports for demo sites, use aggressive caching, or show progressive results.

2. Gemini API Chain Is Fragile

Failure points in the pipeline (Lighthouse → Normalize → Gemini → Display):

● Gemini API rate limits during hackathon traffic spikes.

● Gemini response latency of 3–10 seconds.

● Multiple Gemini calls per report multiply total latency.

What kills you: Demo works locally but fails under pressure due to API slowness or limits.

Mitigation: Cache Gemini responses, reduce to a single AI call, or pre-generate outputs for demo URLs.

3. Gemini Hallucinations Are Obvious and Dangerous

● Suggesting fixes for non-existent issues.

● Inventing or misreporting performance metrics.

● Recommending code changes that could break sites.

What kills you: A knowledgeable judge spots an incorrect explanation, destroying credibility.

Mitigation: Constrain AI to raw audit data, show raw + AI side-by-side, or template explanations heavily.

4. Prioritization Logic Must Be Explainable

● Gemini-based ranking is unreliable and non-deterministic.

● Sorting purely by Lighthouse scores is not true prioritization.

● Lack of clear logic invites judge skepticism.

What kills you: You cannot explain why issue X ranks above issue Y.

Mitigation: Use simple, explicit rules (severity × user impact × ease of fix) and explain them clearly.

5. Firebase / Firestore Overkill

- Auth and security rules add unnecessary complexity.
- Shareable links require routing, storage keys, and cleanup logic.

What kills you: Time wasted on infra instead of core experience.

Mitigation: Avoid auth, use in-memory or simple storage, prioritize demo stability.

Demo-Killing Scenarios

- Judge enters a slow or complex site → Lighthouse times out.
- HTTP or unreachable URL causes Lighthouse failure.
- Gemini produces an obviously incorrect explanation.
- Too many features demonstrated poorly.

What Actually Scores Points in Hackathons

- Reliable, deterministic demo (≈50%).
- Clear and understandable core idea (≈30%).
- One impressive or novel insight (≈20%).

Brutal Truth: Weakest Links

- Real-time Lighthouse execution latency.
- Gemini reliability during live demos.
- Over-scoped extension preview feature.
- Unclear differentiation from 'Lighthouse + AI summary'.

Specific Recommendations

Do This:

- Pre-run and cache reports for 5–10 demo sites.
- Cache or pre-generate Gemini responses.
- Make the demo fully deterministic.
- Build 1–2 features extremely well.

Do NOT Do This:

- Run real-time Lighthouse during the pitch.
- Chain multiple Gemini calls.
- Overbuild the extension preview.
- Pretend prioritization is smarter than it is.