Hackathon project evaluation

India JS_FMCG_Asian_Paints.pptx
Presentation

You are an expert hackathon jury member, startup evaluator, and AI research reviewer.
You must evaluate the following project submission strictly and objectively.
This evaluation will be used for shortlisting finalists in a national-level hackathon.

You must score based ONLY on merit, not on presentation fluff or emotional language.

Your task is to:

Detect plagiarism and idea similarity

Judge originality and innovation

Analyze technical depth

Evaluate real-world feasibility and scalability

Judge efficiency and optimization

Verify prototype authenticity

Analyze README/PPT clarity

Detect overuse of buzzwords without substance

Identify missing pieces

Provide a final numeric score and verdict

You must be strict. Assume that 90% of submissions are low quality and must be filtered out.

✅ INPUT FORMAT

You will be given:

Problem Statement

Solution Description

Architecture / Flow

Tech Stack

README.md or PPT Content

GitHub Repo Summary (if provided)

Screenshots / Demo Description (if provided)

✅ EVALUATION CRITERIA & WEIGHTAGE (100 POINTS TOTAL)
1. Problem Understanding & Relevance (10 Points)

Does the solution directly address the exact problem?

Are constraints (domain, scale, users, government/MSME/etc.) respected?

Is this solution targeted or generic?

Score out of 10.

2. Originality & Innovation (20 Points)

Is the idea fundamentally new?

Is this just a known startup idea wrapped in new words?

Is the workflow novel?

Is the AI usage meaningful or cosmetic?

Score out of 20.

3. Technical Depth & Architecture (15 Points)

Is there a real system design?

Are models, APIs, pipelines, databases clearly defined?

Is the architecture production-grade or only conceptual?

Score out of 15.

4. Prototype Authenticity & Implementation (15 Points)

Is there real working code or only mock UI?

Are endpoints real?

Is the ML model actually trained or fake?

Score out of 15.

5. Feasibility & Scalability (10 Points)

Can this realistically be deployed?

Cost, cloud usage, compute feasibility?

Can it scale beyond a demo?

Score out of 10.

6. Efficiency & Optimization (10 Points)

Is compute efficient?

Is over-engineering avoided?

Are pipelines optimized?

Score out of 10.

7. README / PPT Technical Clarity (10 Points)

Is the explanation precise?

Are steps reproducible?

Is setup clear?

Are metrics shown?

Score out of 10.

8. Plagiarism, Similarity & Genericness (10 Points)

Does it resemble common hackathon templates?

Is semantic plagiarism detected?

Is it a recycled chatbot/dashboard?

Score out of 10
(10 = Fully original, 0 = Copied/Generic)

✅ MANDATORY OUTPUT FORMAT

You MUST reply strictly in the following format:

🔍 1. PROBLEM UNDERSTANDING SCORE: X/10

(1–2 lines justification)

🚀 2. ORIGINALITY & INNOVATION SCORE: X/20

(Explain what is new or what is common)

🏗️ 3. TECHNICAL DEPTH SCORE: X/15

(Assess real engineering effort)

🧪 4. PROTOTYPE AUTHENTICITY SCORE: X/15

(Fully working / partial / fake)

🌍 5. FEASIBILITY & SCALABILITY SCORE: X/10

(Deployment realism)

⚡ 6. EFFICIENCY & OPTIMIZATION SCORE: X/10

(Resource efficiency)

📘 7. README / PPT CLARITY SCORE: X/10

(Clarity for developers & judges)

🧬 8. PLAGIARISM & GENERICNESS SCORE: X/10

(Original vs recycled)

✅ ✅ ✅ FINAL SCORE: X / 100
✅ SHORTLISTING VERDICT (MANDATORY)

Choose ONLY ONE:

✅ AUTO-SHORTLIST (Top 10%)

⚠ BORDERLINE (Human Jury Review Needed)

❌ REJECT (Low Innovation / Weak Technical Depth)

✅ TOP 5 CRITICAL WEAKNESSES

(List real weaknesses that could cause rejection)

✅ TOP 5 STRENGTHS

(List real strengths)

✅ IMPROVEMENT SUGGESTIONS FOR FINAL ROUND

(Only practical improvements, no motivational talk)

✅ PLAGIARISM / COMMON IDEA WARNING (If Any)

(Explicitly mention if detect similarity with common project types)

🚨 IMPORTANT RULES FOR YOU AS JUDGE:

Do NOT be polite.

Do NOT inflate scores.

Do NOT assume missing parts.

Judge only what is shown.

If something is vague → penalize.

If it sounds like "just another AI app" → penalize heavily.

🔍 1. PROBLEM UNDERSTANDING SCORE: **8/10**

The submission clearly understands the **industrial OEM RFP bottleneck (FMCG / Wires & Cables)** and maps pain points across discovery, spec matching, pricing, and coordination. However, parts drift into **over-generalized "enterprise RFP" framing**, reducing sharpness on FMCG-specific constraints.

🚀 2. ORIGINALITY & INNOVATION SCORE: **11/20**

This is **not a fundamentally new idea**.
It is a **well-executed combination** of:

- RFP automation
- Spec extraction
- SKU matching
- Multi-agent orchestration

All of these **already exist independently** in procurement-tech startups (e.g., RFPIO, Loopio, Icertis + AI add-ons).
The *integration* is decent, but **no novel algorithm, model, or workflow is introduced**.
"Multi-agent" is used heavily but mostly as a **workflow abstraction**, not a research-level innovation.

🏗️ 3. TECHNICAL DEPTH SCORE: **9/15**

Architecture is **reasonably structured**:

- RAG
- Vector DB
- Agent separation
- Microservices

However:

- No concrete **data schemas**
- No **API contracts**
- No **model evaluation methodology**
- No evidence of **custom ML beyond LLM prompting**

This is **system integration**, not deep engineering. Production-grade claims are **partially unsubstantiated**.

🧪 4. PROTOTYPE AUTHENTICITY SCORE: **7/15**

A demo video and drive link are provided, but:

- No public GitHub with verifiable commits
- No proof of trained models
- No logs, metrics, or live endpoints
- Likely **UI + orchestrated prompts**, not a deeply functional backend

This looks like a **semi-functional prototype**, not a hardened system.

🌍 5. FEASIBILITY & SCALABILITY SCORE: **6/10**

Technically deployable, **but financially and operationally optimistic**:

- $200K setup + $12M Year-1 benefit is **not credible without sales pipeline proof**
- PSU portal automation is **legally and operationally fragile**
- Heavy dependency on LLM inference costs (Gemini + Groq)

Scales as a **pilot SaaS**, not yet as an enterprise platform.

⚡ 6. EFFICIENCY & OPTIMIZATION SCORE: **6/10**

Parallel agent execution is good in theory, but:

- No latency benchmarks
- No token optimization strategy
- No cost-per-RFP inference math
- Overuse of agents where deterministic pipelines would suffice

Some **over-engineering risk** is present.

📘 7. README / PPT CLARITY SCORE: **8/10**

The PPT is **clear, structured, and readable**:

- Good flow
- Quantified claims
- Clear user journey

However:

- Metrics are **asserted, not proven**
- No reproducibility steps
- Heavy reliance on business language over technical validation

---

🧬 8. PLAGIARISM & GENERICNESS SCORE: **5/10**

No direct copy-paste plagiarism detected, but:

- Strong resemblance to **standard hackathon "AI RFP automation" templates**
- Heavy buzzword stacking: *agentic, autonomous, intelligence, mic-drop*
- Conceptually similar to many 2024–2025 procurement AI demos

This is **derivative, not copied**.

✅ FINAL SCORE: 60 / 100

✅ SHORTLISTING VERDICT (MANDATORY)

⚠ **BORDERLINE (Human Jury Review Needed)**

✅ TOP 5 CRITICAL WEAKNESSES

1. No verifiable trained ML models or evaluation metrics
2. “Multi-agent” used as a buzzword more than a technical necessity
3. ROI and revenue projections are **unrealistic and unvalidated**
4. Lack of public, auditable GitHub repository
5. Over-claiming autonomy while still requiring heavy human-in-the-loop

---

✅ TOP 5 STRENGTHS

1. Strong real-world problem relevance (industrial OEMs)
2. Clear end-to-end workflow mapping
3. Good domain research and stakeholder interviews
4. Reasonable system integration design
5. Better-than-average PPT clarity compared to typical hackathon entries

---

✅ IMPROVEMENT SUGGESTIONS FOR FINAL ROUND

1. Show **actual SKU-matching accuracy benchmarks** vs baseline
2. Open-source a **minimal but real backend repo** (even partial)
3. Reduce agents → show **why each agent is technically required**
4. Add **cost-per-RFP inference math**
5. Replace ROI hype with **pilot-scale numbers only**

✅ PLAGIARISM / COMMON IDEA WARNING

⚠ **Common AI Hackathon Pattern Detected**

This closely resembles many **“AI-powered RFP / procurement automation”** submissions seen across national hackathons in 2024–2025. While not plagiarized, it is **conceptually crowded** and must prove differentiation through **real metrics and code**.

Final Jury Note:

This project is **above average**, but **not Top-10% yet**. It survives initial filtering but requires **hard technical proof** to advance.