

The screenshot shows the DevPost Tableau Hackathon page. Key details include:

- Deadline:** Jan 12, 2025 @ 2:00pm CST
- Participation:** Online, Public
- Rewards:** \$45,000 in cash
- Participants:** 743 participants
- Who can participate:** Min 18 years old & Eligible Individuals, Above legal age of majority in country of residence, Only specific countries/territories included (Quebec is excluded)
- About the challenge:** Step into the next era of analytics. This is your opportunity to pioneer innovative solutions and tackle real-world challenges.

DevPost - Tableau Hackathon.zip Zip Archive

1 Introduction to Rubrics.pdf PDF

We're kicking off another hackathon journey. Analyze the files. Tell me what you think about Grok and my tentative entry: Agentic Analytics Protocol (AAP). Weigh it against the other ideas Grok had and your own ideas using the Rubrics as a reference. The goal is to of course, as always, take top prize.

Thought for 17m 41s >

Short version: Grok did a nice job mapping the hackathon and surfacing AAP; AAP is clearly the most *ambitious* idea in the pile, but it's also the riskiest. If you want a real shot at the Grand Prize, I'd keep AAP as the core *concept*, but ship it as a focused, working protocol + one strong vertical demo (probably governance or fraud) rather than just a broad "standard."

Below is how I'm thinking about it, using the hackathon rubric + the *Introduction to Rubrics* book as the backbone.

1 Introduction to Rubrics

1. The rubric we're playing to

From Grok's PDF + the Devpost content, the judging grid is basically:

- Innovation & Creativity (Idea Quality) – 40%
- Technical Execution & Functionality – 30%
- Potential Impact – 20%
- UX & Presentation – 10%

And it's using a 3-level scale (Exemplary / Proficient / Developing), exactly like the teaching-rubric model: clear task, scale, dimensions, and performance descriptions.

1 Introduction to Rubrics

So I'll score things mentally as:

- Exemplary (E) – this could plausibly win its category.
- Proficient (P) – solid but not unforgettable.
- Developing (D) – interesting, but under-baked for this contest.

2. What Grok got right (and where it's limited)

The guide & rubric

- The “North Star” Devpost guide is genuinely useful: everything you need about rules, prizes, eligibility, and resources is in one place. Good move.
- The hackathon rubric Grok built from that is well-aligned:
 - Right dimensions and weights.
 - 3-level descriptors that mirror the rubric book’s advice: few dimensions, clear language, focus on what “Exemplary” really looks like.

That’s actually a strong foundation for *your* strategy: you’re starting with an explicit scoring grid instead of vibes.

Grok’s 10 idea seeds

Quick recap:

1. Predictive Maintenance Dashboard (manufacturing)
2. Customer Journey Analyzer (Salesforce CRM)
3. Secure Data Governance Auditor
4. Collaborative Forecasting Hub (sales)
5. Sustainability / ESG Impact Tracker
6. Real-Time Health Analytics for Clinics
7. Dynamic Inventory Optimizer

8. Employee Wellness Insights Platform
9. Fraud Detection Sentinel
10. Educational Adaptive Learning Dashboard

Patterns:

- Most are **vertical “agentic dashboard” apps**: one main domain, semantic layer, some Agentforce, some Slack notifications.
- They score well on **Technical Execution and Potential Impact** if built, but **innovation is usually “good, not insane.”**
- Several are things other teams are very likely to attempt: #1, #2, #4, #7, #9, #10 are hackathon classics with an agentic coat of paint.

So Grok did a competent PM-style brainstorm, but none of these (except AAP) scream “*this is obviously the overall winner*” yet.

3. How those 10 ideas stack up on the rubric (high level)

Group A – Classic vertical apps

(1, 2, 4, 6, 7, 9, 10)

- **Innovation (40%)** – Mostly P
 - They use Tableau Next + Agentforce nicely, but they’re variations on problems every AI hackathon sees (maintenance, journeys, forecasting, fraud, etc.).
- **Technical Execution (30%)** – Potentially E
 - These are straightforward to build with the provided org: one or two data sources, a semantic layer, dashboards, and an action pipeline.
- **Impact (20%)** – P→E depending on domain
 - Fraud (#9), health (#6), and education (#10) have strong narratives.
- **UX & Presentation (10%)** – Up to you; conceptually P.

Net: Strong “category winner” candidates *if very polished*, but high risk of being one of many similar entries. They’re safe but not clearly dominant for Grand Prize.

Group B – Cross-cutting / platform-ish apps

(3, 5, 8)

- #3 – Secure Data Governance Auditor

- Innovation: high-ish ($P \rightarrow E$) if you lean into agentic data governance + compliance.
- Technical: moderate complexity but very feasible.
- Impact: E – governance + security are huge pain points.
- This is probably Grok's best *non-AAP* idea: it maps nicely to Salesforce's real business anxieties.
- #5 – Sustainability Impact Tracker
 - Innovation: P – ESG is hot, but lots of dashboards here already.
 - Impact: high, but judges may prioritise revenue/security use cases.
- #8 – Employee Wellness Insights
 - Nice "people analytics" angle, good for People's Choice, but less obviously strategic to Salesforce than governance or fraud.

Net: #3 is the standout "safer but still strategic" idea. The others are good but less central to the platform story.

4. AAP – Agentic Analytics Protocol

What it is, in my words

AAP as Grok described it:

A plug-and-play protocol module for Tableau Next/Cloud that defines how agents discover each other, authenticate, exchange semantic messages, and build trust across an analytics pipeline (data → analysis → action in Slack).

Key elements:

- Three "dimensions" of the protocol (nicely echoing rubric construction):
 1. Basic building blocks – identity, auth, capability discovery.
 2. Meta-level handoff rules – how agents call each other, with semantic mapping of concepts like "churn" or "risk."
 3. Semantic trust layer – logging/scoring of agent actions (zero-trust-ish ledger).
- Application: shipped as an extension API module with a demo "agent swarm" (e.g., a sales pipeline optimizer that spans Data 360 → Tableau Next → Agentforce → Slack).

This is much more of a **platform standard** than an app.

How AAP scores on the hackathon rubric (if you execute a focused MVP)

My rough target scoring, if you build a working prototype + demo use case:

1. Innovation & Creativity (40%) – clear E

- AAP is genuinely novel: you're not just building a dashboard, you're proposing the **missing interop layer** for Tableau's "agentic analytics" story.
- It directly aligns with Salesforce's deeper agenda from the 5-Whys section: multi-agent, AI-native workflows that unlock future products, not just nice charts.

2. Technical Execution & Functionality (30%) – currently D → can be P/E

- Right now it's conceptual. If you:
 - Implement a **small but real protocol surface** (e.g., one registry service, one "hand-off" spec, one trust log).
 - Show **2–3 agents** actually using it end-to-end inside a Salesforce org, it jumps to **Proficient**, maybe **Exemplary** if the implementation is surprisingly clean.

3. Potential Impact (20%) – high P → E

- Judges love platform plays *if* they can see the "so what?".
- You must anchor it in a concrete use case (e.g., data governance, fraud, or sales optimization) and articulate:
 - "Without AAP you build one-off agents; **with** AAP you get reusable multi-agent pipelines and auditability."

4. UX & Presentation (10%) – depends entirely on you

- To hit E here, you want:
 - A simple **visual of the agent network / protocol flow**.
 - A crisp **<5 min story**: problem → AAP concept → live demo → "here's what's possible next."

5. AAP vs Grok's other ideas

Compared to the vertical apps

- Vertical apps (maintenance, journeys, fraud, etc.) are:
 - **E in Execution / P in Innovation** if done well.
- AAP is:
 - **E+ in Innovation / P? in Execution** if you land the MVP.

Given Innovation is 40% of the score and Salesforce is trying to make Tableau Next *the* agentic analytics platform, AAP has a strictly higher upside than any of those verticals *if* you can make it look real.

Compared to the best “safe” idea: Data Governance Auditor (#3)

- #3 has:
 - Innovation: P→E
 - Technical: E (feasible)
 - Impact: E
 - But it's still “*a very good app*,” not a protocol.
- AAP has:
 - Innovation: stronger story and more “this could be a future standard.”
 - Impact: potentially larger (multi-domain).
 - Risk: much higher complexity and more ways to fumble the demo.

My take:

- If your goal is “**strong chance of some prize**”, Governance Auditor alone is a good path.
- If your goal is “**swing for Grand Prize**”, AAP (properly scoped) is the better bet.

6. How I'd shape the actual entry

6.1. Scope AAP down to a *small but complete* protocol

Think in rubric terms: clearly define the **task, dimensions, and descriptors** for AAP itself:

- **Task:** Enable 2–3 agents to cooperate on one business flow using a shared protocol.
- **Dimensions** (for v0.1):
 1. **Service Registry & Capability Discovery** – one endpoint that lists agents and their skills.
 2. **Handoff Contract** – a standard JSON schema for “tasks” and “results” (with semantic labels).
 3. **Trust / Audit Log** – a simple log table that records who called what, with what data and outcome.

If those 3 work reliably in one org, you already have a compelling protocol MVP.

6.2. Pick ONE flagship use case on top of AAP

This is where I'd blend Grok's ideas with my own.

Best candidates:

- **Secure Data Governance Auditor on AAP** (my top recommendation)
 - Agents:
 - **Scanner Agent** – walks Data 360 / Tableau metadata, flags sensitive fields.
 - **Policy Agent** – checks access vs policies, creates “violations.”
 - **Remediation Agent** – opens a case or posts to Slack with recommended fix.
 - AAP ties them together so:
 - Scanner → Policy → Remediation → Slack happen via your protocol, not bespoke glue.

OR

- **Fraud Detection Sentinel on AAP**
 - Same pattern but with transactions, anomaly agent, investigation agent, escalation agent.

Both are:

- Very **on-brand** for Salesforce (risk, compliance, revenue protection).
- Good showcases of why **multi-agent workflows** matter.

6.3. Design the UX / demo around the *protocol*, not just the app

- **Developer view:** a simple UI (or even a Tableau dashboard) that:
 - Lists registered agents, their capabilities, and last health check.
 - Shows a graph of current pipelines and recent handoffs.
- **Business view** (governance/fraud dashboard):
 - Risk scores over time.
 - Recent auto-remediations with traceable audit entries (“Scanner → Policy → Remediation”).

Your video flow:

1. “Here’s the pain” (governance/fraud is fragmented, hard to trust AI).
2. “Here’s AAP: a simple protocol that lets agents discover each other, collaborate, and be audited.”
3. Live demo: kick off a pipeline; show the agents; show the dashboards and Slack action.
4. Close with roadmap: “AAP can power *any* Tableau Next multi-agent workflow.”

6.4. Use the rubric book as an internal compass

The book's big message is: **good rubrics clarify expectations and speed feedback.**

1 Introduction to Rubrics

You can treat AAP like a rubric:

- Define **Exemplary / Proficient / Developing** for:
 - Protocol completeness,
 - Demo robustness,
 - Business clarity.
- After each work session, literally score yourself and adjust – just like they suggest for teaching improvement.

7. So... should you stick with AAP?

If you have the energy to push through a more complex build, **yes**:

- Grok's vertical ideas are solid, but conventional.
- AAP is the only thing in the set that feels like a "**this could be the TCP/IP of Tableau agents**" story.
- Marrying AAP to a concrete, high-stakes use case (governance or fraud) gives judges both **the vision and the proof** they need.

If you'd like, next step we can:

- Turn AAP v0.1 into an explicit spec (endpoints, JSON schemas).
- Sketch the exact agent roles + data model for the governance/fraud use case.
- Draft the 5-minute video script in a way that hits every cell of the hackathon rubric.

Sources



alrighty. I'll give you the mic now. Come up with your top 10 best ideas to take grand prize, use the rubric, and rank them in descending order from likeliest to win.

Thought for 12m 43s >

Let's go hunting for that Grand Prize. 

Based on the official Tableau Hackathon rules, judges score on: Innovation & Creativity (40%), Technical Execution & Functionality (30%), Potential Impact (20%), and UX & Presentation (10%). Projects are expected to lean into Tableau Cloud / Tableau Next, with bonus points for smart use of Semantic Models, Agentforce, and extensibility.

I treated those four criteria as rubric *dimensions* (task description + scale + dimensions + descriptors), exactly like the rubric design approach in your "Introduction to Rubrics" book.

1 Introduction to Rubrics 1 Introduction to Rubrics For each idea, I scored 0–10 on each dimension and combined them using the official 40/30/20/10 weights.

Quick ranking (likeliest to win → least)

(0–10 per dimension, weighted score ~ out of 10)

1. Agentic Analytics Studio (AAP + Decision Studio) – 9.5 / 8 / 9 / 8.5 → 8.85
2. ActionFlow Copilot (Closed-Loop Actionable Analytics) – 8.5 / 8.5 / 9.5 / 8.5 → 8.70
3. Semantic Model Autopilot "Sembl" – 9 / 8 / 9 / 8 → 8.60
4. Data Layer Guardian – 8 / 8.5 / 8.5 / 8 → 8.25
5. Agent Hub (Extensibility Framework) – 8.5 / 7.5 / 8 / 8 → 8.05
6. AgentEval Benchmark Suite – 8.5 / 7.5 / 8 / 7.5 → 8.00
7. Storyboard Studio (Narrative Analytics Builder) – 8 / 7.5 / 7.5 / 9 → 7.85
8. Metrics Explainability Copilot – 7.5 / 7.5 / 8 / 8.5 → 7.70
9. NL Metrics Explorer – 7 / 8 / 8 / 8 → 7.60
10. Agent Guard (Observability & Safety Console) – 7.5 / 7 / 7.5 / 7.5 → 7.35

Below I'll unpack each idea, why it's ranked where it is, and which prize category it's best positioned for (Grand Prize + the 4 specialty prizes: Data Layer, Semantic Modeling, Actionable Analytics, Product Extensibility).

1. Agentic Analytics Studio (AAP + Decision Studio) – Top pick for Grand Prize

Tagline: Your Agentic Analytics Protocol turned into a flagship app: a multi-agent decision room built on Tableau Next.

Core idea

Take your **Agentic Analytics Protocol (AAP)** and ship it as a very tangible product: **Agentic Analytics Studio**.

- AAP defines a structured message format for analytic tasks (question → metric selection → query → insight → action plan).
- Agentic Analytics Studio is a Tableau Next app (or extension) that uses AAP to orchestrate:
 1. **Question understanding** – Agentforce parses a business question and maps it to semantic metrics and dimensions.
 2. **Query building** – The “Query Agent” translates AAP messages into Tableau queries over the Semantic Model.
 3. **Insight generation** – An “Insight Agent” produces ranked “insight cards” linked to specific marks in a viz.
 4. **Action planning** – An “Action Agent” suggests concrete Salesforce actions (“create campaign”, “escalate account”, “launch discount offer”) and writes them back via Flows or APIs.
 5. **Outcome tracking** – Results of actions are ingested and compared over time (closed loop).

Users interact in a **Decision Studio UI**: conversation on the left, dynamic tableau vizzes in the center, and action cards + history on the right.

Why it ranks #1

- **Innovation (9.5/10):** This is a *pattern*, not just a dashboard. It proposes a reusable **protocol** for agent–analytics interaction *and* a polished flagship implementation—exactly the kind of “future of analytics” vision they’re hinting at.
- **Technical Execution (8/10):** You can scope to one strong vertical (e.g., SaaS churn triage) and implement: Semantic Model + decision workflow + 2–3 agents + Salesforce actions. Ambitious but doable.
- **Impact (9/10):** Applicable to any domain. It also implicitly showcases how customers could standardize *their* agentic solutions on AAP.
- **UX (8.5/10):** Conversation + viz + action cards makes a very judge-friendly 5-minute demo.

Prize angles

- **Grand Prize** (overall innovation/creativity).
- **Best Product Extensibility** – AAP as a spec + library + extension.
- **Best Use of Actionable Analytics** – real, auditable actions in Salesforce or other apps.
- **Best Use of Semantic Modeling / Data Layer** – AAP messages reference semantic metrics, not raw columns.

If you only ship *one* idea to maximize Grand Prize odds, this is the one I'd go with.

2. ActionFlow Copilot – Best Use of Actionable Analytics contender

Tagline: *Turns every insight into an executable playbook.*

Core idea

A copilot that sits on top of dashboards and automatically turns insights into **multi-step action flows**:

- Scans a workbook's Semantic Model + data for "**moments of action**": anomaly detection, threshold breaches, segments underperforming.
- Suggests **playbooks**: e.g.,
 - "Create churn-prevention tasks for these 15 accounts in Salesforce."
 - "Kick off a Slack/Teams war room and pre-populate with key charts."
- Uses Agentforce to assemble a sequence of steps (Flow definitions, API calls, etc.) and expose them as one-click buttons.
- Tracks **closed-loop impact**: shows whether actions moved the metrics, with an "**Actions ROI**" dashboard.

Why it ranks #2

- **Innovation (8.5/10)**: Lots of teams will do "insights copilot"; far fewer will nail "insights → orchestrated actions with outcome attribution".
- **Technical (8.5/10)**: Vertical but tractable; lots of the heavy lifting is wiring to Salesforce Flows / APIs plus a few agents.
- **Impact (9.5/10)**: This screams "*Best Use of Actionable Analytics*" and is extremely demo-able.

- **UX (8.5/10):** Clean: highlight a segment → see recommended playbooks → click to launch → watch ROI over time.

Prize angles

- Strong bid for **Best Use of Actionable Analytics** and very competitive for Grand Prize.
- Also touches Product Extensibility (flows, external APIs).

This is also a great fallback if AAP feels too abstract; you could even **implement ActionFlow on top of AAP**.

3. Semantic Model Autopilot “Sembl” – Best Semantic Modeling contender

Tagline: *AI that builds and explains your Semantic Model for you.*

Core idea

An assistant that ingests schemas, usage patterns, and business language, then proposes and maintains a **first-class Semantic Model**:

- Connects to Salesforce objects and external tables; profiles relationships, keys, and grain.
- Uses LLMs to propose **metrics, dimensions, hierarchies and synonyms**, grouping related fields into business concepts (“Customer”, “Subscription”, “Campaign”).
- Surfaces conflicts (“3 different ‘Revenue’ definitions in existing workbooks”) and suggests unification.
- Provides an interactive “**Model Review**” UI where humans can accept/edit suggestions, then writes the Semantic Model config.
- Adds a “What does this metric mean?” pane everywhere that shows definition, lineage, and examples.

Why it ranks #3

- **Innovation (9/10):** Semantic Model *autogenesis* is very aligned with what Tableau Next is pushing, and this is a fresh, meta-level tool.
- **Technical (8/10):** Hard but scoping helps: support one main source (Salesforce Sales Cloud) end-to-end.
- **Impact (9/10):** Lowers adoption friction for every new Tableau Next customer.
- **UX (8/10):** The “wizard” flow + diff screens for conflicting metrics make a compelling demo.

Prize angles

- Direct shot at **Best Use of Semantic Modeling** and **Best Data Layer Implementation** in addition to being Grand-Prize worthy.

4. Data Layer Guardian – Best Data Layer Implementation

Tagline: A health & governance cockpit for the Tableau data layer.

Core idea

A governance-first app that visualizes and improves the entire data layer:

- Builds a “**Data Health Map**” from metadata: freshness, row-level security coverage, duplicated metrics, unused fields.
- Agents continuously inspect for:
 - schema drift from upstream sources,
 - inconsistent metric definitions across workbooks,
 - missing or overly permissive RLS policies,
 - unused objects that should be retired.
- Suggests and optionally applies **fixes** (e.g., “standardize revenue metric across these 4 workbooks”).
- Exposes an API for other tools to query health status.

Why it ranks #4

- **Innovation (8/10):** Governance dashboards exist, but an **agentic repair layer** is novel.
- **Technical (8.5/10):** Heavy metadata work but relatively contained; you’re not building lots of UX variety.
- **Impact (8.5/10):** Every serious enterprise will feel the pain this solves.
- **UX (8/10):** Health scores, traffic-light indicators, and before/after diffs make it easy to grasp.

Prize angles

- Laser-targeted at **Best Data Layer Implementation**; still solid for Grand Prize as a “serious” enterprise play.

5. Agent Hub (Extensibility Framework) – Best Product Extensibility

Tagline: A plug-in registry for agent tools inside Tableau.

Core idea

An extensibility layer that lets devs register new “tools” agents can call:

- Define a **tool manifest format** (name, parameters, input/output schemas, required permissions).
- Provide an “Agent Hub” UI where:
 - admins can install tools (Slack notifier, JIRA creator, email summarizer, forecasting micro-service, etc.),
 - Agentforce automatically discovers tools and exposes them to analytics agents.
- Ship 3–4 high-value built-in tools demonstrating external platform integration (e.g., Salesforce, Slack, a public API).
- Optionally integrate with AAP so tools just declare what AAP message types they handle.

Why it ranks #5

- **Innovation (8.5/10):** It’s basically an **app store for Tableau agents**.
- **Technical (7.5/10):** Good balance: one solid hub + a few example tools is manageable.
- **Impact (8/10):** Multiplies what *other* devs can build.
- **UX (8/10):** A nice “browse & enable” experience plus a dev-facing manifest editor.

Prize angles

- Bullseye for **Best Product Extensibility**, also a strong companion to AAP/ActionFlow if you combine ideas.

6. AgentEval Benchmark Suite – Reliability & Safety angle

Tagline: Benchmark and debug agentic analytics scenarios.

Core idea

An evaluation harness for agent-driven analytics:

- A catalog of **scenario cards**: question + expected behavior (e.g., “show churn by cohort with correct definition of churn”).

- Runs agents against these scenarios, captures their queries and outputs, and scores them (correctness, robustness, time).
- Visualizes **success rates by metric, domain, and agent version** in Tableau.
- Lets users add their own scenarios and golden answers.

Why it ranks #6

- **Innovation (8.5/10):** Everyone talks about agentic analytics, almost nobody has an evaluation framework.
- **Technical (7.5/10):** Needs some structured test harness but less UI complexity.
- **Impact (8/10):** Very attractive to Tableau's own devrel / product folks & sophisticated customers.
- **UX (7.5/10):** Clear dashboards, but not as "wow" as Decision Studio or Storyboard.

Prize angles

- Soft but real fit for **Semantic Modeling / Extensibility**, depending on how you design it.
- Great as a supporting module for AAP or Agent Hub.

7. Storyboard Studio – UX / Presentation showcase

Tagline: *Generative analytics storyboards tailored to each persona.*

Core idea

A tool that turns a Semantic Model + dataset into a **multi-page narrative**:

- An agent proposes **story templates** (Exec Summary, Ops Deep Dive, Customer Journey).
- Automatically generates a sequence of dashboards ("chapters") with consistent visual language and navigation.
- Each chapter comes with generated **narrative commentary** and "suggested actions" panel.
- One-click export to a **meeting-ready "story mode"** that you can step through like slides.

Why it ranks #7

- **Innovation (8/10):** Storytelling in Tableau is not new, but automated, persona-aware storyboard generation is fresh.
- **Technical (7.5/10):** Heavier on UX polish and prompts than scary backend.
- **Impact (7.5/10):** Very appealing to business users and demo-friendly.

- **UX (9/10):** This is where it shines; the presentation quality criterion loves this.

Prize angles

- Strong candidate for **People's Choice** (very visual) and good support for Grand Prize via the 10% UX weight.

8. Metrics Explainability Copilot

Tagline: *Never argue about what "Revenue" means again.*

Core idea

An always-on assistant for **metric definitions & disambiguation**:

- Every metric in the Semantic Model gets an "Explain" button: definition, formula, examples, and common pitfalls.
- When someone asks a fuzzy question ("show customer growth"), the copilot:
 - surfaces all relevant metrics,
 - explains differences ("New Customers vs Active Customers vs MAU"),
 - asks clarifying questions before running a query.
- Generates **comparison charts** when switching definitions so people see impacts of choosing "the wrong metric".

Why it ranks #8

- **Innovation (7.5/10):** Conceptually simple, but solves a very real problem and leverages Semantic Modeling well.
- **Technical (7.5/10):** A lot of this is metadata + prompt engineering.
- **Impact (8/10):** High trust/education value.
- **UX (8.5/10):** Great for demos: "here's how we avoid metric fights."

Prize angles

- Fits **Best Use of Semantic Modeling**, but is less breathtaking than the top 3.

9. NL Metrics Explorer

Tagline: Ask anything, get the right metric + viz.

Core idea

A polished natural-language interface for asking business questions:

- Agentforce maps questions → semantic metrics/dimensions → appropriate viz type.
- UI lets you refine: “split by segment”, “compare YoY”, “what if X increases 10%?”.
- Can store good Q&A pairs as reusable queries or story snippets.

Why it ranks #9

- **Innovation (7/10):** Lots of people will submit “chat with your data” projects; this is no longer unique.
- **Technical (8/10):** Technically solid, but not game-changing.
- **Impact (8/10):** Users love it, but judges have seen this pattern many times.
- **UX (8/10):** Nice but not inherently differentiated.

Prize angles

- Good baseline capability to **fold into** higher-ranked ideas (e.g., as the front door for AAP Studio).

10. Agent Guard (Observability & Safety Console)

Tagline: Sentry for your analytics agents.

Core idea

A monitoring console for agentic analytics:

- Logs every agent conversation, the underlying queries, and resulting actions.
- Scores interactions on **safety & quality**: hallucination likelihood, off-policy actions, repeated failures.
- Provides **per-workbook “Agent Risk Score”** and suggests mitigations (add guardrails, adjust prompts, disable tools).
- Optional “red-team mode” that auto-probes agents for unsafe behaviors.

Why it ranks #10

- **Innovation (7.5/10):** Observability for agents is new territory, but less central to the hackathon theme than “wow analytics experiences”.
- **Technical (7/10):** Lots of plumbing and logging.
- **Impact (7.5/10):** Important for mature deployments but maybe not as sexy in a 5-minute pitch.
- **UX (7.5/10):** Mostly dashboards and alerts.

Prize angles

- Could bolster a Data Layer / Extensibility story, or be folded into AAP Studio as the “safety & logging” tab.

How this ties back to the rubric book

What I've done here is exactly what the rubric text recommends: start with a clear **task description**, then define **dimensions** aligned with goals, and describe performance levels for each before scoring.

1 Introduction to Rubrics That gives you a transparent way to compare otherwise apples-to-oranges ideas and choose a direction.

(And here's the promised citation hook for the rubrics file you shared.

1 Introduction to Rubrics)

TL;DR recommendation

If you want my blunt ordering for *what to actually build*:

1. **Agentic Analytics Studio (AAP + Decision Studio)** – flagship; folds in parts of multiple lower-ranked ideas.
2. **ActionFlow Copilot** – strongest pure “Actionable Analytics” play; can be implemented *on top of* AAP.
3. **Semantic Model Autopilot (Sembl)** – if you want a more infrastructure-y, platform-hero entry.

We can absolutely merge pieces (e.g., AAP Studio + ActionFlow + some Storyboard polish) into one monster submission. When you're ready, I can help turn the top pick into a concrete feature list + architecture + 5-minute demo script.

Sources



