NUVEPRO'S

HACKATHON STORIES 2026

AI HACKATHON CHRONICLES

WHEN HUMANS AND AI BUILT TOGETHER

15 TEAMS. 24 HOURS. ENDLESS IDEAS.

A COLLECTIVE FROM TEAM NUVEPRO

AccountPulse – When Sales Brains Met GenAI

Team : Adarsh EA, Shashi Kiran, Priyadarshan Bhatt, Prasenjeet Maitra
Authored by : Adarsh EA

For the recent Nuvepro Hackathon at Clover Greens Resort,

I teamed up with Shashi, Prasenjeet, and Priyadarshan from the sales team. I had a few invites from other teams too, but I intentionally chose to work with them.

They might call themselves "not very technical," but what they bring to the table is something every product truly needs—clear ideas, real-world requirements, and sharp business context. Also, I must admit, the promises of on-demand beverages and shoulder massages sounded like solid motivation!

We spent a good amount of time brainstorming the idea, with Shashi Kiran owning the core concept. The problem statement was clear and relatable: sales teams spend countless hours manually researching edtech and enterprise companies in target regions, identifying the right stakeholders, and preparing for outreach. It's time-intensive, repetitive, and error-prone. We envisioned a GenAI-powered application that could automate web searches, build company profiles, extract services, discover relevant stakeholders via LinkedIn, and present a ranked list to make prospecting faster and smarter.

On Day 1, We named our project "AccountPulse," a name suggested by Shashi. The sales team walked me through their workflow in detail. Since sales wasn't exactly in my usual domain, it took some time to fully absorb the nuances.

PD's understanding of LinkedIn data extraction and prospecting strategies was particularly helpful in shaping the data-gathering approach. I converted all those discussions into a detailed requirement document using GenAI, shared it with the team, and refined it through multiple feedback cycles.

Along the way, I picked up quite a few sales jargons and strategies. Coming from an engineering background, this cross-functional exposure genuinely expanded my perspective.

Once the requirements were clear, I crafted a structured, phase-wise prompt strategy to generate the code more accurately. I used Kilo Code powered by Claude Opus, and I was fairly confident we could pull this off. True to their word, the team was incredibly supportive. Every 30 minutes, Prasenjeet and PD would check in to see if I needed anything. That kind of energy makes a huge difference during a hackathon.

By the end of Day 1, I had a working prototype with dummy data and walked the team through a demo. The workflow was accepted with a few tweaks, and I went to sleep feeling optimistic.

Day 2 was a different story.

Our presentation was initially slotted for noon, and by 9 a.m., I still didn't have the full implementation ready. That made me a bit anxious, especially when I realized over breakfast that many teams had already wrapped up their builds.

I rushed back and went into execution mode. I first integrated web search to fetch a list of companies and display them on the frontend, with caching to improve performance for repeated queries. Next, I implemented service extraction by scraping company websites—this part was relatively straightforward.

"The real challenge came when we needed to fetch stakeholders from LinkedIn based on specific roles within selected companies."

Accessing LinkedIn member details typically requires Navigation API subscription, which I didn't have.

For a moment, I thought this might block us.

Then I explored alternatives and realized that the search tool I was already using for web search could also be leveraged for LinkedIn searches.

Even with a free-tier limit of 200 searches, it was more than sufficient for a working demo. I fed this requirement into Kilo Code, the necessary modules were generated, and everything clicked into place.

By around 11:30, the complete workflow was functional. Meanwhile, the rest of the team was polishing the presentation.

When I demoed the final version to the team,

the smiles said it all. By the time our actual presentation slot came at 3:30 pm, we were confident and aligned. Although we didn't make it to the winning category, as a team we were genuinely satisfied with what we had built in such a short span.

More than anything, this hackathon reminded me how powerful cross-functional collaboration can be.

Engineering plus sales, backed by GenAI, can move incredibly fast when there's clarity, trust, and a lot of energy in the room.

"Truly an amazing experience"

Team ATOM8

Team : Akshay, Raju, Suprith, and Anushree
Authored by : Akshay

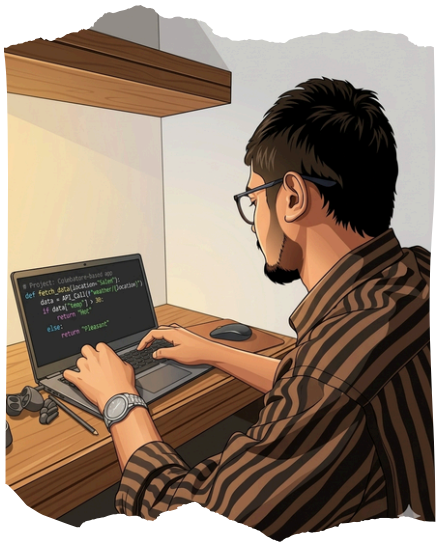
We were a team of four and we named our team ATOM8.

From the start, we were clear that we wanted to build something practical and useful, so we decided to work on a Validation Code Generator to simplify test case creation.

The hackathon was quite intense for us. We spent most of our time together in a room, working continuously day and night. There was a lot of brainstorming, coding, debugging, and refining happening all the time. Every small improvement felt like a big win, which kept us motivated.

What made the experience even better was the collaboration, not just within our team but also with others around us. We interacted with different teams, shared ideas, and picked up new perspectives.

"That environment really added to the overall learning experience"

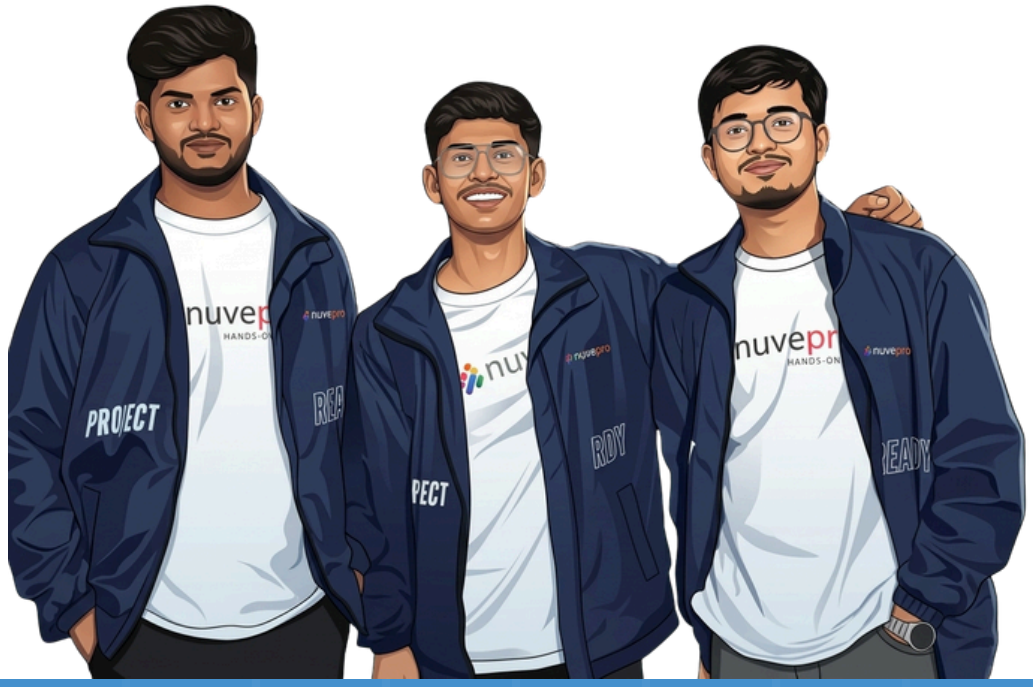


We did face a few challenges while building the solution,

Especially when it came to aligning everything and making sure it worked smoothly. But by dividing tasks and supporting each other, we managed to move forward without getting stuck for too long.

Overall, it was a great experience for all of us.

It wasn't just about the final solution, but the teamwork, late-night efforts, and the energy throughout the hackathon that made it memorable for Team ATOM8.



From Vision to Velocity: Our Team's Journey at the Hackathon

The Hackathon Portal

Team : Priya G, Arun Reddy, Prajwal S, Manoj B. R, Dileep Kumar Gundlapalli
Authored by : Priya G

The Setting: Inspiration Meets Innovation

On February 6th, our team arrived at the Clover Greens Golf Course and Resort.

While the lush greens and the calm atmosphere of the resort were inviting, we knew we weren't there for a vacation.

We were there to celebrate our company's anniversary by doing what we do best: solving problems.

The energy shifted the moment we walked into the party hall for the inauguration. Mr. Rajan's opening session set the stakes high. He spoke about AI moving beyond mere experimentation into the realm of "inference, performance, and scale." His words grounded us, reminding us that the "hype" doesn't matter unless you can build something that survives and scales in the real world. That became our North Star.

The Problem: The "Spreadsheet" Nightmare

As the hackathon officially kicked off after lunch, we looked at the event itself for inspiration.

We realized that managing a hackathon—tracking teams, ideas, and submissions, is often a chaotic mess of disorganized spreadsheets and lost emails.

"We decided to solve the very problem we were living in: The Hackathon Portal".

The Midnight Grind: Powered by AI

Our goal was to see how efficiently we could leverage the current AI boom to build a Minimum Viable Product (MVP) in record time. We didn't just "use" AI;

"we treated Google Gemini as our sixth team member"

While the resort grew quiet at night, our corner of the hall was buzzing. Our developers were sprinting through Java Spring Boot and React.js, while our cloud experts ensured our architecture was robust and secure. Having our COO, Arun Reddy, in the trenches with us was a game-changer; it kept our focus on the user journey and the business value, preventing us from getting lost in the "code weeds."

The Challenges & Breakthroughs

Working through the night wasn't without its hurdles. Integrating JWT-based security and managing real-time team invite codes under a tight deadline tested our patience. However, the synergy was electric. Whenever we hit a wall, we'd lean on AI-assisted debugging to pivot quickly.

By the time Mr. Kashi gave his AI talk on Day 2,
We weren't just listening to theories, we were living them. We had built a centralized, role-based platform that automated team formation, restricted sizes for better coordination, and provided admins with a real-time dashboard of event stats.

The Result: A Celebration of Scalability

As we headed into the Anniversary Celebrations and Awards Ceremony on the final evening, we felt a profound sense of accomplishment. We didn't just build an app; we proved that with the right mix of human expertise and AI assistance, we could cut development time in half without sacrificing quality.

This hackathon was a reminder that at our company, we don't just talk about the future of AI, we build it, one overnight sprint at a time.

Simplifying Complex Azure Policy Dependencies

Team : Karthik, Ashwija, Sudheer and Sirisha
Authored by : Ashwija

The Nuvepro anniversary this year felt different right from the start

There was a sense of excitement in the air—not just because it was a celebration, but because it came with a challenge. Instead of the usual events, this time we had a hackathon. Every team was asked to build something using AI, and that instantly set the tone. It wasn't just about participation; it was about creating something meaningful.

Our team—Ashwija, Karthik, Sirisha, and Sudheer

Came together around noon when the hackathon officially began. Like most teams, we started with a mix of enthusiasm and uncertainty. We had our topic in mind: enabling Azure services along with their dependent services. It sounded straightforward at first, but as we started breaking it down, we realized the depth of what we had taken on. Dependencies, configurations, permissions—everything had to align perfectly.

The first few hours were all about brainstorming and structuring our approach. There were moments of silence where each of us was thinking deeply, and then bursts of discussion where ideas flowed rapidly. Slowly, things started taking shape. We divided responsibilities but stayed closely connected, constantly checking in with each other.

As evening turned into night, the real test began

The energy shifted—not lower, but different. There was a quiet determination in the room. We were no longer just participating in a hackathon; we were fully invested in solving the problem. Coffee cups started piling up, screens stayed bright, and conversations became more focused.

Working through the night was both challenging and memorable. There were moments when things didn't work as expected—configurations failed, dependencies didn't align, and we had to retrace our steps. At times, it felt frustrating, but no one gave up. Instead, we supported each other. If one person got stuck, another jumped in. If someone felt tired, someone else brought the energy back.

What stood out the most was the teamwork

There was no pressure, no blame—just collaboration. We celebrated small wins, like getting a service to work correctly or resolving a tricky issue. Those small moments kept us going.

One of the most interesting and challenging parts of our project was enabling Azure services along with their dependent services. At first, we assumed it would be a simple enablement process, but as we went deeper, we realized how interconnected everything was.

Each Azure service relies on multiple underlying components—network configurations, identity and access management, storage dependencies, and sometimes even region-specific constraints.

We had to carefully map out these dependencies to ensure that when a primary service was enabled, all its required supporting services were also configured correctly.

For example, enabling a compute service wasn't just about turning it on—it required proper networking setup, role assignments, and storage configurations to function seamlessly.

There were moments when a service would fail to start, We had to trace back through layers of dependencies to identify what was missing. This process taught us the importance of understanding cloud architecture beyond the surface level. We also explored ways to automate this dependency handling, so that users wouldn't have to manually configure each component.

As we progressed, we started building logic that could intelligently identify required dependencies and enable them in the correct sequence. This not only improved efficiency but also reduced the chances of errors. It was satisfying to see how something complex could be simplified through the right approach.

By early morning, exhaustion started to show, but so did satisfaction. We had built something we were proud of. It wasn't just about completing the task; it was about the journey we went through together. The late-night discussions, the problem-solving, the laughter in between—it all became part of the experience.

Looking back, the hackathon wasn't just an event. It was a reminder of what a team can achieve when everyone is aligned and motivated. It brought out creativity, resilience, and collaboration in the best way possible.

"This anniversary celebration will definitely be remembered—not just for what we built, but for how we built it, together".

Git Squeaky

Team : Manjunath Kulkarni, Ujwala, Gaurav Kanaujiya, Chandra Prakash
Authored by : Chandra Prakash

Team Roles

- Idea Generation: Manjunath Kulkarni
- Environment Configuration & Testing: Ujwala
- Development, Bug Fixes, & Workflow Validation: Gaurav Kanaujiya
- Project Manager (Discussion, Design, Test Scenarios, & Presentation): Chandra Prakash

Our team was driven by a high level of enthusiasm to make this project fully functional, bug-free, and capable of covering all use cases. Each member brought a unique perspective to their tasks, including implementation, coding, testing, and design guidance.

We began our discussions as soon as the group was formed.

Over lunch, we started building our prototype and workflow, identifying the necessary tools and platforms. A major challenge was developing an AI-based solution, as no one on the team had prior experience with AI. Ultimately, we decided to work with Gemini AI.

By tea time, we had prototyped a webpage with several core functions. To keep our minds fresh during the intense development, design changes, and testing phases, we played chess as a mental break. Before dinner, Ujwala and Manjunath Kulkarni conducted two rounds of rigorous testing and successfully identified several bugs. Gaurav Kanaujiya did a quick bug fixes and ready for another round of testing.

There were moments when a service would fail to start, We had to trace back through layers of dependencies to identify what was missing. This process taught us the importance of understanding cloud architecture beyond the surface level. We also explored ways to automate this dependency handling, so that users wouldn't have to manually configure each component.

After dinner, we met with other teams to exchange thoughts. We were determined to achieve our goal and provide stiff competition; unlike other teams that focused only on concepts, we were committed to a live demonstration. The team worked late into the night, finally calling it a day at 2:00 AM.

The next morning, we brought our laptops to breakfast for an end-to-end prototype demo and found that the workflow was perfect. By this time, the team had jelled exceptionally well, and a strong bond had developed between us. We prepared our PowerPoint presentation and synchronized it with our live demo workflow.

"The final presentation went very well"

The team was thrilled with the successful live demo, which we concluded exactly at the 2-minute mark.

The Night That Built More Than Just Code

Team : Amit Yadav, Dharmender, Naveen
Authored by : Dharmender

It started with team selection and project ideas being thrown around

Everyone trying to find something that clicks. We landed on ours. An MCP server for NuveLink. Simple on the surface. Not so simple once we actually started building it.

The goal was clear.

We wanted to build an MCP that could create and manage multiple labs for each user—something that could actually be used in a real platform.

"Not just creation. But full control.Start. Stop. Delete. Configure. Everything"

Teams formed. Roles got divided.

And just like that—we were in. I teamed up with Amit and Naveen. Naveen focused on building the APIs. Amit took charge of the frontend. And I worked on the MCP backend—creating APIs and making sure everything connected and actually worked together.

At first, things felt manageable.

Then the bugs showed up.

There was this chatbot-related flow I had worked on—something that should have worked. It didn't. Something was off. Responses weren't consistent. Behavior wasn't reliable. We kept trying to fix it. Again. And again.

Somewhere in between all this

Dinner time happened. And for a moment, everything paused. I stepped out and saw something interesting–People walking out of their rooms one by one. Same tired faces. Same stretched backs. Same "we've been grinding for hours" look. No one really said it out loud...But you could feel it. Everyone was in the same fight. Different projects. Same pressure. And then–just like that–Everyone went back in. And the night? It disappeared.

While others slowed down, we kept going. Debugging. Fixing. Testing. Rebuilding. Laughing at bugs. Celebrating small fixes like they were major wins. That strange mix of exhaustion and excitement that only shows up during hackathons.

Morning came faster than expected.

Presentation day. We were sitting at the end. And even then–Amit was still working. Not small tweaks. Actual improvements. New features. Right until the last moment.

Then our turn came. We presented our MCP for NuveLink–A system that enables users to create multiple labs, manage them efficiently, and perform key actions like starting, stopping, deleting, and configuring labs seamlessly.

"Not just an idea. A working system".

We answered every question.

Explained our decisions. Walked them through the architecture. But in the end–We didn't make it to the top 3. And honestly? It didn't feel like a loss.

Because what we built overnight–The pressure we handled, The system we designed, The problems we solved–That stayed.

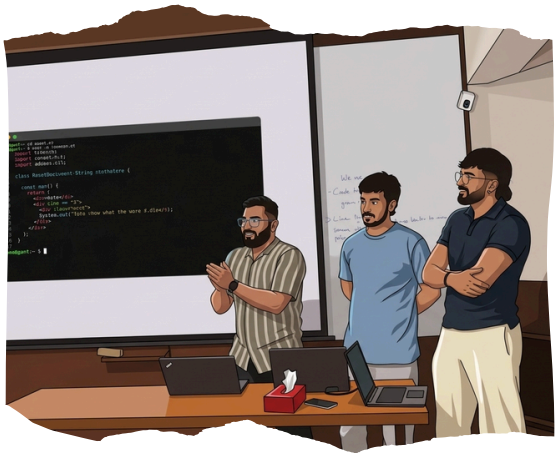
It wasn't just about rankings. It was about building something real, under pressure, with a team that didn't quit. One night. One MCP. A lot of bugs. A lot of learning. And a reminder–Sometimes the real win isn't the result.

"It's the version of you that comes out after the grind".

From Idea to Automation: Building an AI-Powered Sandbox Setup

Team : Darshan, Sai Lokesh, Dhruvakumar D L
Authored by : Dhruvakumar D L

It started as an unexpected twist,



A hackathon announced right in the middle of what was supposed to be a quiet break. Within a short time, that curiosity turned into genuine excitement.

The problem statement was one we all understood well—reducing the time it takes to build labs at Nuvepro. Setting up environments had always been a detailed, manual process: installing tools, configuring dependencies, tweaking setups based on client needs. It worked, but it wasn't fast, and it certainly wasn't scalable.

Somewhere in the early discussions, an idea emerged that immediately caught everyone's attention. "What if an AI could do this for us?"

That question set everything in motion.

The goal became clear

To build a system where an AI model could interpret client requirements and handle installations, configurations, and environment setup automatically. Instead of manually executing every step, we wanted the model to understand intent and act on it.

> "From that point on, it didn't feel like work—it felt like building something exciting".

The development phase turned into a continuous cycle of brainstorming and experimentation. Darshan, Sai, and I spent hours breaking down the problem, exploring approaches, and refining the idea. Every discussion opened up new possibilities, and every small success pushed us further.

Of course, no such journey is complete without its share of challenges.

There were moments when things didn't behave as expected—responses from the model were off, configurations didn't execute correctly, and flows had to be redesigned. But instead of slowing us down, these challenges made the process more engaging. Each problem solved felt like unlocking another piece of the system.

And then there was Moyukh.

At some point, we lost count of how many times we knocked on his door. Questions, edge cases, validations—we brought everything to him. If persistence was part of the process, we certainly leaned into it. To his credit, he stayed patient through it all, helping us navigate the tricky parts and keeping us grounded when things got complex.

Then came the moment that made it all worthwhile.

A response from the LLM. Clean. Correct. Exactly what we needed. We ran the configuration. It worked.

There was a brief pause as it sank in, followed by that unmistakable "hurray" moment. The system we had imagined was now real-capable of taking a client's request and turning it into a working environment.

What began as an unexpected hackathon had turned into an experience we genuinely enjoyed—building, learning, solving, and pushing boundaries as a team.

That night, after everything finally worked, we shut our laptops with a sense of satisfaction that only comes from solving something meaningful.

And then, for the first time in days,

We slept—completely at peace with what we had built.

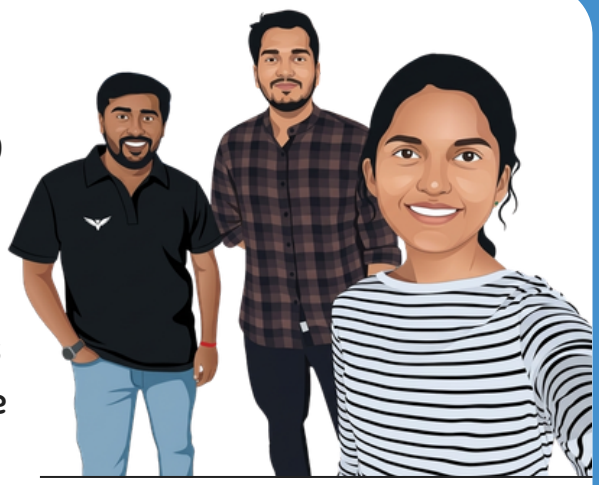
AWS Management Hub

Team : Harsha K, Kunal Srivastava, Kavyashri K R
Authored by : Kavyashri K R

The Beginning

When the announcement came that Nuvepro Technologies was hosting a company-wide internal hackathon, the energy was immediate. For us, it wasn't just a competition – it was an opportunity to build something meaningful, work under pressure, and see what we were capable of as a team. We were in from the moment we heard about it.

Meeting the Team



When we looked around, teams were still forming and ideas were being explored. We took that as our cue. We connected with several colleagues, had genuine conversations, and found people who matched our enthusiasm for building. That's what we were looking for – teammates who were as fired up about the hackathon as we were.

We came together quickly and started discussing what we'd build and how. Even when one teammate was swapped in, the transition was smooth – the new addition brought fresh energy and fit right in.

The Problem We Wanted to Solve

Our team had two compelling ideas on the table.

The first was an AI-powered system that analyzes customer support tickets.

The second was an AI-powered unified multi-cloud platform – a single source of truth spanning AWS, Azure, and GCP environments. Both were ambitious and genuinely useful.

Building the Idea

We threw ourselves into brainstorming mode. With laptops open and notebooks filling up with diagrams and ideas, the cafeteria became our war room. The team was wonderfully collaborative – asking what resources were needed, offering API keys and subscriptions, and making sure everyone had what they needed to build.

Even when one of us stepped away to attend a regular AI Cohort session, the rest kept the momentum going. The brainstorming never stopped, and neither did the building.

The Chaos Phase

With two strong ideas in hand – and more surfacing from our sessions – we moved fast, pivoted often, and explored a lot of directions at once. Context-switching is never easy, but in hindsight, it was part of our learning. We were building something real under pressure, and that kind of experience is hard to replicate.

We barely slept. But we kept going. And we came out of it with a much sharper sense of how to align quickly, commit to a direction, and execute.

The Final Stretch

We converged on our final idea: the AWS Management Hub – a platform designed to give teams clear, centralized control over their cloud infrastructure. We divided the demo responsibilities cleanly: one of us would open with an introduction, another would walk through the live product demo, and the third would close with a strong conclusion.

Demo Day

Demo day had its own energy. We signed up early, which gave us a great slot. When the time came, we got up, walked the panel through what we'd built, and delivered a clear, confident demo. The AWS Management Hub spoke for itself.

The day had a few unexpected turns – but we handled them with patience, adapted without losing composure, and kept our focus on what mattered: showing the panel a product we were proud of.

That experience reminded us how important it is to stay grounded and empathetic when things move fast – and to understand that everyone on the team is doing their best.

What We Took Away

Hackathons teach you things no classroom can. Here's what stayed with us:

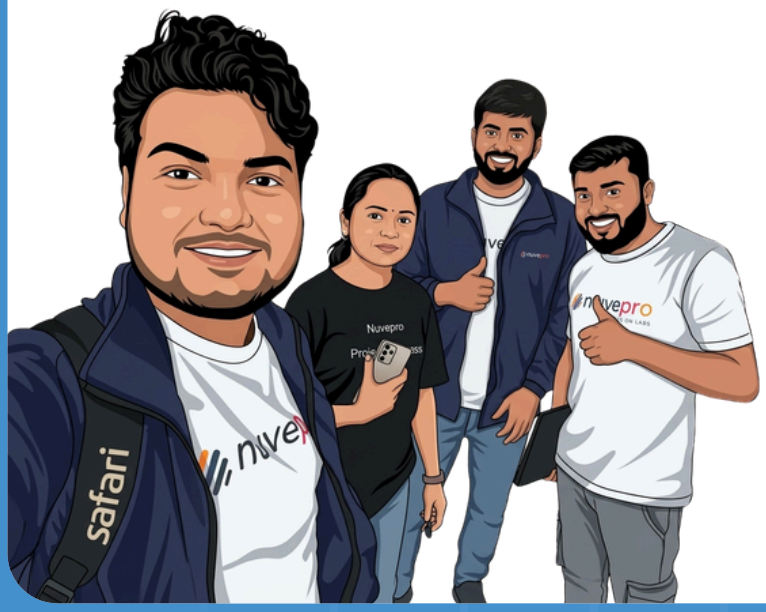
- Commit to a plan before you build – clarity of direction multiplies your output.
- Late-night building under a deadline is its own kind of fuel – stressful, yes, but genuinely fun.
- Patience and empathy aren't soft skills – they're what keep a team together when pressure peaks.

Looking Forward

The hackathon planted a seed. One of the ideas we explored – building automation for the support team – is something we're genuinely excited to take further. We walked away not just with a demo, but with a clearer sense of what we want to build next, and who we want to build it with.

When Bugs, Coffee & Chaos Built an AI Idea

Team : Kalavathi, Rishab, Mahadev, and Rudra
Authored by : Kalavathi



It all began at the very last moment.

While most teams were already formed and racing ahead, our team came together just in time.

Meet our squad – Kalavathi, Rishab, Mahadev, and Rudra – four minds, one unexpected team, and zero time to overthink.

The Start

While every other team jumped straight into coding, we did something... unusual. We paused. We grabbed coffee, We went for a cycle ride, We laughed, relaxed, and reset our minds And only at 7 PM, when most teams were already deep into development, we just began.

Brainstorming Under Mosquito Attack

We gathered, energized and focused. What started as a casual brainstorming session soon turned into an intense deep dive. Ideas flew, laughter echoed–and somehow, mosquito bites became the spark for our concept: AI Bug-Based Learning. Time disappeared.

We were so immersed that we completely forgot dinner – eventually becoming the last team to eat.

Midnight Hustle (Almost!)

Post dinner, the real grind began. Rishab and I jumped into development—building, breaking, retrying. Flow after flow, idea after idea. It wasn't perfect, but it was evolving.

Meanwhile, Mahadev and Rudra briefly disappeared for a "drink break" (because why not).

We waited... and waited... until finally, around 12:30 PM – midnight, we called it a day.

The Comeback Morning

Next morning – fresh start from all four of us. Breakfast done, Laptops open, Focus mode: ON And something amazing happened...

Within just one hour, everything clicked:

"The application flow | The presentation points | The confidence"

Pressure builds teams. Moments define them

Presentations began at 12 PM. And guess what? We were the last team to present. With a mix of excitement and nervous smiles, we stepped forward... and delivered.

"Not perfect. Not polished. But real. Feedback received. Mission complete".

What We Truly Won

Did we win the hackathon? That's not the point.
Because what we gained was bigger:

- We learned how AI can be explored creatively.
- We built, learned, laughed—and delivered.
- And most importantly... we grew as a team.

Final Thought

sometimes the best ideas come between coffee breaks, cycle rides, and mosquito bites.

Perfection is overrated, learning is not.

Hackathons aren't just about winning – they're about growing.

In the end, this hackathon wasn't just about building an application – it was about discovering how we work, fail, adapt, and rise together. From last-minute beginnings to last-stage presentations, from coffee breaks to code breaks, every moment shaped our journey. We didn't chase perfection; we embraced the process, the bugs, the chaos, and the laughter. And somewhere between mosquito bites and midnight pauses, we found something far more valuable than just a solution – we found confidence, connection, and a story we'll carry long after the hackathon ends.

Not the fastest start. Not the smoothest journey.

But definitely a memorable one.

A Night of Bugs, Birthdays, and Breakthroughs

Team : Thanushree, Akshay Parihar, Manasa
Authored by : Manasa

It started in the evening

Ideas being pitched, teams forming, and just enough confidence to think we had it under control. We didn't. Our idea got shortlisted.

"Two teams, one room"

Different ideas—but the same pressure, the same sleepless grind.

We started with a simple but frustrating truth:

GenAI teams move fast, but when it comes to using research papers in real engineering decisions, most just don't have the time. So they guess, experiment, and often end up dealing with hallucinations, instability, and wasted effort.

We wanted to fix that.

Our idea was ambitious

A system that reads a GenAI codebase, detects what's being used—RAG, agents, prompting—and then connects it to relevant research. Not just that, but explains in plain language what might break, why it breaks, and how to fix it.

"We validated the idea, got the green signal, and got to work".

Akshay and Thanushree went deep into the backend. I took charge of the frontend—while also juggling helping other teams set up tools. It was chaotic, but that's what hackathons are. "By night, the real grind began".

Somewhere in between debugging and building,

We realized it was Akshay's birthday. So naturally, we went on a late-night cake hunt. After running around and realizing brownies were gone (tragic), we managed to secure one big slice of chocolate cake. At midnight, we pulled him out with a fake excuse and surprised him.

Next thing he knows—cake, chaos, and birthday bumps. He survived. Barely. Then, straight back to work.

By 1 AM, Most people were asleep. Only a few of us were still pushing through. And then came the loop.

Every 30 minutes, Thanushree's alarm would go off. She'd wake up, clearly exhausted and already dealing with a headache, look at me, and ask: "Bro... ayta?"

My answer stayed the same: "Nope."

She'd go back to sleep. Alarm again. Same question. Same answer. Meanwhile, the headache? Only getting worse—but the commitment to waking up every 30 minutes somehow stayed strong.

By the third or fourth round, this stopped being a conversation and started becoming a pattern.

Me and Surendra were watching this like it was a live system. Same alarm. Same movement. Same question. Same answer.

At some point,

Surendra—who wasn't even part of this loop initially—got so frustrated with the repetition that he ended up with a headache too. Not from coding. Not from lack of sleep. From Thanushree's 30-minute question cycle. And that's when it clicked. Next time the alarm went off, we were ready. We just looked at her, waiting. She just started to move- And before she could even ask, both of us said: "Not done."

She paused for a second... processed it... and just went back to sleep. No question. No confirmation. Just acceptance.

At that point, we weren't just developers anymore. We had basically become a prediction system—responding before the input even arrived.

This continued till 5 AM. At that point, it wasn't just discipline It was a full system running on pure stubbornness. At some point, exhaustion hit all of us. We crashed for a bit, surrounded by a chorus of perfectly out-of-sync snoring that honestly could've been turned into music. But the real tension came later.

While upgrading the backend,

The Claude tool we were relying on basically said, "I'm done helping you." Perfect timing. That was our moment of uncertainty. No shortcuts now—just us, low on sleep, slightly frustrated, and completely out of backup plans. And we still had to make it work.

So we did. We pushed through, fixed what mattered, and somehow got everything working. When it was time to present, we didn't just show a product—we showed a solution to a real, painful problem. Something teams actually need.

The judges saw it. Both teams—working side by side in that same room—ended up securing 3rd place. Not bad for a night powered by one slice of cake, 30-minute alarms, growing headaches, broken tools... and

"A surprisingly accurate human prediction system"

AI Smart Coach for Moodle

Team : Manjunath A, Anand , Nagaraj
Authored by : Manjunath A

Initially, we were unsure whether our idea would be shortlisted, so it came as a surprise when it was selected. The next major challenge was implementing the solution within just one and a half days.

The journey from start to finish was the most memorable part

Balancing tight deadlines with the excitement of building something meaningful. On one hand, we were racing against time; on the other, we wanted to enjoy the process. We managed to do both.

We started by breaking down the problem statement into smaller, achievable tasks and distributed them among the team.

As tasks began getting completed within hours, we realized how effectively AI tools could accelerate development and turn ideas into working solutions.

Challenges we faced were:

- Strict time constraint (1.5 days to build the solution)
- Limited knowledge in building chatbots initially
- Less experience with frontend technologies
- Limited understanding of vector databases

Solution:

We built a specialized AI assistant to enable a seamless learning experience. The goal was to design an ethical, course-specific AI Smart Coach (chatbot) that helps learners quickly find relevant information from course materials within the Moodle platform.

"We broke down the problem statement into phases and then used different tools to generate the solution".

Tools & Tech Stack Used:

We leveraged advanced AI tools such as Claude, GitHub Copilot, Gemini, and Antigravity to accelerate development and improve solution quality.

Frontend: Next.js | Backend: Python (FastAPI) |
Architecture: RAG (Retrieval-Augmented Generation using Pinecone)

Contributors:

Manjunath (Backend & AI)
Worked on building the chatbot logic and APIs using FastAPI. Implemented the RAG pipeline and integrated Pinecone to store and retrieve embeddings. Focused on making the chatbot give accurate, context-based responses.

Nagaraj (Frontend)
Developed the user interface using Next.js. Designed a clean and user-friendly layout and connected the frontend with backend APIs for smooth interaction.

Anand (Integration & Presentation)
Handled integration between frontend and backend to ensure everything worked end-to-end. Also worked on the presentation, project flow, and final pitch.

Behind the Scenes – Our AI Hackathon Journey

Team : Vaishnavu C V & Manoj K S
Authored by : Manoj K S

What started as a simple idea

Quickly turned into one of the most intense, exciting, and rewarding experiences for our team.

For this AI Hackathon,

"Our goal was clear: Automate test case creation
for the Moodle LMS using PlayWright MCP".

It sounded straightforward at first, but as we began breaking it down, we realized the depth of the challenge. Generating meaningful, structured, and reliable test cases using AI required not just technical skill, but also clarity of thought and constant iteration.

From the very beginning,

The energy within the team was high. There was a strong sense of curiosity and determination. We spent hours discussing different approaches – how to structure the inputs, how to make the outputs useful, and how to ensure the solution actually solves a real problem. Every discussion opened up new possibilities, and at the center of it all was Vaishnavu.

Vaishnavu truly was the driving force – the one who took ownership of the idea and pushed it forward with clarity and confidence. Watching the way solutions were identified, broken down, and implemented was inspiring.

It wasn't just about coding; it was about thinking through the problem deeply and building something meaningful.

The most memorable part of the journey

Was the night before the hackathon submission. We were all in deep discussion until around 1:00 AM, refining ideas, questioning assumptions, and aligning on the approach. There was no sense of fatigue – just focus and excitement. Even after we paused for the night, the momentum didn't stop.

At around 3:00 AM, while most of us were resting, Vaishnavu woke up and got back to work – quietly turning all those discussions into reality. That phase was where everything came together. Piece by piece, the entire solution started taking shape – from concept to execution.

By the time the hackathon began,

We had a working end-to-end solution ready. Seeing it all come together after hours of discussion, effort, and persistence was incredibly satisfying. It wasn't just about completing a project – it was about the journey of building it.

The challenges we faced were real:

- Translating a conceptual idea into a working AI-driven system
- Ensuring the generated test cases were practical and usable
- Managing time effectively under pressure

But each challenge pushed us to think better, collaborate more, and move faster.

Overall, the experience was nothing short of amazing. It was a perfect blend of problem-solving, and passion. The late-night discussions, the early morning execution, and the shared excitement made this hackathon truly special for us.

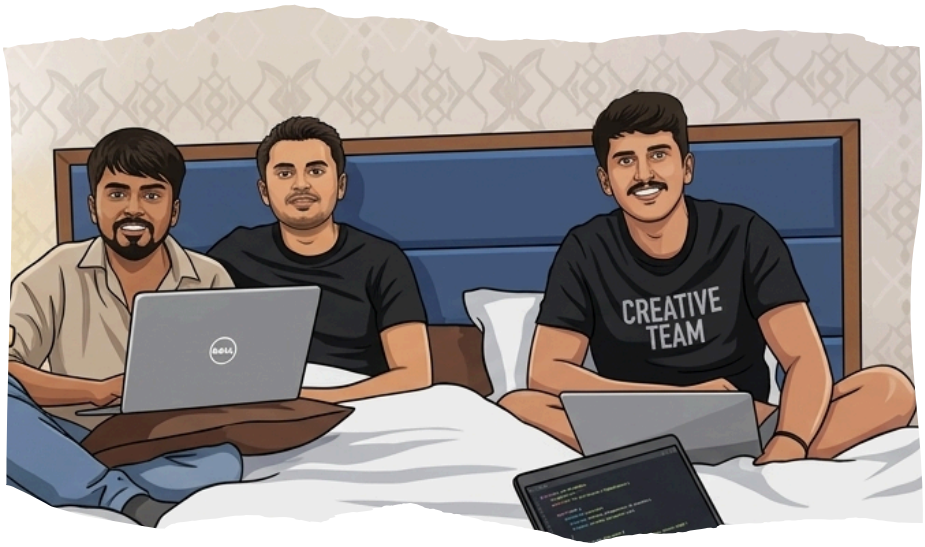
In the end, what we built mattered – but what we experienced as a team mattered even more.

The Outing Hackathon: Tokens, Stars, and Splashes

Team : Lithin, Vishal, Syed Noor and Arun V
Authored by : Noor

In the glow of laptops under starry golf course skies,

Team Lithin's "outing" hackathon blurred the lines between work and play. We weren't trying to boil the ocean or solve a massive company-wide crisis; instead, we targeted a specific, persistent pain point for our sales team. Our lightbulb moment was clear: build an AI-driven automation tool for cost monitoring to make their lives easier.



What started as a relaxing escape

Quickly turned into a puzzle-solving marathon as we divided and conquered. Lithin laid down the foundation by setting up the infrastructure and data sources. Noor played the crucial role of gatekeeper, carefully wrangling our strict $100 budget cap on API keys. Vishal dove deep into the heavy lifting of data handling and coding, while I took on the UI and automation to tie the whole system together.

But it wasn't a smooth ride. After a full day of travel and soaking in AI seminars, we were officially in "enjoy mode."

With a party vibe running in the backend of our brains,

we kept hitting walls where our tired minds just refused to tap into their usual debugging talent. When Vishal's brain finally "switched off," I tried to rally us to untangle the code, but the exhaustion was real. Even with the hotel AC humming in the background, the shared josh was being tested by stubborn bugs.

Day 2 brought demo chaos, and we needed a hard reset. "Demo can wait.... we're swimming!" we declared. Poolside, water dripping, the stress melted away. We rebooted, laughed off the presentation slides. That quick rebellion cleared the fog entirely.

We traded the pool for a quick shower

Walked into the presentation completely refreshed. Vishal took the stage for the demo and absolutely nailed it. He flawlessly fielded the judges' questions, showing off a working cost monitoring solution that did exactly what we set out to do. The judges loved it so much that their main feedback was a push to take it further and integrate the solution with the finance team.

> "These moments shine brightest off-stage: dim-room focus, unspoken teamwork, and that collective sigh when the code finally clicks".

This event remind us that innovation thrives not just in the spotlight, but in stolen swims, starry code sessions, and the thrill of cracking a problem together.

From Not Knowing Where to Start to Winning Third Place: THE STORY OF NUVECOACH

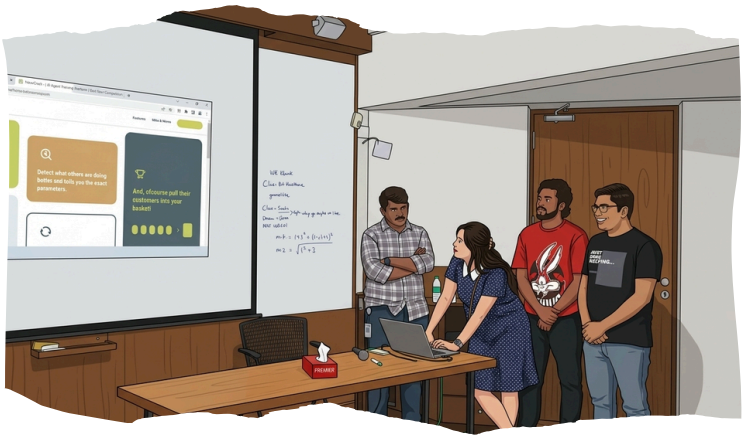
Team : Surendra, Harsh, Kevin, Shavez and Shivpriya

Authored by : Shivpriya

Going into the hackathon,

I thought we were prepared. I already had a marketing idea in mind something structured and understood, something I knew I could execute if I have a few technically aware colleagues along with me who can assist me in this direction.

But the moment the teams were announced, everything changed. We had an unexpected, intense conversation with Kashi KS and Giridhar. Kashi introduced us to an idea of an agent that sits on the top of the agent and that completely shifted our direction.



He gave us a new idea which we named as NuveCoach. Not another marketing tool, but something far more ambitious: a system that could

"Train marketing agents to perform better over time".

And while the idea sounded exciting, in that moment, it honestly felt overwhelming.

We looked at each other thinking:

How do we even begin to build something like this? That was our biggest challenge. We had no ready plan anymore, our idea was complex, unfamiliar, and the timeline was unforgiving. We were expected to build a prototype, validate it, and present it: all within hours. At one point, we were genuinely close to giving up and just calling it a night.

But somewhere in the middle of that chaos, things shifted. I had a long conversation with Rajesh and Moyukh, and it helped me reframe everything. There was no harm in attempting something new. We didn't need all the answers, but I simply wanted to start. That one thought changed how we approached the night.

"The First Win: We stopped overthinking and started building"

Surendra took charge of bringing the agent to life, translating a vague concept into something tangible. Harsh worked on creating a marketing agent that could simulate real campaign performance using metrics like CTR, CPC, and ROAS. Kevin ensured that everything we built could be clearly seen and understood through design and presentation. Shavez helped us with the hosting which helped to train chat bot agent. And I focused on making the idea real beyond just us.

Once we had something working,

I reached out to a few marketer friends and asked them to test it. Their response was immediate: curiosity, excitement, and a genuine interest in the idea of an "agent trainer."

It told us we weren't just building for a hackathon, we were building something that could make sense in the real world if we build it as an offering of Nuvepro.

What we built was NuveCoach.

A system where marketing agents don't just execute campaigns, they learn from them. It observes competitor campaigns, benchmarks performance, and continuously improves outcomes like CTR, CPC, and ROAS.

"Almost like a personal trainer, but for your ad agents".

Somewhere around 3 or 4 in the morning,

The intensity of the night gave way to something else. Exhaustion turned into laughter. We went from being completely stressed and unsure... to laughing at the most random things. Harsh even started walking around, checking in on other teams, which quickly turned into these spontaneous "status check + comedy sessions.", I remember Kevin started playing songs, Surendra started bringing in some chai.

It was chaotic, sleep-deprived, and completely unfiltered but it brought all of us closer and made us more confident.

And when I look back at the experience, this is what stands out the most:

- We started off completely unsure.
- We almost gave up before we even began.
- We chose an idea that felt bigger than what we could handle.

And yet..."We walked away with the 3rd place".

That night made me believe yet again, it's not important to think about the result, what's more important is if you try for it or not.

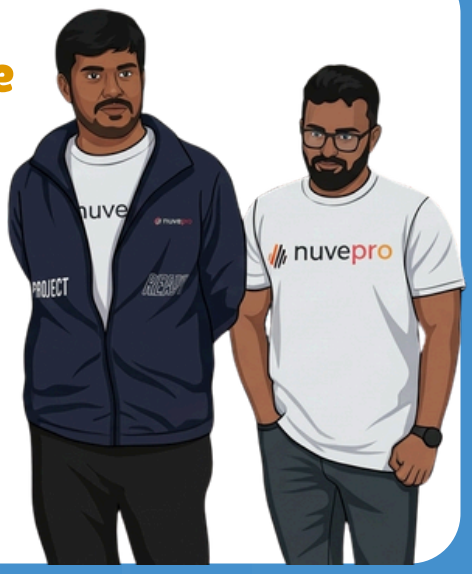
"NuveCoach wasn't just built that night. It was figured out.
And that's what made it truly worth it".

From a Simple Question to First Place

Team : Siva, Santosh & Sagar

Authored by : Siva

Our hackathon journey began even before the event itself



With a moment of curiosity after seeing the hackathon announcement email. Instead of immediately thinking about building something new, I spent time reflecting on a simple question:

Are we fully using the data we already have?

That thought led to an interesting realization

We had a significant amount of clean and consistent usage data, yet we had never deeply analyzed it to derive predictive insights. This sparked an idea using usage patterns to identify potential customer churn risk.

The concept felt practical and closely connected to real business value, so I submitted it, without expecting it to be shortlisted.

As the hackathon approached, I assumed I would eventually join another team if their ideas were selected. So, it came as a surprise when, on the day of the event, my idea was announced as one of the finalists but with one challenge: I didn't yet have a team.

Since the format required three members, things came together quickly. Santosh volunteered to join, bringing enthusiasm and support, and shortly after, the hackathon panel assigned Sagar to complete our team.

"What started as an individual idea suddenly became a collaborative journey".

Once the hackathon began,

We focused first on clearly defining the problem and shaping how we wanted AI to approach it. A significant portion of our time went into refining prompts and experimenting with different ways to structure the solution. This phase helped us align our thinking and ensured we were solving the right problem rather than rushing into development.

Using Claude Code, we translated our approach into a working prototype. The speed at which AI enabled iteration was impressive, but we quickly realized that demonstrating real impact required meaningful data. To make the solution tangible, we incorporated six months of historical usage data into the model, allowing us to showcase how churn insights could be generated from actual patterns.

With the prototype ready, we presented our solution to the panel and waited for the results. For us, completing the journey itself already felt rewarding collaborating as a newly formed team, experimenting with AI-driven development, and turning an idea into a functional demonstration within a short time.

During the results announcement, as third and second places were revealed without our team being called, we assumed our journey had ended there. Then came the final announcement. It took a moment to register what had happened, only after seeing everyone around us reacting did we realize that our team had been announced as the "first-place winner".

The experience reinforced an important lesson:

Innovation doesn't always begin with complex ideas; sometimes it starts with asking simple questions and exploring familiar problems from a new perspective. The hackathon showed us how collaboration, clarity of thought, and the effective use of AI tools can accelerate experimentation and bring ideas to life quickly.

More than the win itself,

The journey highlighted the value of curiosity, teamwork, and learning through exploration ,making the entire experience both memorable and meaningful.