Hackathon Challenge 1: Event Creator's Ticketing System

The Problem

Right now, using NFT tickets is way too hard for 99% of people. Event organizers want counterfeit-proof, collectible tickets, but they don't know how to write a smart contract. They're stuck with old platforms that charge high fees and still have issues with scalping and fraud. Web3 feels locked away from them.

Your 24-Hour Mission:

Your challenge is to build the tool they actually need. Create a simple, no-code platform that lets any creator launch their own NFT ticket sale in minutes. You're not just building one ticket; you're building the factory that lets anyone build their own.

Required MVP

- **The "Event Creator" Dashboard:** A super simple, friendly webpage. The host connects their wallet, fills out a form ("Event Name," "Price," "Total Tickets"), and maybe uploads a poster.
- **The "Launch" Button:** When the host clicks "Launch," your platform works in the background to deploy a brand-new, secure ERC-721 contract just for their event.
- **The Shareable Link:** The platform instantly generates a beautiful, simple link (like your-app.com/food-fest-2025). Attendees go there, connect their wallets, and mint their tickets.
- **The "Digital Bouncer" App:** A separate, simple "gate" page for the event staff. They just enter an attendee's wallet address and get a big, clear "GREEN - Welcome!" or "RED - Sorry!"

Bonus Goal

- Give the host a simple snippet of code (`<iframe>` or similar) they can copy-paste, letting them put the "Buy Ticket" button directly on their own website.

Hackathon Challenge 2: The "Web3 Loyalty Card" for Businesses

The Problem

Paper punch cards are a pain. They get lost, feel cheap, and nobody likes carrying them. Small business owners want to reward their regulars in a way that feels special and modern, but they're stuck. They need a way to give customers a true "VIP" membership they can't lose and a "Free Coffee" coupon that's more than just an email.

Your 24-Hour Mission:

Your job is to build a simple, plug-and-play platform that lets any small business (like a cafe, bookstore, or online creator) roll out their own web3 loyalty program without writing a single line of code.**Required MVP**

- **The "Owner's Toolkit":** A clean dashboard for the business owner. They connect their wallet and easily design their program:
    - A **"VIP Membership Pass"**: A non-transferable Soul-Bound Token (SBT) that proves someone is a true regular.
    - A **"Redeemable Coupon"**: A regular NFT eg for a "Free Muffin" or discount.
- **The "Launch" Button:** A simple "Go" button that deploys a new, secure contract just for their shop.
- **The "Cashier's App":** The most important part! A simple page for the employee at the register to:
    - **Check:** Enter a customer's wallet address to see if they have the "VIP Membership Pass" (and give them their 10% discount).
    - **Redeem:** Find a customer's NFT(like a "Free Coffee NFT") and click "Redeem" to burn or use it, preventing reuse.
- **The "Reward" Tool:** A simple form in the owner's dashboard to paste a customer's wallet address and instantly send them a new reward NFT.

Hackathon Challenge 3: The Global Peer-to-Peer Skill Swap

The Problem

The cost and time commitment of formal education are often too high, leaving vast amounts of valuable **human capital locked up** in individuals. People want to learn practical skills from real experts but don't want to deal with high course fees or complex scheduling. Existing platforms are transaction-based (money for course) or too casual (forums), failing to facilitate a structured, direct, and mutually beneficial **exchange of expertise**. The barrier to finding a perfect, fair learning partner is too high.

Your 24-Hour Mission:

Build the world's most effective peer-to-peer skill-matching and exchange platform. The core is an intelligent algorithm that matches users based on what they **Offer** and what they **Want** to learn, plus a simple in-platform workspace for the actual teaching.

Required MVP

- **The "Offer & Want" Profile:** A simple user onboarding flow where a user lists one skill they **Master and will teach** (e.g., *Advanced Excel Modeling*) and one skill they **Want to learn** (e.g., *Beginner Python for Data*).
- **The "Perfect Match" Engine:** An automated backend that constantly matches users with a high **Skill A (Offer)** for a user with a high **Skill B (Want)** and vice-versa (high **Skill B (Offer)** for a user with high **Skill A (Want)**).

- **The In-Platform Classroom:** Once a match is accepted, the platform provides a shared, synchronous workspace with:
    - A **Live Video/Audio Call** option.
    - A **Collaborative Whiteboard** tool for diagrams and code snippets.
    - A Persistent **Chat** thread for communication.
- **The "Trust Score" System:** A simple mechanism where partners must mutually **confirm completion** of the exchange, automatically triggering a **Rating and Review** prompt for both the skill they taught and the effectiveness of the exchange.

Bonus Goal

- Build a **Scheduler Integration** that checks both users' time zones and schedules a 1-hour session directly into their calendars (e.g., Google Calendar, Outlook) with the meeting link.

Hackathon Challenge 4: The AI-Powered Lecture Generator

The Problem

Educators, trainers, and content creators waste countless hours transforming dense information into engaging, modern, and visually appealing teaching material. They need custom videos, dynamic slides, and professional voiceovers, but the process of scripting, animating, recording, and editing is slow and expensive. The tools available are either too basic (PowerPoint) or require deep video editing expertise.

Your 24-Hour Mission:

Create a simple, text-to-educational-asset tool. A user should be able to input a **single prompt** (e.g., "Explain the process of photosynthesis for 10th-grade students") and receive a complete, ready-to-present lecture package in minutes.

Required MVP

- **The "Prompt-to-Curriculum" Input:** A single text box where the user inputs the topic, target audience, and desired length (e.g., "5-minute video").
- **The Dual-Asset Generator:** Your platform's core AI logic must instantly produce two linked assets:
    - **The Slide Deck:** A downloadable presentation (e.g., PDF or Google Slides export) with structured content, clear headings, and placeholder images/diagrams for each point.
    - **The Video Script & Voiceover:** A complete script derived from the slides, paired with an AI-generated, natural-sounding voiceover and synced to automatically scroll through the slide content.
- **Visual Customization:** A simple panel allowing the user to select the **Visual Theme** (e.g., *Minimalist*, *Chalkboard*, *Corporate*) before generation.
- **Download & Share:** The ability to download the generated video (e.g., MP4) and/or a

shareable link to the full presentation.

Bonus Goal

- Include a feature to generate **3 Quiz Questions** (multiple choice) relevant to the content of the lecture, ready to be copied and pasted into a learning management system.

Hackathon Challenge 5: The Collegiate Inbox Navigator

The Problem

The student college experience is chaotic. Critical academic information like scholarship deadlines, class cancellations, assignment feedback PDFs, exam schedules, and course announcements is scattered across multiple platforms. Most of it lives in a neglected **university-provided Gmail inbox**, while the rest hides in **Google Classroom**, where assignments and announcements often go unnoticed.As a result, students miss deadlines, lose crucial documents, and feel overwhelmed trying to manually cross-reference their inbox, Google Classroom, and calendar.

Your 24-Hour Mission:

Develop a smart application that acts as a **personal, AI-powered academic assistant**, bringing together insights from both **Gmail** and **Google Classroom** into one actionable dashboard

Required MVP

- **Secure Gmail Integration:** A robust connection to a student's college Gmail & classroom account for reading and processing emails.
- **The "Critical Path" Dashboard:** A single, clean page that summarizes all crucial information:
    - **Upcoming Deadlines:** A list of assignments and exams extracted from emails, with a clear countdown timer.
    - **Key Document Repository:** A dedicated section that automatically identifies and organizes all **PDFs, DOCX, and PPT files** attached to emails, categorized by course name.
    - **Schedule Changes/Alerts:** A dynamic feed of emails containing keywords like "Cancelled," "Rescheduled," "Urgent Notice," or "Room Change."
- **Smart Categorization:** An algorithm that uses AI/keyword matching to tag emails and attachments by their corresponding **Course/Subject** (e.g., *'Calculus 101'*, *'Organic Chemistry Lab'*).
- **One-Click Calendar Sync:** For any detected event (exam, lecture, meeting), a simple button to **instantly add it** to the student's personal calendar (e.g., Google Calendar).

Bonus Goal

- A **"Summarize This"** button that appears next to any long email (over 200 words) from a professor, using a large language model to generate a 3-sentence summary of the main points and action items.

Hackathon Challenge 6: The Automated Lab Grader (The "Digital TA")

The Problem

Teaching large university technical classes is plagued by manual, repetitive grading. Instructors spend more time checking syntax and edge cases than helping students learn concepts. Feedback is slow, often taking days or weeks, which prevents students from learning from their mistakes quickly. Existing Learning Management System (LMS) tools lack robust, automated code execution and objective evaluation, leading to inconsistent scores across teaching assistants (TAs) and different lab sections.

Your 24-Hour Mission:

Develop an accessible, centralized platform for university labs that acts as an objective, real-time code judge. The goal is to liberate instructors from manual grading and give students instant, accurate performance feedback and clear ranking.

Required MVP

- **Instructor Assignment Toolkit:** A secure dashboard where an instructor can:
    - Define a new assignment (Title, Deadline, Language support: e.g., Python, C++, JavaScript).
    - Upload a set of **Test Cases** (input file and expected output file) with assigned point values for each.
- **Student Submission & Instant Feedback Loop:** A clean student-facing page where a student can:
    - Upload their solution file (e.g., lab1.py).
    - Receive a score immediately, showing which **public test cases** passed and which failed, along with execution time.
- **Live Coding Editor (In-Browser):** An integrated editor where students can write, run, and test small code snippets before final submission, ensuring a smooth development workflow.
- **Centralized Analytics:** A section for the instructor to view overall class performance, including submission times, average score, and a **ranking/leaderboard** of all students for motivation (with optional anonymous display).

Bonus Goal

- Implement a basic **Plagiarism/Similarity Checker** that compares the submitted code against all other student submissions for that assignment and flags potential

high-similarity matches for the instructor to review.

Hackathon Challenge 7: The Mobile Packet Hunter

The Problem

Cybersecurity professionals require constant, deep network visibility, but traditional tools like Wireshark are confined to the desktop. When conducting mobile penetration tests or field network analysis using frameworks like **Kali NetHunter**, there is a critical need for a high-performance, mobile-native application capable of **Deep Packet Inspection (DPI)**. The current gap is an efficient, on-the-go tool for real-time traffic capture and forensic analysis using only an Android device, without sacrificing critical inspection depth.

Your 24-Hour Mission:

Build a standalone Android-compatible module, optimized for the Kali NetHunter ecosystem, that enables network engineers and ethical hackers to capture, filter, and analyze packet data directly from their mobile device in real-time.

Required MVP

- **Real-Time Capture Module:** The ability to initiate and stop live traffic capture from the device's **WiFi and Mobile Data** interfaces, running efficiently in the background without overwhelming mobile resources.
- **The Traffic Dashboard:** A landing screen displaying vital real-time metrics: Packets/Second, Total Data Captured, and a visual breakdown (e.g., a simple pie chart) of the top **Protocols** (TCP, UDP, ICMP, DNS).
- **Granular Packet Inspector (DPI):** A list view of all captured packets. Tapping a packet opens a detail pane showing:
    - Source IP, Destination IP, Source Port, Destination Port.
    - Protocol and Layer information.
    - A hex/ASCII view of the **raw packet payload** (the Deep Packet Inspection component).
- **Advanced Filtering Controls:** Dedicated input fields allowing users to instantly filter the displayed packet list based on **Source/Destination IP**, **Port**, or a specific **Protocol** (e.g., filter only for HTTP traffic).

Bonus Goal

- Implement a **Predefined Security Filter** section that allows the user to quickly select common security filters (e.g., "Show all packets using default HTTP ports," "Flag non-standard ICMP traffic") to immediately detect suspicious activity.