Of course. Here is a comprehensive document for a fortnightly GenAI Hackathon, including the strategic execution plan and a list of recurring problem statements categorized by difficulty.

Fortnightly GenAI Hackathon: Strategic Execution Plan

1. Vision & Strategic Goals

Vision: To cultivate a vibrant culture of innovation and continuous learning by leveraging the power of Generative AI. Our fortnightly hackathons will serve as a recurring platform for employees to experiment, collaborate, and rapidly prototype solutions to real-world business challenges.
Strategic Goals:

- **Accelerate Innovation:** Rapidly generate and test new ideas and product features.
- **Upskill Workforce:** Provide a hands-on, practical environment for employees to learn and apply GenAI skills.
- **Foster Collaboration:** Break down departmental silos by encouraging cross-functional team formation.
- **Solve Business Problems:** Crowdsource solutions for internal process inefficiencies and external customer-facing challenges.
- **Identify Talent:** Discover and nurture employees with a passion and aptitude for AI development and product thinking.

2. Hackathon Framework

- **Frequency:** Fortnightly (every two weeks).
- **Duration:** 2 Days (e.g., Thursday 9:00 AM - Friday 4:00 PM).
- **Participants:** Open to all employees. Teams should be 2-4 members. Cross-departmental teams are highly encouraged.
- **Theme:** Each hackathon will have an optional theme to focus efforts (e.g., "Customer Experience AI," "Internal Process Automation," "Data-Driven Decisions").

3. Fortnightly Execution Timeline (The 2-Week Cycle)

This timeline ensures a smooth, repeatable process with minimal overhead.
Week 1: Planning & Announcement

- **Day 1 (Monday):**
  - **Post-Mortem & Planning (1 hour):** The Hackathon Committee meets to review feedback from the previous event and select the theme and problem statements for the upcoming one.
- **Day 2 (Tuesday):**
  - **Official Announcement:** An organization-wide email and Slack/Teams message is sent out.
  - **Content:** Theme, dates, registration link, and the list of problem statements.
- **Day 3-5 (Wednesday - Friday):**
  - **Registration Drive:** Gentle reminders to register.

    - **Team Formation:** A dedicated channel is used for individuals looking for teams. Committee helps facilitate matchmaking.

Week 2: Execution & Wrap-Up

- **Day 8 (Monday):**
    - **Finalize Logistics:** Confirm mentor availability, judging panel, and any required cloud/API resources.
- **Day 11 (Thursday - Hackathon Day 1):**
    - **9:00 AM:** Kick-off call. Welcome, review rules, theme, and judging criteria.
    - **9:30 AM:** Hacking begins!
    - **Throughout the day:** Mentor check-ins and support on the dedicated communication channel.
- **Day 12 (Friday - Hackathon Day 2):**
    - **12:00 PM:** Code Freeze / Submission Deadline. Teams must submit a link to their code repository (e.g., GitHub) and a short presentation (5 slides max).
    - **1:00 PM - 3:00 PM:** Project Demonstrations. Each team gets 5 minutes (3 min demo + 2 min Q&A).
    - **3:00 PM - 3:30 PM:** Judges Deliberation.
    - **3:30 PM:** Winners Announcement & Closing Ceremony.
- **Following Monday (Day 1 of next cycle):**
    - **Feedback & Highlights:** Send a survey to all participants. Share a summary of winning projects and key highlights with the organization. Begin the next cycle.

4. Roles & Responsibilities

- **Hackathon Committee (Core Team):** Responsible for overall planning, communication, and execution.
- **Mentors / Subject Matter Experts (SMEs):** Senior engineers, data scientists, and product managers who provide technical and business guidance to teams during the event.
- **Judges:** A panel of 3-5 leaders (technical and business) responsible for evaluating projects based on set criteria.

5. Judging Criteria

Projects will be scored on a scale of 1-10 across four categories:

1. **Innovation & Creativity (25%):** How novel and creative is the application of GenAI?
2. **Technical Implementation (35%):** How well is the solution built? Is the use of the GenAI model(s) effective? Is the demo functional?
3. **Business Value & Impact (30%):** Does this solve a real, meaningful problem for the company or its customers?
4. **Presentation & Demo (10%):** How clearly and effectively did the team present their project?

6. Tools & Resources

- **Communication:** A dedicated Slack/Teams channel (e.g., #genai-hackathon).
- **Code Management:** A shared GitHub/GitLab organization.
- **GenAI Platforms & APIs:**

    - OpenAI API (GPT-4, GPT-3.5-Turbo)
    - Google AI Platform (Vertex AI, Gemini models)
    - Hugging Face (Transformers, Hub)
    - LangChain / LlamaIndex for building applications
- **Cloud Infrastructure:** Provide pre-approved access to cloud credits (AWS, GCP, Azure).

Recurring GenAI Problem Statements Library

This library can be used to inspire participants. New problems can be added over time.

Category 1: Simple (Difficulty: ★☆☆)

(Focus: Using a single API call or a basic chain. Great for beginners.)

1. **Email Subject Line Generator:**
    - **Description:** Create a tool that takes the body of an email as input and suggests 3-5 catchy and professional subject lines.
    - **GenAI Task:** Text Generation.
2. **Meeting Summarizer:**
    - **Description:** Build a service that accepts a raw text transcript of a meeting and generates a concise summary with key decisions and action items.
    - **GenAI Task:** Text Summarization.
3. **Social Media Post Creator:**
    - **Description:** Given a topic or a link to a blog post, generate a short, engaging post formatted for Twitter, LinkedIn, or Instagram.
    - **GenAI Task:** Text Generation with persona/style adaptation.
4. **Code Commenter:**
    - **Description:** Create a script that takes a function or block of code and automatically adds explanatory comments.
    - **GenAI Task:** Code Generation/Explanation.
5. **Internal Acronym Explainer:**
    - **Description:** Build a simple chatbot where users can ask "What is [internal acronym]?" and the bot provides the full form and a brief explanation.
    - **GenAI Task:** Basic Q&A.

Category 2: Intermediate (Difficulty: ★★☆)

(Focus: Combining models, using Retrieval-Augmented Generation (RAG), or building a simple workflow.)

1. **"Chat with our Docs" Bot (RAG):**
    - **Description:** Ingest our internal product documentation or knowledge base into a vector database. Build a chat interface that allows employees to ask questions about our products and get answers based *only* on the provided documents.
    - **GenAI Task:** Retrieval-Augmented Generation (RAG).
2. **Customer Feedback Classifier & Summarizer:**
    - **Description:** Create a pipeline that ingests customer feedback (from surveys, support tickets, app reviews). First, classify each piece of feedback (e.g., Bug Report, Feature Request, Positive Feedback). Then, for each category, generate a

summary of the key themes for the week.

- **GenAI Task:** Classification and Summarization.

3. **Resume Screener:**
   - **Description:** Build a tool that takes a job description and a candidate's resume (in PDF or text format). The tool should output a summary of the candidate's fitness for the role and a score based on how well their skills match the requirements.
   - **GenAI Task:** Information Extraction and Analysis.
4. **SQL Query Generator:**
   - **Description:** Create an interface where a non-technical user can describe the data they want in plain English (e.g., "Show me all customers from Bengaluru who purchased Product X in the last month"), and the tool generates the corresponding SQL query.
   - **GenAI Task:** Natural Language to SQL (NL-to-SQL).
5. **Personalized Onboarding Plan Generator:**
   - **Description:** For a new hire, take their role (e.g., "Software Engineer"), team ("Platform Team"), and experience level ("Junior") as input. Generate a personalized 30-day onboarding plan with recommended documents to read, people to meet, and initial tasks.
   - **GenAI Task:** Complex Text Generation with context.

Category 3: Difficult (Difficulty: ★★★)

(Focus: Building agentic systems, fine-tuning models, multi-modal analysis, or complex, multi-step chains.)

1. **AI Agent for Internal Support:**
   - **Description:** Design an AI agent that can autonomously resolve simple IT or HR support tickets. The agent should be able to understand the user's request, ask clarifying questions, and use tools (e.g., call a (mock) API to reset a password or check vacation balance).
   - **GenAI Task:** Agentic AI, Tool Use.
2. **Fine-Tuning a Model for Company-Specific Language:**
   - **Description:** Gather a dataset of our company's internal documents (wikis, project specs, marketing copy). Use this dataset to fine-tune a pre-trained language model (like Llama-3 or GPT-3.5-Turbo). Demonstrate that the fine-tuned model performs better on company-specific tasks (e.g., drafting internal emails) than the base model.
   - **GenAI Task:** Model Fine-Tuning.
3. **Multi-Modal Competitor Ad Analysis:**
   - **Description:** Build a system that takes a competitor's video ad as input. It should transcribe the audio, describe the key visual scenes, and analyze the sentiment of the music. Combine these three streams of information into a comprehensive report on the ad's messaging and potential impact.
   - **GenAI Task:** Multi-modal understanding (Video/Vision + Audio + Text).
4. **Proactive Sales Opportunity Detector:**
   - **Description:** Create a system that monitors public news feeds, press releases, and financial reports for our key accounts. Use a GenAI model to identify trigger events (e.g., company expansion, new funding round, leadership change) and generate a concise "Opportunity Alert" for the relevant account executive, including a

suggested email draft to restart conversations.

- **GenAI Task:** Information Extraction, Reasoning, and Text Generation.

5. **Automated Pull Request (PR) Reviewer:**
   - **Description:** Develop a GitHub Action that uses GenAI to review new pull requests. The AI should check for adherence to coding style guides, identify potential bugs or logic flaws, and suggest improvements, leaving comments directly on the PR. This goes beyond simple linting.
   - **GenAI Task:** Advanced Code Analysis and Generation.