Problem Statements for Datahack 3.0

1. AI-Driven Financial Planning for Healthcare Costs

Description:
Develop an AI-driven tool that helps individuals and families predict and plan for healthcare costs over a lifetime, including scenarios like aging, chronic illness, or unforeseen medical expenses.

Judging Criteria:
1. Innovation (20 points)
- Originality: How unique is the approach taken to predict healthcare Originality costs?
- Use of AI: How effectively does the tool leverage AI techniques (e.g., machine learning algorithms, predictive analytics)?

2. Accuracy and Reliability (20 points)
- Model Performance: How accurate are the predictions made by the AI model?
- Data Sources: What data is used for training the model, and how reliable is it? Are assumptions clearly stated?

3. Experience (20 points)
- Interface Design: Is the user interface intuitive and user-friendly? Does it facilitate easy navigation and understanding?
- Accessibility: How accessible is the tool for various demographics, including those with limited tech experience?

4. Comprehensiveness (20 points)
- Scenarios Covered: Does the tool address a wide range of scenarios (aging, chronic illness, emergencies)?
- Financial Insights: Are there useful insights or recommendations provided alongside predictions (e.g., savings plans, insurance options)?

5. Presentation and Communication (20 points)
- Clarity: Is the project presented clearly and coherently? Are technical concepts explained in an understandable manner?
- Engagement: How well does the team engage the judges during the presentation, demonstrating enthusiasm and knowledge?

2. Space Debris Tracking and Mitigation

Description:

Design a system for tracking space debris over Indian territory using existing observational data and predicting potential collision paths with satellites, proposing mitigation strategies.

Judging Criteria:

1. Technical Feasibility (20 points)
   - Implementation: How practical and achievable is the proposed system? Are the technical challenges addressed?
   - Use of Data: How effectively does the solution utilize existing observational data? Are data sources clearly identified?

2. Accuracy and Reliability (20 points)
   - Prediction Models: How accurate are the predictions regarding potential collision paths? Is there a validation method in place?
   - Real-Time Tracking: Does the system offer real-time tracking capabilities, and how reliable is the data?

3. Mitigation Strategies (20 points)
   - Proposed Solutions: Are the proposed mitigation strategies for space debris effective and feasible?
   - Impact Assessment: How well does the project assess the potential impact of the debris on satellite operations?

4. User Interface and Usability (20 points)
   - Interface Design: Is the user interface designed for ease of use, allowing stakeholders (e.g., satellite operators, researchers) to access information easily?
   - Visualization: Are the data visualizations (e.g., collision predictions, debris tracking) clear and informative?

5. Collaboration and Outreach (20 points)
   - Engagement with Stakeholders: Does the project demonstrate an understanding of the needs of stakeholders in the space industry (e.g., satellite operators, regulatory bodies)?
   - Public Awareness: How well does the project promote awareness of space debris issues among the general public or specific communities?

Event Idea No.1

Data Dash: Break the Code

Theme: "Hack the System"
The team has been trapped in a digital vault by a hacker group. They must solve technical challenges in 4 stages to regain control and unlock the virtual door to escape. Each role has a specific puzzle tailored to their expertise.

Premise:
A hacker group has breached your company's system and locked the team in a virtual vault. Each team member must use their unique technical skills to solve challenges that will bring down the system's firewall and unlock the door.

Time Limit & Hints

- Time Limit: 30 minutes.
- Hints: One hint per puzzle to make it easier, but deduct 5 minutes for each hint used

---

Stage 1: Binary Puzzle

- Challenge: The firewall is locked with binary encryption. The Decoder must break the binary code to open the first level.
- Puzzle: Present the team with a binary sequence:
  Example: 01101111 01110000 01100101 01101110 (which translates to "open").
- Solution: The Decoder converts the binary to text ("open"), unlocking the first part of the system.

Stage 2: Debugging a Code Snippet

- Challenge: The Debugger must fix an error in the code to disable the next security measure.
- Puzzle: Provide a snippet of code with syntax or logic errors:

```
def subtract_numbers(a, b):
    print(a - b
```

  Error: A missing parenthesis is preventing the code from running.
- Solution: The Debugger fixes the code:

```
def subtract_numbers(a, b):
    print(a - b)
```

Stage 3: Cryptography Puzzle

- Challenge: The system uses an encrypted message that must be decrypted to get the next clue.
- Puzzle: The Cryptographer receives a Caesar cipher message (shifted by 3):
  Example: "Frgh" (which translates to "Code").
- Solution: The Cryptographer decodes the message ("Code"), and the system grants access to the next stage.

Stage 4: SQL Security Challenge (Final Unlock)

- Challenge: The Analyst must prevent an SQL injection attack to stop the hacker's attempt to steal data and unlock the vault door.
- Puzzle: Provide an SQL query with a security vulnerability:

SELECT * FROM users WHERE username = '" + user_input + "';

Problem: The query is susceptible to SQL injection.

- Solution: The Analyst rewrites the query using parameterized queries:

SELECT * FROM users WHERE username = ?;

Once this query is secured, the final firewall is brought down, unlocking the room and allowing the team to escape.

Once the last solution is entered, a message can appear on-screen like:

"Firewall Deactivated. Room Unlocked!"

Judging Criteria

- Whichever team solves it in the least amount of time wins

Event Idea No.2

FeatureQuest: Unlocking Insights

Event Overview

- Introduce the purpose of the competition: to encourage participants to create innovative features from raw datasets to enhance machine learning model performance.

Competition Structure

- Team Size: 2-4 participants per team.
- Time Limit: 4-8 hours to complete the task.
- Data Provided: A raw dataset related to user interactions, sales, or similar.

Challenges

- Participants will analyze the provided dataset and create new features.
- They will evaluate the impact of these features on model performance using metrics like accuracy and precision.

Judging Criteria

- Improvement in model performance after adding new features.
- Creativity and originality of the features created.
- Documentation and explanation of the feature engineering process.