Sejal Test Task:

Objective:

Build a mini AI Agent-based system to simulate or organize a 1-day student hackathon, designed to:

- Use automation and AI agents to handle outreach, registration, and judging
- Create engagement across 2–3 colleges via outreach efforts
- Lay the foundation for a scalable hackathon assistant agent

Role & Context:

As an AI & DS student leader and organizer, this task blends her technical, organizational, and outreach skills. She will build an intelligent system (and mini event plan) that:

- Uses AI to simulate hackathon planning
- Shows initiative in peer outreach and college-level coordination
- Includes end-to-end flow: planning → registration → judging → wrap-up

Duration:

72 hours of focused work, to be submitted within 7 calendar days.

Deliverables (6 Sections):

1. Hackathon Design Brief

- Name + theme (e.g., “CampusCoders: AI for Real Life”, “HackAI24”)
- 1-day agenda: Opening → Build → Submit → Judging → Closing
- Judging criteria: usefulness, creativity, teamwork, tech stack, clarity
- Categories: AI/ML, Gaming, Web3, Open Innovation
- Format: 2-page PDF or Notion doc
-

2. HackaAgent Prototype

- Build a basic AI-powered agent using Python or Streamlit (no UI styling required)
- Core Features:

    - Add/View problem statements
    - Team registration (name, members, emails)
    - Upload or link project submission
    - Judges assignment and score entry
- Data can be stored in JSON, Firebase, or Supabase
- Output: GitHub repo + README + small demo video or walkthrough (2–3 mins)

3. Outreach Snapshot

- Identify 2–3 nearby colleges/peer groups
- Show evidence of contacting:
    - Screenshots of WhatsApp/Telegram/Instagram DMs
    - Small blurb/poster sample for reaching out
    - If they show interest: try to form real or simulated 5-team list
- Include a list of student groups or relevant communities approached

4. AI Agent Support Plan

Design 3 simple agents as part of the hackathon ecosystem:

- MentorBot – answers coding/idea doubts using GPT-3.5
- ReminderBot – sends event reminders (or pre-written messages)
- JudgingBot – helps review and summarize projects
  (Just descriptions needed, actual code optional.)

Bonus:

- Add a “Hackathon Generator” — simple GPT-3.5 prompt to generate 5 challenges by theme

5. Execution Simulation

- Share:

- A simulated or actual Discord/Telegram/WhatsApp group screenshot
- 5 team names + members (fictional or real)
- 3 sample project ideas
- Optional: selfie/post organizing or creating the teams

6. Wrap-Up Plan

- Where would teams submit? (Google Form, GitHub links, email)
- How would winners be selected?
- How can the system scale to 50+ teams later?
- Suggested improvements for future events

Submission Format:

Send via WhatsApp or Google Drive folder:

- 1 PDF/Notion plan
- Screenshots (group chats, mock posters, outreach)
- GitHub repo with working code and README
- Small video walkthrough (2–3 mins) of what was built (screen-recorded is fine)
- In-person or live demo (if required)

Evaluation Criteria (Out of 10):

Criteria	Weight
Hackathon Structure + Creativity	25%
Outreach and Team Formation	25%
Agent Simulation and AI Use	20%
Screenshots + Execution Proof	20%
Presentation + Submission Quality	10%

Tech Stack Suggestions:

- Frontend: Streamlit, CLI, or Notion
- Backend: Supabase, Firebase, or JSON
- Agent Logic: GPT via LangChain/OpenAI wrapper
- Tools: Notion, Canva, WhatsApp, Discord, Google Forms

Learning Requirement:

In addition to the task, Sejal must explore the following resources in her 7-day timeline. These cover AI agents and Reinforcement Learning (RL):

Learning Resources:

Note: Links may break over time; if so, search the video title or keywords on YouTube

1. AI Agents – Concept & Examples

- What are AI Agents? (freeCodeCamp)
- LangChain Agent Explained
- AI Agents 101 – Prompt Engineering & Tools

2. Reinforcement Learning for Beginners

- RL Concepts – Simplified (StatQuest)
- Beginner-Friendly RL Tutorial – Python
- Q-Learning Intro (Simple)

Keywords to search if links fail:

- “AI agent vs chatbot”
- “LangChain agent beginner”
- “Reinforcement learning basics Python”
- “AI workflow agent tutorial”