Session - 2&3

Date: September 10, 2025.

Practical API Integration & Workflow Automation

Topics: API Authentication (Keys & OAuth), Docker Concepts, Cloud Platform Integration (Google Cloud), and Hands-on Workflow Automation with n8n.

Understanding Core Integration Concepts

This part of the session focused on the foundational concepts required to securely connect different applications and services across the web.

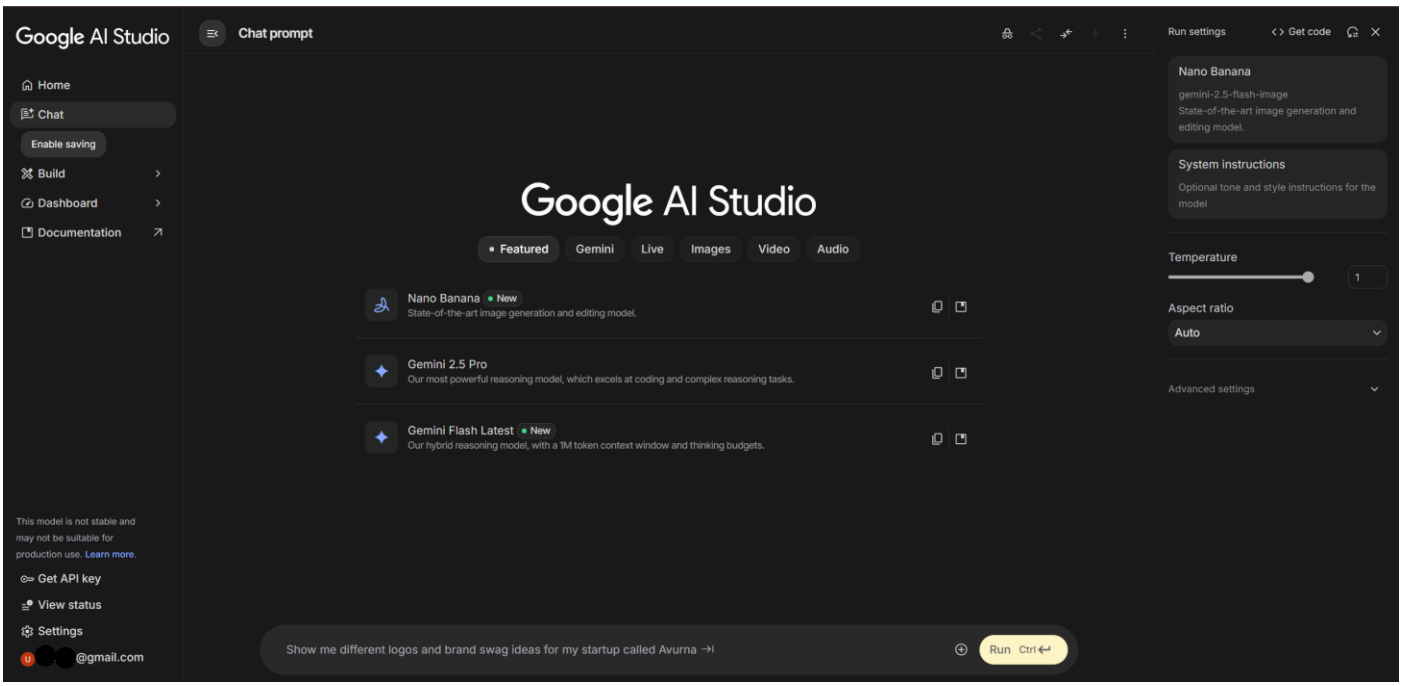
1. Authentication vs. Authorization

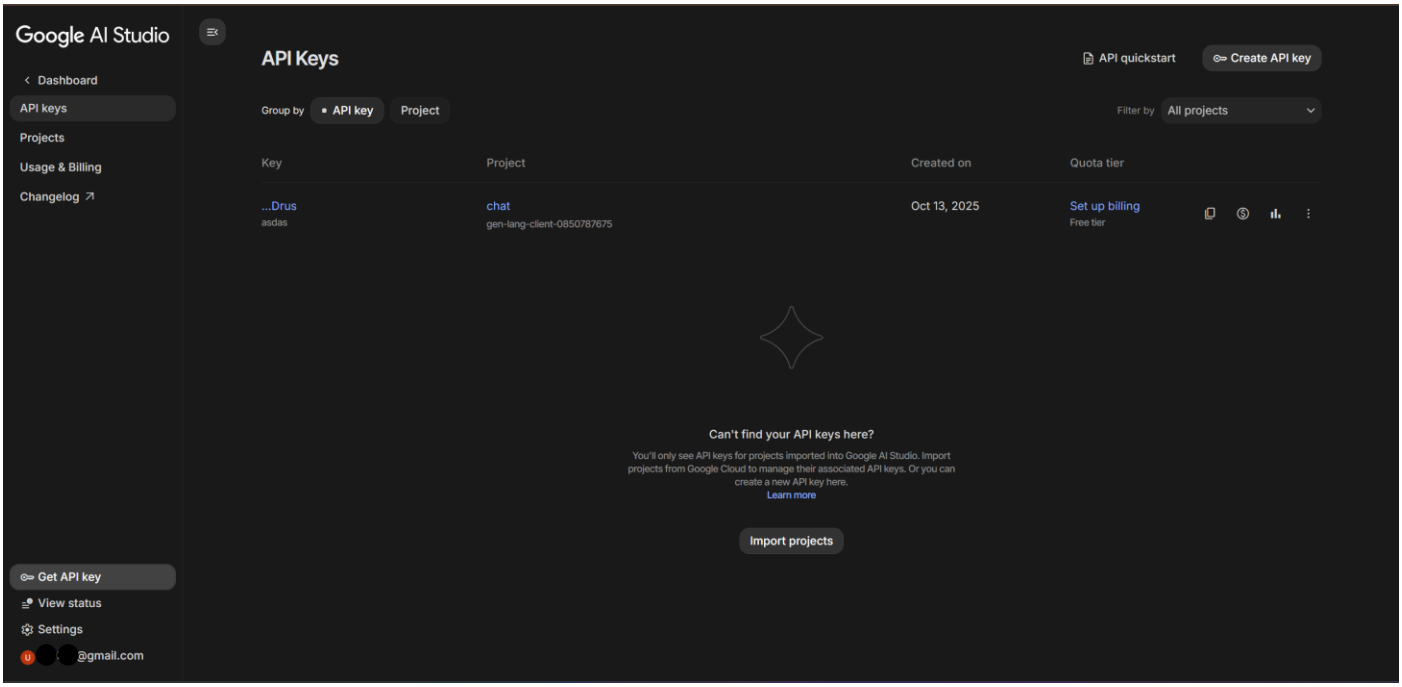
Before connecting services, we need to manage permissions. This involves two key ideas:

- **Authentication:** This is the process of **verifying who you are**. When you log in with a username and password, you are authenticating yourself. In the world of APIs, an **API Key** often serves this purpose.
- **Authorization:** This is the process of **determining what you are allowed to do** after you've been authenticated. Just because you are a valid user doesn't mean you can access everything. OAuth handles secure delegated access (authorization), and with OpenID Connect, it can also handle authentication.

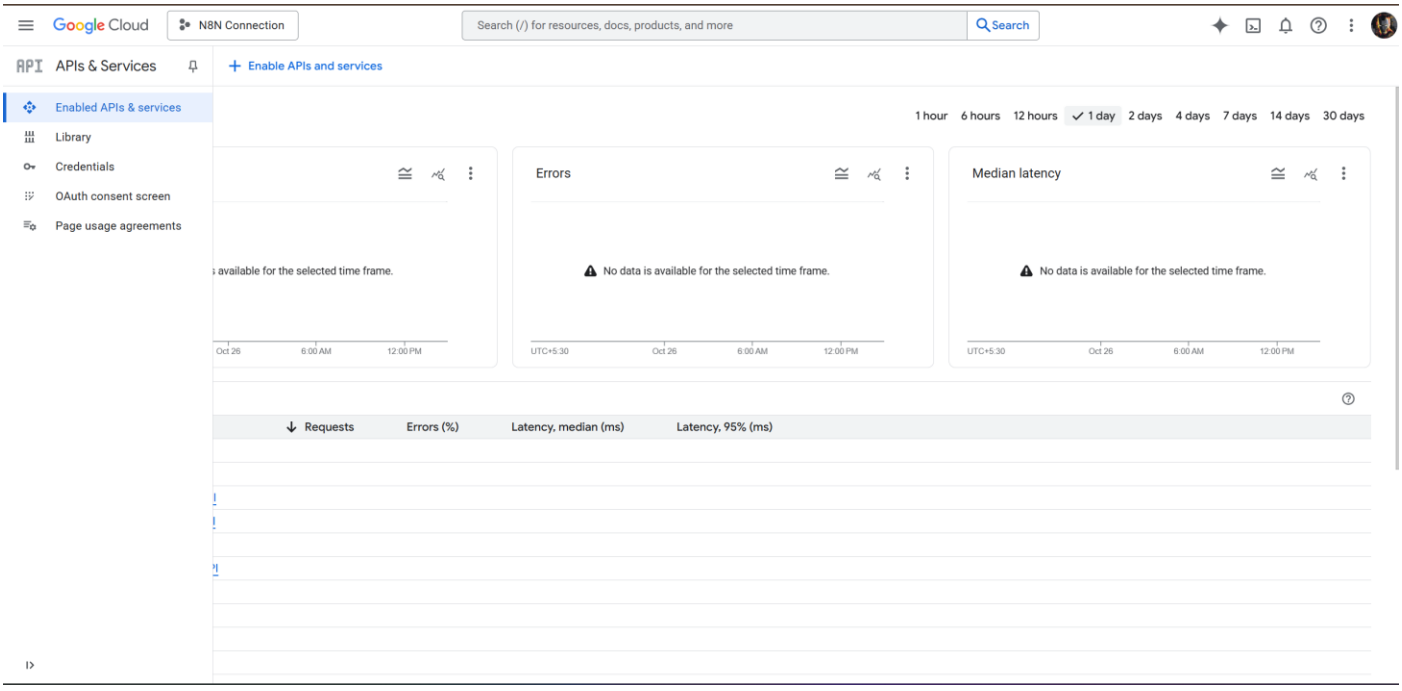
2. API Keys & OAuth

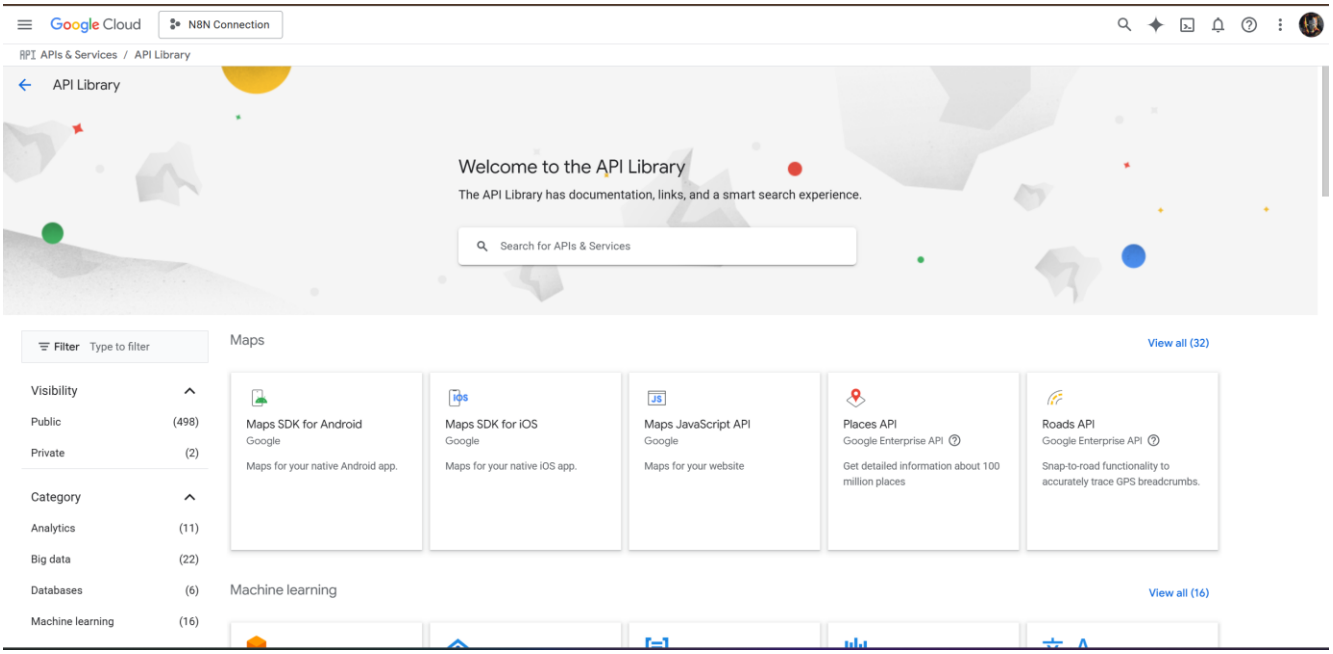
- **API Keys:** Think of an API Key as the **key to your house**. It's a single, simple secret that you give to an application to grant it access. It's straightforward but less flexible. If the key is stolen, someone has full access.

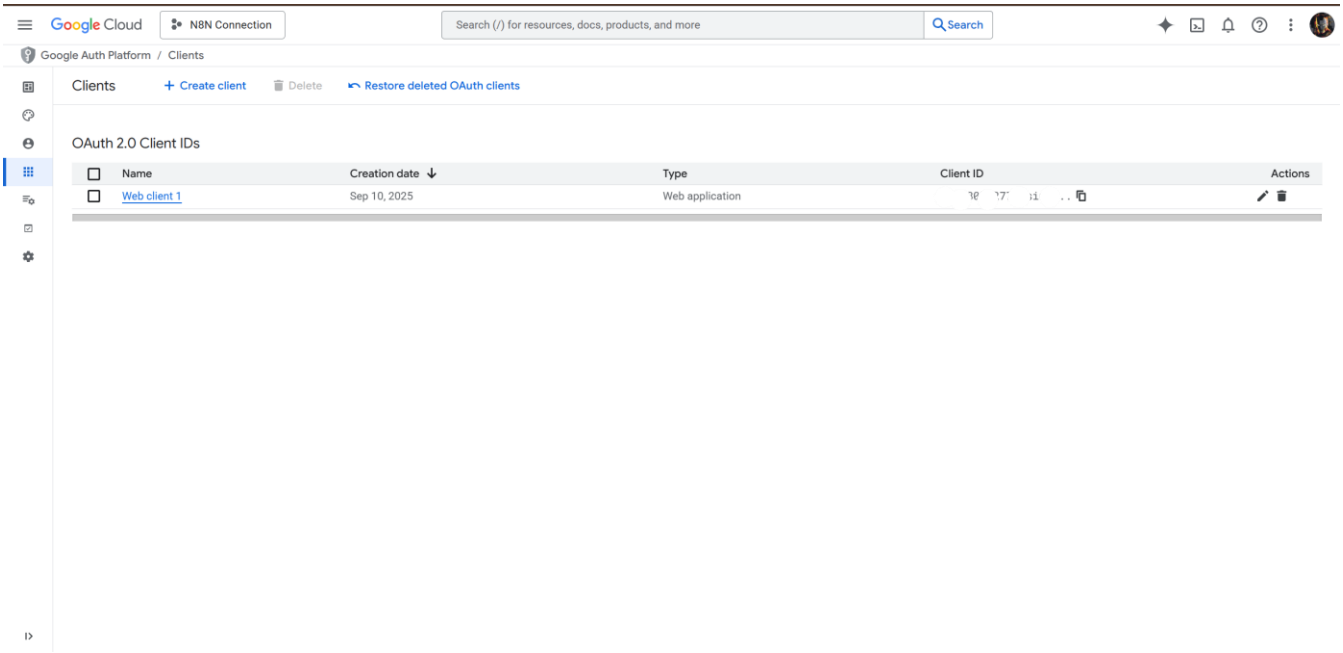


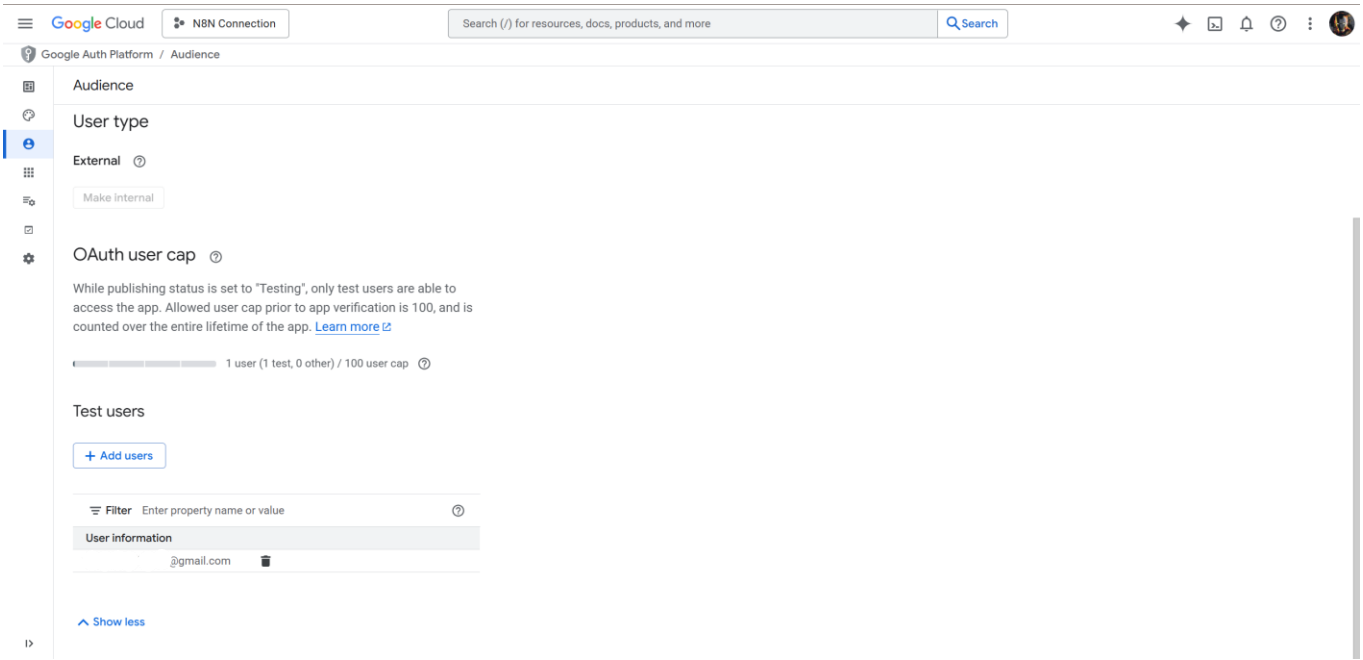


- **OAuth (Open Authorization):** Think of OAuth as a **hotel key card or a valet key**. It grants specific, limited permissions for a certain amount of time without you having to share your actual password (the master key). When an app asks, "Can this application post to your Twitter feed?", and you click "Allow," you are likely using OAuth. It's more secure because you can revoke access for a single app without changing your main password, and you control exactly what it's allowed to do.



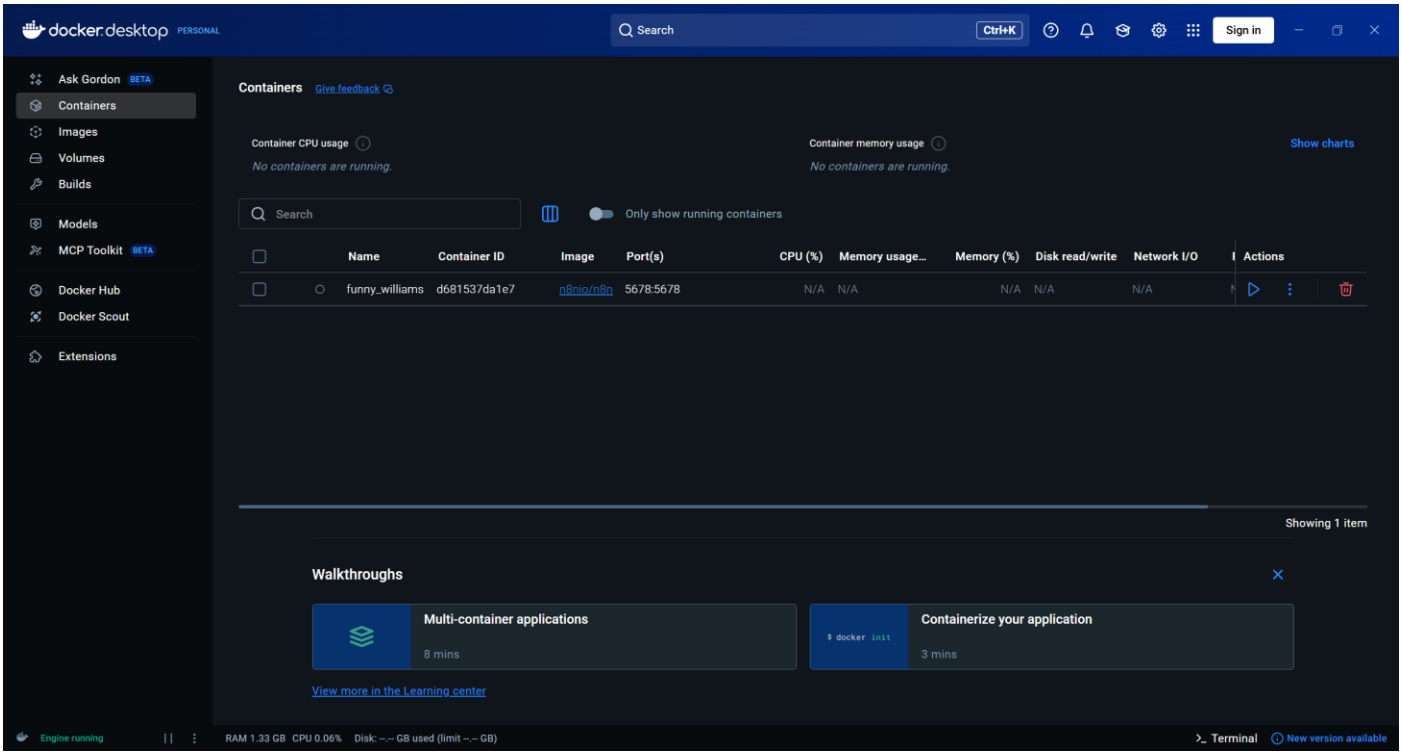

Google Cloud
N8N Connection
APIs & Services / API Library
API Library
Welcome to the API Library
The API Library has documentation, links, and a smart search experience.
Search for APIs & Services
Filter Type to filter
Visibility
Public (498)
Private (2)
Category
Analytics (11)
Big data (22)
Databases (6)
Machine learning (16)
Maps
View all (32)
Maps SDK for Android
Google
Maps for your native Android app.
Maps SDK for iOS
Google
Maps for your native iOS app.
Maps JavaScript API
Google
Maps for your website
Places API
Google Enterprise API
Get detailed information about 100 million places
Roads API
Google Enterprise API
Snap-to-road functionality to accurately trace GPS breadcrumbs.
Machine learning
View all (16)


Google Cloud
N8N Connection
Search (/) for resources, docs, products, and more
Search
Google Auth Platform / Clients
Clients
Create client
Delete
Restore deleted OAuth clients
OAuth 2.0 Client IDs
Name
Creation date
Type
Client ID
Actions
Web client 1
Sep 10, 2025
Web application


Google Cloud
N8N Connection
Search (/) for resources, docs, products, and more
Search
Google Auth Platform / Audience
Audience
User type
External
Make internal
OAuth user cap
While publishing status is set to "Testing", only test users are able to access the app. Allowed user cap prior to app verification is 100, and is counted over the entire lifetime of the app. Learn more
1 user (1 test, 0 other) / 100 user cap
Test users
Add users
Filter Enter property name or value
User information
@gmail.com
Show less

3. Docker and Containerization

- **The Problem:** A common issue in development is the "it works on my machine" problem, where code runs perfectly for one developer but fails for another due to differences in operating systems, software versions, or other settings.
- **The Solution: Docker Containers.** A Docker container includes **application + dependencies**, but not a full OS kernel. That’s why they’re lightweight compared to VMs.
    - **Analogy:** A Docker container is like a standardized shipping container for software. It doesn't matter if you're moving it by truck, train, or ship; the contents inside are protected and work the same way everywhere.
    - **In this session,** understanding Docker helped the team collaboratively resolve setup and integration challenges.

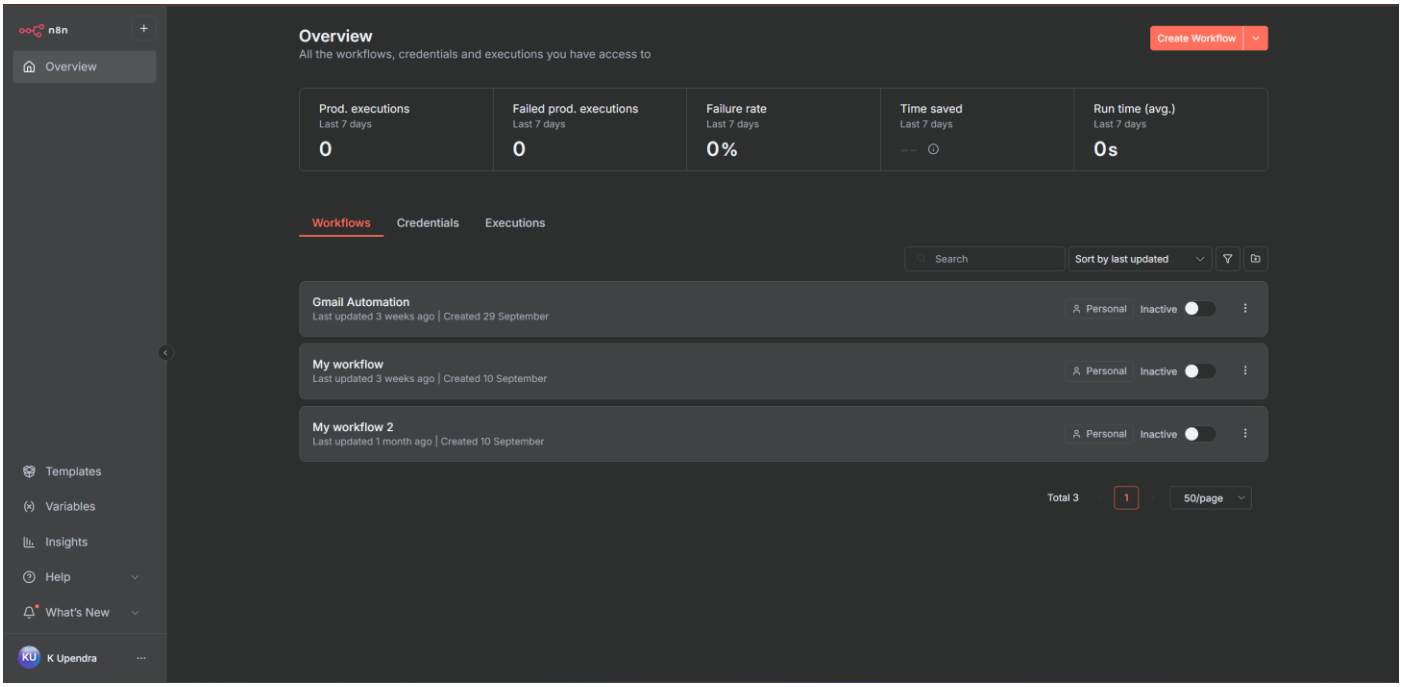


Hands-On Automation with n8n

The session then moved to practical, hands-on work building and testing automated workflows using n8n, a visual automation tool.

What is n8n?

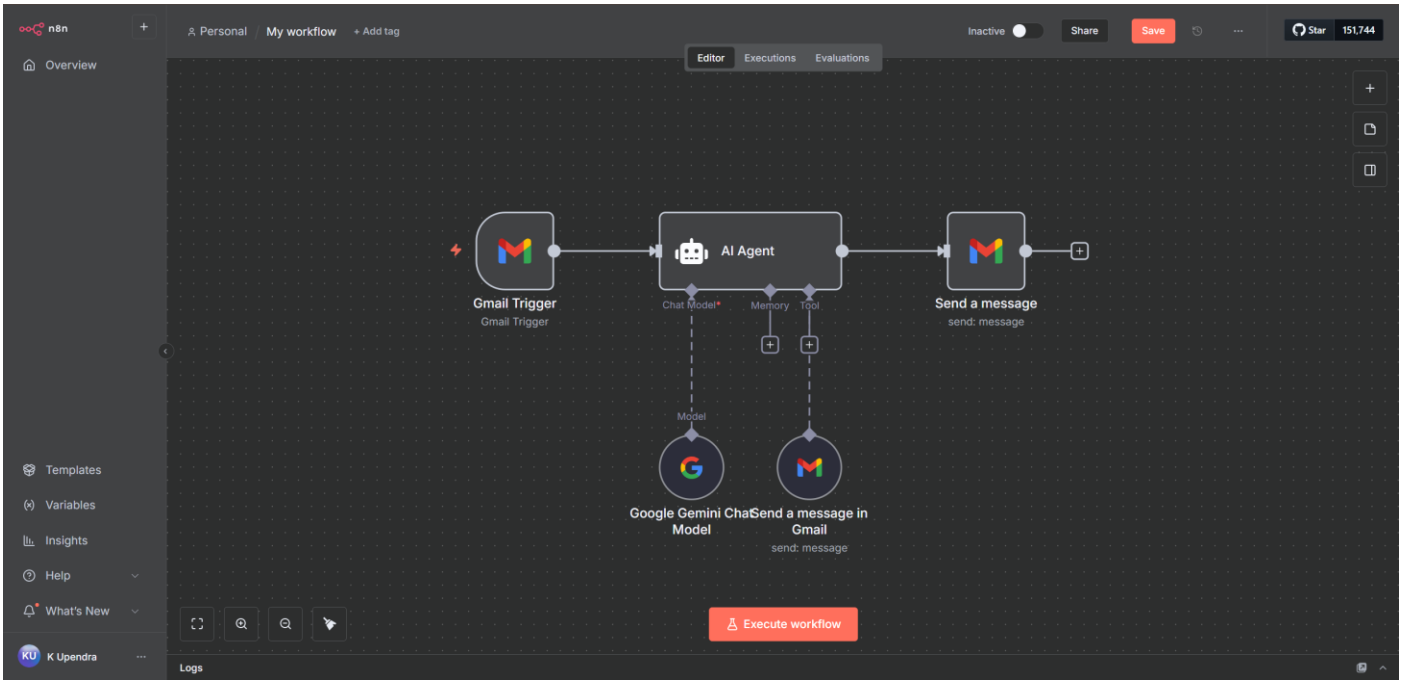
n8n is a workflow automation tool that allows you to connect different applications and automate tasks without deep coding knowledge. You connect "nodes" (which represent apps or functions) to create a sequence of actions. For example: "When I receive a new email in **Gmail** (Node 1), get the attachment and send a message to **Telegram** (Node 2)."



Google Cloud & Gmail API Integration

This lab focused on connecting a cloud service to an n8n workflow.

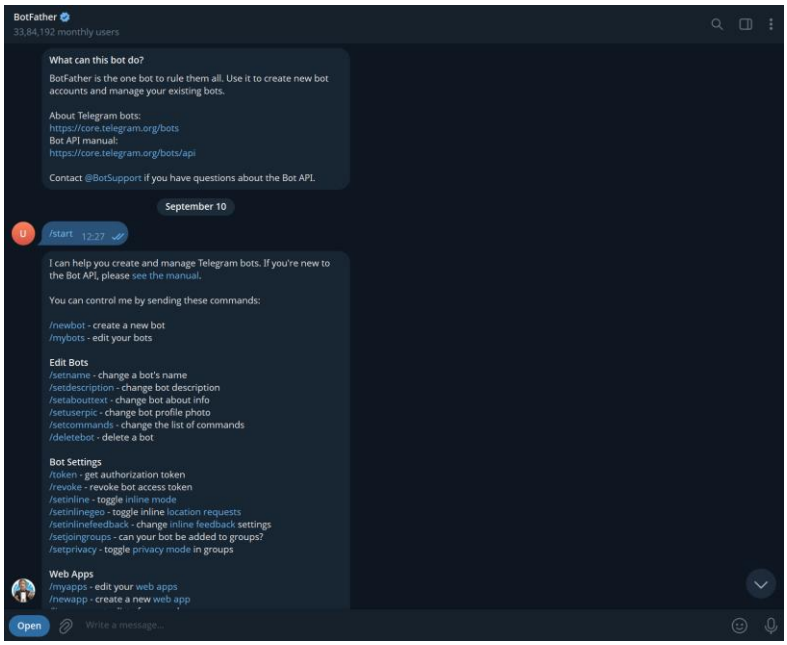
- **Step 1: Project Setup in Google Cloud.** Each participant created a new project within the Google Cloud Platform. This acts as a central place to manage resources and permissions.
- **Step 2: Enable the Gmail API.** Within the project, the Gmail API was located and enabled. This is the explicit step of giving permission for external tools to interact with Gmail data.
- **Step 3: Connect to n8n.** The credentials from the Google Cloud project were used to configure the Gmail node in n8n, establishing a successful connection.

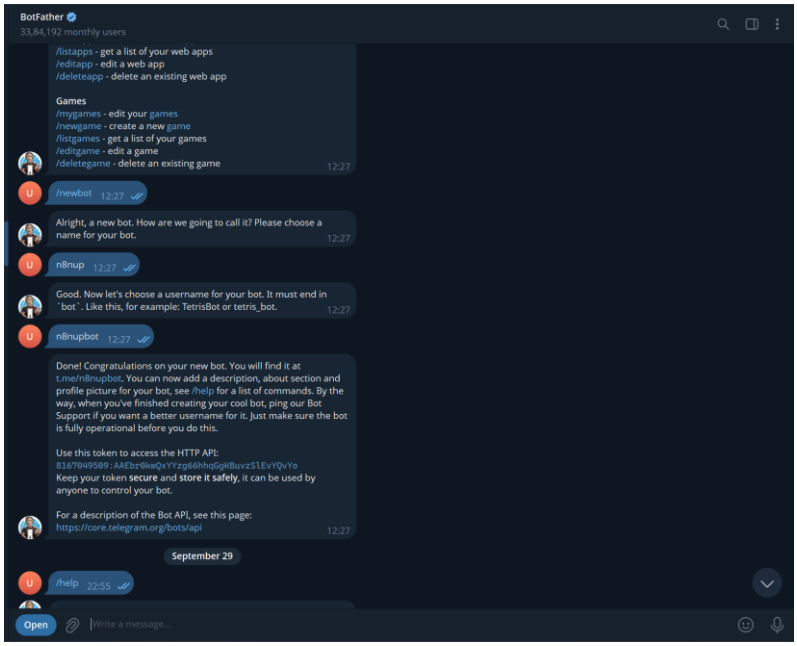


Telegram Bot and API Integration

This lab demonstrated how to integrate a messaging service.

- **Step 1: Create a Telegram Bot.** Participants created a new Telegram bot using Telegram's "BotFather".
- **Step 2: Generate an API Key.** The BotFather provides a unique API key (token) for the new bot. This key is used to authenticate and control the bot.
- **Step 3: Integrate with n8n.** The Telegram API key was used to set up the Telegram node in n8n, allowing the workflow to send and receive messages through the bot.





AI Theory to Real-Time Implementation

Topics: T-Shaped Developer Model, Agentic AI, Transformer Architecture, and n8n Workflow with Gemini AI.

The T-Shaped Developer Model

The **T-Shaped Developer** model was introduced as a framework for professional growth. Imagine the letter 'T':

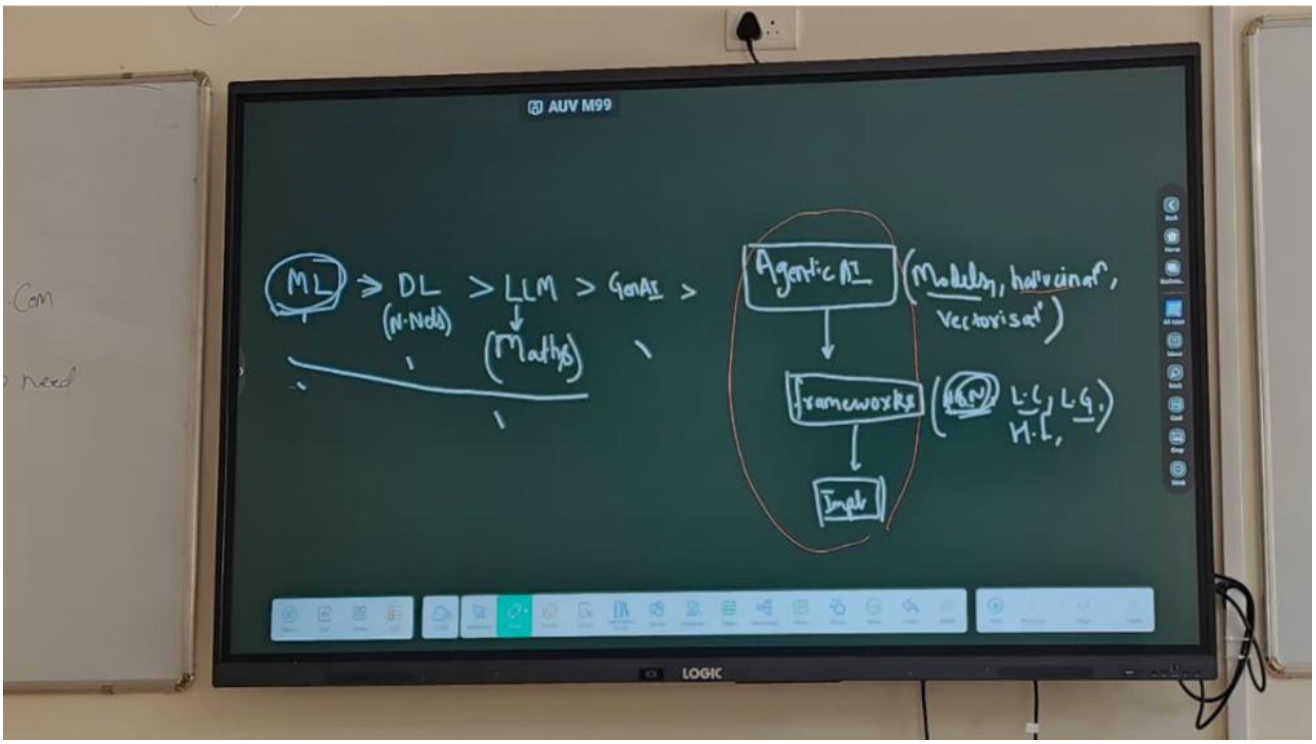
- **The Horizontal Bar**: Represents a **broad knowledge** across many different domains—like front-end development, back-end, databases, and cloud services.
- **The Vertical Bar**: Represents **deep specialization** and expertise in one specific area.

What is Agentic AI?

Agentic AI = AI that not only responds but can **act independently**, use tools, and pursue goals with minimal human prompting. The session explored how these agents rely on four key components:

1. **Input**: The data or prompt that starts the task.
2. **Chat Models**: The "brain" (like Gemini,ChatGPT) that processes information.
3. **Memory**: The ability to recall past interactions to maintain context.
4. **Tools**: The ability to use other applications or APIs to take action in the real world.

The discussion also covered advanced topics like Agentic AI frameworks, how to manage AI "hallucinations" (when an AI generates incorrect information), and vectorization.



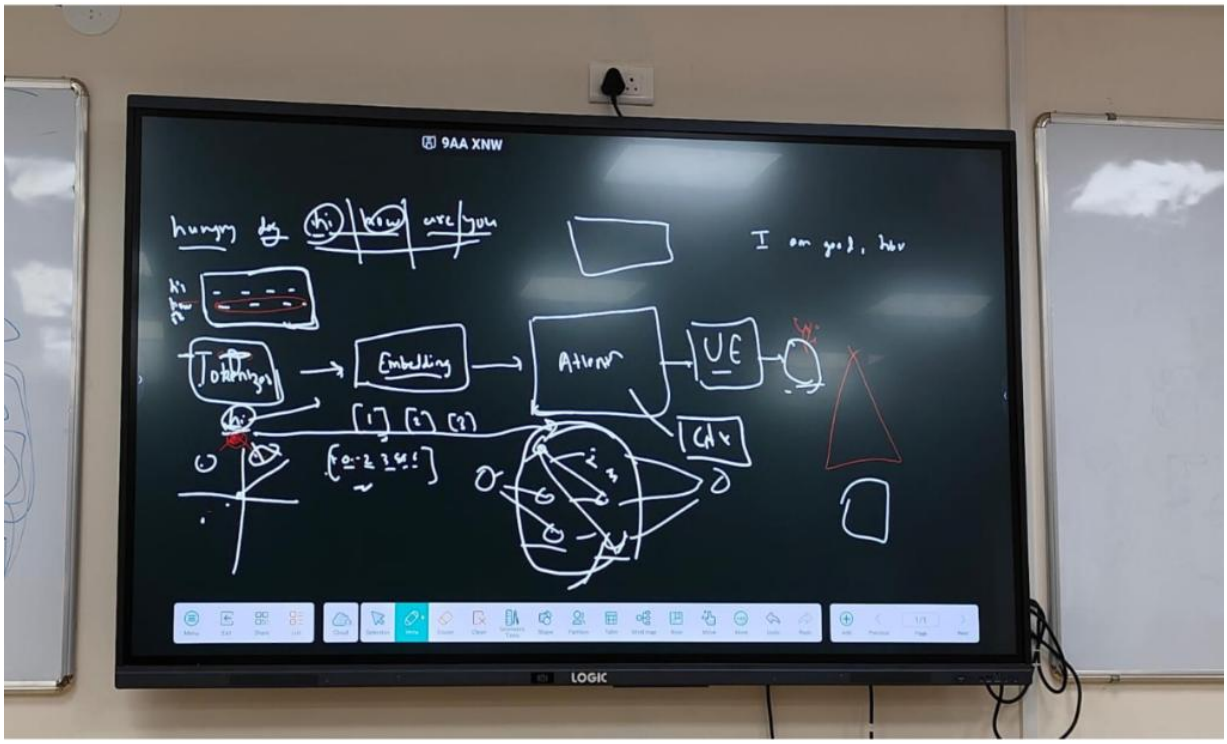
AI Brain: Transformer Architecture

To understand how models like Gemini, ChatGPT work, the session broke down the core concepts of the **Transformer architecture**.

Think of the process like a master chef preparing a complex dish from a written recipe (your prompt):

1. **Tokenization**: The chef first reads the recipe and **chops the ingredients** into small, manageable pieces. This is tokenization, where input text is broken down into machine-readable units called tokens. [8]

2. **Embedding**: The chef understands the **flavor profile and characteristics** of each chopped ingredient. This is embedding, where each token is converted into a numerical vector—a list of numbers that represents its meaning and relationship to other words.
3. **Attention Mechanism**: As the chef cooks, they **pay more attention** to key ingredients that define the dish's final taste. This is the attention mechanism, which allows the model to weigh the importance of different tokens in the input, focusing on the most relevant context.
4. **Transformer Layers**: The dish goes through **multiple cooking stations** (sautéing, simmering, baking), with each station refining the flavor. These are the transformer layers—multiple stacked blocks that process the embeddings and their relationships over and over to build a deep understanding.
5. **Unembedding**: Finally, the chef **plates the finished dish** beautifully. This is Final step = linear projection from hidden states → probability distribution over vocabulary → decoding into text to form the output.



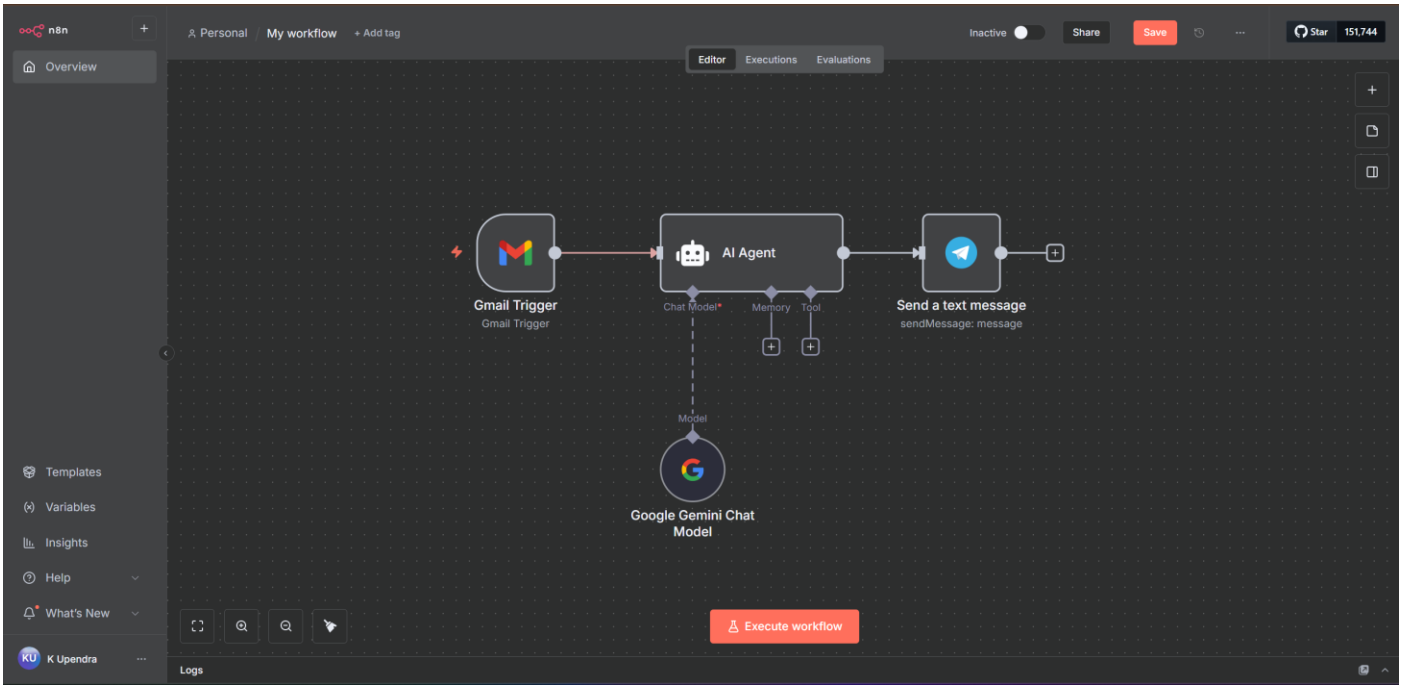
Hands-On Lab: Building a Real-Time AI Assistant

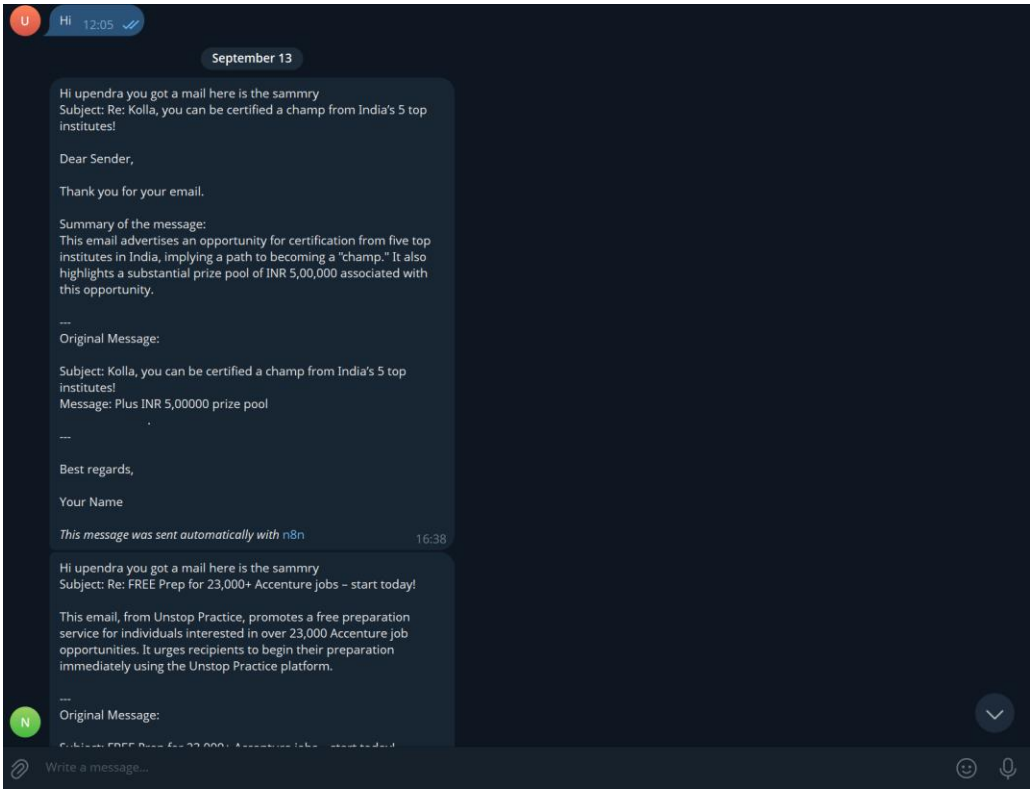
This practical lab involved building and deploying a live n8n workflow that uses AI to automate email handling.

The Workflow: **Gmail Trigger → AI Agent (powered by Google Gemini Chat Model) → Telegram**

- **Node 1: Gmail Trigger**: This node starts the entire workflow automatically whenever a new email is received.
- **Node 2: AI Agent (with Gemini)**: This is the core of the workflow. It takes the content from the incoming email and feeds it to the Gemini Chat Model. [15]The model is given a prompt (e.g., "You are an intelligent assistant. Read this email and craft a summary.") to guide its response.
- **Node 3: Telegram**: This final node takes the AI-generated summary from the Gemini model and sends it as a message to a specified Telegram chat.

The entire workflow was tested node-by-node to ensure functionality before being deployed live.





Assignments

1. Focus on **GitHub best practices**. You are to properly structure your GitHub repository, rename files logically, push all your work, and rely on CLI commands. **Crucially, ensure no API keys are pushed to the repository.**
2. Begin formal **project planning**. Create a new folder named project1-n8n and inside it, create detailed documentation outlining your project plan. This should include the tools you'll use, the workflow (flow), and the project's value proposition. After documenting, you can begin implementation.
3. Continue progress on the **full-stack application assignment**.