Introduction

Data: https://github.com/tiger-openai-hackathon/ai-build/tree/main/data.

Data Description: The data is about the recycling's role, resource conservation and environmental .

Problem statement: We want to develop an interactive AI-powered chatbot that will aid in educating users on the benefits of "Recycling" and answer questions related to the same.

Overview: We'll build a chatbot powered by Azure AI services such as Azure AI Search, Azure Open AI, and Azure AI Studio. This chatbot will be capable of ingesting any type of document, responding to questions based on only the information available in the documents, and handling multi-turn conversations. Assuming this is our first AI chatbot project, we will walk through the steps in detail. We'll use Azure AI Studio to develop this solution and deploy a sample interactive web application.

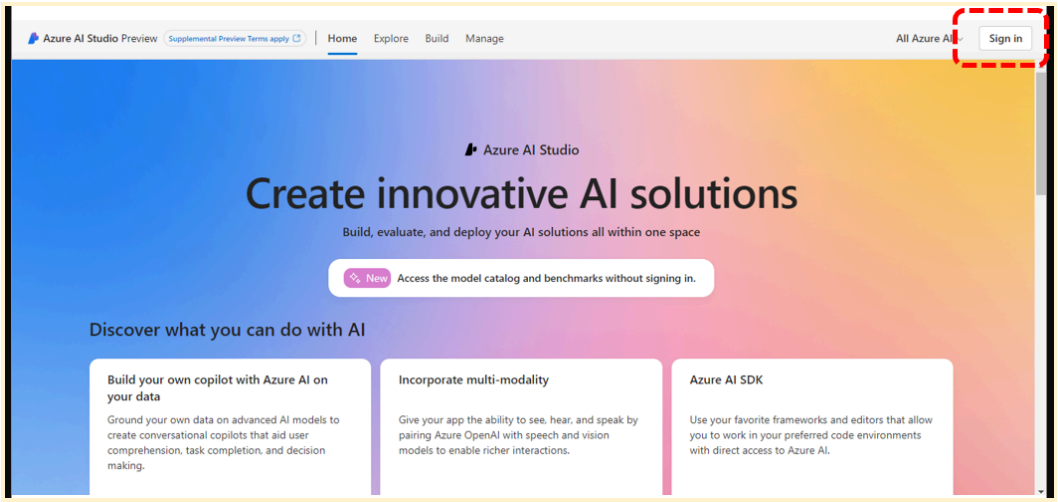
If time persists, we'll also look into how to change prompts, and parameters like top_k and what impact it will have on the responses and we will try to give you a glance at other developer features such as prompt flow, and evaluation flow.

Steps for creating your first AI project

1. Log in to the Azure AI Studio Portal

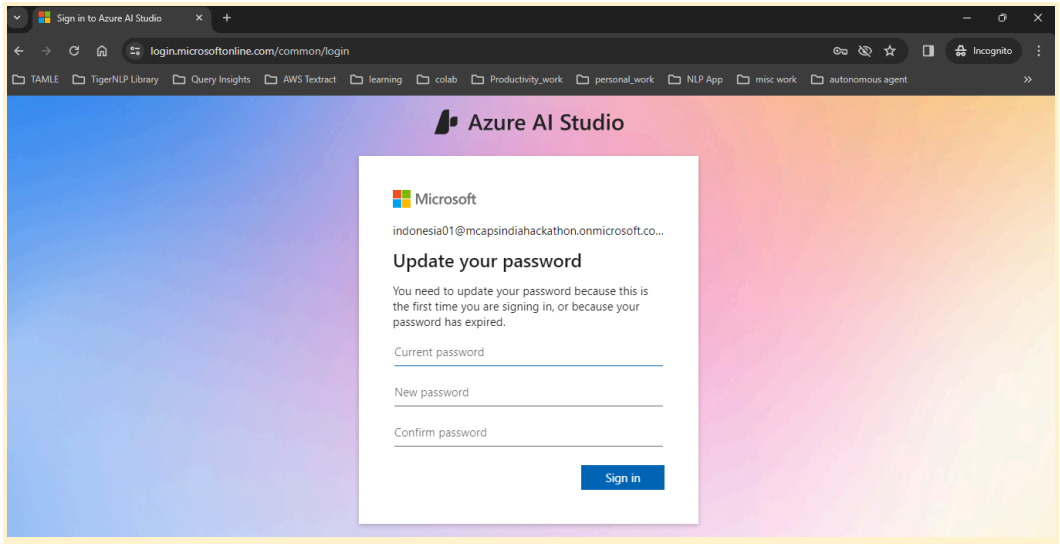
a. Please go to Azure AI Studio Portal from your browser (please use Incognito/ private mode to avoid clashes with your work profile)

b. Click on the Sign-in Button

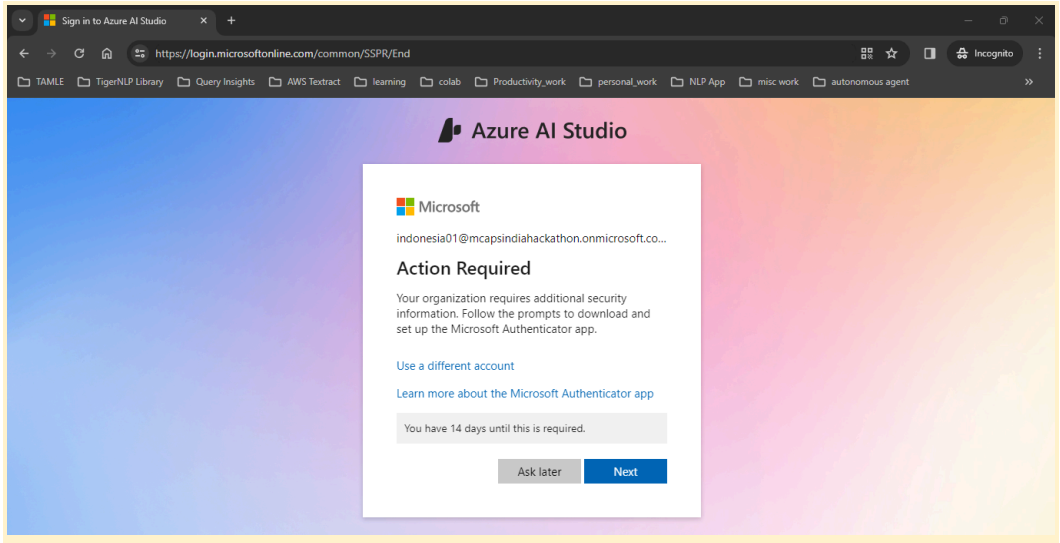


c. Enter the credentials shared with you at the registration desk. (Please keep your slip with you till the end of the event).

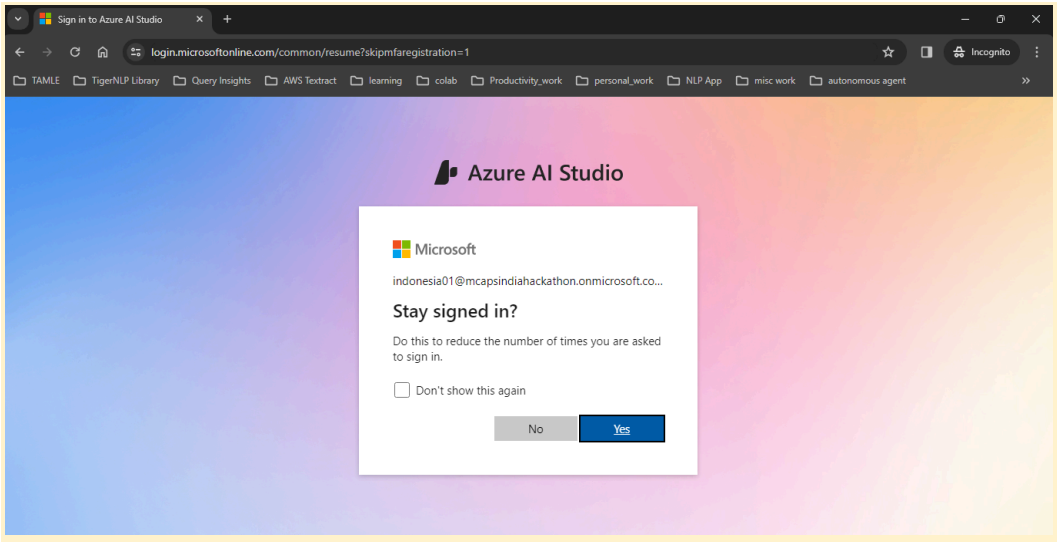
d. Setup a new password



Setup MFA/authenticator : Skip this step by clicking on **"Ask Later"**

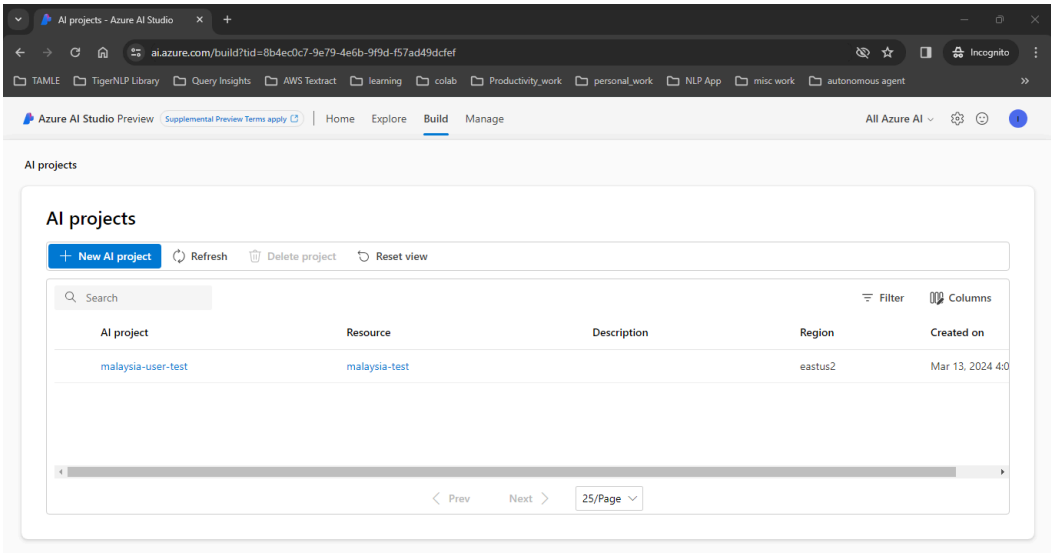


e. Click on **Yes** in the next step



2. Create a new project

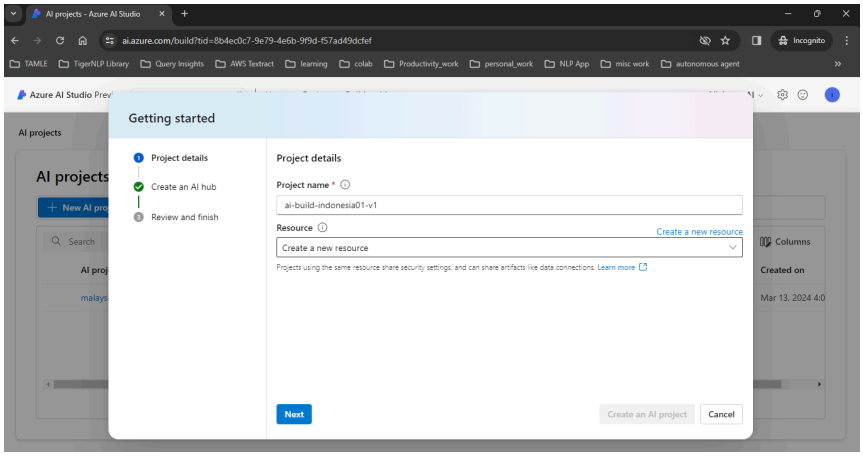
a. Go to the **Build** tab on the top and click on the **New AI Project** button



b. In the **Project details** tab

i. Add an appropriate project name (“ai-build-” + login ID + version number. Ex: “ai-build-malaysia01-v1”)

1. Under the **Resource**, click on **Create a new resource**. It should take you to **Create an AI hub** section



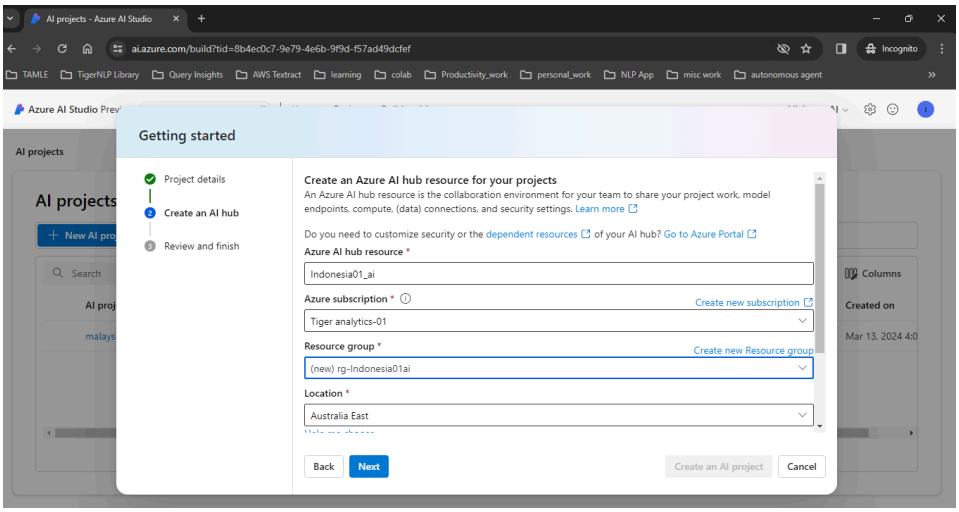
c. In the **Create an AI hub** tab

i. Under the **Azure AI hub resource**, by default it will be populated with your user ID, If not please add an appropriate resource name (userid+"_ai". Ex: "Indonesia01_ai")

ii. Under the **Azure subscription**, select the default entry (It will look like **Tiger analytics-<number>**, Ex: Tiger analytics-01**)**

iii. Under the **Resource group**, select "(new) + <the name you added in the **Azure AI hub resource**>" which will be displayed by default

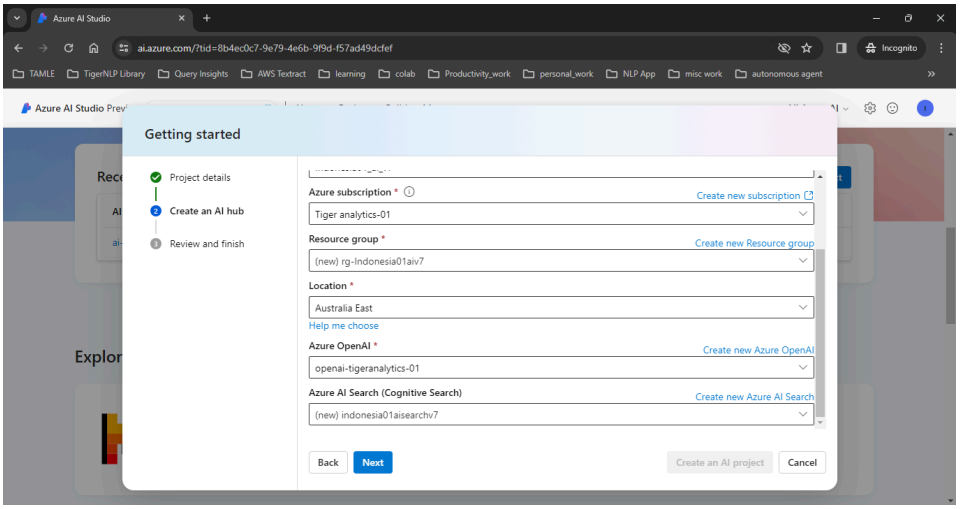
iv. Under the **Location**, select **Australia east**



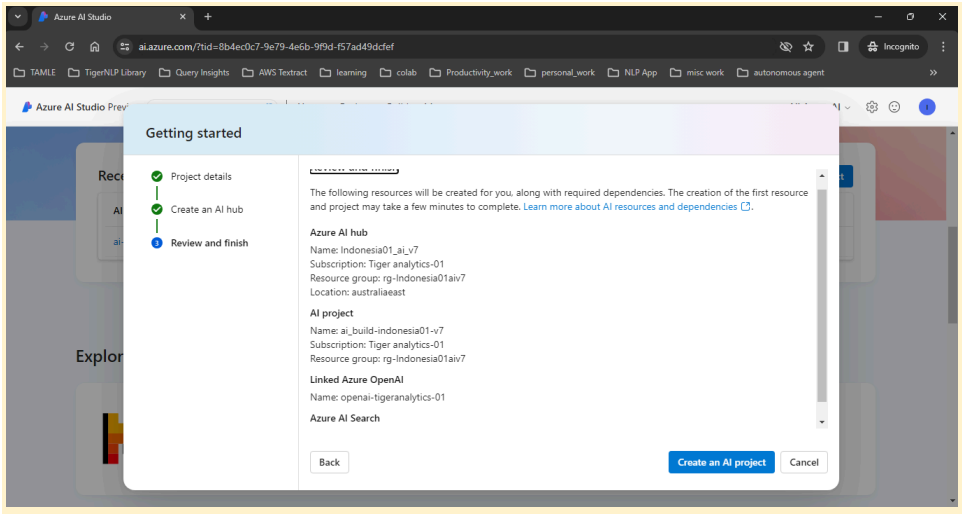
v. Under the **Azure OpenAI**, select the dropdown and select **the available Azure Open AI resource.** It will be in a format **openai-tigeranalytics-<number> Ex: openai-tigeranalytics-01**

vi. Under **Azure AI search (Cognitive Search)**, click on **"Create new Azure AI Search"**, Add the Azure AI search Name in the following format

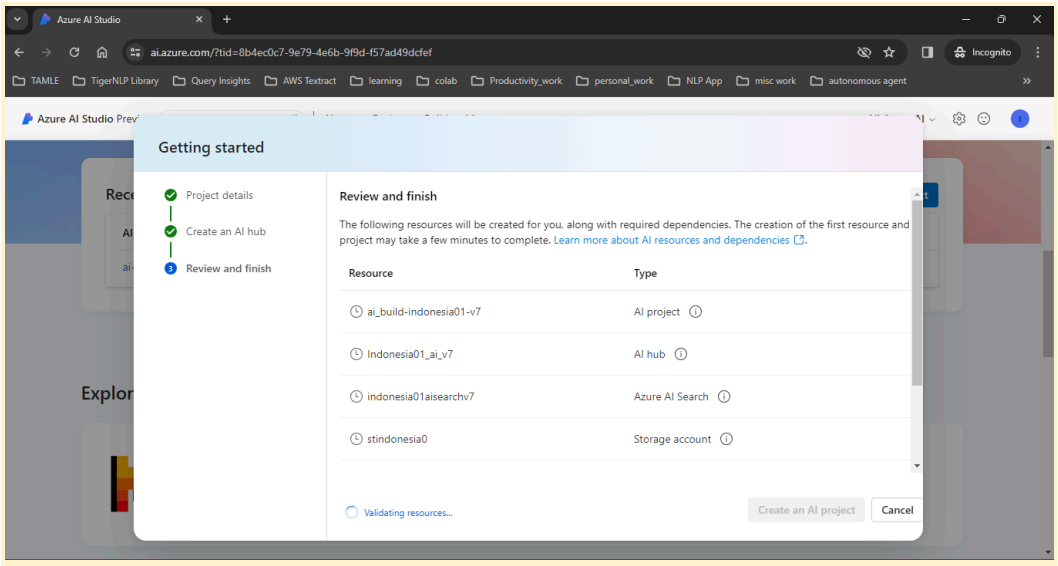
“useridaisearch”. Ex: **“Indonesia01aisearch”** and click **“Next”**



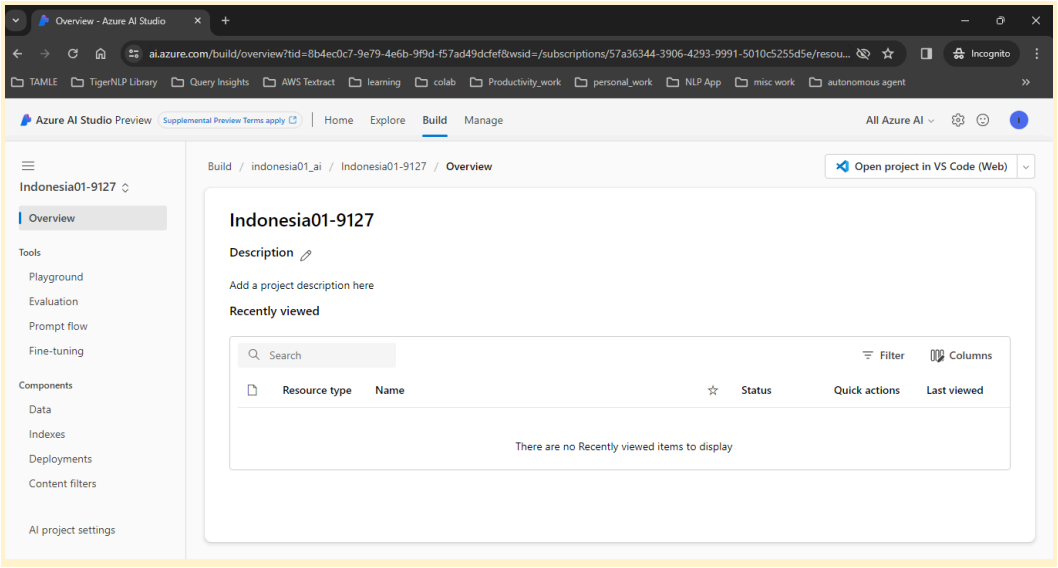
vii. Review the details and Click on **“Create an AI project”.**



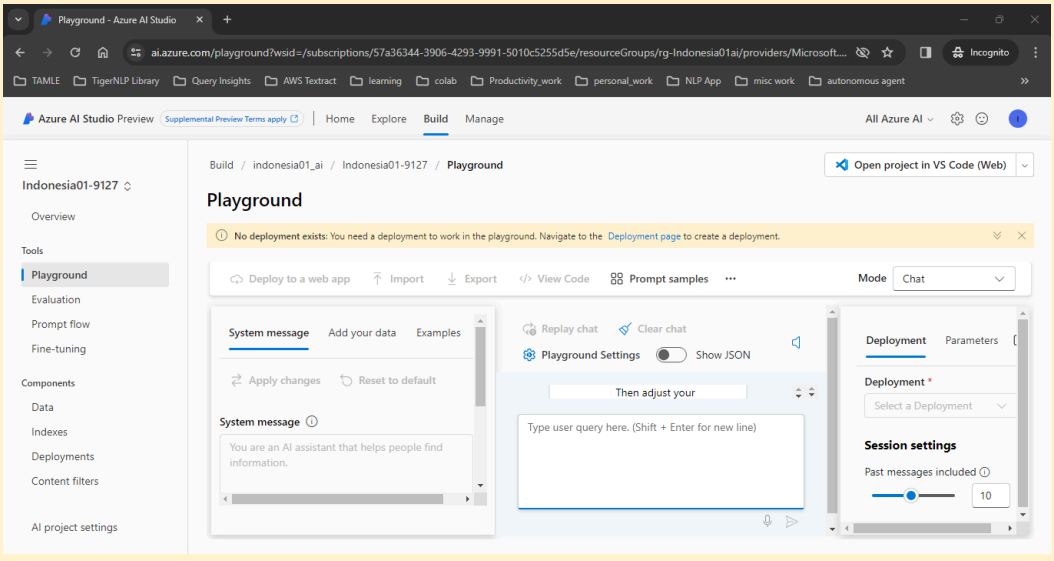
d. In the **Review and Finish** tab please wait while the Azure services are created for you. This might take a couple of minutes



e. It will automatically redirect you to the Project **Overview** page. Click on the **Playground** to start with the next step

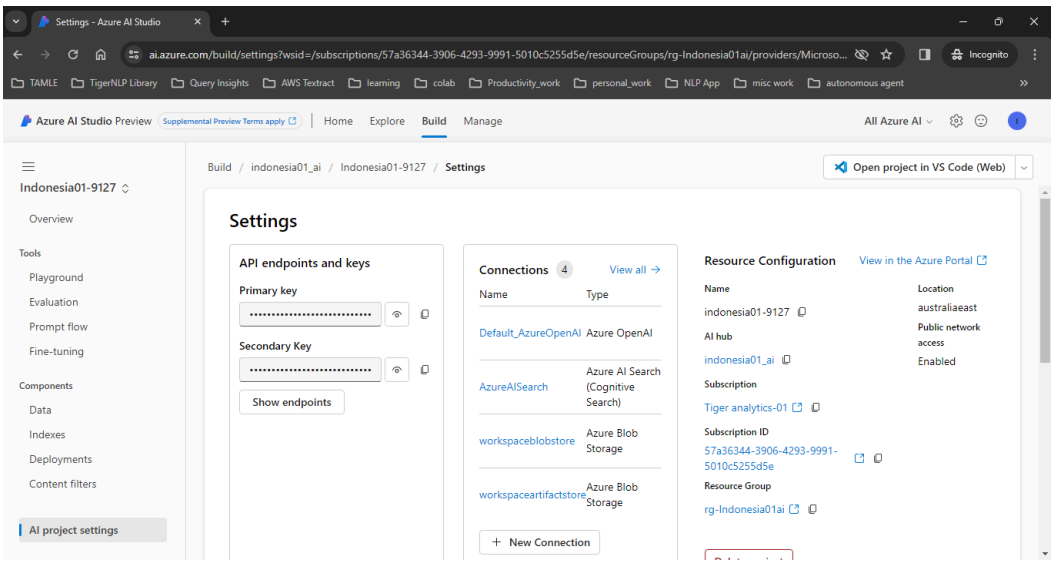


f. You will see the Playground in the screenshot below



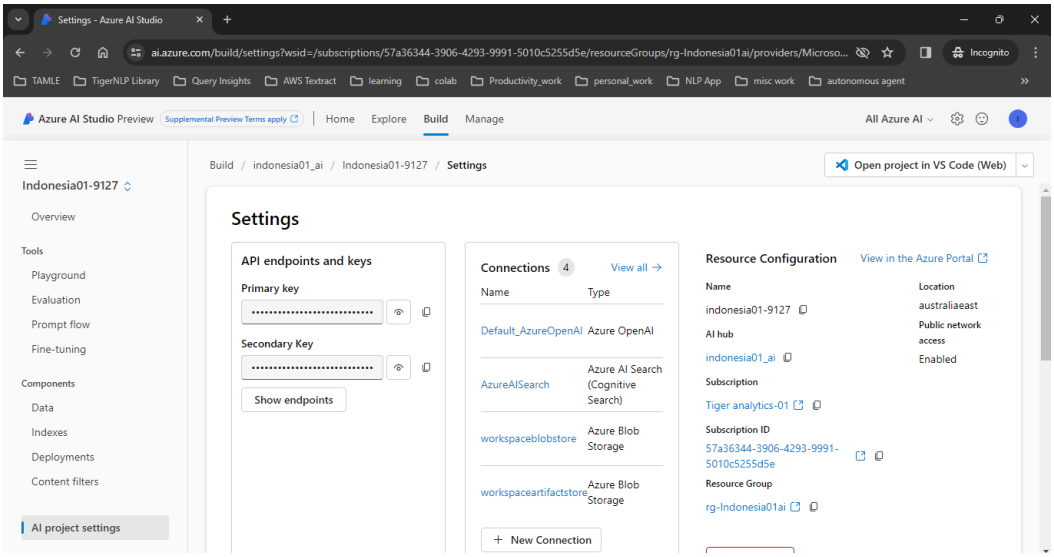
3. Add a connection to the Azure OpenAI service

a. On the left-hand side panel of the project playground, click on the option **AI Project Settings** and then click on **New connection** in the **Connections** pane



b. In the API base add the Azure Open AI endpoint and in the API key add the key present in the sheet given to you at the registration desk.

c. Add the Input Connection name in the following format **“userid+azureopenai”. Ex: Indonesia01azureopenai** and click on **Create connection**



d. In the **Session settings** under the same, select 10. This indicates how many previous conversations Azure OpenAI adds in the context while responding to the current question. 10 indicates that it retains the context from the previous 5 user questions and the system responses
e. Select **Mode** as Chat from the dropdown above the **Deployments** section
f. Add a screenshot

4. Test the connection

a. Ask a question in the playground chat box (Ex: “Tell me about Azure OpenAI services”)
b. If you get a system response, the connection is set and we’re good to go to the next step
c. Add a screenshot
d. The system is now set up to respond to general user questions. We can add our data in the backend to unlock the power of Azure AI studio, which is the ability to generate responses from the sources provided by the user

5. Add your data

a. Click on **Add your data** and click on **Add a data source**
b. In the **Data source** tab
    i. Under the **Select data source** dropdown, click on **Azure Blob Storage (preview)**
    ii. Under **Subscription**, select **Microsoft Azure Sponsorship**
    iii. Under **Select Azure Blob storage resource**, select Add blob name where the data is already uploaded for this event. You can also find the data here
    iv. Under **Select storage container**, select the Add container information here
    v. Under **Select Azure AI Search resource**, click on Abin Joseph / Pushvinder Rohtagi - please add what to do here

vi. Under **Enter the index name**, add an appropriate name to your index ("data-index-" + id + your name + version, Ex: "data-index-malaysia001-srikanth-v1")
vii. Under **Indexer schedule**, select **Once**
viii. Check on **Add vector search to this search resource**
ix. Under **Select an embedding model**, select **Azure OpenAI - text-embedding-ada-002**
x. Check on I acknowledge that connecting to an Azure AI Search account will incur usage to my account
xi. Click on Next
xii. Add a screenshot

c. In the **Data management** tab
   i. Under the **Search type** dropdown, select **Hybrid (vector + keyword)**
   ii. Check on I acknowledge that adding vector embeddings account will incur usage to my account
   iii. Click on Next
   iv. Add a screenshot

d. In the **Review and finish** tab
   i. Review the storage details to check everything is correct and click on **Save and close**
   ii. Add a screenshot

e. Please wait while the data is indexed and stored in the database
   i. You should see a message on the screen saying Ingestion in progress
   ii. Please wait while the Preprocessing of the files and the indexing are completed
   iii. Add a screenshot
   iv. Once this is done, you should see your index in the **Indexes** tab on the left-hand side. The status should say **Ready** with a green check mark next to it
   v. Add a screenshot

f. Add a screenshot

6. Create an index for your data

a. Add details on what indexing is
b. You'll see a green play button
c. Wait till it completes (add what it shows after it finishes)
d. I forgot the next step, but it was doing something after the index
e. Wait till it completes (add the next step that comes up also)

7. Prompt

a. Define the system prompt in the prompt box and click on **Apply changes** for the changes to be reflected in the system responses
b. Sample prompts can be found in **Prompt Samples**

8. Start asking questions

a. Check on the responses from the data and ask questions. Some sample questions on the data can be found here (Please click on **view raw**, and it will download the Excel to your local machine).
b. Please make sure to click on the **Clear chat** button before asking an unrelated follow-up question in the chat box
c. Add a screenshot

9. Tweak the parameters

a. If you want the responses to come not only from the source data alone but also from the external knowledge of the Azure OpenAI models, you can click on **Advanced settings** under **Add your data** tab and uncheck **Limit responses to your data content**
b. Strictness and Retrieved documents
c. Tweak the model parameters from the **Parameters** tab Add a link to the definitions of parameters
d. Make changes to the prompts to change the response to the way you like (Ex: We can add to give the response in bullet points or a happy tone). Remember to click on **Apply changes** after changing the prompt. You can click on the **Replay chat** button to get the responses to the existing questions on the updated prompt
e. Define variables by using the **Add variable** section for easy prompt changes. For example, tone can be set as a variable in the prompt whose value can be changed easily across different runs. Add screenshots
f. You can also **import** and **export** the chat settings (prompts, examples, parameters) to a json. You can try the same using this json (add json link). This feature is useful for sharing your workspace settings while working as a team
g. You can also click on View Code to get an executable code in the programming language of your choice with the Azure OpenAI, data, and index connections set already

10. Deploy the model as a web application

a. Click on “Deploy as a web App”
b. Enter a name for the web app
c. Select the subscription, resource group, and location (please refer to the slip)
d. Select the “Free” pricing plan
e. Check the acknowledgment related to pricing
f. Hit deploy and wait for a few minutes (this might take ~10 minutes) till the deployment is complete
g. Check the link available somewhere and open it in your browser
h. You can start asking questions in the web application. People who have access to your AI Studio project will also be able to access the application via URL and ask questions. All the logs, user questions, and system responses are stored in a database

11. Playground settings
    a. You can change the language, and enable speech-to-text, and text-to-speech capabilities in the Playground settings tab. These use Azure AI services in the backend
12. Open project in VS Code (Web)
    a. You can also view the entire project in VSCode web IDE and get responses in the terminal
13. Show JSON
    a. You can look at the JSON of the chat transcript. This is formatted for the API calls
14. Prompt flow
15. Evaluation
16. Examples

Note: The Azure service accesses used in this event are only valid during the event. Please contact your respective Microsoft account teams to follow up on the continued access or further questions.