REST API – Automated Email sending

**Purpose:**

The aim of this automation is to seamlessly integrate emails notifications into data engineering pipelines, enabling automatic delivery of reports, dashboards, or insights to recipients. By leveraging Databricks and OAuth authentication, this solution ensures secure, efficient, and fully automated email distribution without manual intervention.

A diagram of a computer software

AI-generated content may be incorrect.

**1. Azure App Registration**

* **What Happens:**
  + An **Azure App Registration** is created in **Entra ID.**
  + The app is granted necessary **permissions (Mail.Send)** to send emails on behalf of the authenticated user (see below for information on user details)
* **Key Credentials:**
  + **Tenant ID:** Identifies the Azure AD instance.
  + **Client ID:** Uniquely identifies the registered application.
  + **Client Secret:** Used for authentication in OAuth workflows.

**2. OAuth Authentication**

* **What Happens:**
  + The OAuth flow is initiated using the credentials.
  + A **Python script** is used to authenticate and retrieve an access token from **Microsoft’s Identity Platform**.
  + The token is used to authorize API calls.

**3. Databricks Processing**

* **What Happens:**
  + The **Python script** is executed within **Databricks**, leveraging the access token to send emails via **Microsoft Graph API**.
  + The email content, recipients, and attachments are dynamically processed.

**4. Azure Services & Email Distribution**

* **What Happens:**
  + Databricks sends the email to **Azure services** (Enterprise App).
  + The email is delivered to the designated recipients.
  + Email distributor
    - TBC on name
    - Type of account: Cloud
    - Not to be setting using an account which is hybrid due to API restrictions
    - Account to be a cloud account only to make use of Exchange Rest API.

**5. Email Delivery**

* **What Happens:**
  + The email is successfully sent to users.

**Testing**

1. Ensure the Azure App Registration has the correct API permissions
2. Verify the Client ID, Tenant ID, Client Secret are access securely
3. Test the MFA on the account
4. Test OAuth Connection – Ensure status code = 200
5. Send a test email with attachement
6. Send test email with no attachement
7. Send test email to 1 recipient
8. Send test email to 2 or more recipients
9. Validate email delivery – Ensure it is received (if not checks logs)

**Security implementations**

* Email account password will be securely stored in **Bitwarden**
  + Only access by Data Engineers
* IDs (Client ID, Tenant ID, Client Secret) to be stored in **Azure Key Vault**
  + Only access by Data Engineers (RBAC secured)