**Authentication and Role Assignment Flow**

This document outlines the authentication process and access control logic for users logging in via Microsoft AD, including token handling, user registration, and role assignment.

**1. Login and Authentication Flow**

* **Frontend Initiation:**
  + The frontend requests a Microsoft AD login URL from the backend.
  + The backend generates the login URL and returns it to the frontend.
* **User Authentication:**
  + The user is redirected to the Microsoft login page.
  + Upon successful authentication, Microsoft redirects back to a predefined callback URL with an authorization code.
* **Token Exchange and User Info Retrieval:**
  + The frontend sends the redirect URL (containing the code) to the backend.
  + The backend exchanges the code for access and refresh tokens.
  + Using the access token, the backend retrieves the user's profile information (e.g., name, email, AD ID).

**2. Token Handling and Validation**

* **Token Management on the Client:**
  + Access and refresh tokens are issued to the frontend upon successful authentication.
  + The **access token** is used to authenticate API requests from the frontend to the backend.
  + The **refresh token** is used by the frontend to obtain new access tokens when the existing one expires.
  + **Best practice:** store tokens securely on the client side (e.g., access token in memory or HTTP-only cookie, refresh token in HTTP-only cookie).
* **Per-Request Token Validation:**
  + For every protected API call, the backend validates the access token using Microsoft’s public signing keys (JWKS).
  + This ensures the authenticity of the token and verifies claims such as expiration and intended audience.

**3. User Registration and Access Initialization**

* **User Lookup and Creation:**
  + After authentication, the backend checks if the user exists in the internal database.
  + If the user is new, they are registered with basic profile information and default access settings.
* **Country and Brand Mapping:**
  + A CSV file provided by Bart defines the mapping between users and their associated countries and brands.
  + This file is regularly updated and referenced to assign appropriate data access to the user.

**4. Role Assignment Process *(as discussed with Tuba)***

* **Default Role:**
  + Upon first login, users are assigned the default lowest-permission role.
* **BI Analyst Role Upgrade:**
  + Users have the ability to upgrade their role to **BI Analyst** if needed.
* **Restricted Role Assignment:**
  + The **Senior Manager** role is limited to users listed in a CSV file provided by Tuba.
  + A separate CSV, also from Tuba, defines users who should have **HITL (Human-in-the-Loop)** privileges.
  + These roles are assigned only if the user appears in the respective approved lists.

**5. Security Considerations**

* All communication between frontend, backend, and Microsoft AD is secured via HTTPS.
* Token validation is enforced on each request to ensure integrity and prevent unauthorized access.
* Role and access configurations are centrally managed and reviewed periodically via CSV-based governance.

**API Design Overview**

**1. Framework and Performance**

* **Framework:** FastAPI is used as the primary web framework.
* **Asynchronous Endpoints:** All routes are implemented using async functions to maximize throughput and handle high concurrency.

**2. Modular LLM Querying**

* **Separation of Concerns:** The logic for querying large language models (LLMs) is encapsulated in dedicated modules.
* **Prompt Engine:** A pluggable prompt-engine component allows easy customization and reuse of prompt templates.
* **Adapters:** Support for multiple LLM providers via adapter interfaces, enabling seamless provider switching.