Microsoft Azure Guide – Mentor Management

Overview

The project uses Microsoft Azure OAuth for logging in through university Microsoft accounts. For authentication to work, the Microsoft Client Secret must be valid and correctly set inside the backend codebase—not just created on Azure.

Whenever a token expires, you must generate a new one and update it in the backend.

1. Generate a New Microsoft Azure Token

1. Log in to Azure Portal.

2. Go to: Home → App Registrations.

3. Search for the application named "PWEB".

4. Open it, then navigate to: Manage → Certificates & Secrets (Tokens and Secrets).

5. Create a New Client Secret.

6. Copy the value of the new token (you will not be able to view it again later).

Note: Creating the token alone does not update the app. You must insert it into the codebase.

2. Redirect URLs (Authentication Settings)

Under Manage → Authentication in the Azure App Registration, you must configure the redirect URLs. These URLs tell Microsoft where to send the user after successful login.

The project uses the following redirect URLs:

- https://127.0.0.1:8000/authapp/microsoft/callback/

- https://10.8.0.56:8000/authapp/microsoft/callback/

- https://34.170.92.44:8000/authapp/microsoft/callback/

Each environment (local development, internal network, and production server) must be registered here so authentication works properly across all deployment setups. If you are not using one of these sets of IP and Port, the authentication and login will not work.

*More redirect URLs can be created. Based on current URLs as of 12/1/25.

3. Update the Token in the Backend

After generating the new token:

1. Navigate to the backend folder in the project:

```
 Fall-2025-Capstone-MMS/
  └── backend/
      └── myproject/
```

2. Open settings.py.

3. Scroll to line 182 (or search for MICROSOFT_CLIENT_SECRET).

4. Replace the existing value with the new token you copied.

Example:

```
MICROSOFT_CLIENT_SECRET = "your_new_token_here"
```

5. Save and exit.

4. Completion

Once the new secret has been added to settings.py and the Redirect URLs are correctly configured in Azure, the authentication system is fully updated. Users should now be able to log into the frontend using Microsoft OAuth.