Setting Up OAuth for Gmail and Obtaining Credentials

Step 1: Create a Project in Google Cloud Console

1. **Access Google Cloud Console**: Go to <u>Google Cloud Console</u>.

2. **Create a New Project**: Click on the project dropdown near the top of the page and then click on "New Project". Give your project a name and click "Create".

Step 2: Enable Gmail API

1. **Select Your Project**: Make sure your new project is selected in the top bar.

2. **Enable APIs and Services**: Navigate to "APIs & Services > Dashboard" and click on "+ ENABLE APIS AND SERVICES".

3. **Search for Gmail API**: In the API Library, search for "Gmail API" and select it.

4. **Enable Gmail API**: Click the "Enable" button to enable Gmail API for your project.

Step 3: Create Credentials

1. **Navigate to Credentials**: Go to "APIs & Services > Credentials" in the sidebar.

2. **Configure the OAuth Consent Screen**:

   - Click on "CONFIGURE CONSENT SCREEN".

   - Select "External" for User Type and click "Create".

   - Fill in the required fields like App name, User support email, and Developer contact information.

   - Click "Save and Continue" and go through the scopes and test users sections, adding what's necessary for your application.

3. **Create Credentials**:

   - In the Credentials tab, click on "+ CREATE CREDENTIALS" at the top and select "OAuth client ID".

- Choose "Web application" as the Application type.

- Click "Create". You will then see your client ID and client secret.

Step 4: Download Credentials JSON

1. **Download the JSON File**:

   - In the Credentials tab, find the OAuth 2.0 Client IDs section.

   - Click the download button (an icon of a down arrow) on the right side of the newly created client ID.

   - Save this JSON file as `credentials.json` in your project directory.

Step 5: Use Credentials in Your Application

1. **Integrate with Your Application**:

   - Include the `credentials.json` file in your project.

   - Use this file to initialize the OAuth process in your application code.

Step 6: Testing and Verification

1. **Run Your Application**: Execute your application. On the first run, it should prompt you to log in with your Google account and consent to the requested permissions.

2. **Verify Token Generation**: After consenting, a token should be generated, which your application will use to authenticate future requests.

Note: Keep your client secret confidential. For a production application, you will need to go through the verification process to have your application verified by Google, especially if you are accessing sensitive scopes.