

# Phase 7: Integration & External Access

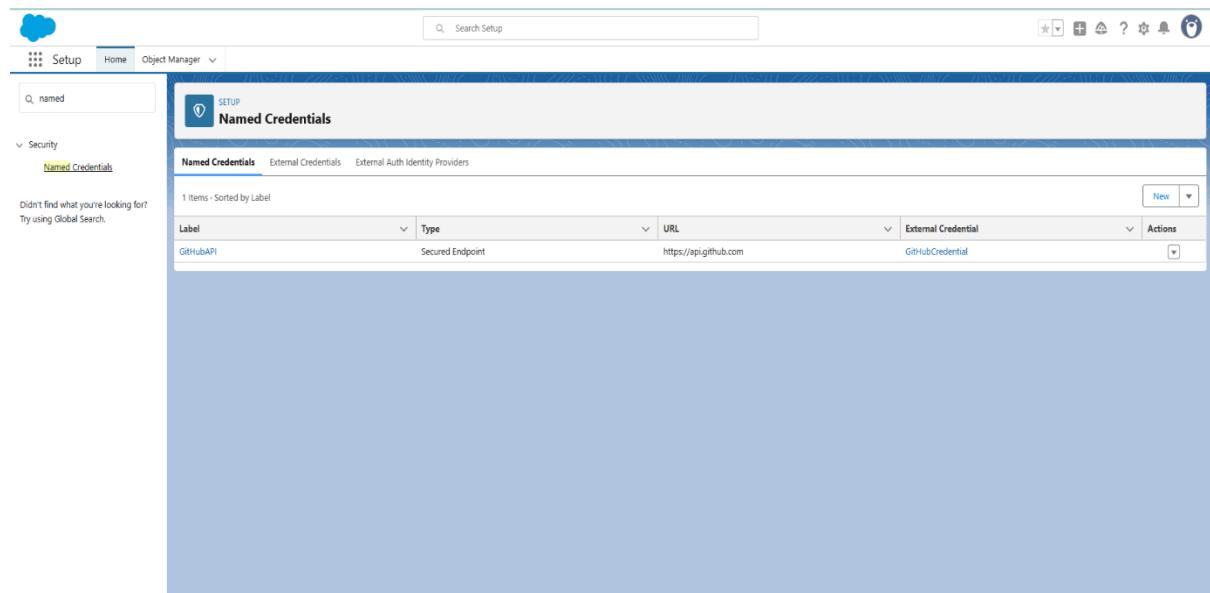
## 1. Named Credentials

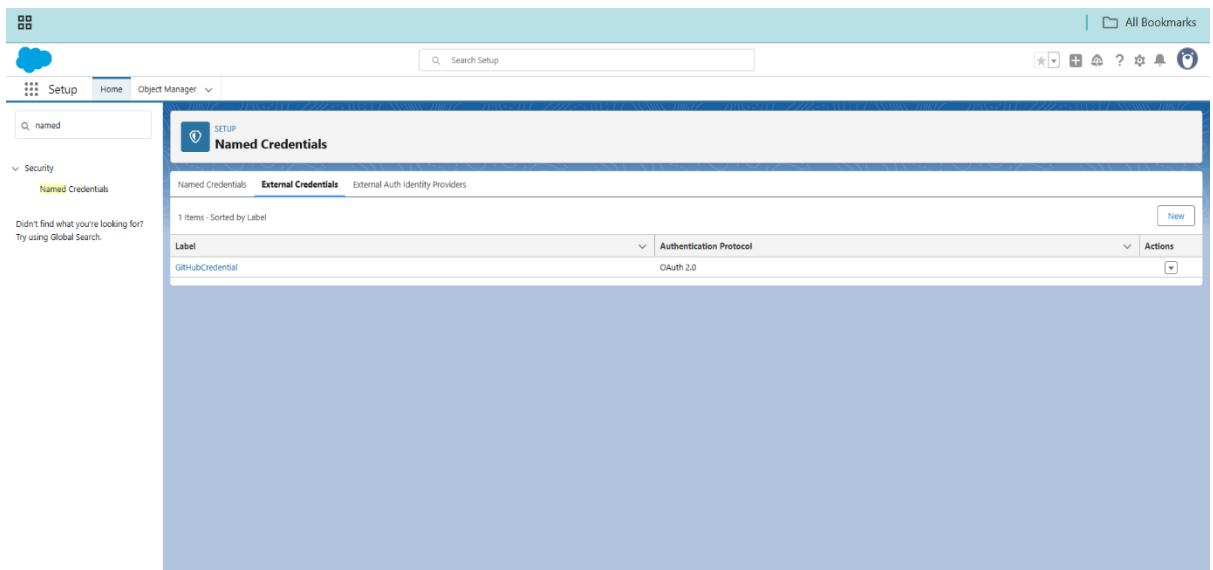
### Purpose:

Named Credentials provide a secure way to store authentication details for external services and simplify callouts from Apex or Flows.

### Implementation Details:

- **Named Credential Name:** GitHubAPI
- **External Credential:** GitHubExternalCredential
- **Authentication Protocol:** OAuth 2.0
- **Authentication Provider:** ExternalAuthIDE
- **URL:** <https://api.github.com>
- **Scopes:** read:user user:email
- **Callout Enabled:**  Yes
- **Test:** Verified using Apex test class NamedCredentialTester → received HTTP 200 response.





## 2. External Services

### Purpose:

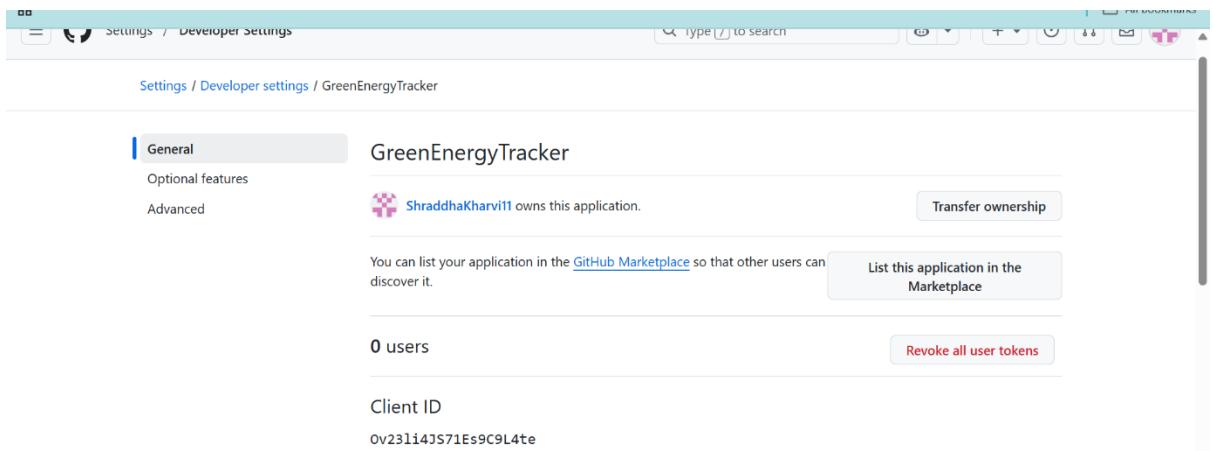
Integrate REST/SOAP services declaratively using Salesforce External Services without writing Apex code.

### Implementation in Project:

- Registered external services (e.g., GitHub APIs, other REST endpoints) using OpenAPI specification.
- Enabled declarative actions in Flows and Apex for automation.

### Outcome:

External Services are available in Flows for automated CO<sub>2</sub> data fetch and reporting.



### **3. Web Services (REST/SOAP)**

#### **Purpose:**

Expose Salesforce data to external systems or fetch data from external APIs.

#### **Implementation:**

- **REST API:**
  - Apex REST endpoints for CarbonFootprint\_\_c and Subscription\_\_c objects.
  - Methods: GET (fetch), POST (create), PATCH (update).
- **SOAP API:**
  - Exposed for legacy integration systems if needed.
  - WSDL generated and tested via SOAP client.

#### **Outcome:**

External systems can securely read/update CO<sub>2</sub> and subscription data.

### **4. Callouts**

#### **Purpose:**

Programmatically access external APIs from Apex.

#### **Implementation:**

- Named Credential used for authentication.
- Apex example for GitHub:

```
HttpRequest req = new HttpRequest();
req.setEndpoint('callout:GitHubAPI/user');
req.setMethod('GET');

Http http = new Http();
HttpResponse res = http.send(req);
System.debug(res.getBody());
```

## **6. Change Data Capture (CDC)**

### **Purpose:**

Track changes to Salesforce records in real-time.

### **Implementation:**

- Enabled CDC on **CarbonFootprint\_\_c** and **Subscription\_\_c** objects.
- Listeners capture create/update/delete events.
- Used in combination with Platform Events for analytics dashboards.

### **Outcome:**

All critical object changes are captured in real-time for integration.

## **7. OAuth & Authentication**

### **Purpose:**

Secure authentication for external system access.

### **Implementation:**

- Auth Provider: ExternalAuthIDE (GitHub OAuth 2.0)
- Callback URL: <https://<org>.my.salesforce.com/services/authcallback/ExternalAuthIDE>
- Scopes: read:user user:email
- Named Credential linked to External Credential for OAuth token management.

### **Outcome:**

Users can authenticate securely; Apex callouts automatically handle OAuth tokens.

## **8. Remote Site Settings**

### **Purpose:**

Allow Salesforce to access external endpoints not covered by Named Credentials.

### **Implementation:**

- Added <https://api.github.com> as Remote Site (required if Named Credential not used).
- Verified connectivity via test Apex callouts.

### **Outcome:**

External services accessible securely; callouts succeed without errors.