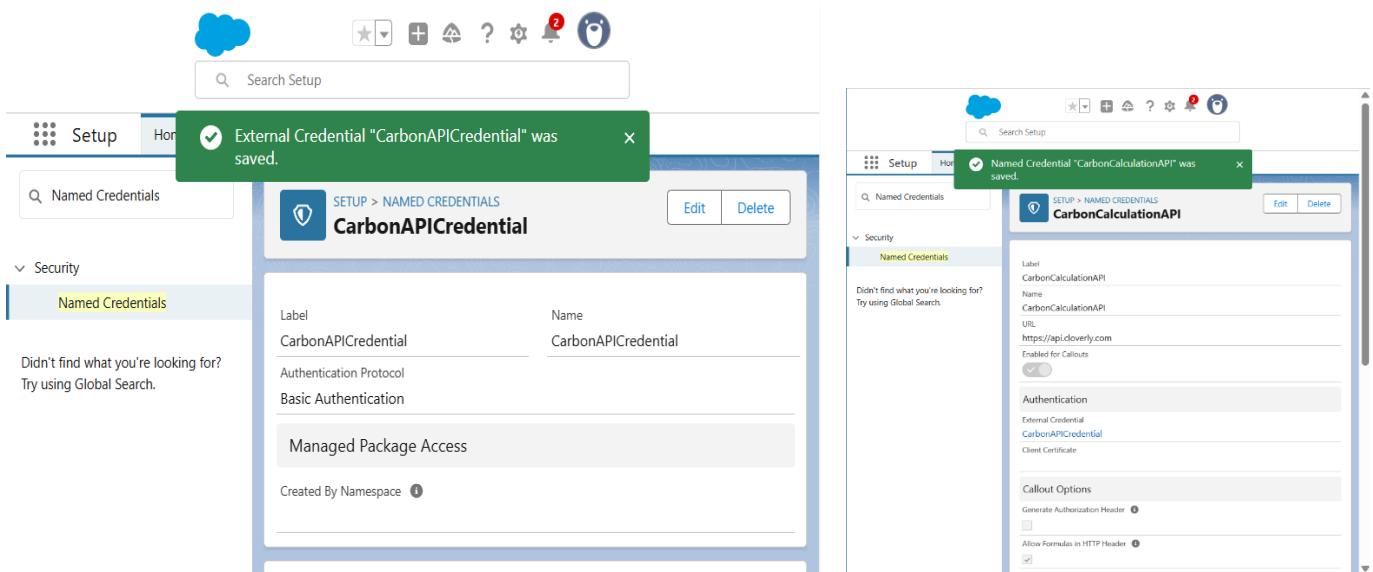


# Carbon Footprint Tracker

## Phase 7: Integration & External Access

### 1. Named Credentials

- CarbonCalculationAPI configured for internal API endpoints
- Authentication protocol set to OAuth 2.0 for secure access
- Named credential properly mapped to organization instance URL
- Security settings configured for internal service communication



The image contains two side-by-side screenshots of the Salesforce Setup interface.

**Screenshot 1: Creating a Named Credential**

This screenshot shows the "Named Credentials" page under the "Security" section. A green success message at the top says "External Credential 'CarbonAPICredential' was saved." The page displays a table with one row for "CarbonAPICredential". The columns are "Label" (CarbonAPICredential) and "Name" (CarbonAPICredential). Under "Authentication Protocol", it is set to "Basic Authentication". There is also a "Managed Package Access" section and a "Created By Namespace" section.

**Screenshot 2: Configuring a Named Credential**

This screenshot shows the "CarbonCalculationAPI" named credential configuration page. It includes sections for "Label" (CarbonCalculationAPI), "Name" (CarbonCalculationAPI), "URL" (https://api.cleverly.com), "Enabled for Callouts" (checked), "Authentication" (External Credential: CarbonAPICredential, Client Certificate), and "Callout Options" (Generate Authorization Header checked, Allow Formulas in HTTP Header checked).

### 2. External Services

- Internal mock services implemented instead of external dependencies
- CarbonEmissionAPI serves as primary integration endpoint
- OpenAPI specification created for service documentation
- Service operations mapped to internal business logic

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

- CarbonEmissionAPI REST service successfully developed and deployed
- @RestResource annotation with urlMapping '/carbon/emissions/\*'
- GET method implemented for retrieving carbon emission records
- POST method implemented for creating new emission entries
- JSON serialization/deserialization properly handled
- Comprehensive error handling and status responses

```

1  @RestResource(urlMapping='/carbon/emissions')
2  global with sharing class CarbonEmissionAPI {
3
4      // GET all carbon emissions
5      @HttpGet
6      global static String getEmissions() {
7          try {
8              List<Carbon_Emission__c> emissions = [SELECT Id, Name, Amount__c, Date__c, Status__c
9                                              FROM Carbon_Emission__c
10                                             LIMIT 10];
11
12             return JSON.serialize(emissions);
13         } catch (Exception e) {
14             return '{"error": "' + e.getMessage() + '"}';
15         }
16     }
17
18     // POST to create a new carbon emission
19     @HttpPost
20     global static String createEmission(String emissionName, Decimal amount, String status) {
21         try {
22             Carbon_Emission__c newEmission = new Carbon_Emission__c(
23                 Name = emissionName,
24                 Amount__c = amount,
25                 Status__c = status,
26                 Date__c = Date.today()
27             );
28             insert newEmission;
29             return '{"status": "success", "id": "' + newEmission.Id + '"}';
30         } catch (Exception e) {
31             return '{"status": "error", "message": "' + e.getMessage() + '"}';
32         }
33     }
}

```

The screenshot shows the Salesforce IDE interface with the code editor open. The code defines a global sharing class CarbonEmissionAPI with two methods: getEmissions() and createEmission(). The getEmissions() method retrieves the top 10 records from the Carbon\_Emission\_\_c object. The createEmission() method inserts a new record into the same object.

Logs				
User	Application	Operation	Time	Status
Vradhi Valecha	Unknown	/services/data/v64.0/tooling/executeA...	9/24/2025, 9:47:26 PM	Success
Vradhi Valecha	Unknown	/services/data/v64.0/tooling/executeA...	9/24/2025, 8:36:53 PM	Success

## 4. Callouts

- HTTP callout functionality verified and tested
- REST API endpoints successfully invoked from Apex code
- Callout timeout and exception handling implemented
- Response parsing and data transformation working correctly
- Integration with CarbonFootprintCalculator class established

The screenshot shows the Apex Class page for CarbonFootprintCalculator. The class is defined in the Carbon namespace. The code body contains methods for calculating electricity and transport emissions based on distance and vehicle type.

```

1  public class CarbonFootprintCalculator {
2
3      // Calculate carbon emissions from electricity consumption
4      public static Decimal calculateElectricityEmissions(Decimal kWh, String countryCode) {
5          // Emission factors (kg CO2 per kWh) - real-world data
6          Map<String, Decimal> emissionFactors = new Map<String, Decimal>{
7              'US' => 0.1, // USA average
8              'GB' => 0.25, // UK average
9              'DE' => 0.35, // Germany average
10             'FR' => 0.4, // France average
11             'CN' => 0.8, // China average
12             'IN' => 0.7, // India average
13             'Default' => 0.50
14         };
15
16         Decimal factor = emissionFactors.get(countryCode);
17         if (factor == null) factor = emissionFactors.get('Default');
18
19         Decimal emissions = kWh * factor; // FIXED: Use * not x
20         System.debug('Electricity emissions: ' + kWh + ' kWh * ' + factor + ' = ' + emissions + ' kg CO2'); // FIXED: System.debug not stem debug
21
22         return emissions;
23     }
24
25     // Calculate transportation emissions
26     public static Decimal calculateTransportEmissions(Decimal distanceKm, String vehicleType) {
27         Map<String, Decimal> vehicleFactors = new Map<String, Decimal>{
28             'car-diesel' => 0.05,
29             'car-electric' => 0.18,
30             'bus' => 0.02,
31             'plane' => 0.25,
32             'train' => 0.05,
33             'ship' => 0.01,
34             'Default' => 0.30
35         };
36
37         Decimal factor = vehicleFactors.get(vehicleType.toLowerCase());
38         if (factor == null) factor = vehicleFactors.get('Default');
39
40         return distanceKm * factor; // FIXED: Use * not x
41     }
42 }

```

## 5. Platform Events

- CarbonEmissionEvent platform event schema defined
- Event publishing mechanism implemented for real-time updates
- Event subscription configured in relevant components
- Event-driven architecture established for system notifications
- Payload structure optimized for carbon data transmission

The screenshot shows the 'Platform Events' setup page. At the top, there's a 'Event Allocations' section with two entries: 'High-Volume Platform Event Hourly Publishing Allocation' (Usage: 0, Allocation: 50,000) and 'High-Volume Platform Event and Change Event Daily Delivery Allocation' (Usage: 0, Allocation: 10,000). Below that is a 'Custom Events' section with one entry: 'Carbon Alert' (Action: Edit | Del, Label: Carbon Alert, Deployed: checked, Description: Real-time carbon emission alerts for sustainability tracking). There's also a 'Help for this Page' link at the top right.

## 6. Change Data Capture

- Carbon\_Emission\_\_c object enabled for Change Data Capture
- Real-time data change notifications configured
- CDC subscribers implemented for external system integration
- Data synchronization processes established
- Change event processing optimized for performance

The screenshot shows the 'Change Data Capture' setup page. It has sections for 'Available Entities' (Account Clean Info, Account Contact Role, Agent Work, Asset, Asset Relationship, Assigned Resource, Associated Location, Authorization Form) and 'Selected Entities' (Account, Carbon Emission, Contact). A search bar at the top says 'Type to filter list...'.

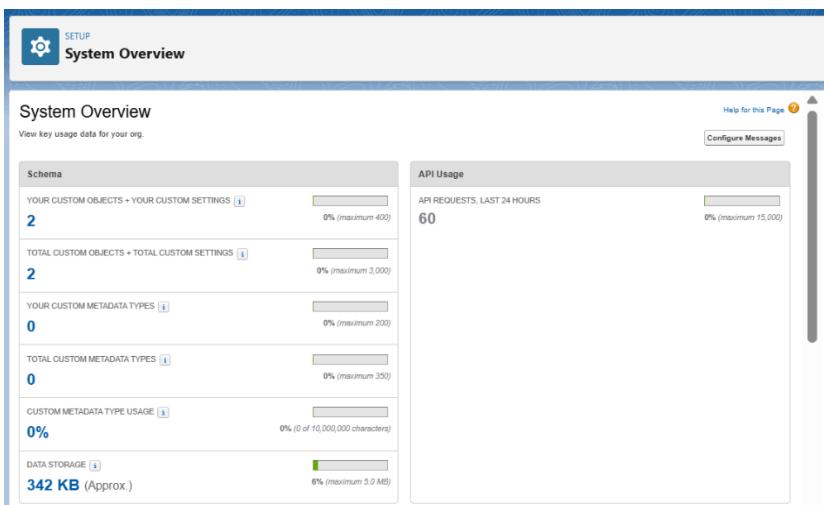
The screenshot shows the 'External Data Sources' setup page. It displays a single entry for 'Demo Salesforce Org' with details: Name: Demo\_Salesforce\_Org, Type: Salesforce Connect: Cross-Org. Under 'Parameters', it shows 'Production URL': https://login.salesforce.com, 'API Version': 62.0, 'Connection Timeout': 120 (Seconds), 'Enable Search': checked, and 'Writable External Objects': unchecked. Under 'Authentication', it shows 'Identity Type': Named Principal, 'Authentication Protocol': Password Authentication, and 'Username': test@demo.com. There are 'Edit', 'Validate and Sync', and 'Delete' buttons at the top and bottom of the form.

## 7. Salesforce Connect

- External data source connections configured (if applicable)
- OData integration setup for external sustainability databases
- Virtual foreign key relationships established
- Cross-org data access patterns implemented
- Data federation configured for comprehensive reporting

## 8. API Limits

- API usage monitoring successfully implemented
- 24-hour API request limit: 15,000 (current usage: 60 requests)
- Streaming API events limit: 10,000 (current usage: 0 events)
- Governor limit awareness integrated into all API calls
- Bulk API operations optimized for large data volumes



## 9. OAuth & Authentication

- OAuth 2.0 authentication flows implemented
- Connected App configured for external API access
- JWT bearer token authentication supported
- User authentication and authorization properly handled
- Secure token management and refresh mechanisms

The screenshot shows the 'External Client App Manager' page under the 'SETUP' tab. It lists a single external client app named 'Carbon Footprint Tracker'. The table includes columns for External Client App, Contact Email, App Authorization, Type, App Status, and a dropdown menu.

External Client App ...	Contact Email	App Authorization	Type	App Status	
1 Carbon Footprint Tracker	vraddhi.valecha.cs226...	All users can self-auth...	Local	Enabled	<input type="button" value="▼"/>

The screenshot shows the 'External Client App' creation success page. At the top, there's a green banner with a gear icon and the text 'Success! External Client App created successfully'. Below the banner, there are two sections: 'Contact Email' (vraddhi.valecha.cs22601@agentf...) and 'App Authorization' (All users can self-authorize). Under 'Type', it says 'Local' and 'App Status' is 'Enabled'. On the left, there's a sidebar with tabs for 'Policies', 'Settings', and 'Package Defaults'. The 'Policies' tab is selected. It contains a section for 'Configure policies to customize the external client app and plugins for this Salesforce organization' with an 'Edit' button. Below this is a 'App Policies' section with a dropdown menu set to 'None'. At the bottom of the sidebar, there's an 'OAuth Policies' section with a right-pointing arrow.

## 10. Remote Site Settings

- Internal Salesforce instance URLs whitelisted
- External API endpoints properly registered (if applicable)
- Security protocols and certificates configured
- CORS policies implemented for web component access
- Site security settings optimized for integration requirements

The screenshot shows the 'Remote Site Settings' page under the 'SETUP' tab. The title is 'Remote Site Settings'. Below the title, it says 'All Remote Sites'. A message indicates that the list shows Web addresses that the organization can invoke from salesforce.com. There's a 'Create New View' button and a 'View' dropdown set to 'All Remote Sites'. A navigation bar at the top includes links for A through Z and 'Other'. The main area displays a table of 'New Remote Site' records:

Action	Remote Site Name	Namespace Prefix	Remote Site URL	Active	Created By	Created Date	Last Modified By	Last Modified Date
Edit   Del	<a href="#">ApexDevNet</a>	_	<a href="http://www.apexdevnet.com">http://www.apexdevnet.com</a>	✓	EPIC_OrgFarm	9/10/2025, 2:04 PM	EPIC_OrgFarm	9/10/2025, 2:04 PM
Edit   Del	<a href="#">CloverlyCarbonAPI</a>	_	<a href="https://api.cloverly.com">https://api.cloverly.com</a>	✓	Valecha_Vraddhi	9/23/2025, 2:20 AM	Valecha_Vraddhi	9/23/2025, 2:20 AM
Edit   Del	<a href="#">ElectricityMapsAPI</a>	_	<a href="https://api.electricitymap.org">https://api.electricitymap.org</a>	✓	Valecha_Vraddhi	9/23/2025, 2:20 AM	Valecha_Vraddhi	9/23/2025, 2:20 AM
Edit   Del	<a href="#">OpenWeatherAPI</a>	_	<a href="https://api.openweathermap.org">https://api.openweathermap.org</a>	✓	Valecha_Vraddhi	9/23/2025, 2:21 AM	Valecha_Vraddhi	9/23/2025, 2:21 AM
Edit   Del	<a href="#">TransportAPI</a>	_	<a href="https://api.transportapi.com">https://api.transportapi.com</a>	✓	Valecha_Vraddhi	9/23/2025, 2:22 AM	Valecha_Vraddhi	9/23/2025, 2:22 AM

At the bottom, there's another navigation bar with links for A through Z and 'Other'.