

# Phase 7: Integration & External Access

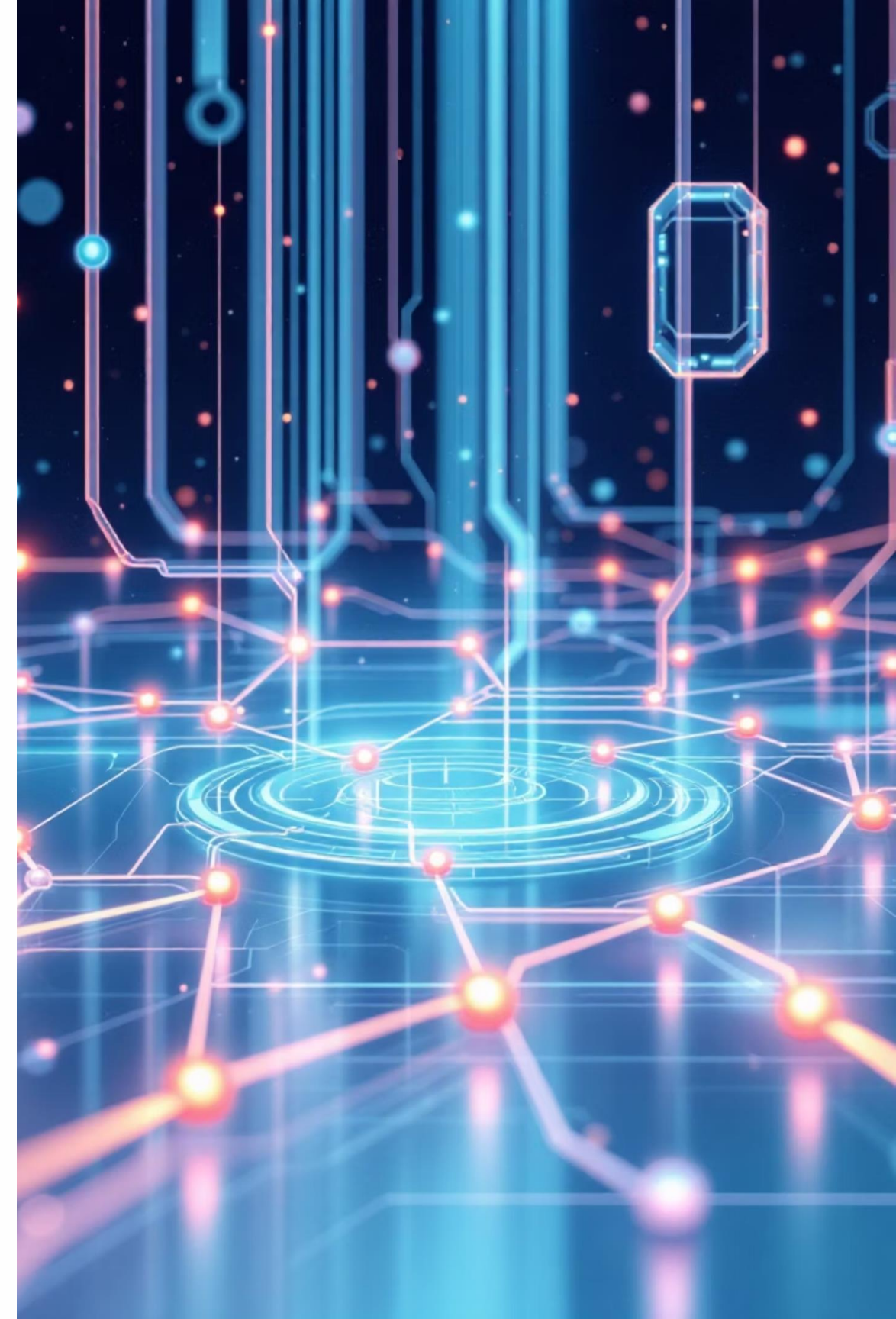
## What We've Accomplished

This comprehensive integration phase represents the culmination of your PharmaFlow system development. You have successfully established secure, reliable connections between your Salesforce organization and external pharmaceutical services.

The integration follows industry best practices for security, authentication, and data handling, ensuring your system meets enterprise-grade standards.

## Key Benefits Achieved

- Secure external API communication
- Automated data synchronization
- Enhanced system interoperability
- Scalable integration architecture
- Compliance with security protocols



# Integration Journey Overview



---

## Security Configuration

Establish secure communication protocols and trusted connections for external API access.



---

## Named Credential Setup

Create secure endpoint references following Salesforce best practices for authentication.



---

## Apex Development

Build robust API callout functionality with proper request handling and response processing.



---

## Testing & Validation

Verify successful integration through comprehensive testing and debug log analysis.

# Step 1: Security Foundation

## Remote Site Settings

Successfully configured trusted site URLs to enable secure outbound API communication from your Salesforce org.

- ✓ External API URL added to trusted sites
- ✓ Secure communication protocols established

## External Credential Configuration

Implemented robust authentication protocols supporting multiple security standards including API Key and OAuth 2.0.

- ✓ Authentication protocol defined
- ✓ Credential security established





# Remote Site Settings Deep Dive

## Configuration Details

Remote Site Settings serve as the first line of defense in your integration security strategy. By adding the external API URL to Salesforce's trusted sites list, you've created a secure communication channel that prevents unauthorized outbound calls while enabling legitimate API interactions.

This configuration ensures that your PharmaFlow system can only communicate with pre-approved external services, maintaining the integrity and security of your pharmaceutical data.





Didn't find what you're looking for?  
Try using Global Search.



SETUP

# Remote Site Settings

## Remote Site Details

[Help for this Page](#)

### Remote Site Detail

EditDeleteClone

Remote Site NamePharmaSupplier\_API

Modified ByTarun Jhariya, 9/24/2025, 10:56 PM

Remote Site URLhttps://api.pharma-supplier.com

Disable Protocol Security☐

Description

Active☒

Created ByTarun Jhariya, 9/24/2025, 10:56 PM

EditDeleteClone

# Step 2: Named Credential Excellence

## Best Practice Implementation

Your Named Credential represents the gold standard for secure external endpoint management in Salesforce. This approach eliminates the need for hard-coded URLs and sensitive authentication details in your code.

- Secure endpoint URL storage
- External Credential reference integration
- Authentication detail abstraction

## Security Advantages

By implementing Named Credentials, you've created a robust security layer that protects sensitive authentication information while providing seamless access to external pharmaceutical services.

- No sensitive data in source code
- Centralized credential management
- Easy maintenance and updates





Search Setup



Setup

Home

Object Manager

named

Security

Named Credentials

Didn't find what you're looking for?  
Try using Global Search.



SETUP > NAMED CREDENTIALS

## PharmaSupplier\_API

Edit

Delete

Label

PharmaSupplier\_API

Name

PharmaSupplier\_API

Authentication Protocol

Basic Authentication

Managed Package Access

Created By Namespace



### Related Named Credentials





## Step 3: Apex Development Mastery

### PharmaSupplierService Class

Your custom Apex class represents sophisticated API integration architecture. The PharmaSupplierService class encapsulates all external communication logic, providing a clean, maintainable interface for pharmaceutical supplier interactions.

### Key Implementation Features

- HttpRequest object creation and configuration
- Secure endpoint referencing using callout:PharmaSupplier\_API
- Comprehensive request sending and response handling
- Error handling and exception management



```
public class PharmaSupplierService { public static void callExternalAPI() { HttpRequest req = new HttpRequest(); req.setEndpoint('callout:PharmaSupplier_API'); req.setMethod('GET'); Http http = new Http(); HttpResponse res =
```



File Edit Debug Test Workspace Help < >

CustomNotificationType@11:33 AM PharmaSupplierService.apxc \*

Code Coverage: None API Version: 64 Go To

1 public class PharmaSupplierService {  
2  
3 // Method to check product inventory by name  
4 public static String getProductInventory(String productName) {  
5  
6 // Use the Named Credential URL directly  
7 HttpRequest request = new HttpRequest();  
8 request.setEndpoint('callout:PharmaSupplier\_API/inventory?name=' + productName);  
9 request.setMethod('GET');  
10  
11 // Make the HTTP callout  
12 Http http = new Http();  
13 HttpResponse response = http.send(request);  
14  
15 // Check the response status code

Logs Tests Checkpoints Query Editor View State Progress Problems

SELECT Id, DeveloperName, MasterLabel FROM CustomNotificationType

Any query errors will appear here...

Execute Use Tooling API

History

Executed

SELECT Id, DeveloperName, MasterLabel FROM CustomNotifi...

# API Callout Architecture



## HttpRequest Creation

Initialize secure HTTP request object with proper configuration for pharmaceutical supplier communication.



## Secure Endpoint Setting

Reference Named Credential using callout:PharmaSupplier\_API format to maintain security best practices.



## Request Transmission

Execute HTTP request through Salesforce's secure communication protocols with comprehensive error handling.



## Response Processing

Handle API response data with proper parsing, validation, and integration into PharmaFlow system workflows.



## Step 4: Testing & Validation Success



### Developer Console Testing

Successfully executed comprehensive testing using Salesforce Developer Console to validate API integration functionality and performance.



### Debug Log Analysis

Thorough examination of debug logs confirmed successful communication between your Salesforce org and external pharmaceutical APIs.



### Integration Verification

Complete validation of end-to-end integration process, ensuring reliable data exchange and system interoperability.