Salesforce Project Implementation Phases with Concepts (Admin + Developer)
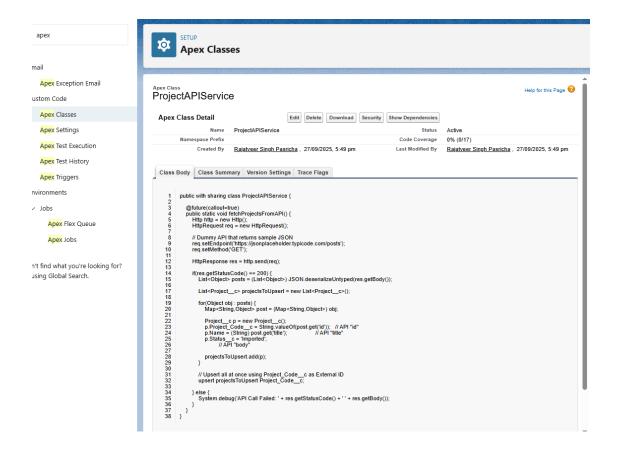
Problem Statement: Construction companies often struggle to manage communication among multiple stakeholders such as clients, project managers, vendors, contractors, and subcontractors. Since updates, approvals, and requirements are exchanged through scattered channels like phone calls, emails, and spreadsheets, important information is often lost or delayed. This lack of a centralized system results in miscommunication, project delays, cost overruns, and reduced client satisfaction.

Proposed Solution : A tailored Construction CRM solution is needed to streamline communication, track tasks, manage vendor/contractor interactions, and provide real-time project updates, thereby improving efficiency, accountability, and client trust.

Phase 7: Integration & External Access

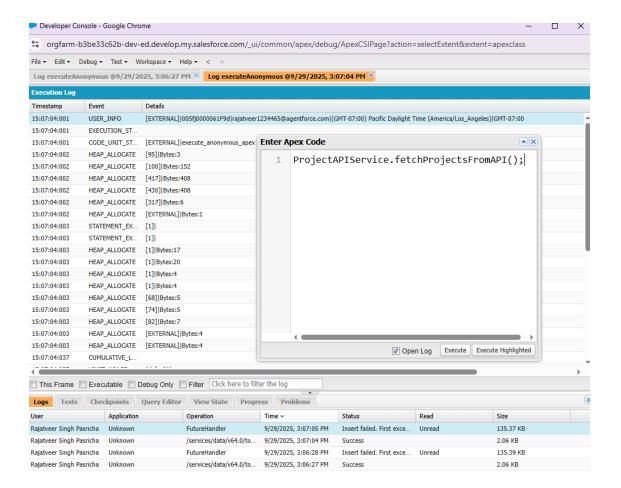**Here I have displayed an example of using API and calling out API services. I'll do further integration in the advance stages of projects.**

- **NAMED CREDENTIALS**
- **EXTERNAL SERVICES**
- **WEB SERVICES (REST/SOAP)**
- **CALLOUTS**

mail

Apex Exception Email

ustom Code

Apex Classes
Apex Settings
Apex Test Execution
Apex Test History
Apex Triggers

nvironments

✓ Jobs

Apex Flex Queue
Apex Jobs

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

SETUP
Apex Classes

Apex Class
ProjectAPIService

Help for this Page

Apex Class Detail   Edit Delete Download Security Show Dependencies

Name	ProjectAPIService	Status	Active
Namespace Prefix		Code Coverage	0% (0/17)
Created By	Rajatveer Singh Pasricha , 27/09/2025, 5:49 pm	Last Modified By	Rajatveer Singh Pasricha , 27/09/2025, 5:49 pm

Class Body Class Summary Version Settings Trace Flags

```
1    public with sharing class ProjectAPIService {
2
3        @future(callout=true)
4        public static void fetchProjectsFromAPI() {
5            Http http = new Http();
6            HttpRequest req = new HttpRequest();
7
8            // Dummy API that returns sample JSON
9            req.setEndpoint('https://jsonplaceholder.typicode.com/posts');
10           req.setMethod('GET');
11
12           HttpResponse res = http.send(req);
13
14           if(res.getStatusCode() == 200) {
15               List<Object> posts = (List<Object>) JSON.deserializeUntyped(res.getBody());
16
17               List<Project__c> projectsToUpsert = new List<Project__c>();
18
19               for(Object obj : posts) {
20                   Map<String,Object> post = (Map<String,Object>) obj;
21
22                   Project__c p = new Project__c();
23                   p.Project_Code__c = String.valueOf(post.get('id'));   // API "id"
24                   p.Name = (String) post.get('title');          // API "title"
25                   p.Status__c = 'Imported';
26                       // API "body"
27
28                   projectsToUpsert.add(p);
29               }
30
31               // Upsert all at once using Project_Code__c as External ID
32               upsert projectsToUpsert Project_Code__c;
33
34           } else {
35               System.debug('API Call Failed: ' + res.getStatusCode() + ' ' + res.getBody());
36           }
37       }
38   }
```

Apex Class
ProjectAPIService

Help for this Page

Apex Class Detail   Edit Delete Download Security Show Dependencies

Name	ProjectAPIService	Status	Active
Namespace Prefix		Code Coverage	0% (0/17)
Created By	Rajatveer Singh Pasricha , 9/29/2025, 2:34 AM	Last Modified By	Rajatveer Singh Pasricha , 9/29/2025, 2:34 AM

Class Body Class Summary Version Settings Trace Flags

```
1    public with sharing class ProjectAPIService {
2
3        @future(callout=true)
4        public static void fetchProjectsFromAPI() {
5            Http http = new Http();
6            HttpRequest req = new HttpRequest();
7
8            // Public test API - returns fake posts as sample data
9            req.setEndpoint('https://jsonplaceholder.typicode.com/posts');
10           req.setMethod('GET');
11
12           HttpResponse res = http.send(req);
13
14           if(res.getStatusCode() == 200) {
15               // Deserialize JSON into a list of maps
16               List<Object> posts = (List<Object>) JSON.deserializeUntyped(res.getBody());
17
18               List<Project__c> projectsToInsert = new List<Project__c>();
19
20               for(Object obj : posts) {
21                   Map<String,Object> post = (Map<String,Object>) obj;
22
23                   // Create a new Project__c record in Salesforce
24                   Project__c p = new Project__c();
25
26                   // Map API data to existing writable fields
27                   p.Name = (String) post.get('title');      // map title → Name
28                   p.Project_Status__c = 'Imported';          // default status
29
30
31                   projectsToInsert.add(p);
32               }
33
34               // Insert all projects at once
35               if(!projectsToInsert.isEmpty()) {
36                   insert projectsToInsert;
37               }
38
39           } else {
40               System.debug('API Call Failed: ' + res.getStatusCode() + ' ' + res.getBody());
```

orgfarm-b3be33c62b-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage?action=selectExtent&extent=apexclass

File ▾   Edit ▾   Debug ▾   Test ▾   Workspace ▾   Help ▾   <   >

Log executeAnonymous @9/29/2025, 3:06:27 PM ✕   **Log executeAnonymous @9/29/2025, 3:07:04 PM** ✕

Execution Log

Timestamp	Event	Details
15:07:04:001	USER_INFO	[EXTERNAL]|005fj0000061F9d|rajatveer1234465@agentforce.com|(GMT-07:00) Pacific Daylight Time (America/Los_Angeles)|GMT-07:00
15:07:04:001	EXECUTION_ST...	
15:07:04:001	CODE_UNIT_ST...	[EXTERNAL]|execute_anonymous_apex
15:07:04:002	HEAP_ALLOCATE	[95]|Bytes:3
15:07:04:002	HEAP_ALLOCATE	[100]|Bytes:152
15:07:04:002	HEAP_ALLOCATE	[417]|Bytes:408
15:07:04:002	HEAP_ALLOCATE	[430]|Bytes:408
15:07:04:002	HEAP_ALLOCATE	[317]|Bytes:6
15:07:04:002	HEAP_ALLOCATE	[EXTERNAL]|Bytes:1
15:07:04:003	STATEMENT_EX...	[1]|
15:07:04:003	STATEMENT_EX...	[1]|
15:07:04:003	HEAP_ALLOCATE	[1]|Bytes:17
15:07:04:003	HEAP_ALLOCATE	[1]|Bytes:20
15:07:04:003	HEAP_ALLOCATE	[1]|Bytes:4
15:07:04:003	HEAP_ALLOCATE	[1]|Bytes:4
15:07:04:003	HEAP_ALLOCATE	[68]|Bytes:5
15:07:04:003	HEAP_ALLOCATE	[74]|Bytes:5
15:07:04:003	HEAP_ALLOCATE	[82]|Bytes:7
15:07:04:003	HEAP_ALLOCATE	[EXTERNAL]|Bytes:4
15:07:04:003	HEAP_ALLOCATE	[EXTERNAL]|Bytes:4
15:07:04:037	CUMULATIVE_L...	

Enter Apex Code

```
1   ProjectAPIService.fetchProjectsFromAPI();
```

☑ Open Log   Execute   Execute Highlighted

☐ This Frame   ☐ Executable   ☐ Debug Only   ☐ Filter   Click here to filter the log

Logs | Tests | Checkpoints | Query Editor | View State | Progress | Problems

User	Application	Operation	Time ▾	Status	Read	Size
Rajatveer Singh Pasricha	Unknown	FutureHandler	9/29/2025, 3:07:05 PM	Insert failed. First exce...	Unread	135.37 KB
Rajatveer Singh Pasricha	Unknown	/services/data/v64.0/to...	9/29/2025, 3:07:04 PM	Success		2.06 KB
Rajatveer Singh Pasricha	Unknown	FutureHandler	9/29/2025, 3:06:28 PM	Insert failed. First exce...	Unread	135.39 KB
Rajatveer Singh Pasricha	Unknown	/services/data/v64.0/to...	9/29/2025, 3:06:27 PM	Success		2.06 KB

REMOTE SITES:

All Remote Sites

Help for this Page ❓

Below is the list of Web addresses that your organization can invoke from salesforce.com. To add another Web address, click New Remote Site.

View: [All Remote Sites ▾]   Create New View

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | **All**

New Remote Site

Action	Remote Site Name ↑	Namespace Prefix	Remote Site URL	Active	Created By	Created Date	Last Modified By	Last Modified Date
Edit	Del	ApexDevNet	_	http://www.apexdevnet.com	✓	Pasricha, Rajatveer Singh	18/09/2025, 9:56 am	Pasricha, Rajatveer Singh	18/09/2025, 9:56 am
Edit	Del	JSONPlaceholder	_	https://jsonplaceholder.typicode.com	✓	Pasricha, Rajatveer Singh	27/09/2025, 5:45 pm	Pasricha, Rajatveer Singh	27/09/2025, 5:45 pm

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | **All**

API LIMITS:

Configure Messages

API Usage

API REQUESTS, LAST 24 HOURS

132

0% *(maximum 15,000)*

imum 400)

um 3,000)

imum 200)

imum 350)