Azure DevOps Integration – PR Review API Documentation

# 🔒 Authorization

Named Credential Used: callout:Azure\_Ado\_User

This is used for all PR-related callouts.

Headers:

Content-Type: application/json-patch+json  
Connection: keep-alive  
X-HTTP-Method-Override: PATCH

# 🔹 A. Pull Request Reviewer Callout (GET Azure DevOps PR Reviewers)

Purpose: Retrieves reviewers’ approval status from Azure DevOps for a PR

HTTP Method: GET

Endpoint Format:

https://dev.azure.com/{organization}/{project}/\_apis/git/repositories/{repo}/pullrequests/{prId}/reviewers?api-version=5.1

Apex Method: makeCallout(String prURL, String truncatedUserName)

Sample Constructed URL:

callout:Azure\_Ado\_User/{project}/\_apis/git/repositories/{repo}/pullrequests/{prId}/reviewers?api-version=5.1

## 📥 Sample API Response (Azure DevOps)

{  
 "value": [  
 {  
 "id": "abc123",  
 "uniqueName": "vstfs\\username",  
 "vote": 10,  
 "isRequired": true  
 },  
 {  
 "id": "def456",  
 "uniqueName": "vstfs\\optionaluser",  
 "vote": 5,  
 "isRequired": false  
 }  
 ]  
}

## 🔄 Review Status Mapping

Azure DevOps 'vote' values are mapped to Copado review actions as follows:

10 => Approved => Approved  
5 => Approved with suggestions => Approved  
0 => Opened => (null)  
-5 => Waiting for author => (null)  
-10 => Rejected => Unapproved

Other Mapped Fields:

- PR\_Submitter\_\_c: true if isRequired = false and no vote

- Required\_Reviewer\_\_c: true if isRequired = true

## 🔁 Flow Summary

1. When a copado\_\_Pull\_Request\_Review\_\_c record is inserted, the getFullUserNameCallout() method is triggered (via onAfterInsert()).  
2. It parses the PR URL, fetches reviewers from Azure DevOps.  
3. It updates Salesforce records with:  
 - Review status (Approved / Unapproved)  
 - Reviewer type (Required / Optional / PR Submitter)  
4. Duplicate review records are also identified and removed safely.