

Copado - Code Scan Integration

Install the Extension package

Pre-requisites

1. Install [Copado v20.0](#) or higher
2. Install Copado Quality Tools extension v1.14 or higher
3. Install Copado DX Extension v3.3 or higher
4. Install the Copado CodeScan integration from the [DevOps Exchange Listing](#)

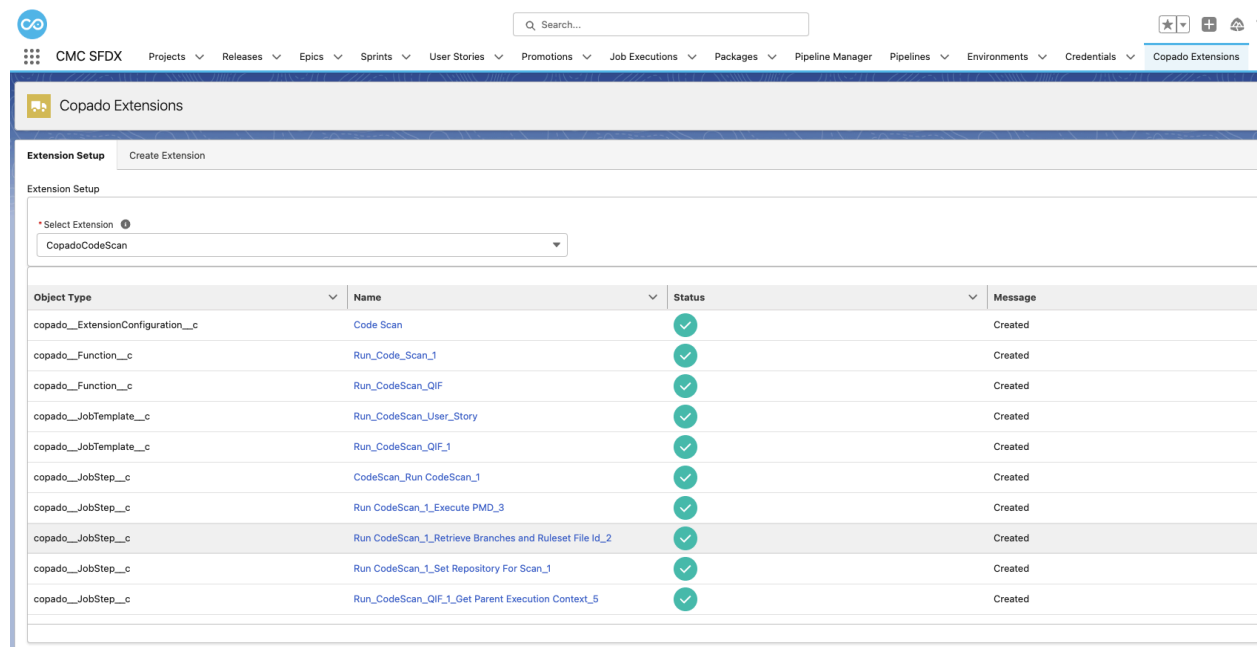
Picklist Values

Create the Following Picklist values

- Object: Extension Configuration, Field: Extension Tool, Value: CodeScan
- Object: Quality Gate Rule Condition, Field: Platform, Value: SFDX
- Picklist Value Set: Copado Test Tool, Value: CodeScan

Create The Functions and Job Templates

Navigate to the “Copado Extensions” tab, select “CopadoCodeScan” and press the button “Generate Extension Records”.



The screenshot shows the Copado Extensions interface. The top navigation bar includes the Copado logo, a search bar, and various tabs like CMC SFDX, Projects, Releases, Epics, Sprints, User Stories, Promotions, Job Executions, Packages, Pipeline Manager, Pipelines, Environments, Credentials, and Copado Extensions. The 'Copado Extensions' tab is active, showing a sub-tab 'Extension Setup' with a 'Create Extension' button. Below this, there's a dropdown menu labeled '* Select Extension' with 'CopadoCodeScan' selected. A table displays the generated extension records with columns: Object Type, Name, Status, and Message. All records show a 'Created' status with a green checkmark.

Object Type	Name	Status	Message
copado__ExtensionConfiguration__c	Code Scan	✓	Created
copado__Function__c	Run_Code_Scan_1	✓	Created
copado__Function__c	Run_CodeScan_QIF	✓	Created
copado__JobTemplate__c	Run_CodeScan_User_Story	✓	Created
copado__JobTemplate__c	Run_CodeScan_QIF_1	✓	Created
copado__JobStep__c	CodeScan_Run CodeScan_1	✓	Created
copado__JobStep__c	Run CodeScan_1_Execute PMD_3	✓	Created
copado__JobStep__c	Run CodeScan_1_Retrieve Branches and Ruleset File Id_2	✓	Created
copado__JobStep__c	Run CodeScan_1_Set Repository For Scan_1	✓	Created
copado__JobStep__c	Run_CodeScan_QIF_1_Get Parent Execution Context_5	✓	Created

Note that if multi-currency is not enabled, records will show in red because of the missing field but the records were created successfully anyways.

Configure your Pipeline

Create the Following Pipeline System Properties and set the corresponding values

CS_SERVER
CS_PROJECT_ID
CS_TOKEN
CS_ORGANIZATION


Set the right values, make sure the CS_TOKEN is set to Hidden

The screenshot displays two sections of the Copado interface. The top section, titled 'Pipeline test pipeline', shows a list of pipeline steps with their corresponding SFDX commands and checkboxes for enabling them. The bottom section, titled 'System Properties (4)', shows a table of system properties with columns for Property Name, API Name, Value, Hide Value, Environment, Parent Id, and User. The CS_TOKEN property is highlighted with a red circle, indicating it should be set to Hidden.

Property Name	API Name	Value	Hide Value	Environment	Parent Id	User
CS_SERVER	CS_SERVER	https://[redacted].codescan.io	<input type="checkbox"/>		a0Y7S000002zOnyUAE	
CS_PROJECT_ID	CS_PROJECT_ID	[redacted]	<input type="checkbox"/>		a0Y7S000002zOnyUAE	
CS_TOKEN	CS_TOKEN		<input checked="" type="checkbox"/>		a0Y7S000002zOnyUAE	
CS_ORGANIZATION	CS_ORGANIZATI...	[redacted]	<input type="checkbox"/>		a0Y7S000002zOnyUAE	

Configure the Quality Gate

Navigate to the Quality Gate Rules tab and create a new record as follows. Note that the Type field will be populated automatically upon save. The global value set Test Tool should have a value for CodeScan as part of this package. It can be created manually if necessary.


Quality Gate Rule
Code Scan

Tool	Type	Execution Sequence	Copado Actions List	Status
CodeScan	CodeScan	After	Commit;	Inactive

Details

Conditions

Information

Quality Gate Rule Name

Code Scan

Owner

Federico Larsen

Tool

CodeScan

Type

CodeScan

Status

Inactive

Trigger

Execution Sequence

After

Copado Actions

Commit

System Information

Created By

Federico Larsen, 19/10/2022, 20:34

Last Modified By


Federico Larsen, 19/10/2022, 20:34

Set the conditions so that it only applies to Pipelines with Platform = SFDX . This picklist value can be added manually if necessary.

Once saved, press the button “Activate” on the Quality Gate Rule record.

You are all set.

To test the configuration, just commit a user story on a SFDX platform Pipeline, and the Commit Action will enforce Code Scan after every commit.


SFDX Commit

View Job Execution

>

Commit

Success

Type: Function Step result: R00121

19/10/22, 16:56:40

>

Code Scan

Success

Type: Test Step result: R00122

19/10/22, 16:57:39

Job Execution Result

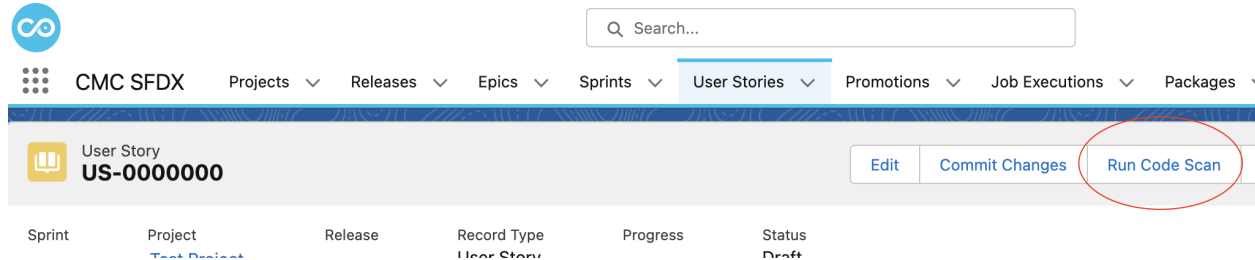
Job Execution Status: Successful

Completed

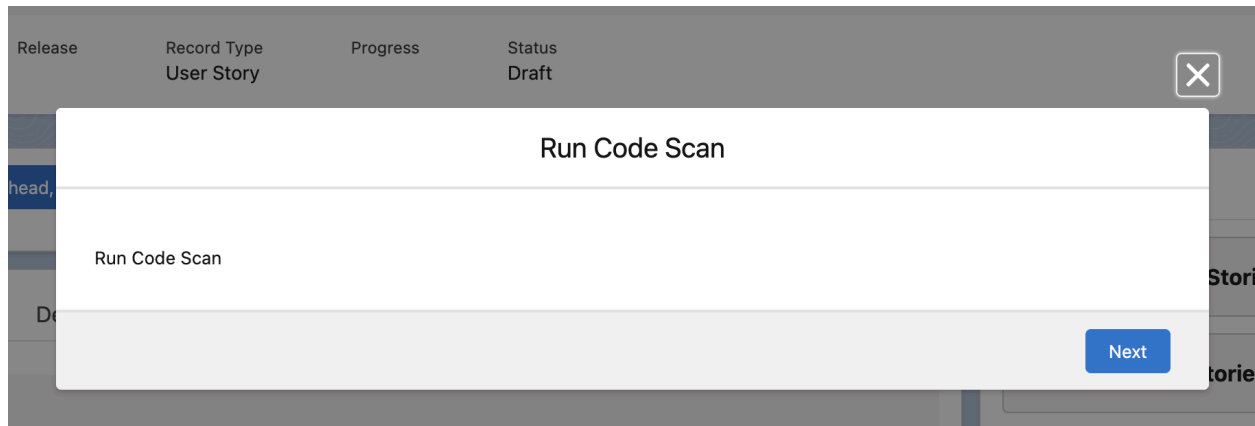
19/10/22, 21:28:55

Run a scan from the User Story at any time (optional)

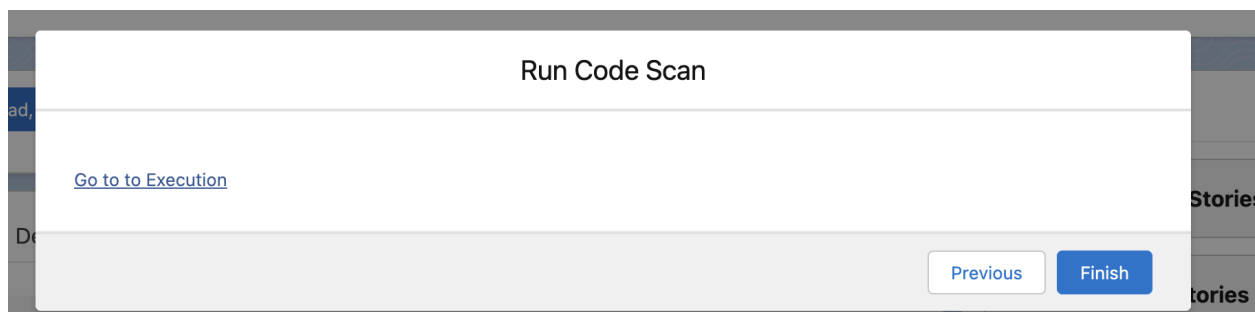
Add the Quick action “Run Code Scan” to your User Story Lightning Record Page.




Press the button on the User Story to run a new scan on the Feature Branch



After pressing the Next button, the execution starts and the progress can be followed from the Execution link.



Logs are available from the Result record related to the execution and the json output of the scan is attached to the result.



CMC SFDX

Projects

Releases

Epics

Sprints

User Stories

Promotions

Job Executions

Packages

Pipeline Manager

R00000

More

Result

R00000

Edit

Delete

```

4 19:24:17 param ORGANIZATION = 
5 19:24:17 param BRANCH = 
6 19:24:17 param BRANCH = feature/US-0000000
7 19:24:18 CopadoFunction INFO setting status 'cloning repo...'
8 19:24:19 CopadoFunction INFO fetching/checking-out branch 'feature/US-0000000' in directory '.'
9 19:24:19 CopadoFunction INFO Current home directory =/root/
10 19:24:19 Warning: Permanently added 'product-int-ssh.cvc.copado.com,34.75.194.150' (ECDSA) to the
    list of known hosts.
11 19:24:19 Hi there! You've successfully authenticated with the deploy key named Copado Labs, but
    Gitea does not provide shell access.
12 19:24:19 If this is unexpected, please log in with password and setup Gitea under another user.
13 19:24:20 From product-int-ssh.cvc.copado.com:flarsen/test
14 19:24:20 * branch feature/US-0000000 -> FETCH_HEAD
15 19:24:20 * [new branch] feature/US-0000000 -> origin/feature/US-0000000
16 19:24:20 Switched to a new branch 'feature/US-0000000'
17 19:24:20 Branch 'feature/US-0000000' set up to track remote branch 'feature/US-0000000' from
    'origin'.
18 19:24:21 CopadoFunction INFO setting status 'running codescan...'
19 19:24:30 INFO: Scanner configuration file: NONE
20 19:24:30 INFO: Project root configuration file: NONE
21 19:24:30 INFO: SonarQube Scanner 3.3.0.1492
22 19:24:30 INFO: Java 11.0.16 Ubuntu (64-bit)19:24:30 INFO: Linux 4.4.0 amd64
23 19:24:32 INFO: User cache: /root/.sonar/cache
24 19:24:36 INFO: SonarQube server 8.5.0
25 19:24:36 INFO: Default locale: "en US". source code encoding: "US-ASCII" (analysis is platform

```

Function

Run CodeScan

Job Step

Run CodeScan

Job API Name

Run_Code_Scan_1

External Job ID

Context ID

a0v7S000000bulpQA!

External Result Link

Files (2)

Function Logs for a1K7S0000019UPqUAM.txt

18 Oct 2022 • 10KB • txt

result.json

18 Oct 2022 • 7KB • json

If the execution is OK, meaning no quality gate is violated in Code Scan, the step is Successful and Failed otherwise.

Edit

Delete

New

Order

Run CodeScan from User St...

View Job Execution

Run CodeScan

18/10/22, 21:24:13

Success

Type: Function Step result: R00000

Job Execution Result

Job Execution Status:

18/10/22, 21:26:09

Successful

Completed