

Copado User Story Lifecycle

Video explanation

From 00:00 to 03:41 Full Story Lifecycle of a successful scenario

From 03:41 to 04:33 Show a failed scenario

Written Steps

The screenshot shows the Copado Developer interface for a User Story (US-0004464). The interface is divided into several sections: a top navigation bar, a main action bar with buttons like 'Open Org', 'Commit Changes', 'Add Apex Tests', 'Manage Apex Tests', 'Validate Changes', and 'Guided Deploy', and a main content area. The main content area is divided into a 'Plan' section (with sub-sections like 'Build', 'Test', 'Deliver', 'Related') and a 'Project Management' section. The 'Plan' section shows a progress bar with stages: Monthly Dev 1 - 3 ahead, 15..., Monthly INT, INT, UAT, and Production. The 'Project Management' section shows a table with columns for Title, Record Type, User Story, and Documentation Complete. The 'Commit Changes' button is highlighted with a green '1'. The 'Add Apex Tests' button is highlighted with a green '2'. The 'Validate Changes' button is highlighted with a green '3'. The 'Guided Deploy' button is highlighted with a green '4'. There are also green annotations on the right side of the interface, such as 'Ensure Apex Tests are added and executed providing 80% of coverage: Only if the story requires apex test coverage', 'Note: Remember to create a PR for review from the next successful validation (not merge)', and 'Use this instead of the "Promote and Deploy" checkbox'. A 'Child User Stories (0)' section is also visible on the right.

1. **Commit the changes through the Commit quick action**
2. **If you have done Apex changes or other changes that require Apex test coverage (i.e Flow changes)**
 - a. Add the tests class, where either
 - i. commit the apex classes (if these are modified) through the standard commit
 - ii. if the apex classes just need to provide coverage but no tests are added/modified, add them through the "Add apex tests button"
 - b. Run the tests, using the **Manage Apex Tests** button
3. **Validate the changes, through Validate Changes button.** This will create a new validation deployment. Some important points:
 - a. If you get a conflict error during the deployment, go to the created validation promotion record (in the User Story "Deliver" section), and click on "Review merge conflicts". Note that profile conflicts are set to be automatically resolved and won't be displaying.
 - b. **IMPORTANT: The changes of any conflict solved automatically by Copado will also display in this Pull Request. Reason why it's very important to review it, to ensure correct changes (for example in profiles).**
 - c. If the validation fails, check the error in the validation deployment (in the User Story "Deliver" section you have the promotion record and inside of it the deployment record), perform the necessary commits to fix it, and then trigger a new validation.
 - d. **If the validation works, create a pull request from the latest validation deployment**

The screenshot shows a table titled 'Deployments' with the following columns: ACTION, DEPLOYMENT NAME, DATE, SOURCE CREDENTIAL, % COMPLETED, STATUS, and SCHEDULE. There is one row of data:

ACTION	DEPLOYMENT NAME	DATE	SOURCE CREDENTIAL	% COMPLETED	STATUS	SCHEDULE
1 Edit Dell Pull Request	CD: Deploy - US-0004464 → Monthly INT		Copado - Monthly Dev 1	0	Draft	Deploy now

A green arrow points to the 'Edit | Dell Pull Request' link in the first row.

