Accelerating Development with SAP Business Application Studio

Unit 1

SAP Business Application Studio

Lesson 1: Challenges and Market Trends

- Role of developers in achieving digital transformation
- High productivity tooling
  - Work efficiently
  - Focus on most challenging projects

Lesson 1: Challenges and Market Trends

- Lack of highly skilled professionals
- Factors that impact performance
  - Technical debt
  - Excess of different tools
  - Fragmented technology stacks

Lesson 1: High Productivity Tools

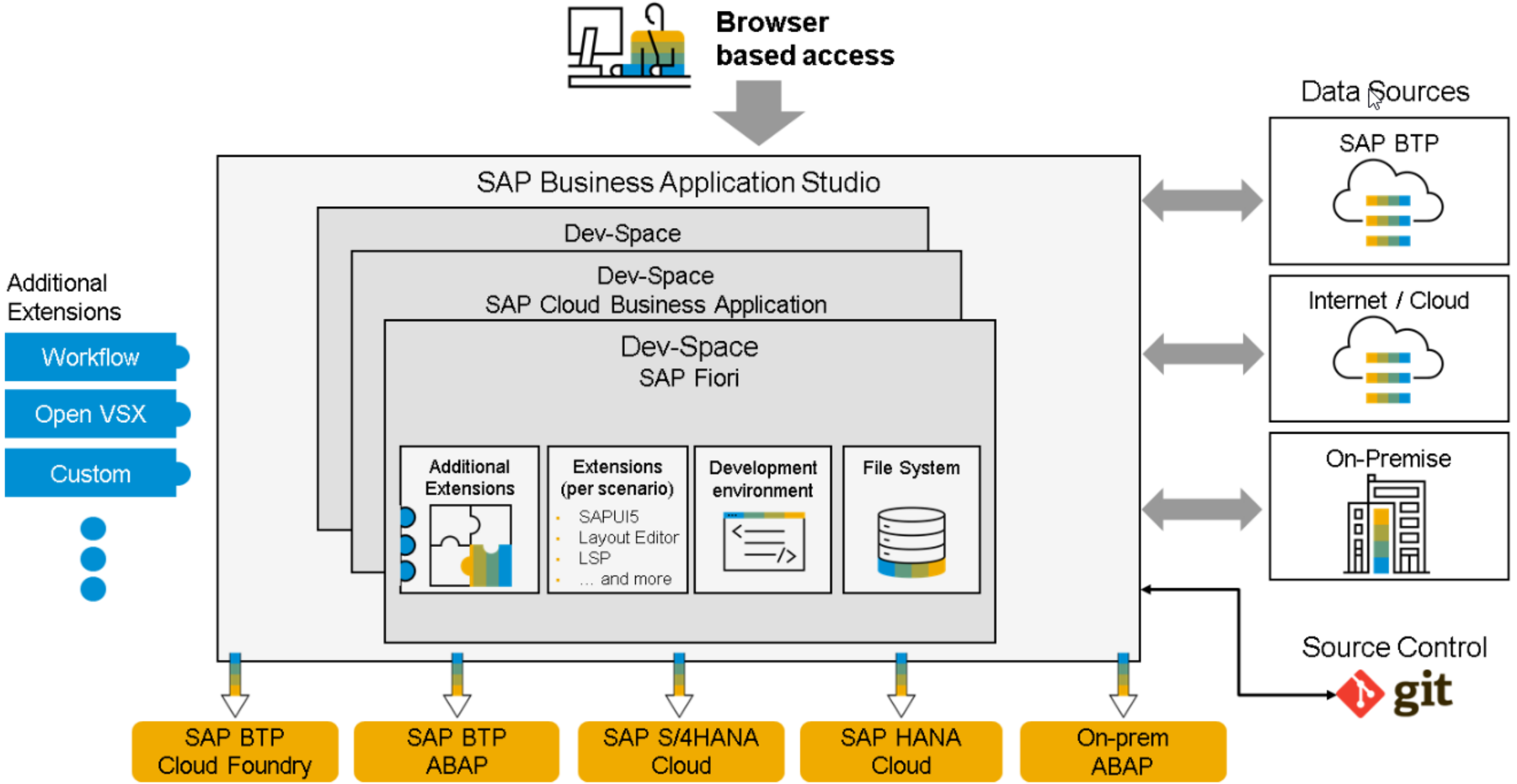
- Benefits of high productivity tools
  - New development paradigm – Low code / No code development
    - Combine traditional development with visually guided development
    - Suited for part time developers, professional developers
  - Reduce time intensive coding
  - Reuse instead of rebuild
  - Cost efficient
  - Speed to market (25% gain in speed)
  - Improved staff retention

Lesson 2: Introducing SAP BAS

- SAP Business Application Studio
  - Develop extensions, business applications on SAP Business Technology Platform (SAP BTP)
  - Provides all tools for modern cloud development
    - Especially tools required for business applications in SAP ecosystem
  - Offered as SAP BTP service. Provides 2 types of application plans
    - Free (contains some restrictions)
    - Standard edition
  - Access it using internet browser. Similar to Microsoft Visual Studio Code

Lesson 2: Introducing SAP BAS

- Dev spaces
  - Isolated environments in cloud
  - Contains tools and preinstalled runtimes + built-in extensions
  - Additional extensions from Open VSX Registry
  - Implemented as a Kubernetes Pod
  - List of Dev Space Types
    - SAP Fiori
    - Full Stack Application using Productivity Tools
    - Full Stack Cloud Application
    - SAP HANA Native Application
    - SAP Mobile Application
    - Basic


Browser based access
Data Sources
SAP BTP
Internet / Cloud
On-Premise
SAP Business Application Studio
Dev-Space
Dev-Space
SAP Cloud Business Application
Dev-Space
SAP Fiori
Additional Extensions
Extensions (per scenario)
SAPUI5
Layout Editor
LSP
... and more
Development environment
File System
Additional Extensions
Workflow
Open VSX
Custom
Source Control
git
SAP BTP Cloud Foundry
SAP BTP ABAP
SAP S/4HANA Cloud
SAP HANA Cloud
On-prem ABAP

Lesson 2: Benefits of SAP BAS

- Centrally administered with tools, repositories, system access, company policies
- Offers set of productivity tools
- Consistent development experience
- Integrates with existing SAP solutions, systems and services

Lesson 3: Setting up SAP BAS

- Prerequisites
  - Global account enabled
  - Entitlements in global account
    - SAP Business Application Studio
    - SAP Build Work Zone
    - SAP HANA Cloud
    - SAP Mobile Services (optional)
    - Cloud Foundry runtime

- **Note:** *Global Account Administrator* role required to create new subaccount

Lesson 3: Adding Entitlements

Access to SAP Business Application Studio

- Manual subscription
- Booster
  - Set of guided interactive steps
  - Sets up basic entitlements and subscriptions
  - Note: To enable deployment, you need additional entitlements
    - SAP HANA Cloud
    - SAP HANA Schemas & HDI Containers
    - Cloud Foundry Runtime

Lesson 3: Booster

Lesson 3: Adding more entitlements

- SAP Build Work Zone
- SAP HANA Cloud
- SAP HANA Schemas & HDI Containers
- Mobile Services
- Cloud Foundry Runtime

Lesson 3: Subscribe to services

- SAP Build Work Zone
- SAP HANA Cloud
- Mobile Services

Assign following role collections

- Launchpad_Admin
- CICD Service Administrator

Lesson 4: Getting Started

- Dev Spaces in SAP Business Application Studio
  - Full fledged dev environment (Private virtual machine)
  - Contains runtimes and tools for specific development scenarios
  - Delivered with predefined set of tools
  - Created in dev space manager
  - Different tools and runtimes are available as extensions
  - Kubernetes technology used to provision and manage containers

Lesson 4 – Dev space characteristics

Automatic installation of extensions

Whenever you create a new dev space, the latest version of the selected extensions is installed. Extensions in existing dev spaces are **updated automatically every time it is restarted**, thus making sure you always use the most up-to-date version of the tools.

Isolated

Dev spaces are completely **isolated** from one another, there is no way to access the file system and processes of one dev space from another dev space, even if both are owned by the same user. This level of isolation, together with the containers approach, enabled us to provide the long-requested terminal access and the ability to run almost any command as if you are working on your own laptop.

Disposable

Unlike your laptop, dev spaces are also **disposable**. If, for some reason, your dev space is messed up, you can always create a new one and easily move your projects to it by downloading them from the previous one and uploading them to the new one. Having your projects always in sync with their Git repositories will make it even easier to switch between dev spaces.

Restorable

To save on resources, dev spaces are stopped after 3 hours of inactivity (1 hour in Trial and free-tier). Don't worry, your dev space, including all the files it contained, will be **restored** once you start the dev space again. However, processes like applications or jobs you were running during your development will have to be restarted manually.

Preserved

Dev spaces will be **preserved** as long as your subscription to SAP Business Application Studio is in place and of course, if not explicitly deleted.

Lesson 5 – Tour of Business Application Studio

- High productivity toolkit
  - Powerful IDE providing full developer flexibility
  - Visual tools for entire end-to-end application development
    - Data models and services
    - SAP Fiori user interfaces

Lesson 5 – High Productivity Toolkit

- Storyboard
- Project Explorer
- Data Models and Services Graphical Editor
- Service Center
- Preview and deployment of your application

Lesson 5 – High Productivity Toolkit

- Terminal
- Command Palette
- File Explorer
- Search
- Git
- Debug
- Outline
- Problems View
- Code Completion

Lesson 5 – User and Workspace Settings

- User Settings
  - Global
- Workspace Settings
  - Specific to project
  - Can be shared across developers

Lesson 5: Tour of Business Application Studio

Features of the application

- Uses a managed approuter
- Uses the HTML5 Application Repository
- Uses data from a remote API
- Built using the Productivity Tools of SAP Business Application Studio

Developing Risk Management Application to extend SAP S/4HANA

Unit 2

SAP Business Application Studio

Lesson 1: Use Case and Architecture of the Application

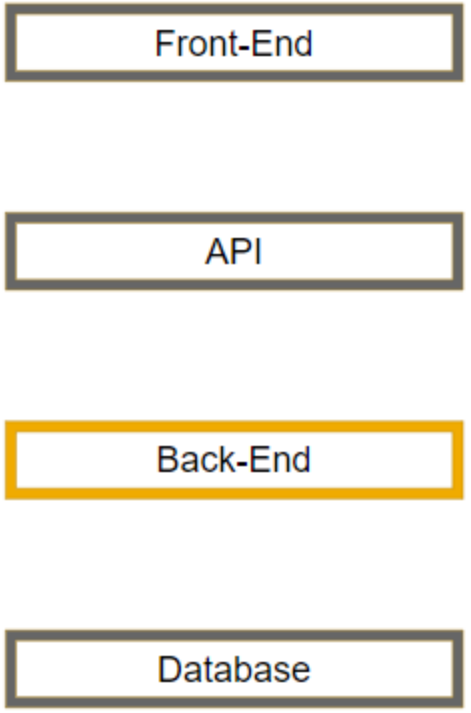
Use Case: Application to manage the risks and mitigations of a company

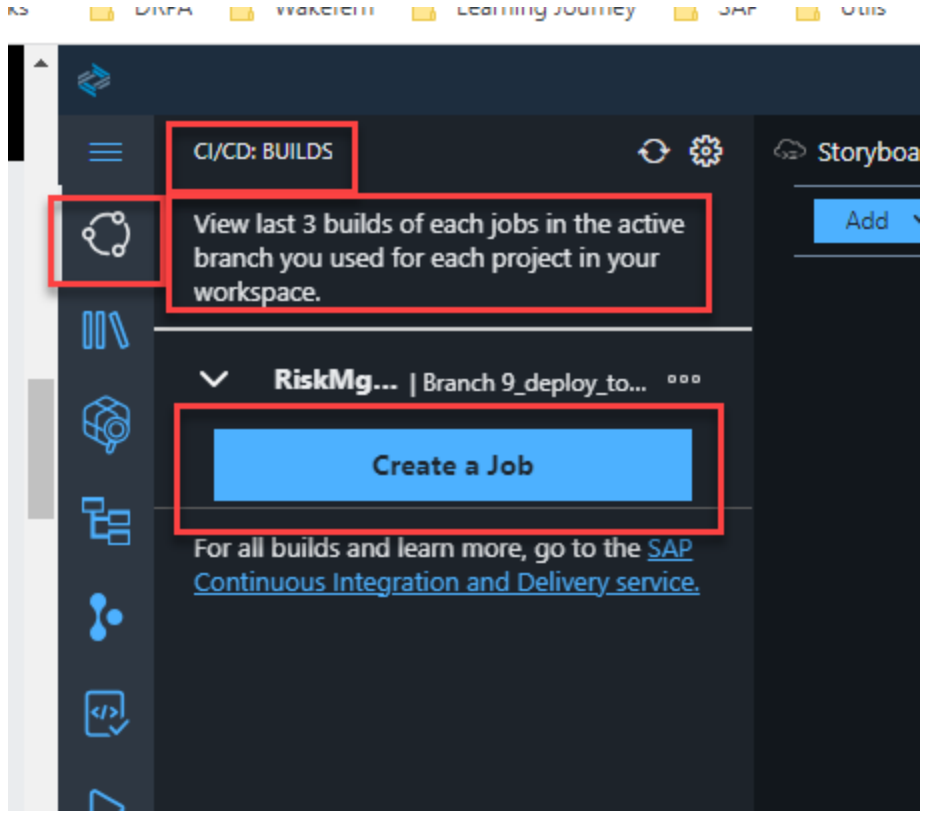
SAP Fiori Elements with List Report Floorplan as User Interface

ODATA REST API generated by CAP app

SAP Cloud Application Programming Model used to model DB entities, implement business logic and define/expose OData services

SAP HANA Cloud Database




CI/CD: BUILDS
View last 3 builds of each jobs in the active branch you used for each project in your workspace.
RiskMg... | Branch 9_deploy_to...
Create a Job
For all builds and learn more, go to the SAP Continuous Integration and Delivery service.
Add

Create a CI/CD Job : RiskMgmt

Create a job in SAP Continuous Integration and Delivery to build your project using a CI/CD pipeline.

Configure your Job 1

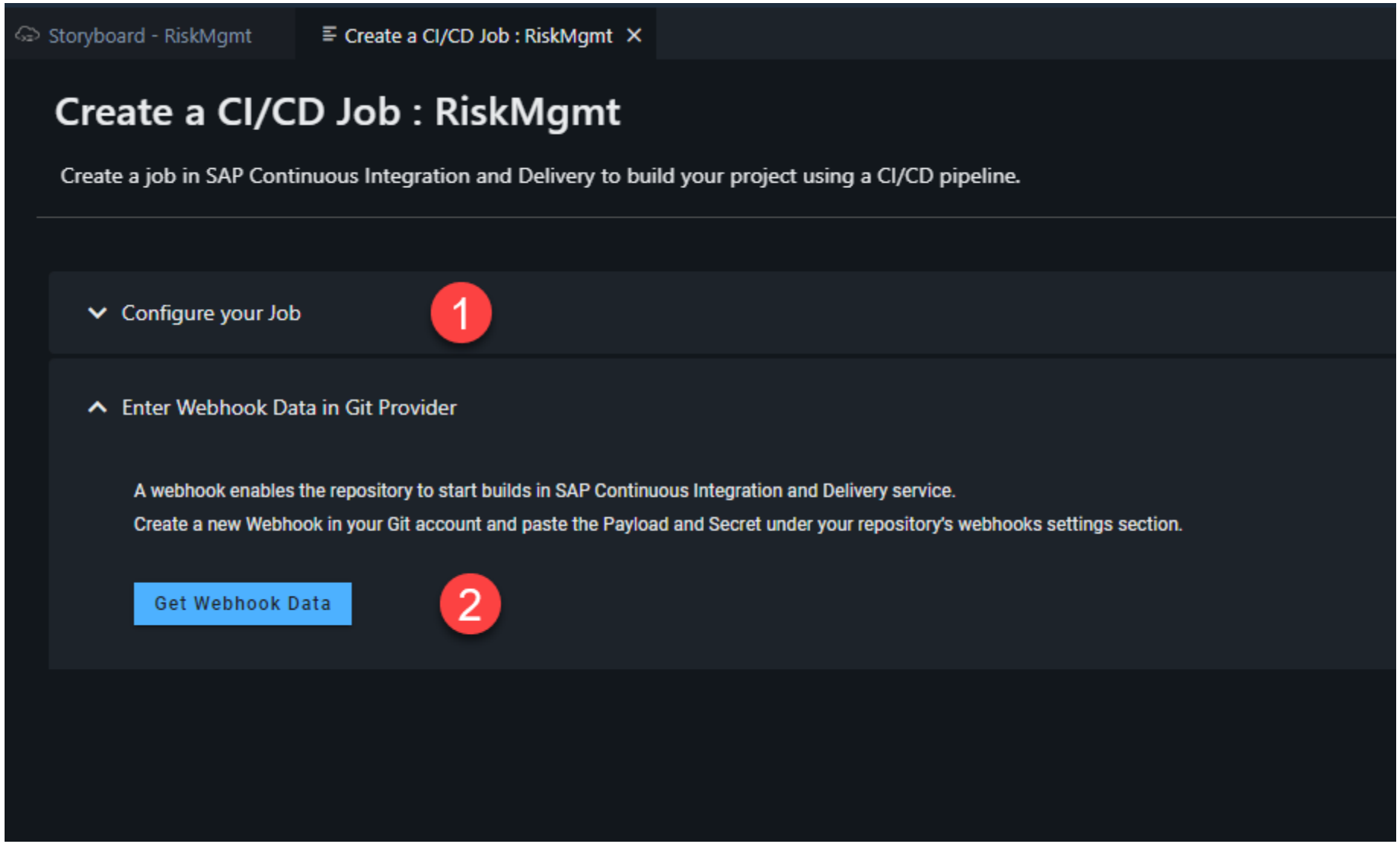
A CI/CD job is a recurring and automatic continuous integration and delivery task. It depends on a pipeline, a source repository, and various configuration settings.

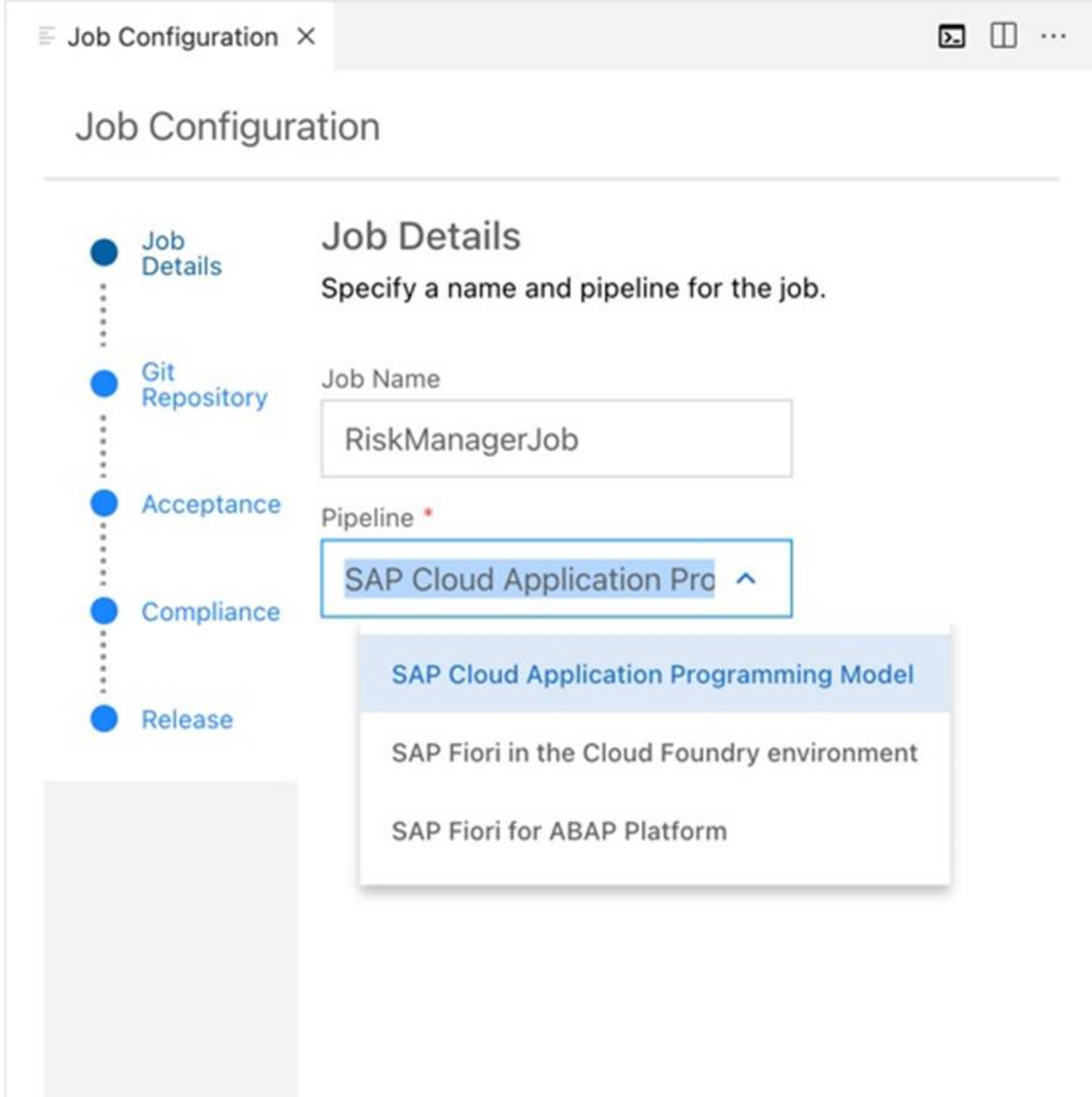
Fill in the required job details and configuration in the wizard to configure the job. The configured job will run the following build steps:

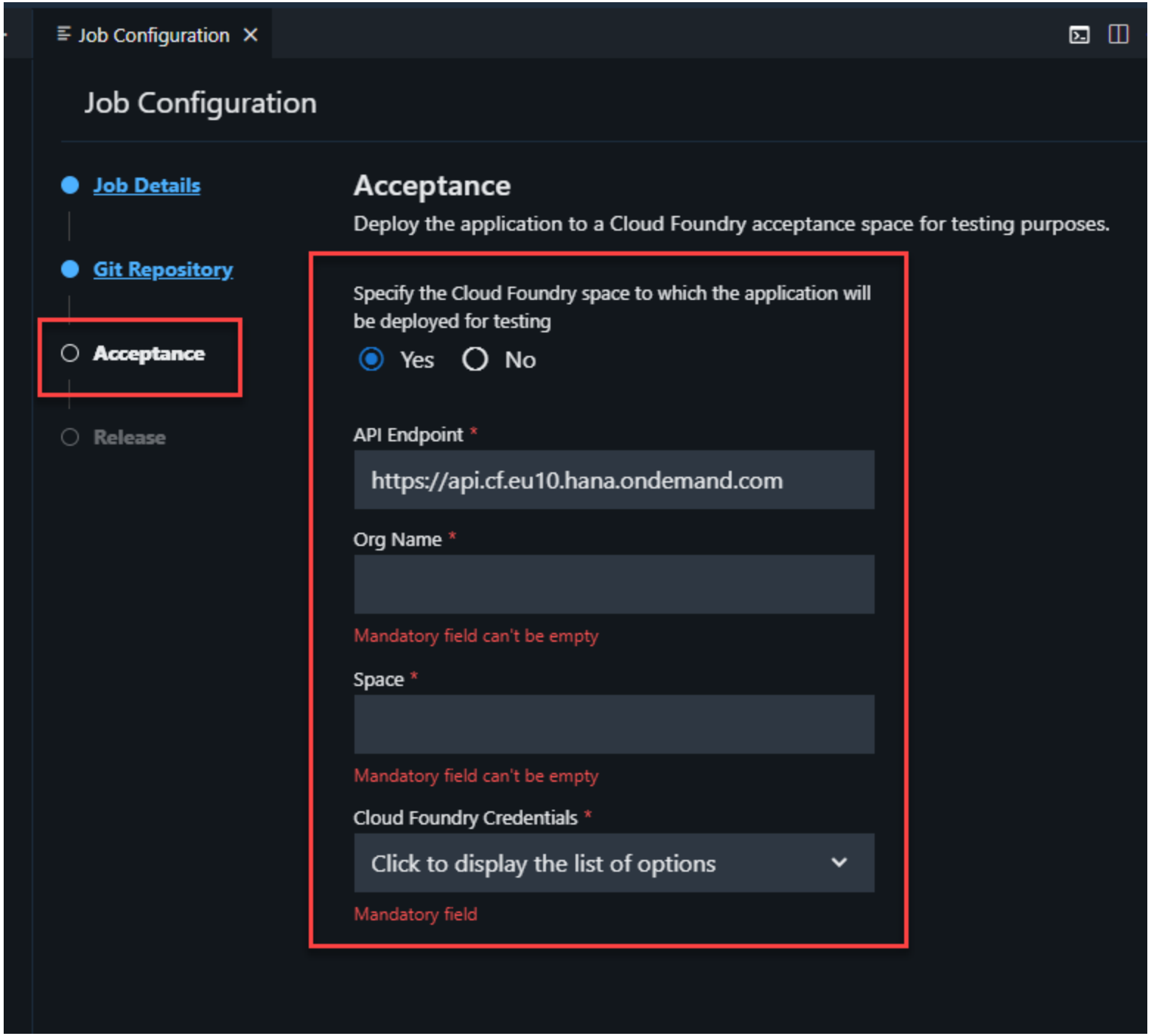
- Build the application using the selected build tool.
- Perform an optional unit test and code scan.
- Perform optional deployment steps:
  a. Deploy the application to the acceptance space (for example, Cloud Foundry) for testing purposes.
  b. Deploy the application to the Cloud Foundry space or ABAP platform as part of the release.
  c. Upload the application artifacts to the Cloud Transport Management service. Using the Cloud Transport Management service, you can implement approval processes for deploying your application. For more information, see Integrate SAP Cloud Transport Management into Your Pipeline.

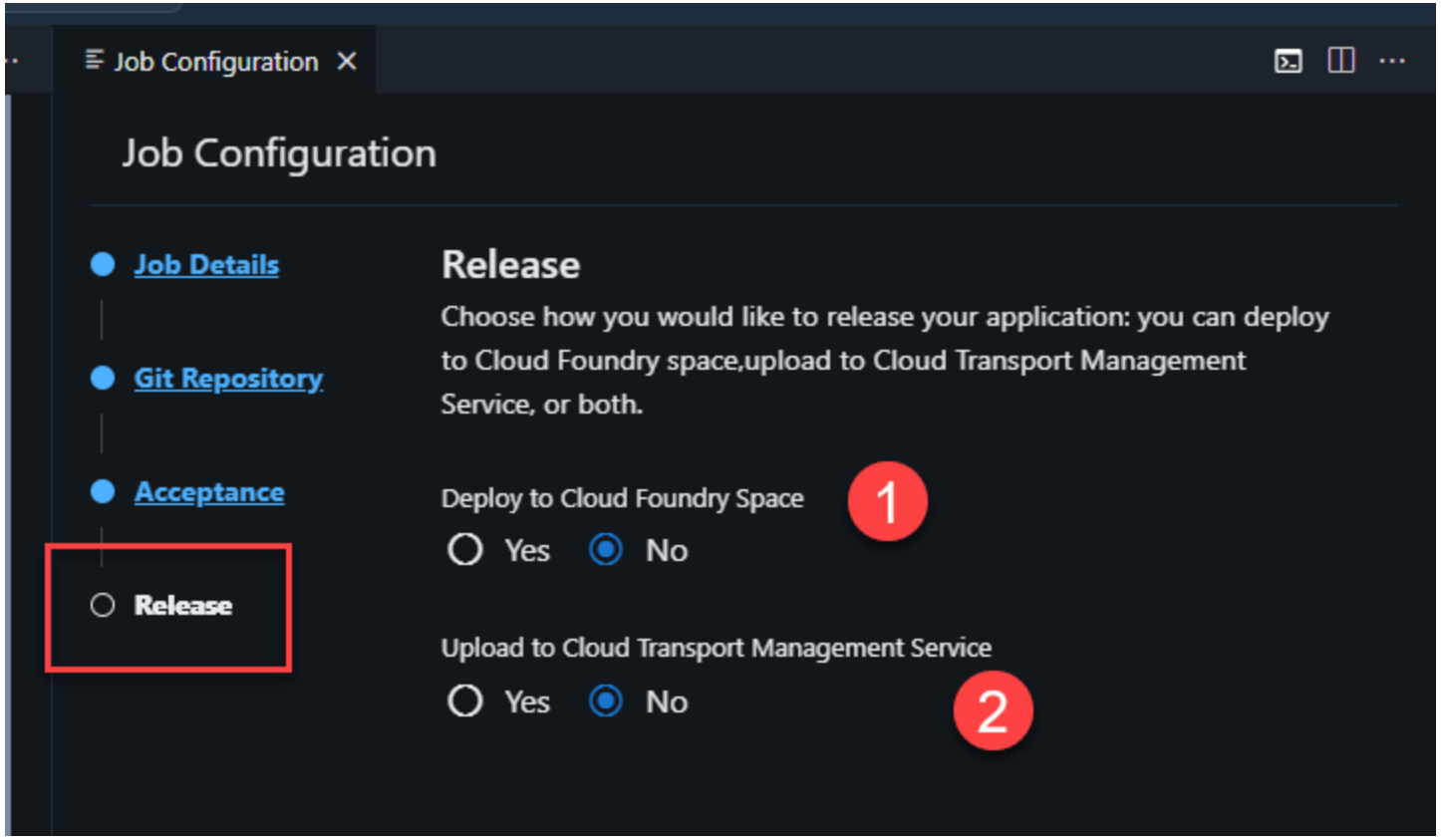
Configure Job

Enter Webhook Data in Git Provider 2


Storyboard - RiskMgmt
Create a CI/CD Job : RiskMgmt
Create a CI/CD Job : RiskMgmt
Create a job in SAP Continuous Integration and Delivery to build your project using a CI/CD pipeline.
Configure your Job
1
Enter Webhook Data in Git Provider
A webhook enables the repository to start builds in SAP Continuous Integration and Delivery service.
Create a new Webhook in your Git account and paste the Payload and Secret under your repository's webhooks settings section.
Get Webhook Data
2


Job Configuration
Job Configuration
Job Details
Git Repository
Acceptance
Compliance
Release
Job Details
Specify a name and pipeline for the job.
Job Name
RiskManagerJob
Pipeline *
SAP Cloud Application Pro
SAP Cloud Application Programming Model
SAP Fiori in the Cloud Foundry environment
SAP Fiori for ABAP Platform


Job Configuration
Job Configuration
Job Details
Git Repository
Acceptance
Release
Acceptance
Deploy the application to a Cloud Foundry acceptance space for testing purposes.
Specify the Cloud Foundry space to which the application will be deployed for testing
Yes
No
API Endpoint *
https://api.cf.eu10.hana.ondemand.com
Org Name *
Mandatory field can't be empty
Space *
Mandatory field can't be empty
Cloud Foundry Credentials *
Click to display the list of options
Mandatory field


Job Configuration
Job Configuration
Job Details
Git Repository
Acceptance
Release
Release
Choose how you would like to release your application: you can deploy to Cloud Foundry space,upload to Cloud Transport Management Service, or both.
Deploy to Cloud Foundry Space
1
Yes
No
Upload to Cloud Transport Management Service
Yes
No
2

Guided Development | Home | Automate Build Triggers ×

Automate Build Triggers with Webhooks

Configure your Git settings so that a build is triggered automatically every time you commit and push changes.

1. Navigate to the [redacted] Git repository.

2. Find out how to create your webhook in your Git provider, as described in the documentation.

3. Create a new webhook using the following data:

1 **Payload URL:** [redacted]

2 **Content type:** application/json

3 **Secret:** [redacted]a0db5e8d0

4. Once the webhook is created, you can commit and push your changes to automatically trigger builds.

Note

In case the build is not triggered automatically, please do the following:

1. Choose the Overflow button beside the job name to open the context menu.
2. Choose *Trigger a build*.
3. In the confirmation dialog, you can choose whether you want your builds to be triggered automatically or manually in the future upon committing changes to your Git repository.