

# SIEMENS

Insights Hub

Developer Cockpit for Subscribers to MindAccess  
Plans

System Manual  
04/2024

Introduction

1

Application Lifecycle

2

User interfaces inside Developer  
Cockpit

3

User rights in "Developer Cockpit"

4

Application types and specifications

5

Workflow of Developer Cockpit

6

Managing an application in  
Developer Cockpit

7

Handing over the application

8

Assigning the application

9


FAQs


10


## Legal information

### Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

 <b>DANGER</b>
indicates that death or severe personal injury <b>will</b> result if proper precautions are not taken.

 <b>WARNING</b>
indicates that death or severe personal injury <b>may</b> result if proper precautions are not taken.

 <b>CAUTION</b>
indicates that minor personal injury can result if proper precautions are not taken.

<b>NOTICE</b>
indicates that property damage can result if proper precautions are not taken.


If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

### Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

### Proper use of Siemens products

Note the following:

 <b>WARNING</b>
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

### Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

### Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

# Table of contents

<b>1. Introduction</b> .....	5
1.1. Introduction.....	5
<b>2. Application Lifecycle</b> .....	7
2.1. Application Lifecycle.....	7
<b>3. User interfaces inside Developer Cockpit</b> .....	9
3.1. User interfaces.....	9
<b>4. User rights in "Developer Cockpit"</b> .....	12
4.1. User rights in "Developer Cockpit".....	12
<b>5. Application types and specifications</b> .....	13
5.1. Application types and specifications.....	13
<b>6. Workflow of Developer Cockpit</b> .....	16
6.1. Workflow of Developer Cockpit.....	16
<b>7. Managing an application in Developer Cockpit</b> .....	18
7.1. Add an application.....	18
7.2. New version of an application.....	24
7.3. Register an application.....	26
7.4. Assign application Roles and Scopes.....	29
7.5. Create a new customized role for an application.....	32
7.6. Manage API application with message broker.....	33
7.7. Issue application credentials.....	35
7.8. Revoke application credentials.....	37
7.9. View the registered Insights Hub Monitor applications in Insights H...	38
<b>8. Handing over the application</b> .....	40
8.1. Application handover to an operator.....	40
8.2. Managing uploads of an application for Cloud Foundry.....	41
8.3. Preparation release processes for self hosted and without backend ...	44

8.4. Approve uploads of an application.....	46
<b>9. Assigning the application.....</b>	<b>49</b>
9.1. Assign an application to an Operator.....	49
9.2. Updating new configuration of the assigned application.....	50
9.3. Manage promoted apps.....	52
<b>10. FAQs.....</b>	<b>54</b>
10.1. FAQs.....	54

# Introduction

## 1.1 Introduction

The developers can create and manage applications within the platform and use various services. The [MindAccess Developer Plan](#) includes Cloud Foundry Compute Resources for developing applications using various buildpacks.

Before subscribing to the MindAccess Developer Plan, you can start your journey with various basic applications in [Insights Hub Start for Free](#) package. Later, you can upgrade the package to the next level. In Insights Hub Start for Free package, "Cloud Foundry" feature is not activated.

## Overview of Developer Cockpit

Developer Cockpit provides a workbench to create apps and custom-built solutions for Insights Hub. It helps developers to create, host and manage the registered applications on the platform. By registering your application, the application will be made available via a Insights Hub URL and enable you to test your own application.

## Functions and benefits

Developer Cockpit provides the following functionalities:

- Create an application with multiple components and corresponding endpoints.
- Overview of developed applications.
- Configure applications, for example "Application Icon" or "Display Name".
- Secure applications at the endpoint level and manage application specific roles.
- Test the application on your own tenant, before promoting it to an operator.
- Promote and transfer applications from developer to an operator.

## Prerequisites

The following prerequisites must be fulfilled in order to access Developer Cockpit:

- Access Insights Hub Start for Free to own your free tenant

- Subscription to MindAccess Developer Plan
- Log in to Developer Plan Launchpad
- Role: mdsp:core:Developer or mdsp:core:DeveloperAdmin assigned via Settings application

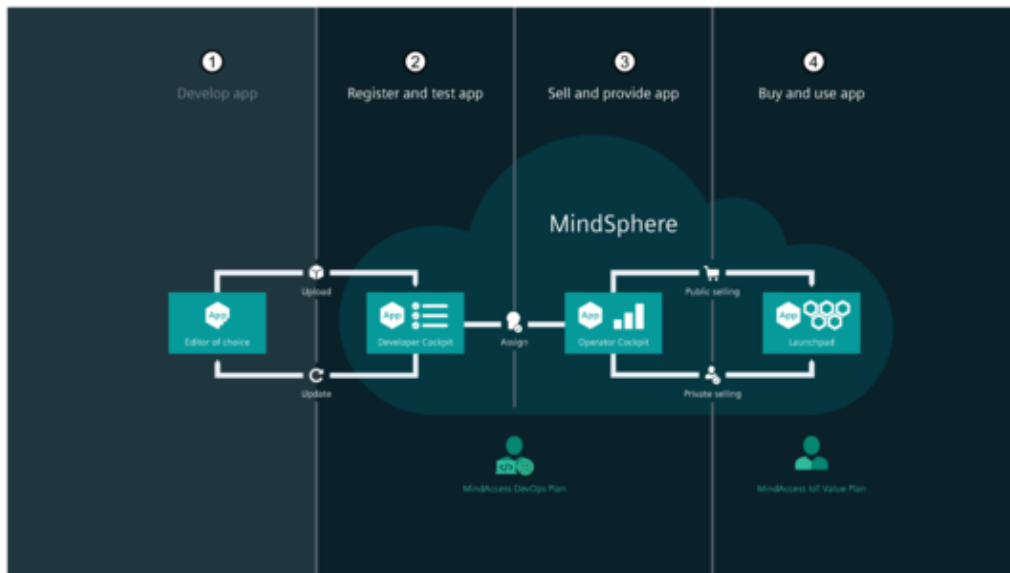
# Application Lifecycle

# 2

## 2.1 Application Lifecycle

The Application Lifecycle depicts the workflow from the development of an application until the productive use of an application. The Application Lifecycle describes the correlation between the developer and the IoT tenant/end customer user.

The following graphic depicts the Application Lifecycle with the respective processes:



① Develop an application:

- The developer develops the application in the local environment

② Register and test the application:

- The developer creates the application in Developer Cockpit
- The developer tests the application inside the Insights Hub environment
- The developer assigns the application to operators

③ Sell and provide the application:

- The operator accepts the application
- The operator registers the application to make it available on his tenant
- Operator Cockpit publishes the application in Industrial IoT Store

④ Purchase and use the application:

- Any IoT Value Plan or Developer Plan owner can make a subscription for applications from Industrial IoT Store

- After the application is provisioned by the operator, it will be available on the tenant
- API applications can be provisioned to other operators.



# User interfaces inside Developer Cockpit

# 3

## 3.1 User interfaces

Developer Cockpit application can be accessed from Insights Hub Launchpad by clicking on the following icon:



Click "Developer Cockpit" icon to display the dashboard screen. The "Dashboard" gives you a quick overview of all available applications in Developer Cockpit. The following graphic shows the Dashboard of Developer Cockpit.

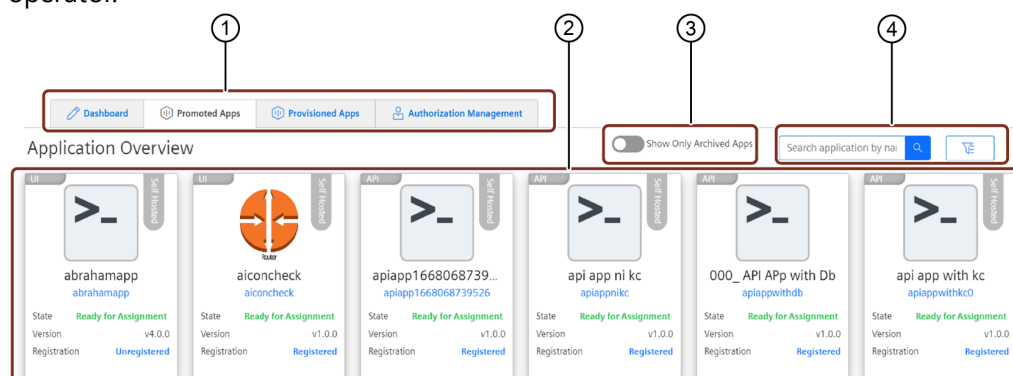
### "Dashboard" User interface



- ① Navigation tabs can quickly navigates through Developer Cockpit
- ② List of available applications
- ③ Search a specific application
- ④ Filter the applications

## "Promoted Apps" User interface

The "Promoted Apps" screen shows all applications which are ready to be handed over to the operator.



- ① Navigation tabs can quickly navigates through Developer Cockpit
- ② List of the promoted applications
- ③ Toggle button to show only archived apps
- ④ Search and filter the promoted applications

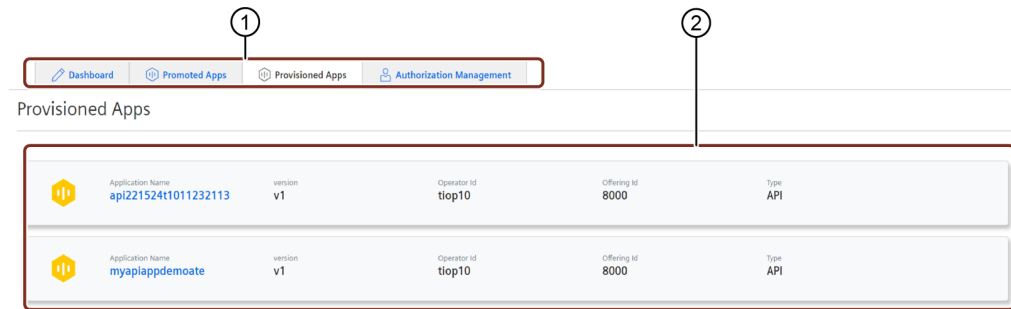


- For the outdated applications, you cannot change the states "Remediation Required" or "Deprecated". These states will remain in the "Promoted Apps" tab.
- The outdated apps with these statuses can be deleted.

## "Provisioned Apps" User interface

3.1 User interfaces

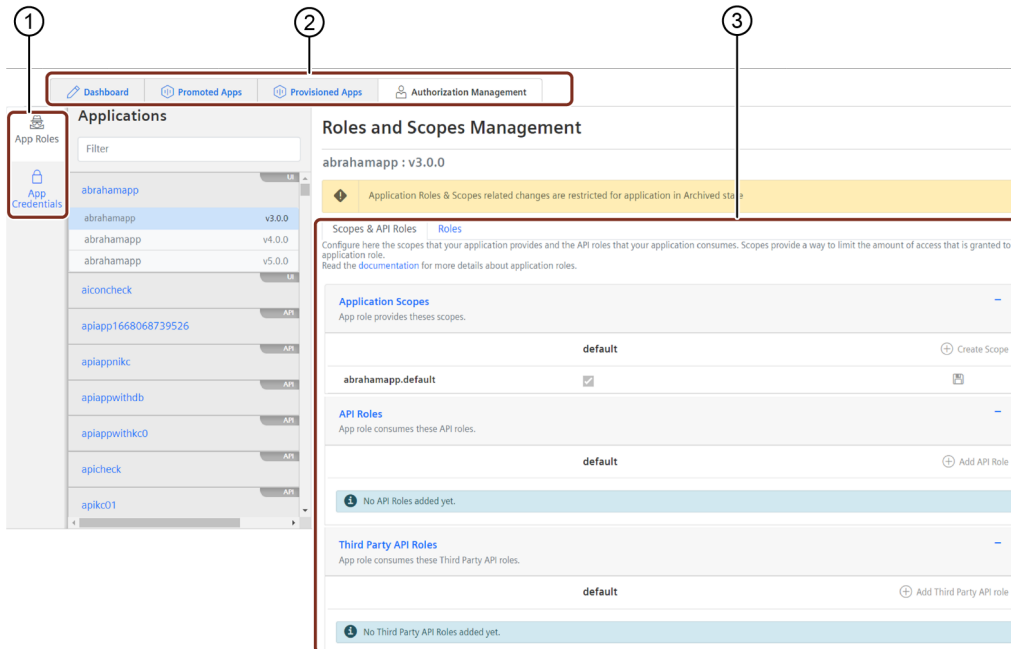
The "Provisioned Apps" screen shows the list of all the third-party API applications provisioned from the other tenants.



- ① Navigation tabs can quickly navigates through Developer Cockpit
- ② List of the provisioned API applications

## "Authorization Management" User interface

The "App Roles" and "Application Credentials" are the two features to manage the authorization of the application. For more information, see [Assign application Roles and Scopes](#).



- ① Navigation tabs available in Authorization Management
- ② Navigation tabs can quickly navigates through Developer Cockpit
- ③ Display area of Roles and Scopes

# User rights in "Developer Cockpit"

# 4

## 4.1 User rights in "Developer Cockpit"

Developer Cockpit is available with two different user roles, Developer and Developer Admin. The key difference between a Developer and a Developer Admin is the ability to transfer the developed application to the application repository and assign the application to one or more operator accounts.

### User rights

The following table shows the assignment of roles:

Right	Role: Developer	Role: Developer Admin
Create applications	✓	✓
Assign roles and scopes	✓	✓
Application credentials create and revoke	✓	✓
Register applications	✓	✓
Promoting applications	✓	✓
Manage assignments		✓
Delete applications	✓	✓

# Application types and specifications

# 5

## 5.1 Application types and specifications

This chapter describes the available types and infrastructure.

### Types of applications

Developer Cockpit allows you to add the following types of applications:

- **Standard:** The standard type of application will be hosted in the Launchpad after the handover is done. Such types of applications provide a user interface for their users.
- **Insights Hub Monitor Plugin:** The created applications will be visible in Insights Hub Monitor.
- **Mobile:** The mobile type of application represents a mobile native application connected to Insights Hub. It will be available via your self-written app on your device. Also it supports app and universal links
- **API:** API application is not possible to use on their own as it always requires one or multiple depending on frontend.
- **Mendix-based applications:** Mendix is a software platform that provides tools for building, testing, and deploying applications. This feature is applicable only for Cloud Foundry and self-hosted applications. When your application is marked with Mendix feature, then the operator gets informed upon the handover that a Mendix license is required to run this application. For more information on Mendix, see [Developing Mendix Apps](#).

When applications are added, they are available in Developer Cockpit dashboard. After the registration of the applications, the standard UI applications are available in Launchpad and the Insights Hub Monitor plugin applications are available in the Insights Hub Monitor dashboard. Mobile type applications are available exclusively for iOS and Android mobile devices.

### Types of application infrastructure

**Cloud Foundry:** These applications will be hosted in the provided Cloud Foundry space. It supports applications that consist of one or more components. Each component is represented by a Cloud Foundry application that requires to be run within Cloud Foundry Space. Currently, only Cloud Foundry applications that have a single Cloud Foundry manifest configuration are supported.

**Self-hosted:** Developer Cockpit provides a platform to integrate your own applications with Insights Hub. The self-hosted applications will be hosted on the private infrastructure.

**None (without infrastructure):** This type of infrastructure does not require any separate backend. Only mobile type of applications can be hosted here, where all logic is managed in the mobile application. However, they are exclusively available only for iOS and Android mobile devices.

## Supported infrastructure for each type of application

The following table describes the supported infrastructure for each type of application:

Type of application	Cloud Foundry	Self-hosted	None (without infrastructure)
Standard application	✓	✓	-
Insights Hub Monitor Plugin application	✓	✓	-
Mobile application	✓	✓	✓
API application	✓	✓	-
Mendix-based applications	✓	✓	-

## States for an application

Any application must go through the following states in sequential order in Developer Cockpit, before being available in Launchpad:

- Adding application
- In Development
- Registration
- Preparation in progress
- Waiting for approval
- Ready for assignment, in case of passed applications
- Application handover
- Check failed, in case of failed applications
- Archived, if the application is not required for time being

For self-hosted applications, once an application reaches the "Ready for assignment" state, they can be assigned to operators. This is because the self-hosted applications are third-party applications and have URLs that are registered with Insights Hub for execution. If any errors are detected during executions, they just require fixes. Hence, no further checks are required after the "Ready for assignment" state, unlike Cloud Foundry (CF) applications.

## States of handover

### In "Dashboard" for the applications:

- In-Development: The first state is the application being in the "In-Development" phase. An application will move to this phase after its creation. The application needs to be then registered.
- Preparation in progress: This is the second phase of the process. After pushing the binaries into the application, the application moves to the state of initiating the upload.
- Waiting for approval: This is the third phase of the application handover. An application reaches this phase when its metadata information is submitted successfully.
- Check in progress: This is the last successful phase before pushing the application to the "Promoted Apps" tab. After an application is ready for its upload, Developer Cockpit submits its requirements for getting approval from the administrator of the requested service plans. You can successfully submit the application for approval only if the application has one role and one scope added in it. The application will be uploaded successfully.
- Check failed: An application goes to this state by the following reasons:
  - Mismatch in meta-data
  - Risk found while scanning the uploaded files

You will get a notification in Developer Cockpit and also via failure mail.



For more information about binary files and manifest file uploads, see [FAQ](#).

### In "Promoted Apps" for the applications:

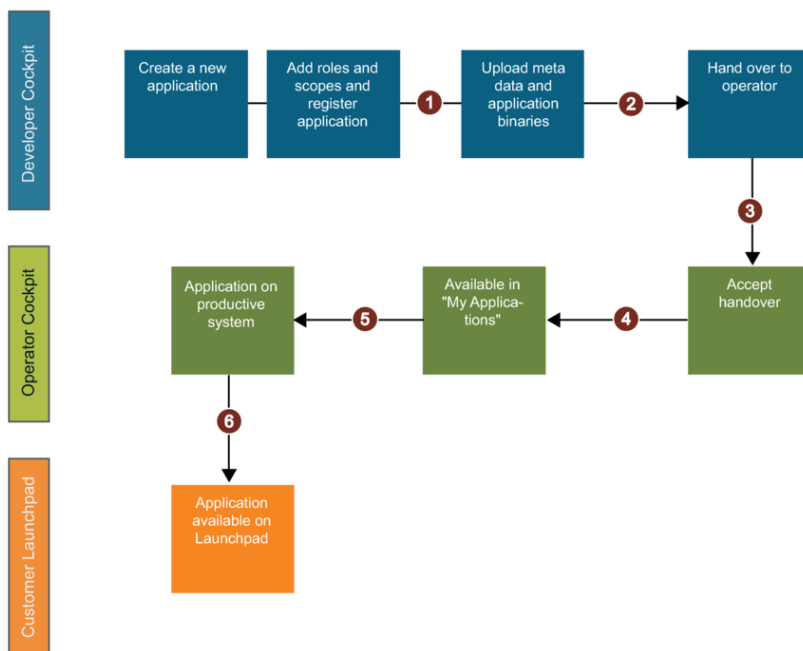
- Ready for assignment: If the upload request of the application is successful, the submitted application is approved, the application is pushed to the "Promoted Apps" with its new state. The application is now ready to be assigned to operators.

# Workflow of Developer Cockpit

# 6

## 6.1 Workflow of Developer Cockpit

The following schema describes the communication between the developer and the operator using Developer Cockpit and Operator Cockpit.



① Add application | Add roles and scopes and register application:

- Use Developer Cockpit to add a new application.
- Add roles and scopes for the added application.
- Use Developer Cockpit for application registration.

② Upload application:

- Upload meta data. Example - the manifest file.
- Enter service names.
- Upload zip-file with application binaries.

③ Hand over application to Operator Cockpit:

- Assign the successfully uploaded application data to a known operator.
- Operator accepts the assignment in Operator Cockpit.

④ Accept handover:



- After acceptance is received from the operator, the developer grants the application to an operator.
- Cloud Foundry application is then deployable for an operator inside Operator Cockpit.



The Self-hosted application can be deployed directly.

⑤ Deployment in productive system:

- In Operator Cockpit, the operator deploys Cloud Foundry application to the productive system.
- Operator can publish this application to Industrial IoT Store, or provision application to known customers.

⑥ Provisioning to Customer:

- In Operator Cockpit, the operator provisions the application to the known customers.
- If the operator has published an application to Industrial IoT Store, a subscription notification to provision to a new customer is received.

# Managing an application in Developer Cockpit 7

## 7.1 Add an application

Adding an application in Developer Cockpit is the first step to start your journey. You can add different types of application based on their app and hosting type. For more information on the supported application types and infrastructures, see ["Application types and specifications"](#).

### Pre-requisites for creating an application

The following requirements apply for integrating your application into Insights Hub:

- Your self-hosted application is reachable via DNS and a valid SSL certificate (self-signed certificates are not allowed).
- Your self-hosted application exposes one health endpoint for internal monitoring.
- Your Cloud Foundry hosted application must have the deployable Cloud Foundry application ready.
- Your mobile application without backend must be installable on a mobile device.

### User Interface

#### "Create Application" screen:

You can add your application to Developer Cockpit by filling the below fields.

The screenshot shows the 'Create Application' form with the following fields and options:

- Display Name:** A text input field with a placeholder 'Display Name'.
- Type:** Radio buttons for Standard (selected), Insights Hub Monitor Plugin, Mobile, and API. A note states: 'Standard applications always provide a user interface for users.'
- Infrastructure:** Radio buttons for Cloud Foundry (selected) and Self Hosted. A note states: 'Application infrastructure type once selected can not be changed.'
- Internal Name:** A text input field with a placeholder 'Internal Name'. A note states: 'The application name will be part of application URL and application specific role names.'
- Version:** A text input field with a placeholder 'v1.0.0'. A note states: 'The display name will be shown beneath the application icon on the Launchpad.'
- Description:** A text area with a placeholder 'Description'.
- App Icon:** A preview of a blue icon with a white symbol, labeled 'App Icon' and 'icon.png'. There is an 'Edit icon' link.
- Endpoints:** A section with a 'Name' field (placeholder: 'e.g. FileService, UIService etc.') and a 'Cloud Foundry Direct URL' field (placeholder: 'https://<name>.apps.eu1-dev.mindsphere.io/<ContextPath>'). There are 'Add/Edit Component' and 'Delete Component' buttons.

Parameter	Description
-----------	-------------

7.1 Add an application

Parameter	Description
Type	You can create four types of applications: Standard, Insights Hub Monitor Plugin, Mobile and API.
Infrastructure	Three types of infrastructures are available: Cloud Foundry, Self-hosted and None (valid for mobile apps).
Display Name	Enter the name of the application that you want to display on the Launchpad. Also, this name is displayed to the Operator. This field does not support any special characters and it is limited to 50 characters only.
Internal Name	By default, the "Internal Name" is auto-filled by entering the display name in lower case and without special characters. The internal name is used later in the URL and must be unique for your tenant. Enter the name for the application with the following conditions: - The name must start with an alphabet. - Only lower-case letters are allowed. - A maximum of 20 alphanumeric characters is allowed.
Version	By default, "v1.0.0" version is auto-filled in the version field and it can be changed as per the requirement. The version supports alphanumeric string data. Spaces in between the nomenclature format are not supported. A maximum of 30 alphanumeric characters is allowed. Use of capital letters for naming versions is not allowed. However, it supports the following special characters: - # - \$ - . - ; - + - - - _ - @ In the version field in Developer Cockpit, if there are any special characters (except "@", "."), the provisioning workflow fails in Operator Cockpit. For API applications, the following format needs to be followed: <code>v&lt;version&gt;.&lt;revision&gt;.&lt;patch&gt;-&lt;label(optional)&gt;</code> e.g. v1.0.0, v2.1.0-alpha - <version> is version, number with range 1-9. - <revision> is revision, number with range 0-99. - <patch> is patch number, number with range 0-99. - <label> is label and optional, case insensitive alphabet.
Mendix-based application	This check box is used to notify the operators that a Mendix license is needed to run the app.
Description	Add a description for the application. This is an optional field. This provides information about the application and will be visible to the operator.

Parameter	Description
Icon	Icon of the application is displayed in Developer Cockpit dashboard screen, Launchpad and will later also be visible to an Operator and IoT customers. By default, an application icon is loaded. Upload a different image to represent the application according to your suitability. The recommended resolution of 512x512 is suitable for both high and low-resolution screens.

**"Components" screen:**

You can add multiple components based on the complexity of your application, for example, you can add the components as shown in the following figure:

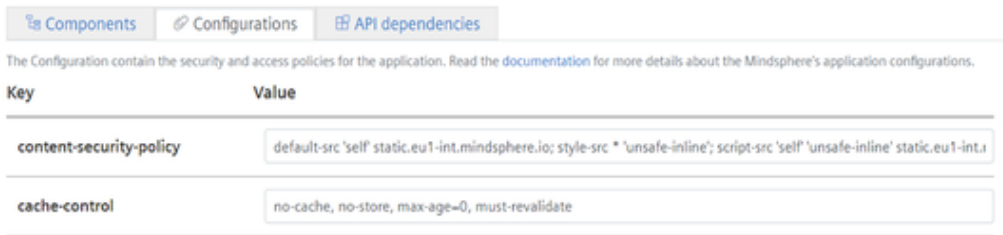


Parameter	Description
Components	Adding components to your application are used to structure your application and register it accordingly in the Industrial IoT Gateway. However, it is mandatory to have a minimum of one component with one endpoint for an application. Add at least one component to your application, if the application is a CF or self-hosted based application. If you select the infrastructure as "None", then the component field will be disabled.
Component Name	Enter the name of the component. The following conditions hold while naming a component: <ul style="list-style-type: none"> <li>- It should not include alphanumeric and '-', '_', ':' special characters.</li> <li>- A maximum of 40 characters are allowed.</li> <li>- The "Name" field of the application in the component tab supports only lower-case letters.</li> <li>- The component name should be same as the application name in the .yml file.</li> </ul> Accordingly, the name your application in the manifest file, or during the push operation. Otherwise, the upload of the application will fail. This is applicable for Cloud Foundry application only.
URL	<ul style="list-style-type: none"> <li>- Enter the URL for the component for a CF application. The URL format should be valid in the format: https://&lt;name&gt;.apps.&lt;region&gt;-dev.mindsphere.io.</li> <li>- Enter the direct URL for each of the components for a self hosted application. The URL format should be valid in the format and not malicious: https://apps.example.com/&lt;ContextPath&gt;.</li> </ul>
Endpoints	The endpoints section will remain disabled until "Name" and Cloud Foundry or Self-hosted direct URL's have values. For new application, the "Edit" button will remain disabled since a new application is being added.

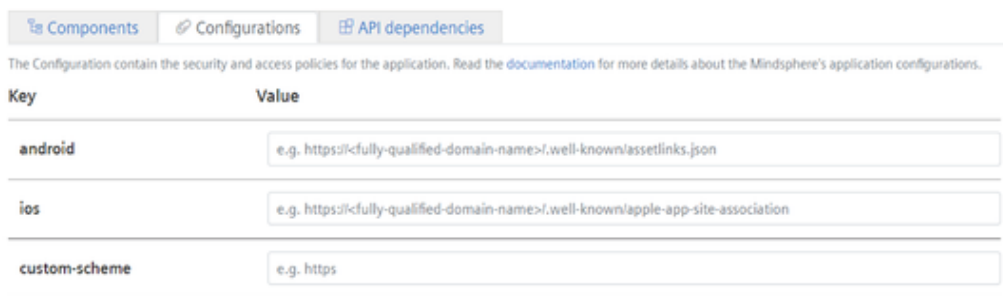
Parameter	Description
Matching Order	<p>The Industrial IoT Gateway selects the first matching endpoint based on the request URL. Currently, there is no proximity matching available.</p> <p>You can add arbitrary paths including wildcards here except /api, as this is reserved for calling the APIs from the browser client of your application.</p> <p>The mapping matches URLs using the following rules:</p> <ul style="list-style-type: none"> <li>?matches one character</li> <li>**matches zero or more directories in a path</li> <li>a specific path, e.g./machines</li> <li>a path with wildcard/machines/*</li> <li>a patch with general wildcard/**that matches also zero or more sub paths.</li> </ul> <p><b>Examples</b></p> <p>/machines/drill-? matches machines/drill-1 but also machines/drill-2 or machines/drill-a /monitoring/machines/drill-* - matches all drill-* endpoints in the /monitoring/machines endpoint /machines/**/list matches all list endpoints underneath the /machines path.</p>

**"Configurations" screen:**

For Standard, Insights Hub Monitor Plugin, and API applications



For Mobile applications

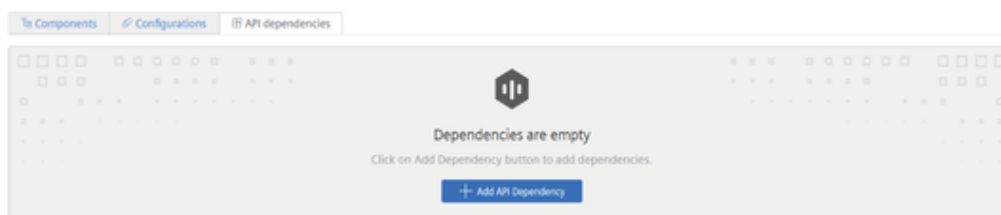


Parameter	Description
Content-Security-Header	<p>Configuring Content Security Policy header, prevents from possible attacks and execution of malicious content or code and makes your application more secure. For more information, see <a href="#">"Configuring Content Security Policy"</a>.</p> <p>By default, the configuration value is taken for the content security policy. The content security policy header value can be only 1000 characters long, rest of the characters will be discarded.</p> <p>The field cannot be empty. If you try to save the application by keeping the field empty, backend service will set the default values.</p>

Parameter	Description
Cache control	<p>With Cache control, each cache will be revalidated before using a cached response to capture the uncacheable responses.</p> <p>By default, the configuration value is taken for the cache control. The cache control value can be only 255 characters long, rest of the characters will be discarded.</p> <p>The field cannot be empty. If you try to save the application by keeping the field empty, backend service will set the default values.</p>
Configurations for applications (For Mobile applications)	<p>Enter the Android App Links and Universal Links configuration information for "Android" and "iOS" respectively as per the specified format.</p> <p>These links seamlessly link to content inside your application. Using these links, you can always give users the most integrated mobile experience, even when your application is not installed on the device.</p> <p>The link configurations are the special URLs which invokes specific screen of native mobile application. These links can be non-HTTP(S) based URLs (called as custom links) or strictly HTTP(S) based URLs. Mobile OS allows multiple applications to register with the same link. Whenever user navigates to any of such links, mobile OS allows the registered user to select one of the applications.</p> <p>For more information on "Android" and "iOS" application links, see <a href="#">"Mobile Application Developer Documentation"</a>.</p> <p>Hosting of assetlinks.json and AASA files: In configurations, the field user can provide the link to the "apple-app-site-association file" or to "assetlinks.json" file which is either hosted in user's domain or storage. Each application can have only one assetlinks.json and one AASA file. The maximum size of each file is 2kb.</p> <p><b>Note:</b></p> <p>The "Configurations" and "Icon" fields for mobile applications are optional.</p>
Android	<p>Domain validation is performed by Android, and it expects a special file assetlinks.json to be hosted under <code>https://&lt;domain&gt;/well-known/</code> folder.</p> <p>The field value can be only 255 characters long.</p>
iOS	<p>Apple expects a special file apple-app-site-association (AASA) under <code>https://&lt;domain&gt;/well-known</code> or directly under <code>https://&lt;domain&gt;</code>.</p> <p>The field value can be only 255 characters long.</p>
Deep link custom-scheme	<p>Domain validation is performed by customized scheme like, "<code>https://&lt;domain&gt;/.</code>" or "<code>mobile://&lt;domain&gt;/.</code>"</p>

### "API dependencies" screen:

For adding API dependencies, click the "Add API Dependency" button. Add the API dependencies for the application from the "API dependencies" pop-up screen.



Parameter	Description
<b>API dependencies for applications (for standard, Insights Hub Monitor Plugin, and mobile applications)</b>	A user can develop backend applications, to support single or multiple UI apps. These applications are called API type apps. The developer can re-use the functional logic in multiple applications. API dependencies for applications enables you to build UI applications on top of already existing API applications from other developers. To use this new functionality, an operator will provision the API app to your developer tenant, and you can continue to use the provided functionality. Currently provisioned apps can be used only with Custom App Credentials and by using Advanced Token Exchange. To learn more about the Advanced Token Exchange, see <a href="#">Accessing /Token APIs</a> .
Add API Dependency	Add the API from the list. Maximum 5 dependencies can be added for an application.
Name	Name of the API dependency
Type	Select "Internal" or "External" or both
Version	Version of the API dependency
Operator	Operator name who provides the API app to you.
Registered	State of the API dependency application is registered or not registered.

## Procedure for adding an application

To add an application, follow the steps:

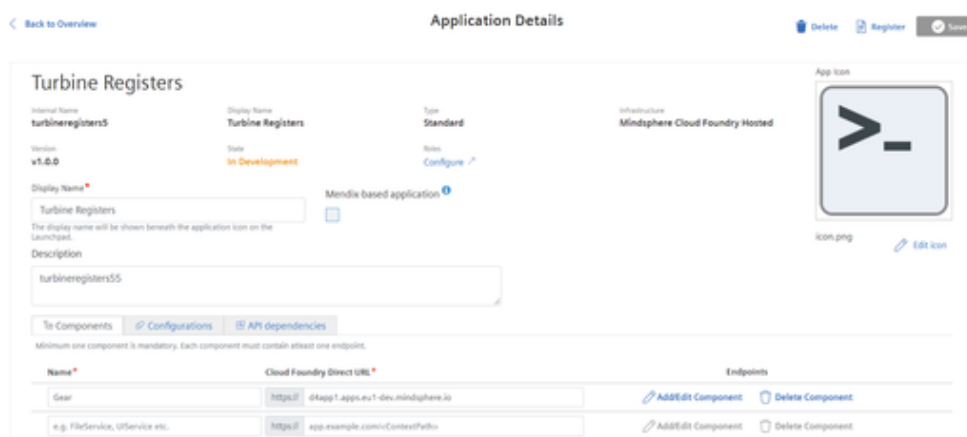
1. Log in to Developer Cockpit.
2. In "Dashboard", click "Add application".
3. Enter the field parameters as per the parameters table.
4. Upload the icon in the "App icon" field.
5. In the "Component" tab, enter the component's name and URL and click "Add Endpoint" to add the component.
6. In the "Configuration" tab, verify the configurations of the application.
7. In the "API dependencies" tab, click the "Add API Dependency" and select the API dependency and then click "Add" (optional).
8. Click "Save" to add the application in the "Dashboard" tab.



- The name and the version of the application cannot be changed after its creation. However, you can change the Application Icon, Display Name, Description and endpoint configuration.
- The configuration of an application can only be changed when it is not registered in the Gateway.
- Only "Third Party API Roles" section is not applicable in API "Authorization Management" tab as APIs cannot access other API applications.

## Result

The application is saved successfully and will be available in the "Dashboard" tab.



## Next steps

- Register the application. For more information on registering applications, see [Register an application](#).
- Configure your application specific Roles and Scopes to provide access to the application. For more information on assigning roles, see [Assign roles and scopes](#).
- Assign yourself at least one application specific role in order to access it. For more information on roles, see [Roles](#) chapter in Settings application.
- If you need to create a new version/revision for an application in Developer Cockpit, see [New version of an application](#).

## 7.2 New version of an application



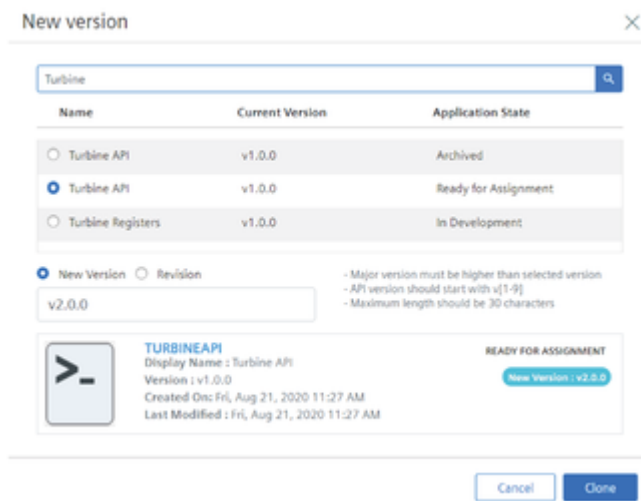
The "New Version" creates a replica of an existing application of any state. The only difference will be the version and the application state. And for API type application, you can also create a revision of the application.

The new application will have the state "In Development" on the creation of a new application version. You can save the time by creating the same application again if it fails at any point. You can modify the cloned application when in the "In Development" state. You can modify all parameters except the app name, app type and hosting type.

## User Interface

### "New version" screen:

You can clone the existing application to create a new version or revision of the application.



## Parameters table


Parameters	Description
Name	<p>Displays the name of the application.</p> <p>For API applications, the following format needs to be followed: v&lt;version&gt;.&lt;revision&gt;.&lt;patch&gt;-&lt;label (optional)&gt; e.g. v1.0.0, v2.1.0-alpha</p> <ul style="list-style-type: none"> <li>- &lt;version&gt; is version, number with range 1-9.</li> <li>- &lt;revision&gt; is revision, number with range 0-99.</li> <li>- &lt;patch&gt; is patch number, number with range 0-99.</li> <li>- &lt;label&gt; is label and optional, case insensitive alphabet.</li> </ul>
Current version or review	Displays the existing version or revision of the application

Parameters	Description
New Version or Revision	<p>You can create either new version or new revision of the application</p> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>- API type application follows the version in the format of v&lt;Major Version&gt;.&lt;Minor Version&gt;.&lt;Patch&gt; {-tag}, where tag is optional.</li> <li>- Revision number is applicable to API type applications only.</li> </ul>

## Procedure for changing version

To change the version or revision of the existing application, follow these steps:

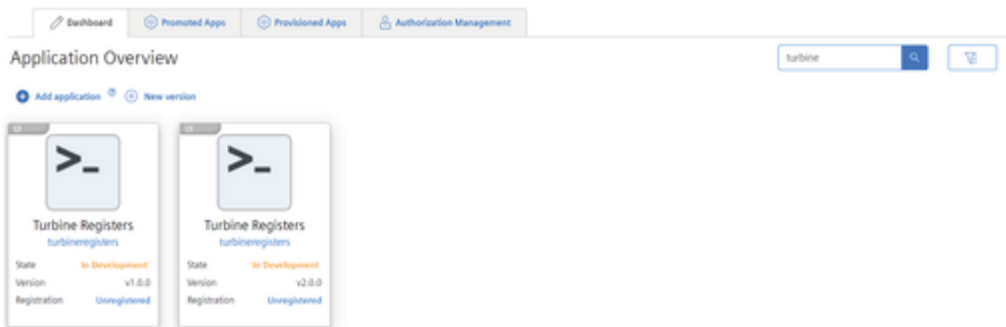
1. In Dashboard tab, click "New version".
2. Select an existing application version.
3. Enter the "New Version" and "Revision" number.
4. Click "Clone".



If you delete the scope after creating a new version or revision of the API application, the scope mapping will be impacted while cloning the application.

## Result

The new version of the application is created, and you can now change the configuration. For example, modifying endpoints etc.



## 7.3 Register an application

In order to use any application, you need to register the application. The registration make your application available and can be accessible via a Insights Hub URL.

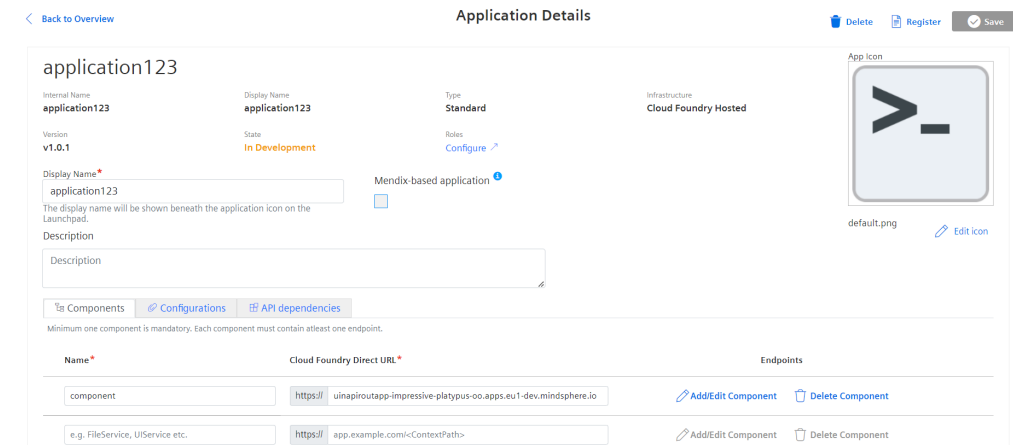
## Prerequisite

- Before proceeding for registration, it is mandatory to create at least one scope in [Roles & Scopes](#) tab for the application.
- For standard, Insights Hub Monitor plugin and mobile applications: It is only possible to register one version of an application. It is not possible to register another version without de-registering the existing version. You can circumvent this limitation by using different application names.
- For API applications: it is possible to register two versions but not two revisions of an application.
- An application cannot be registered, if it does not have any Roles, since this information needs to be provided to the Industrial IoT Gateway.

## User interface

### "Application registration" screen:

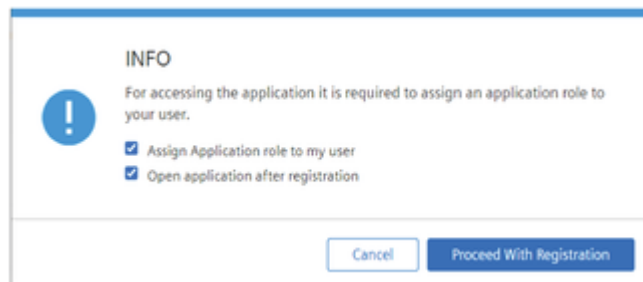
You can register your application in Insights Hub.



## Procedure for registering an application

To register the application, follow these steps:

1. In "Dashboard" tab, select your application and click "Register"



2. Click "Proceed With Registration", to confirm the application registration.

**Additional step for mobile applications:**

After registering the mobile application, application credentials are displayed. With these credentials, other application can interact or call the APIs of mobile application. Also, these credentials provides the security in interacting with other APIs as well. For more information, see [Developing Mobile Apps for Insights Hub](#).

After the application is registered successfully, a "client ID" and a "client secret" appear in the 1. pop up window. Save the credentials for future use since they are displayed only once.



Value of the "client secret" is public for each application. The "Client id" follows the below 2. nomenclature: mobile-client-<tenant>-<app>

In case you want to view the credentials again, you need to de-register and register the application again thereafter, click "Prepare Release". In the "Prepare Release" window, enter 3. the description and click "Submit".

## Result

It is now possible to access the application via a link or from Launchpad and the application will also be lauched in a new browser tab.



Insights Hub Monitor plugin, mobile applications and API applications are not visible on Launchpad.

**Example:**

URL-Schema: <tenantname>-<applicationname>-<tenantname> .  
<region>.mindsphere.io

Using this official URL, you are redirected to the application.

## Next step

- Test your application.

## 7.4 Assign application Roles and Scopes

The application roles and scopes (permissions) are used to define the access for securing your application. For more information, see [Roles & Scopes](#) for applications.

### Management of Roles and Scopes

**Scopes:** A scope is the smallest entity that describes a single permission.

Scopes describe permissions which are listed in the access token as named parameters. When accessing an endpoint or application within Insights Hub, the Identity and Access Management automatically adds the required scopes to the access token if the respective user has access permission.

Scopes must adhere to the following naming convention: {application\_name}.{scope}.

**Roles:** A role is a collection of multiple scopes (permissions). These roles can be assigned to a user.

It can either be assigned to a user via the Settings application or added to an application role to grant access to APIs. For example, the Core role `mdsp:core:iot.timUser` can be added to the application role so that this application can read time series data. This assignment makes all the scopes of the Core role available in your application role.

The scopes can be assigned to one or more default application roles. For more information on API specific roles and the available scopes, see [Developer Documentation](#).

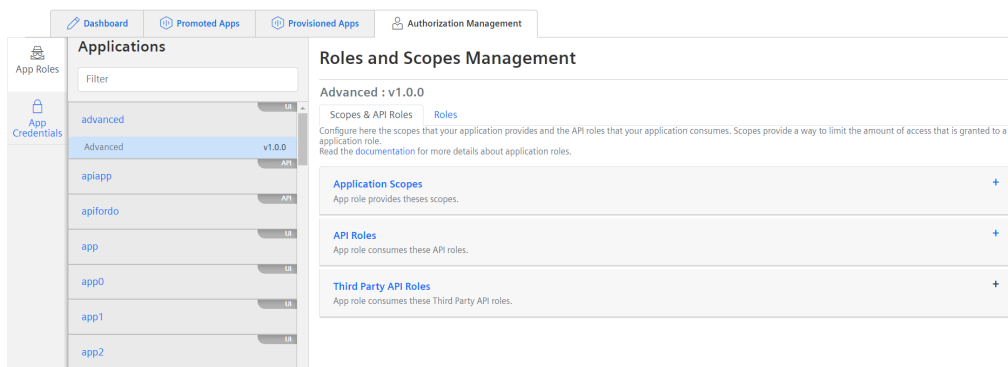


- By default, the application scope with "Admin" role is automatically assigned to a newly created application.
- Once you create another role and scope, you can delete the default role and scope.
- At least, one role and scope is required to access the application.

### User interface of "Roles and scopes Management"

**"Scopes and API Roles" screen:**

In "App Roles" screen, you can assign the required scopes and roles to the application specific roles, for a selected application.



## Parameters table

Parameter	Description
Application Scopes	You can add your application scopes under the "Application Scopes" section. By default, an admin role with a scope is created for an application. You can create the scopes, apart from the default scope.
API Roles	You can add your application scope to the specific API roles under the "API Roles" section.
Third Party API Roles	You can add the dependent third party API role and assign it to the application role under the "Third Party API Roles" section.
Provisioned API Scopes	You can add your application for mapping with the provisioned application scopes under the "Provisioned API Scopes" section.

## Procedure for assigning roles and scopes

To assign roles and scopes for an application, follow the steps:

1. In application overview page, click "Configure" after saving the application.
2. In Authorization Management, select "App Roles" tab.
3. In the "Applications" window, select your saved application from the list.
4. In "Scopes & API Roles" tab, click "Create scope" to enter the scope name and enable the check box for Admin or User or both, as per requirement and click "Save".



- When the mobile type application is created, it is automatically assigned with a default scope: ".\_access" . This default scope "<appname>.\_access" is for mobile application only. This scope cannot be modified or deleted by the user. If you add roles to your application, by default the scope is assigned to the created role.
- Creating at least one scope is mandatory to proceed for the registration of the application.
- Every application specific scope is automatically prefixed with the application name.
- Every role can be found in "Settings" in the following scheme: mdsp:<tenantname>:<application>.<role>.
- Avoid creating scopes which contains the scripting word like, "document.write". As these scopes will create an issue for IAM transactions.

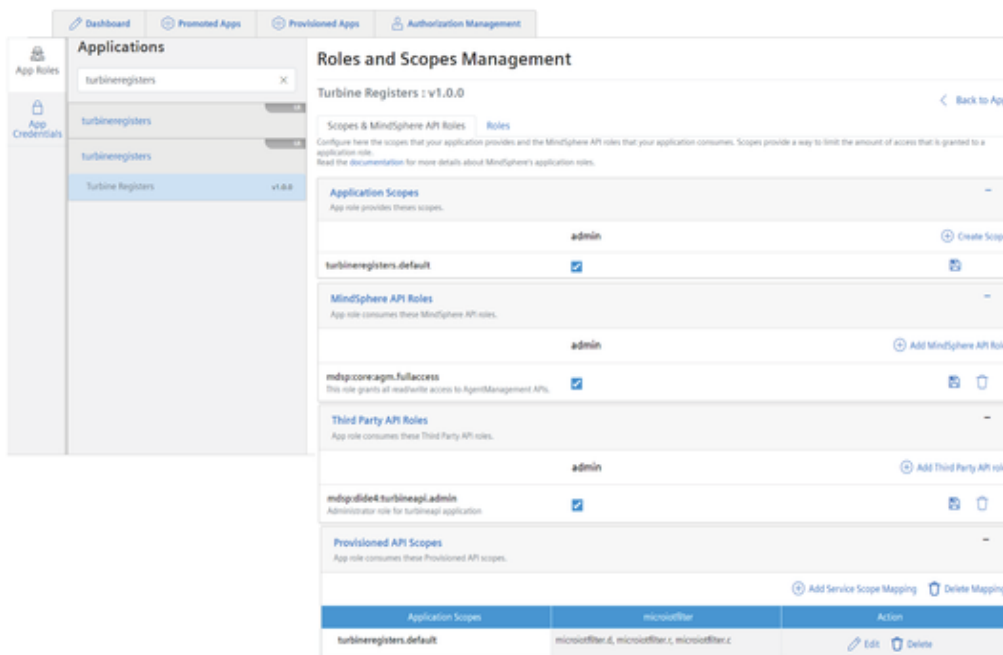
5.Click "Add API Role" and fill the application role name with description from the drop-down menu displayed in the dialog box and click "Save".

6.Click "Add Third Party API role" and fill the Third party API role name with description from the drop-down menu displayed in the dialog box and click "Save".

7.Click "Add Service Scope Mapping" and select an application scope to map the provisioned app scopes to your application and click "Add Mapping".

## Result

The application successfully added with roles and scopes.



Only standard applications with third-party API roles assigned can access API applications.

### Next step

You can access advanced token exchange to provision the APIs. For more information, see [Accessing Industrial IoT /Token APIs](#).

## 7.5 Create a new customized role for an application

By default, "Admin" role is created for an application. You can define new roles other than the provided roles.

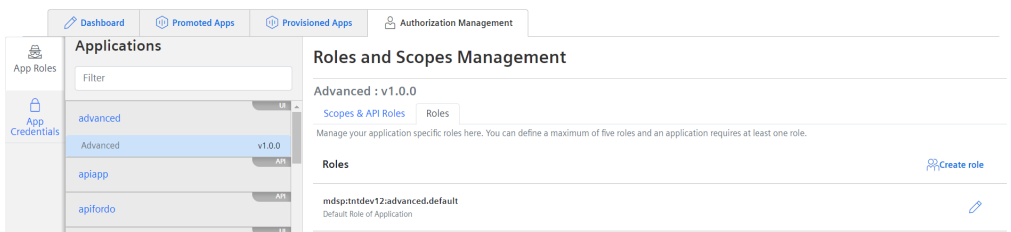
You can delete the default roles as per your requirements. But at least one role needs to be created. This is to ensure that you can access the application.

The cloned applications are assigned with the same set of custom roles, scopes and core roles are inherited from the previous version. For more information on the conditions for creating a custom role, see [Developer documentation](#).

All API (third party APIs and Provisioned APIs) applications can be accessible to standard, Insights Hub Monitor plugin and mobile applications only when third party API roles are provided.

### User interface



**"Customized Roles" screen:**

## Procedure

To create a new customized role for the application, follow these steps:

1. In "Authorization Management" tab, select "App Roles".
2. Select your application from the "Applications" window.
3. Select the "Roles" tab.
4. Click "Create role" to create customized role for your application.
5. Save the newly created customized application role.

## Result

The application successfully added with the customized roles.




For Mendix based application, the admin and user roles are created by default and user scope is added to the created application.

## Next step

- Assign roles to your user via "Settings", in order to access the application. For more information, see [Assign users to a role](#).

## 7.6 Manage API application with message broker

You can subscribe your application with Industrial IoT services by mapping the application scopes with "Message Broker". To subscribe your application with Industrial IoT services, register an endpoint of your application to get notifications from Insights Hub. For more information about "Message Broker", see [Message Broker - Developer Documentation](#).

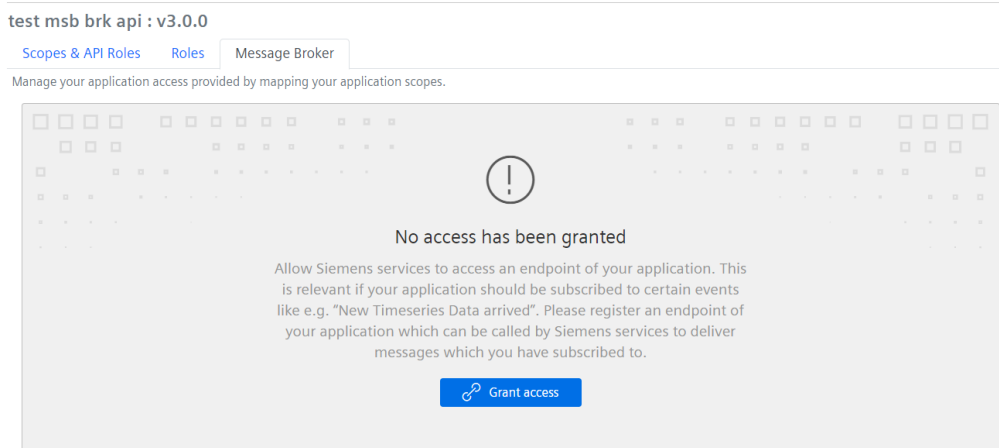
 "Message Broker" feature is applicable for API applications only.

## User Interface

### "Message Broker" screen:

You can grant Industrial IoT services to access your application by mapping the application scope.

#### Roles and Scopes Management



## Procedure

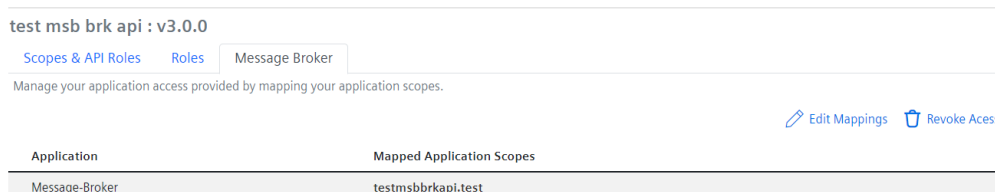
To subscribe your application with Industrial IoT services, follow these steps:

1. In "Authorization Management" tab, select "App Roles".
2. Select your application from the "Applications" window.
3. Select the "Message Broker" tab.
4. Click "Grant access" to map the application scopes.
5. Select the application scopes and click "Save".

## Result

The application is successfully subscribed to Industrial IoT services.

#### Roles and Scopes Management



## 7.7 Issue application credentials

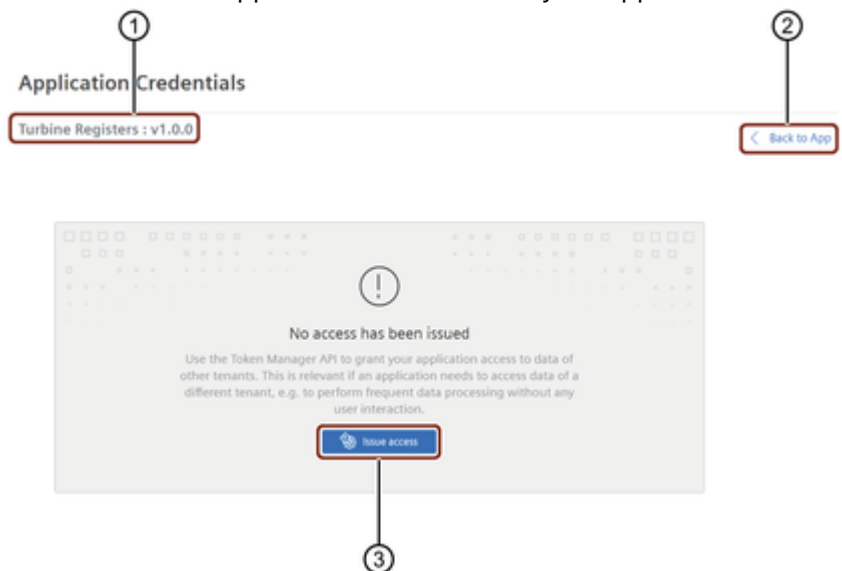
Application Credentials are used for background actions without any user interaction required or, when the data displayed in the UI should be independent from the rights of the App User (example, show all assets on the tenant independent if the App User has the right to access Asset Management itself).

After application credentials are issued to application, you can invoke by Token Manager Service and create a bearer token. For more information, see [Token Manager Service](#).

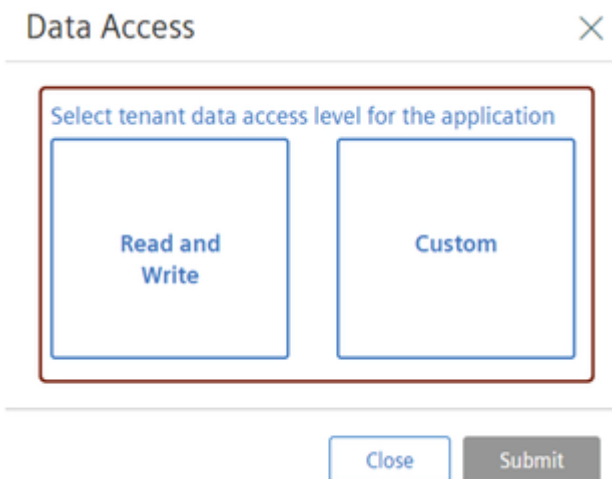
### User interface

"Application Credentials" screen:


You can issue the application credentials for your application.



- ① Application name and version
- ② Redirects back to the App Details page
- ③ Issues Application Credentials



Data Access	Description
Read and Write	In this access, you are allowed to perform all CRUD operations on APIs.
Custom	In this access, the customizable set of core roles can contain any kind of data access level. All APIs can be configured as per the user requirement.

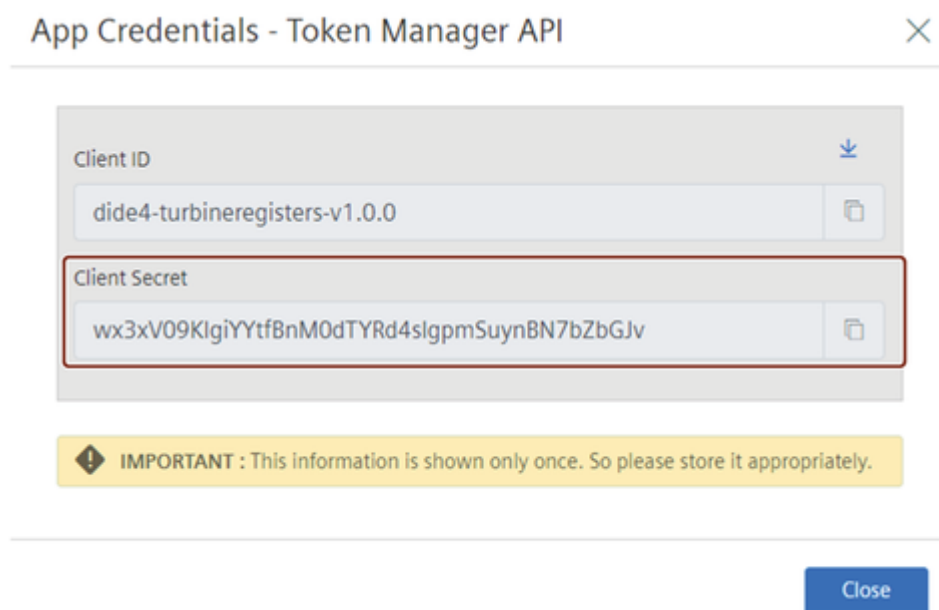


- For API application, application credentials are issued with "Read and Write" data access only.
- If app credentials are created for older version, then it is not possible to changed/remove in new version for API.

## Procedure to issue Application Credentials

To issue application credentials, follow these steps:

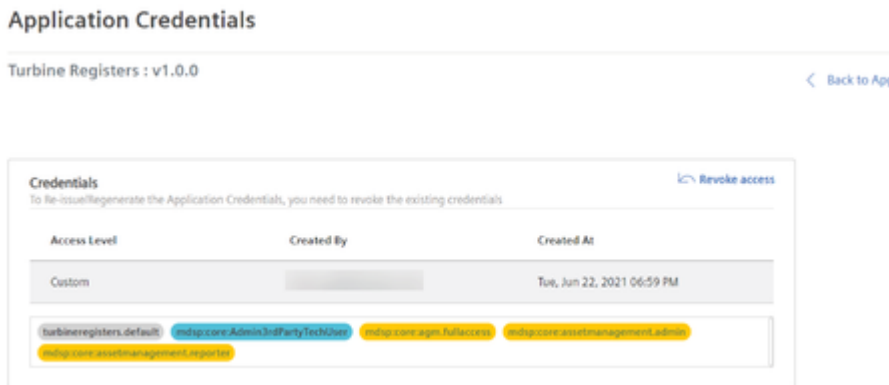
1. In Authorization Management tab, select the "App Credentials".
2. In the the "Applications" window, select your application.
3. Click "Issue Access".
4. In the "Data Access" pop-up, select the data access type and submit.
5. Click "Submit", a "Application Credentials - Token Manager API" dialog box appears with "Client ID" and "Client Secret".



It is important to keep a note of your application credentials since the credentials are only displayed once on the UI during the issuing process.

## Result

The application credentials for the selected application is now generated.



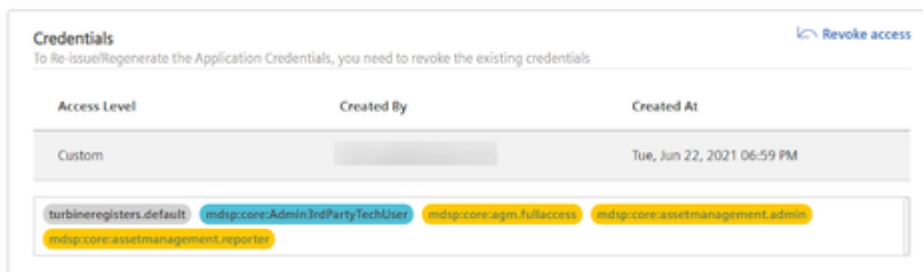
## 7.8 Revoke application credentials

If the user forgets the username and password of the application credentials of the working application, he can re-generate a secret key again by revoking the generated application credential.

### User interface

"Credentials" screen:

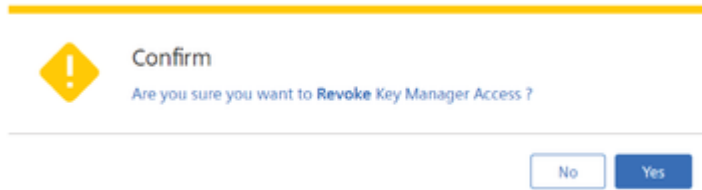
You can revoke the application credential access for your application.



## Procedure to revoke Application Credentials

To revoke the access, follow these steps:

1. In "Application Credentials", click "Revoke access" to revoke the access.
2. Click "Yes" to accept the confirmation.



## Result

The old app credentials will be invalid and you can issue application credentials again for this application.

## Next step

Once the application credentials are generated, the application should be authenticated and authorized to access the APIs. For more information, see ["Authentication & Authorization"](#).

## 7.9 View the registered Insights Hub Monitor applications in Insights Hub Monitor

If you choose to create an application with Insights Hub Monitor Plugin type, you are seamlessly integrating your functionality with the Insights Hub world. Your new functionality will be available in Insights Hub Monitor soon after its registration.

## Prerequisites

The following are the prerequisites to access the Insights Hub Monitor plugin application:

- The application should be registered in Developer Cockpit.
- You need to have tenant admin rights from Settings to view Settings app in your Launchpad.
- To assign the user specific roles, see [Roles](#) in Settings application.



- Insights Hub Monitor application should be available in the Launchpad.



## Procedure to access applications in Insights Hub Monitor

To access the applications in Insights Hub Monitor, follow these steps:

1. Open Insights Hub Monitor.
2. Navigate to `Explore / Assets` to view the Insights Hub Monitor plugin.

## Result

The registered plugin is available in Insights Hub Monitor.

Find more information in the developer documentation of the [Insights Hub Monitor Plugin SDK](#).

# Handing over the application

# 8

## 8.1 Application handover to an operator

### Application upload

You decide the application you want to make available for productive deployment by uploading it together with additional information to an intermediate Application Repository.

The transfer process cannot be withdrawn after the upload. You are notified via "Developer Cockpit" and via email once the transfer of your application has been successful and the application is available to run on the MindAccess Operator Plan Account.

### Preparation for handover

To prepare the handover of the application, follow these steps:

1. Prepare application upload.

This is usually done by the developers.

2. Approve application upload.

This is done by the Developer Admin.

3. Assign the successfully uploaded application to an operator.

4. Accepting the assigned application by an operator in Operator Cockpit.

5. Granting the accepted application to an operator in Developer Cockpit.





- The application must be uploaded as a single zipped archive and a "manifest.yml" file.
- You need to use a single Cloud Foundry "manifest.yml" that contains the configuration of all Cloud Foundry applications that are from your application.
- Use only zip-Archives and not any other compression algorithms for bundling your applications.
- For self hosted and no-infrastructure type applications, single zipped archive and "manifest.yml" are not required.
- For mobile type application, provide the required description. Endpoint URL is not applicable for this type of application.

## 8.2 Managing uploads of an application for Cloud Foundry

To make your app available for the productive use, it needs to be uploaded into the repository. The following chapter shows you how to upload the application.

### Prerequisites

- The upload of the application to the Application Repository for handover to an operator can only be started after your application has been registered. During application registration, additional verification steps are performed in the background, like:
- For Cloud Foundry applications, the zipped binaries and the manifest files of the application should be available. To know more about the binary and the manifest files, see [FAQs](#).
- The zipped file size for binary uploads is limited to 1GB for Region Europe 1.

### User interface

#### "Manage uploads of a Cloud Foundry application" screen

- ① Binary and Manifest files upload
- ② Tools menu

## Parameters table

Parameter	Description
Configure service plan	Configure all Backing Services that your application requires in order to run. The service name needs to match the exact names in the provided "manifest.yml". This additional information is needed as the "manifest.yml" does not contain the information about the type and size of the Backing Services.
Manage service plan uploads	If you close the metadata dialog box by entering the binaries and manifest files, you can configure the service plans later and submit thereafter. This has the configuration for all Cloud Foundry applications which are included in your application. Once the uploading is finished, click "Submit". The application changes its state to "Preparation in progress" state.
Binary	Browse and upload the binary file containing your application.
Manifest	Browse and upload the manifest file containing all necessary configurations for your application.
Service	Select the service from the drop-down menu.
Service Plan	Select the service plan from drop-down menu.

Parameter	Description
Service name	Enter a unique name for the service. This needs to be the same name as in the .yml file
Service List	Displays the list of added services. This will be shown to the operators after assignment of the app
Description	Short summary of the application.
Application credentials	Select the kind of data level access the application requires. These are the following data access level options: <ul style="list-style-type: none"> <li>- None - no action on application credentials</li> <li>- Limited - read and manage assets permissions are granted.</li> <li>- Read and write - both read and write permissions are granted.</li> <li>- Custom - Customizable set of core roles can be configured as per the user requirement.</li> </ul> For more information, see <a href="#">Issue application credentials</a> .

## Procedure

To manage the uploads of the application on Cloud Foundry infrastructure, follow these steps:

1. In Dashboard overview, open the application in the "In Development" state.
2. Click "Manage Uploads" to start the process of uploading the application.
3. In the "Upload" screen, browse and upload the zipped archive that contains the binaries for this application.
4. In the "Upload" screen, browse and upload "manifest" in ".yml" format. After hitting the "Upload" button, a sanity check of the Manifest is performed. It checks if the Manifest contains CF applications for every component that has been configured.
5. You now have two options:
  - You can continue configuring the service plans and submit.
  - While configuring plans, you can upload the binaries and manifest and can submit the uploads later. In this case, the application manages the uploads at the backend. The state of the application changes to "Preparation in progress".
6. Select a "Service" and "Service name" from the drop-down menu and configure your backing services and then click "Add service".
7. Enter the "Description".
8. Select the data access level for the application.
9. Click "Submit".



- Any mismatch in .yaml or entered configuration can lead to a rejection of the app.
- If your application utilizes Cloud Foundry User Provided Services like access to an AWS S3 bucket, you must add this information in the description.

## Result

The application changes its state to "Waiting for Approval".

The screenshot shows the 'Application Details' page for 'Ny adv Application'. The application is in the state 'Waiting for Approval'. The page includes the following information:

- Internal Name:** nyadvapplication
- Display Name:** Ny adv Application
- Type:** Standard
- Infrastructure:** Cloud Foundry Hosted
- Version:** v1.0.0
- Application URL:** https://tntdev12-nyadvapplication-tntdev12.eu1-int.mindsphere.io
- State:** Waiting for Approval
- Roles:** Configure

The 'Information' section states: 'The prepared release version needs to be approved by a user with the Developer Admin role.' The 'Description' section contains: 'My advanced config application'. The 'Endpoints' table is as follows:

Name*	Direct URL*	Endpoints
default	https:// www.domain-name.com	<a href="#">View Application Endpoints</a>

## Next step

After submitting the upload, a user with the Developer Admin role can approve the upload which then triggers the upload to the Operator Cockpit.

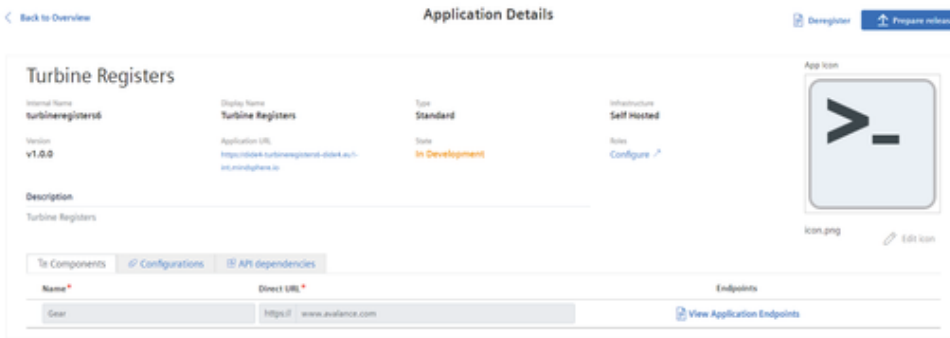
## 8.3 Preparation release processes for self hosted and without backend (mobile) type infrastructure

This is the first phase of the handover process of the registered for the self hosted and without (mobile) infrastructure.

## User interface

"Prepare release of a self hosted and without (mobile) infrastructure" screen:

8.3 Preparation release processes for self hosted and without backend [mobile] type infrastructure



"Prepare release" pop-up:

Prepare release

**Health Endpoint URL**  
   
Health URL consists of the Fully Qualified Domain Name and the Health endpoint, when accessed, it should respond with http status code - 200 (ok).

**Description**  
   
Please describe the purpose of the application, e.g. The application provides machine insights for...

**App Credentials**  
 Please select the data access level that your application requires:    
Your currently configured data access level has been preselected. No credentials are handed over. Only the required data access level is provided to the production system.

Parameters table

Parameter	Description
Health Endpoint URL	Health checking of the selected application URL.
Description	Short summary of the application will be shown to operator.
Application credentials	Select the kind of data level access the application requires. These are the following data access level options: - None - no action on application credentials - Limited - read and manage assets permissions are granted. - Read and write - both read and write permissions are granted. - Custom - Customizable set of core roles can be configured as per the user requirement. Note: For more information on issuing and revoking the application credentials, see <a href="#">Issue and revoke application credentials</a> .

Procedure

To prepare the release, follow these steps:

1. In Dashboard, select the self hosted application.
2. Click "Prepare release" to start the process of approving the application. Once you click "Prepare Release" the state of the application changes to "Preparation in progress".
3. Enter the endpoint URL for health check. The application URLs is available in the "Health Endpoint URL" drop-down menu which are auto-populated when the components are added in a new application.
4. Enter a short description of the application.
5. Select the data access level for the application.
6. Click "Submit".

## Result

The application changes its state to "Preparation in progress".

## Next step

After approval the process, a user with the Developer Admin role can approve the upload which then triggers the upload to the Application Repository. To approve your application and the application upload details, see [Approve uploads of an application](#).

## 8.4 Approve uploads of an application

Once a developer completes the release preparation for an application, a Developer Admin can approve the same. Approve upload confirms the application can be uploaded and scanned in the repository. Thereafter, the app can be assigned to operators for productive use.

## User interface

**"Approve Upload Details" screen:**

You can approve your application by verifying the application upload details as shown below:

**Application Upload Details**

Primary Email\*

Alternate Email\*

Description  
Mechanism of gear control

Service Plan  
Application has no service plans.

API Dependencies

No	API Name	Version
1	turbineapi	v1.0.0

Provisioned API Dependencies

Name	Version	Operator
microiofilter	v2	diop97

turbineregisters0.default mdsip.core.agm.fullaccess mdsip.core.assetmanagement.admin mdsip.core.assetmanagement.reporter mdsip.core.Admin.TndPartyTechUser

submitting, you confirm (i) having evaluated and tested the application as to its technology, functionality, performance, security, and user interface; and (ii) that your application complies with the documentation (in particular with regard to the DevOps Guide available at [www.mindsphere.io/terms](http://www.mindsphere.io/terms)) and any other requirements set out in the MindSphere Agreement or the applicable Order Form.

I Confirm.

Close Discard Approve

## Procedure

To approve the uploads of the application, follow these steps:

1. In "Dashboard" tab, search for the applications that are in the "Waiting for approval" state, and click an application.
2. Click "Approve Upload" to initiate the process of uploading the application.
3. In the "Application Upload Details", add an alternate e-mail address that will be used for contact if any problem occurs and as a contact for production usage.
4. To comply your application with [DevOps Guide](#) and other requirements set in the Master Agreement or applicable Order Form, then select "I Confirm".
5. Click "Approve".

After approving the upload, the application can no longer be changed. If you wish to continue working on this application, then you need to create a new version of the application.

The status of the upload can be seen on Developer Cockpit Dashboard.

## Result

The application state changes to "Ready for Assignment" successfully.

The screenshot displays the 'Application Details' interface for an application named 'Turbine Registers'. At the top, there are navigation links for 'Back to Overview', 'Archive', and 'Deregister'. The application details include:

- Internal Name:** turbineregisters5
- Display Name:** Turbine Registers
- Type:** Standard
- Infrastructure:** MindSphere Cloud Foundry Hosted
- Version:** v1.0.0
- Application URL:** https://bba4-turbineregisters-4bba4-uc1-00.mindsphere.io
- Status:** Ready for Assignment
- Action:** Configure

Below the details, there are tabs for 'Operator Assignment', 'Application Endpoints', 'API Dependencies', and 'Configurations'. The 'Operator Assignment' tab is active, showing a form to assign the application to an operator. The form has two input fields: 'Operator Name' (with a placeholder 'Enter Operator Name (Smart Name)') and 'Email' (with a placeholder 'operator@ mindsphere .com'). An 'Add Assignment' button is located below the form. A note states: 'Only Developer Administrators are allowed to assign applications to Operators'. To the right, the 'Operator Assignment List' is empty, displaying a message: 'No Assignment Available. Yet the application is not assigned to any operator.'

The internal validation takes few minutes or up to some hours depending on the app type and the binary size. Once the scanning has been finished successfully an email is being sent to the specified email addresses and the app will be visible in "Promoted Apps" tab.

## Next step

- After successful upload approval, you can assign the application to an operator. To assign the application, see [Assign an application to an Operator](#).
- You can deregister the application and continue working on a new or different version or revision. To create a new version or revision of the application, see [New version of an application](#).



# Assigning the application

## 9.1 Assign an application to an Operator

After successful upload of an application to the Operator Cockpit, it is then possible to assign the application to an operator account.

The assignment requires 3-way handshake confirmation of the developer and the operator.

### Prerequisites

- The operator account name should be in the lower case, like "operatorname".
- Provide one contact e-mail address of the operator account for notifications.
- User should have "Developer Admin" role to assign the application to the operator.

### User interface

"Assign an application to operator" screen:

The screenshot displays the 'Application Details' page for 'cf ui appli'. The application is in a 'Ready for Assignment' state. Below the details, there is a form to 'Assign application to Operator' with fields for 'Operator Name' and 'Email', and an 'Add Assignment' button. An 'Operator Assignment List' table shows a successful assignment to operator 'dltp432'.

Operator Name	Assignment Status	Action
dltp432	GRANTED	No Action Required

### Procedure

To assign an application to an Operator, follow these steps:

1. In "Promoted Apps" tab, search for the application in "Ready for Assignment" state and select your application.

2. Enter the name of the operator in lower case in the "Operator Name" field.  
Assigning of apps is only possible within the same region.
3. Enter the contact e-mail address.
4. Click on "Add Assignment".
5. Wait for the operator to acknowledge the assignment. The application has to go through the assignment stages and the threefold handshake to get published in Operator Cockpit.
6. After the acceptance by operator and the 3-way-handshake, click "Grant".

## States of actions

Once an operator is assigned to an application, the added operator name and its assignment status are visible in the list of the assigned operators. You will also be able to see the action status of each of the operators.

The operator assignment has the following stages:

1. Assigned: This is the first stage of the assignment process. The application is assigned to an operator. This assigned application is then visible in Operator Cockpit Inbox.
2. Acknowledged: The admin of the Operator Cockpit has to accept the application. The assignment status changes to "Acknowledged". Thereafter, a "Grant" button is displayed under the "Action" heading alongside the "Acknowledged" status in Developer Cockpit application screen.
3. Granted: After clicking the "Grant". The status changes to "Granted".

Once an assignment has been completed, you cannot undo the assignment process. Therefore, a multi-step workflow is recommended in order to prevent accidental assignments. For more information, see [Operator Cockpit](#).



Applications cannot be deleted after the status changes to "Ready for Assignment".

## Result

The application is successfully "Assigned" to an Operator.

## 9.2 Updating new configuration of the assigned application

After successfully assigning the application to Operator Cockpit, it is possible to update the configuration of the application.



To update the configuration of the assigned application, you need to deregister the application in Developer Cockpit.

## Prerequisites

- The new configuration or content-security-policy header information to update in Developer Cockpit.

## User interface

### "Configuration" tab:

The screenshot shows the 'Application Details' page for 'cf ui appli'. The 'Configuration' tab is active, displaying a table of configurations and an 'Operator Assignment List'.

Key	Value
content-security-policy	default-src 'self' static.eu1-int.mindsphere.io; style-src * 'unsafe-inline'
cache-control	no-cache, no-store, max-age=0, must-revalidate

Operator Name	Assignment Status	Action
dltsp432	GRANTED Application assignment granted successfully	No Action Required

## Procedure

To update the configuration of the assigned application, follow these steps:

1. Click "Deregister" the assigned application.
2. In "Configuration" tab, "content-security-policy" field will be enabled.
3. Click "Edit" to enter the new configuration or content-security-policy header information.



You can update "pluginSettings" for Insights Hub Monitor plugin type of applications.

4. Click "Save" to update the configuration in Developer Cockpit.
5. Click "Register".
6. Click "Apply All Operators" to update the configured application for all the assigned operators.

## Result

The application is successfully configured with the new configuration.

## 9.3 Manage promoted apps

Promoted applications are ready to be assigned to an Operator or can be managed under the "Promoted app" tab.

You can archive the application from the "Promoted Apps" tab, before or after it is assigned or granted to an operator. The state of the application will change to "Archived". The application can be un-archived or deleted from the archived state. If the application is un-archived, you can view the application in the "Promoted Apps" tab with the state "Ready for assignment".

## User interface

"Archived application" screen:

The screenshot displays the 'Application Details' page for an application named 'mapp'. The application is in an 'Archived' state. The interface includes a navigation bar with 'Back to Overview', 'Delete', 'Un-Archive', and 'Register' options. The main content area shows the application icon, name, and various attributes: Internal Name (abrahamapp), Display Name (mapp), Type (Standard), and Infrastructure (Self Hosted). Below this, there is an 'Information' section stating 'No actions available for archived applications. Un-Archive application to continue.' At the bottom, there are two tabs: 'Operator Assignment' and 'Operator Assignment List'. The 'Operator Assignment' tab contains a form with fields for 'Operator Name' and 'Email', both of which are disabled. A yellow banner at the bottom of the form reads 'Action not available for archived applications.' The 'Operator Assignment List' tab shows a table with one entry: Operator Name 'dltsp432', Assignment Status 'GRANTED' (indicated by a green box), and Action 'No Action Required'.

## Procedure

To archive, unarchive and delete the applications from the Promoted Apps tab, follow these steps:

1. In "Promoted Apps" tab, search for the application to archive and click on the application.
2. Use the archive-switch to "Archive" or "Unarchive" the applications as needed.

When the promoted apps are archived, it will be de-registered automatically. These applications can be deleted after archiving them, if they are not assigned or granted to an operator. The registered API application cannot be archived unless dependent UI application is de-registered and archived.

## Result

The application is successfully archived.

In "Promoted Apps" tab, click "Show Only Archived apps" toggle button to view the archived application.

## Next step

- After unarchiving the promoted application, the application status changes from "Archived" to "Ready for Assignment". Now, you can assign this application to an Operator. For more information, see [Assign an application to an Operator](#).
- You can deregister the application and continue working on a new or different version. For more information, see [New version of an application](#).

## 10.1 FAQs

### How to test your new application on your own developer tenant?

Once you have created a new application, follow the steps below to make your application available for usage.

- Register the application. For more information on registering applications, see [Register an application](#).

#### Types of application and their availability

- Standard: On new browser for testing.
- Insights Hub Monitor Plugin: In Insights Hub Monitor
- Mobile: Only via app on own mobile device
- API: Only usable with a depending app

### What are the possibilities for the application to be "FAILED" state?

An application might face blocker or may not be able to proceed the states if:

- A self hosted application does not pass the health status check.
- A sanity check takes place while uploading metadata (binaries and manifest files) for a Cloud Foundry application. You will not be able to proceed with the uploads if the metadata uploaded are incorrect. For more information, refer the questions below.
- If metadata uploaded does not match with Operator Cockpit validation. The values configured in upload window must match manifest.yml file configuration. Define the services in the manifest file if service plans are selected from the drop-down menu given in the next section. If you do not select any service, avoid defining services in the manifest file.

### What are the guidelines for zipping the files?

#### Rules to zip and upload binary files:

During the uploading of an application, consider the following factors:

- The zipping of the binary file should be as the below shown example structure.
- The zipped folder should not be empty.
- Except for the corresponding files of the application, there should not be any other additional/ duplicate files in the zipped folder.

**Example scenario to upload zip files:**

Let us consider a zipped file "appbinary.zip".

To upload a zipped file, it should have the following zipping structure:

```
appbinary.zip
|--appbinary
|
|   |--apicomps
|   |   |--apicompl.jar
|   |   |--apicompl2.war
|   |--uicomps
|   |   |--uicompl/dist/
|   |   |--uicompl2/dist/
```

The manifest file has the following structure with "buildpack":

## What are the guidelines to create and upload the manifest files?

**Rules to create and upload a manifest file:**

For uploading a manifest file, the following factors should be considered:

- The "name" parameter in "manifest.yml" should match the name entered in the "Components" section.
- The "path" parameter in "manifest.yml" should point to the correct file/ files.
- The "path" parameter in "manifest.yml" file should be as per the folder structure.
- Do not use the following parameters in the manifest file as they are deprecated:
  - domain
  - domains
  - host
  - hosts
  - no-hostname
  - Use of 'buildpacks' attribute in manifest is not fully supported. Hence use of 'buildpack' is favored. The feature will be updated in future. Refer to [Cloud Foundry Dev Guide](#) for manifest deprecations.
- It is recommended to use random-route: true to avoid path conflicts.
- If a user wants to specify routes for an application, the user has to use routes attribute.



If "buildpack" and "memory" are not specified in the manifest file, auto-deployment option is disabled in Operator Cockpit. "Buildpack" works for both auto & manual deployment.

**Example scenario to upload manifest files:**

The manifest file has the following structure with "buildpack":

①

```

---
applications:
- name: apicompl
  memory: 1024M
  buildpack: java_buildpack
  instances: 1
  path: apicomps/apicompl.jar
  random-route: true
- name: apicomp2
  memory: 1024M
  buildpack: java_buildpack
  instances: 1
  path: apicomps/apicomp2.jar
  random-route: true
- name: uicomp1
  memory: 1024M
  buildpack: staticfile_buildpack
  instances: 1
  path: uicomps/uicomp1/dist/
  random-route: true
- name: uicomp2
  memory: 1024M
  buildpack: staticfile_buildpack
  instances: 1
  path: uicomps/uicomp2/dist/
  random-route: true

```

②

```

---
applications:
- name: apicompl
  memory: 1024M
  buildpack: java_buildpack
  instances: 1
  path: apicomps/apicompl.jar
  random-route: false
  routes:
  - route: apicompl-route1.apps.eul.mindsphere.io
  - route: apicompl-route2.apps.eul.mindsphere.io/testpath
- name: apicomp2
  memory: 1024M
  buildpack: java_buildpack
  instances: 1
  path: apicomps/apicomp2.jar
  random-route: false
  routes:
  - route: apicomp2-route1.apps.eul.mindsphere.io
  - route: apicomp2-route2.apps.eul.mindsphere.io/testpath
- name: uicomp1
  memory: 1024M
  buildpack: staticfile_buildpack
  instances: 1
  path: uicomps/uicomp1/dist/
  random-route: false
  routes:
  - route: uicomp1-route1.apps.eul.mindsphere.io
  - route: uicomp1-route2.apps.eul.mindsphere.io/testpath
- name: uicomp2
  memory: 1024M
  buildpack: staticfile_buildpack
  instances: 1
  path: uicomps/uicomp2/dist/
  random-route: false
  routes:
  - route: uicomp2-route1.apps.eul.mindsphere.io
  - route: uicomp2-route2.apps.eul.mindsphere.io/testpath

```

- ① Manifest with random-route: true and path corresponding to zipping structure
- ② Manifest with random-route: false and with specified routes and path corresponding to zipping structure

## How to use Cloud Foundry hosts in Cloud Foundry manifest file?

It is required to use <appname>- <tenantname> for Cloud Foundry hosts in Cloud Foundry manifest files. You can also use <tenantname> or random routes in Cloud Foundry manifest files.

## What are the throttling limits for the Developer Cockpit?

Developer Cockpit has some technical limitations. The following table provides the information of the technical service limits:



Service	Limit per (request / tenant / 10 minutes)
Maximum limit to create new application	20
Maximum limit to fetch application's metadata for the tenant	200
Maximum limit to update an application	100
Maximum limit to "Register / De-register" an application	20
Maximum limit to clone the application to create new version	10
Maximum limit to get details of specific application	100
Maximum limit to delete an application	20
Maximum limit to get "Roles and Scopes" for an application	100



Technical service limits can be upgraded as per the requirements. For more information, see [MindAccess DevOpsPlan ProductSheet](#).

## Where can I host the self-hosted applications?

The self-hosted applications are hosted on any infrastructure outside of Cloud Foundry. You can add the application hosted URL in the components field. The component URLs cannot end with "mindsphere.io".