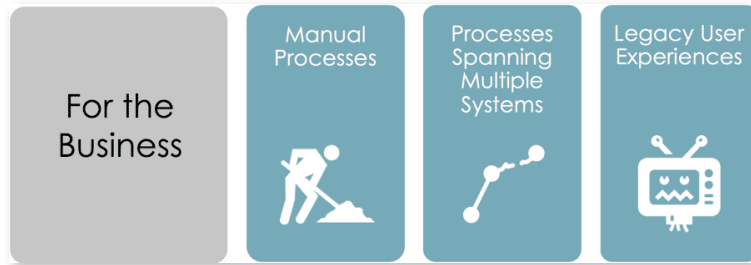




# Common challenges

---

Common application development challenges for the business



Business process owners today face many common problems. There are often so many manual processes across the enterprise. To get information or make a simple request can be a major undertaking. Many of these processes span multiple disconnected systems, data and organizational silos, and geographic regions.

Users today are also so much more demanding in terms of the experiences they expect. In their consumer lives, those experiences are increasingly flexible and effortless with the Amazon and Uber-like options they rely on.

In this lab, the focus will be on learning to build and deploy App Engine apps.

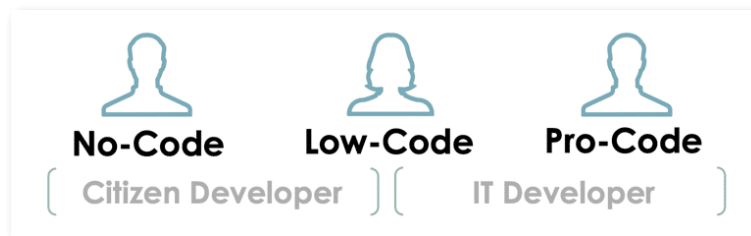
# Solutions

---

## ServiceNow no-code and low-code solutions

ServiceNow has the best tools and capabilities for you to build applications and keep step with the fast pace of your business. ServiceNow leads the way in great no-code and low-code solutions!

ServiceNow allows developers of all skill levels to share the same application within the same environment. For example, a citizen developer can quickly create the initial application, add some basic code, and then pass it to an IT developer to provide advanced capabilities without losing any work.

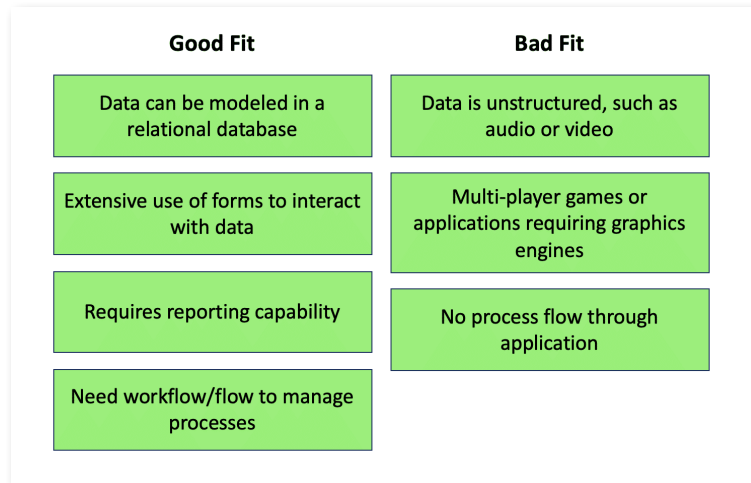


Imagine the benefits when all types of developers are working together based on their strengths. Business process owners, or citizen developers, can start the application development project to target their exact needs on their time frame. If needed, an IT developer can leverage the initial application and apply their skills with more advanced capabilities, like expanded integrations or a more comprehensive user experience.

## Should you build it?

Before you start planning and building an app remember that just because you can build an app does not mean you should build it. The main purpose of a business application is to solve a business problem or meet an improvement opportunity.

Most enterprise decision makers do not realize how much of day-to-day business relies upon custom applications, and many of those applications only partially support business needs or user expectations. Build effective applications that increase responsiveness, productivity, and user satisfaction.



No one understands your business process better than you.  
Make it better and deliver transformation faster with ServiceNow.

You are in the driver seat to build the future of work!

# Lab Structure

---

## Implementation

During this phase, we establish the bedrock by setting up and configuring the App Engine. These activities are conducted as one-time essentials, laying the foundation for an enriched development experience.

## Simulation

Step into the shoes of various App Engine personas to gain comprehensive insights into how this dynamic tool is leveraged by different roles. This immersive experience enhances your understanding of the platform's versatility.

Embark on this transformative journey as we reshape enterprise app development, foster synergy, and empower innovation.


Your commitment to innovation begins here.

# Tips and Tricks

---

## Remember The Copy Button

Look for the button to the right when you see a text box like below . Click the button to copy the value in the box.



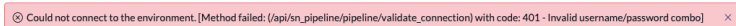
Text to copy

## Username and Passwords

In some of the sections, you will be copying and pasting user credentials.

You MUST use the values provided as that is the password that is already set for that account.

If you do not use the provided username and password, you will get the following error:



⊗ Could not connect to the environment. [Method failed: (/api/sn\_pipeline/pipeline/validate\_connection) with code: 401 - Invalid username/password combo] ✕

## Scope: Global

All activities will be performed in the Global scope unless otherwise specified.

# Exercise: Create Credential Alias

---

Estimated Time to Complete: 5 minutes

## Overview

A [Credential Alias](#) is a username and password used by ServiceNow to log into a remote system.

In this case, it will be used by the Controller instance (Prod) to access other ServiceNow Environments such as Dev, Test, QA, etc.

You will create the Credential Alias with a user account that has already been created for you.

Before Exercise

After Exercise

## Instructions

### CAUTION

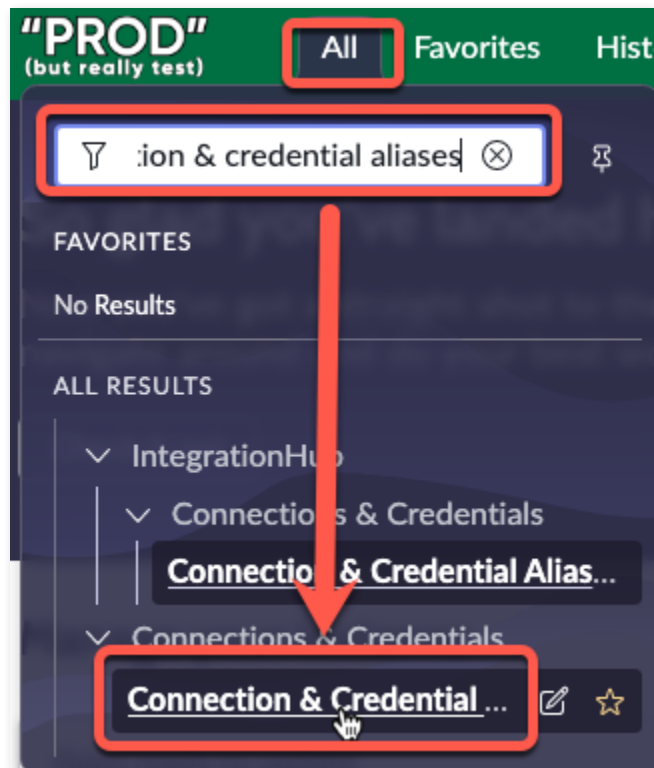
For this exercise, the specific Application Scope is not critical - Global is acceptable. Always check with your Platform Administrator for any company policies around [Connection and Credential Aliases](#).

### DANGER

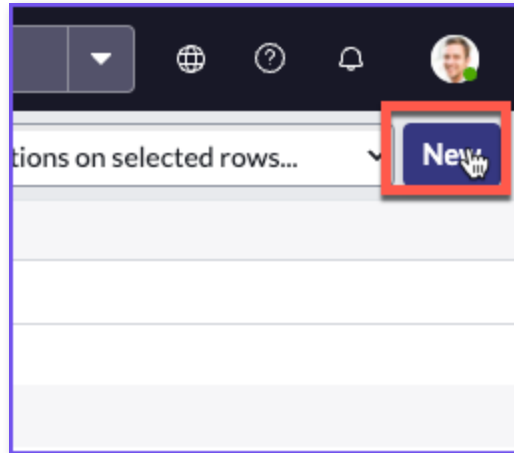
Complete this section in PROD.

1. Log in to PROD.

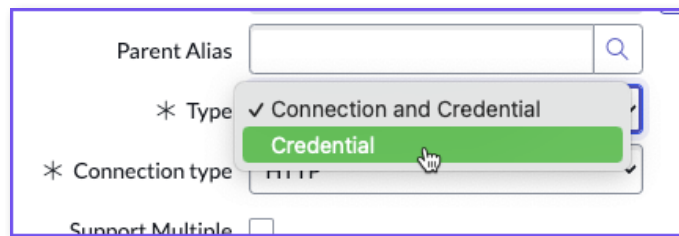
2. Navigate to All >> type Connection & Credential Aliases >> click either Connection & Credential Aliases.



3. Click New in the top-right to create a new Credential Alias record.



4. Set Type to Credential.



5. Set Name to 'Pipeline Credentials' and click Submit.

Name
Pipeline Credentials



\* Name

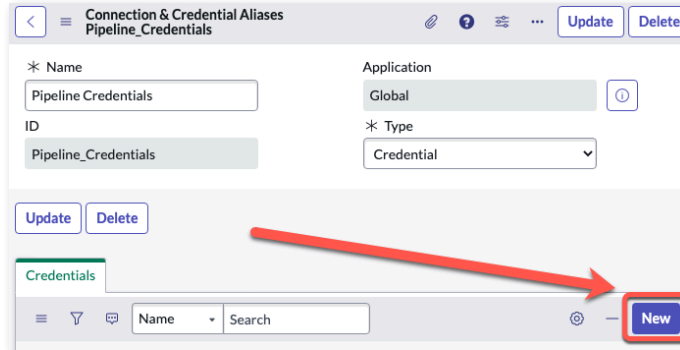
ID

Submit

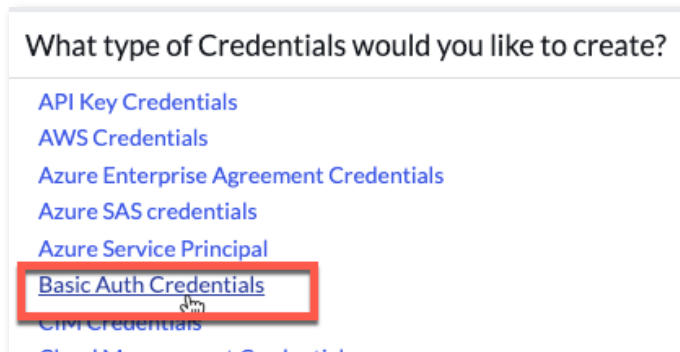
6. Click Pipeline Credentials to open the record.

All	
<input type="checkbox"/>	Name ▲
	<a href="#">CICD</a>
	<a href="#">Docker</a>
<input type="checkbox"/>	<a href="#">Pipeline Credentials</a>
	<a href="#">ServiceNow</a>

7. Navigate to the Credentials Related List and click New to add a credential.



8. Select Basic Auth Credentials.



**!** INFO

You will now input credentials for AEMC to log into a ServiceNow instance. The password for this remote user account is pre-configured. Errors will occur if you do not use the specified password provided below.

9. On the Basic Auth Credentials form, fill in the following values and click Submit.

**!** INFO

Remember to use the copy button to copy the text below.

Name

Pipeline Service Account

Username

svc\_pipeline

Password

ILoveHyperAutomationTimes500!

Name	<input type="text" value="Pipeline Service Account"/>
User name	<input type="text" value="svc_pipeline"/>
Password	<input type="password" value="....."/>

**i** NOTE

The dots in the password field will be shorter after you save the password. This is normal

## Lessons Learned

By completing this exercise, you have successfully achieved the following milestones:

- Gained experience in creating and managing Connection & Credential Aliases in ServiceNow.
- Familiarized yourself with Basic Authentication Credentials setup.

# Exercise: Create Environments

---

Estimated Time to Complete: 10 minutes

## Overview

**Environments** ↗ contain information about a ServiceNow instance and how to access it.

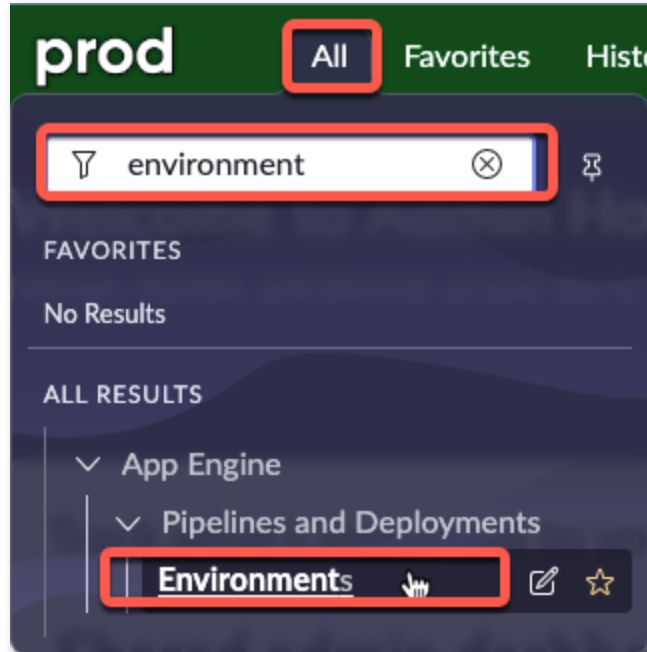
Next, you will create an Environment record for both "Production" and "Development" on the Controller instance where AEMC is installed

Before Exercise

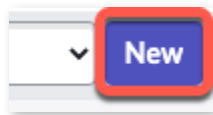
After Exercise

## Instructions

1. Log in to Prod.
2. Click All >> type environment >> click Environments




3. Click New in the top-right.

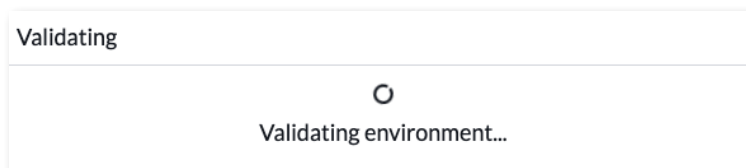
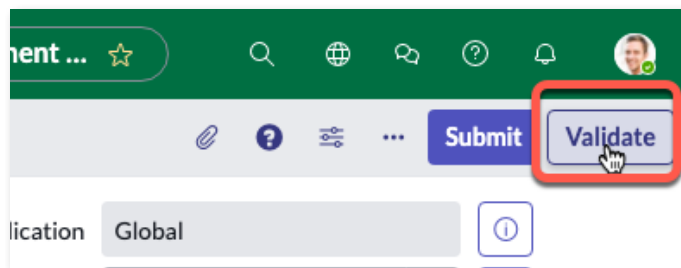


4. Complete the form as below.

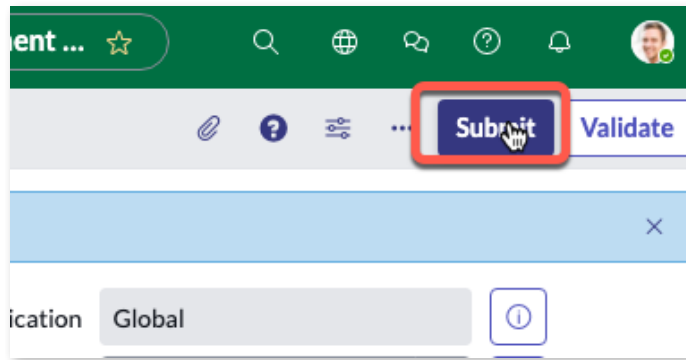
Field	Value
Name	Dev

Field	Value
Instance Type	Development
Dev Instance URL	
Instance credential	Pipeline_Credentials
Is Controller?	UNCHECKED
Instance Id	This will auto populate after clicking Validate.

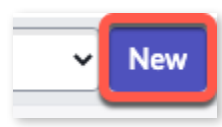
5. Click Validate



6. Click **Submit** to finish creating the environment record.




7. Click **New** in the top right.



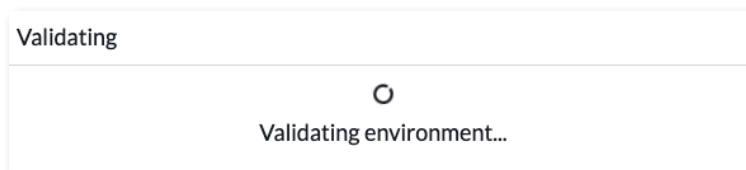
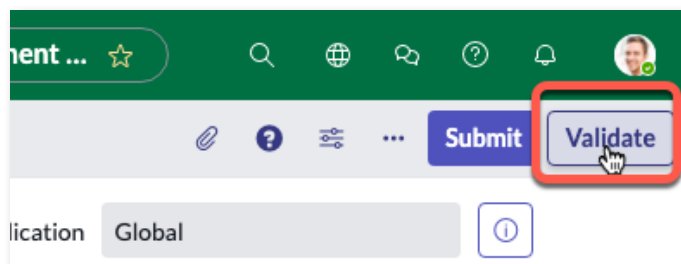
8. Complete the form as below.

Field	Value
Name	Prod
Instance Type	Production

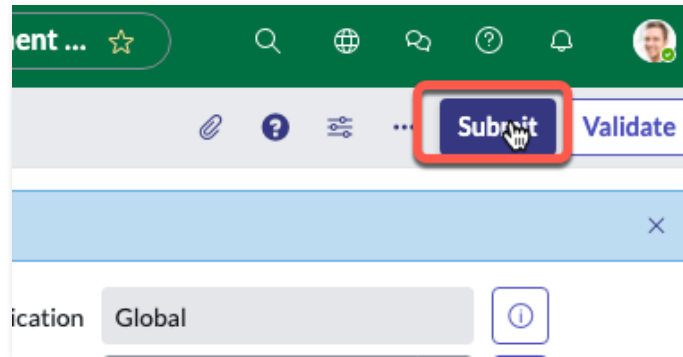


Field	Value
Prod Instance URL	
Instance credential	Pipeline_Credentials
Is Controller?	CHECK THE BOX
Instance Id	This will auto populate after clicking Validate.

9. Click the "Validate" button.



10. Click **Submit** to finish creating the environment record.



## Lessons Learned

In this exercise, you have:

- Understood the function and importance of Environment records in the ServiceNow platform.
- Acquired the knowledge of validating these Environment records.
- Successfully configured Environment records for both Development and Production instances in your Prod environment.

# Exercise: Configure Pipeline in Prod

---

Estimated Time to Complete: 15 minutes

## Overview

[Pipelines](#) ↗ establish the path an application takes from Development to Production.

This facilitates administrators' ability to efficiently transfer applications across different instances in the App Engine Management Center.

The process involves setting up and configuring your pipeline, specifying the [Environments](#) ↗ to include, and determining their position in the pipeline.

The Pipeline is exclusive to the Controller Environment, which in this case is the Production instance.

Before Exercise

After Exercise

Example of a Dev->Test->Prod Pipeline

### ⓘ INFO

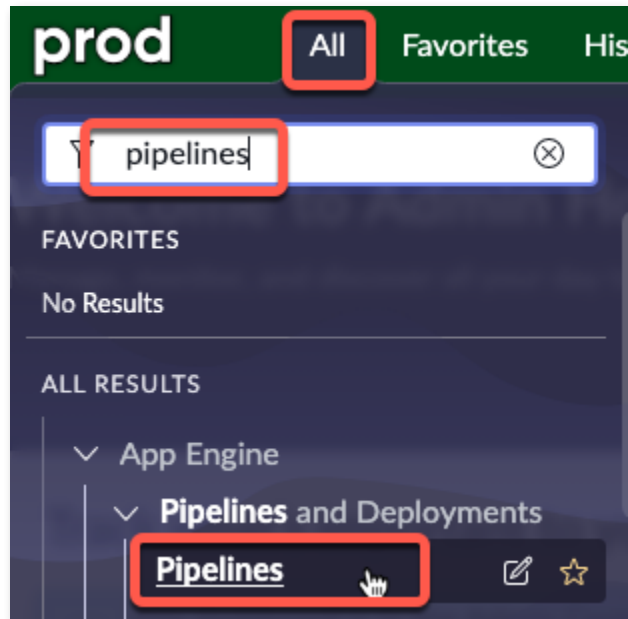
In this Lab, you will only configure Dev and Prod in the Pipeline.  
Pipeline records are only configured in Prod.

## Instructions

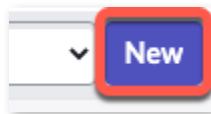
### ! INFO

Complete this exercise in the Global scope.

1. Click All >> type pipelines >> click Pipelines.



2. Initiate a new pipeline: Click New in the top-right corner.



3. Fill out the form as follows:

#	Field	Value
1	Name	Main Pipeline
2	Pipeline Type	Application Deployment
3	Source Environment	Dev
4	Active	Checked

Pipeline New record

1 Name: Main Pipeline

2 Pipeline Type: Application Deploymer

3 Source Environment: Dev

Application: Global

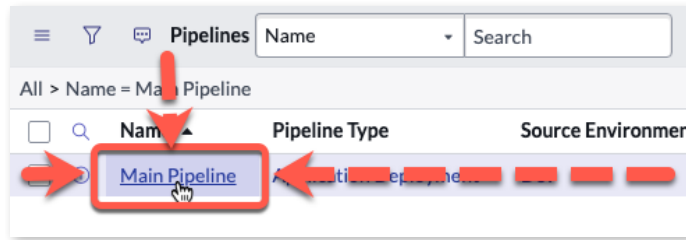
Active:  4

5 Submit

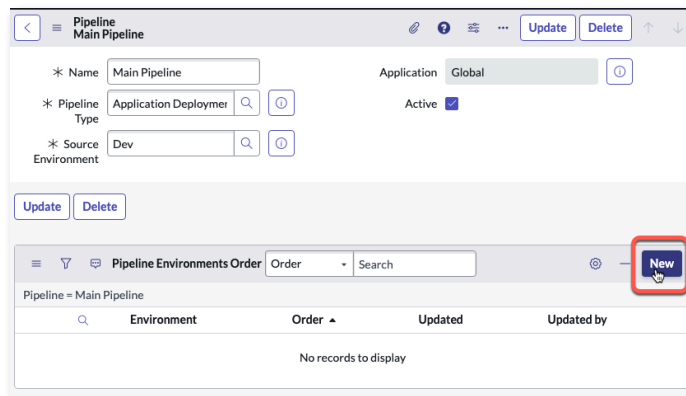
4. Submit the form.

Submit

5. Open the newly created Main Pipeline record.



6. Click New in the Pipeline Environments Order related list.



7. Complete the new form as shown below and click Submit when done.

#	Field	Value
1	Pipeline	Main Pipeline
2	Environment	Prod

#	Field	Value
3	Order	100

The screenshot shows a form titled "Pipeline Environment Order" with a "New record" header. It contains three input fields: "Pipeline" with the value "Main Pipeline", "Environment" with the value "Prod", and "Order" with the value "100". A "Submit" button is highlighted with a red box and a green circle with the number 4. The fields are numbered 1, 2, and 3 respectively.

8. Click Submit.

- Your completed Pipeline record will look like this:

The screenshot shows a table view of the completed Pipeline record. The table has columns for Environment, Order, Updated, and Updated by. The data row shows Environment: Prod, Order: 100, Updated: 2023-07-31 13:04:16, and Updated by: admin. The Environment and Order cells are highlighted with a green box.

Environment	Order	Updated	Updated by
Prod	100	2023-07-31 13:04:16	admin

 DANGER

Make sure your Pipeline configuration matches the screenshot above.  
DO NOT add Dev to the Pipeline Environments Order related list.  
It is the Source Environment and does not get added to the additional environments list.

## Lessons Learned

Through this exercise, we grasped the importance of correctly configuring the Pipeline in the Production instance.



# Exercise: User Provisioning

---

Estimated Time to Complete: 5 minutes

## ! INFO

THIS SECTION IS OPTIONAL AND CAN SAFELY BE SKIPPED IF YOU ARE BEHIND. Skipping steps below will have not prevent you from completing the lab.

## Overview

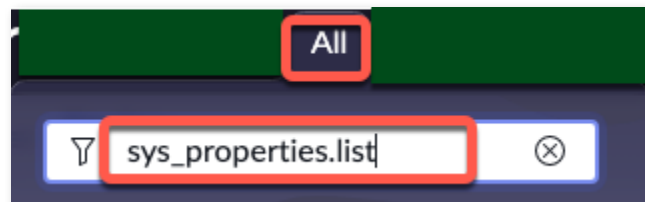
The system property `sn_app_intake.instance_can_provision_user` determines if the current instance can support the automated provisioning of users in the Development environment.

If you prefer to manually add users to groups in Dev, then you would set this property to `false`.

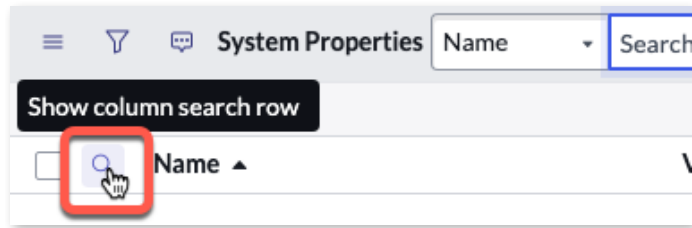
## Instructions

1. Click All >> type `sys_properties.list` >> press ENTER.

```
sys_properties.list
```

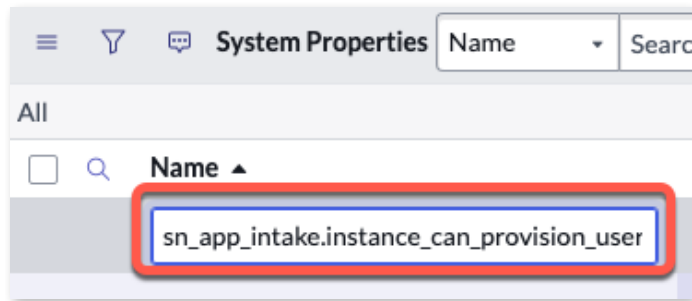


2. Click the magnifying glass to the left of the Name column.

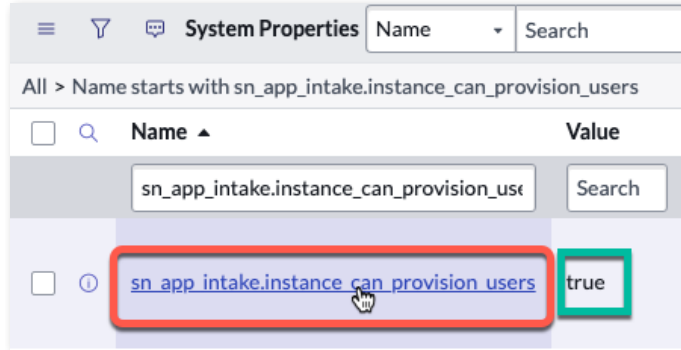


3. Type the name of the system property into the field under the Name column and press ENTER.

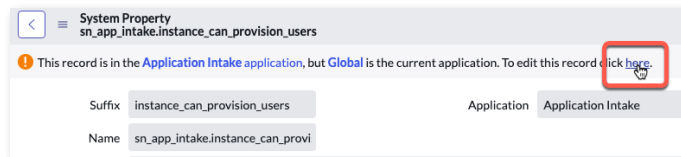
```
sn_app_intake.instance_can_provision_users
```



4. Click the name of the system property to open it. It should already be set to true for you.



5. Click the word **here** to switch from Global to the application scope Application Intake.



6. Change the Value field from 'false' to 'true'.

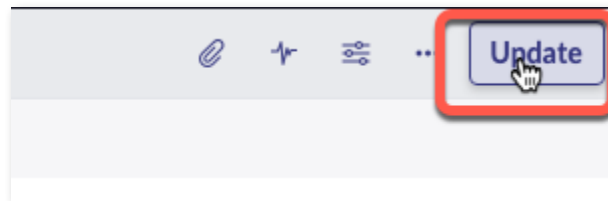
**CAUTION**

**REMEMBER:** This value was already set to 'true' for you as part of the automated lab setup.

The screenshot shows the 'System Property' configuration page for 'sn\_app\_intake.instance\_can\_provision\_users'. The page includes a warning message: 'You are editing a record in the Application Intake application (cancel)'. The configuration fields are as follows:

* Suffix	instance_can_provision_users
Name	sn_app_intake.instance_can_provi
Description	True/False property to determine if the current instance can this instance will be provisioned to another instance
Choices	
Type	true   false
Value	true

7. Click the Update button to save your changes.



## Lessons Learned

This exercise highlighted the importance of understanding and manipulating system properties within ServiceNow.

1. **System Properties:** The task illustrated the role of the system property `sn_app_intake.instance_can_provision_user` in managing automated user

---

provisioning.

- 
2. **User Provisioning:** We understood how enabling or disabling automated user provisioning could impact the process of adding users to groups in the Development environment. Depending on the unique needs and preferences of an organization, the option to manually add users might be chosen over automation.

# Exercise: App Intake - Catalog Item

---

Estimated Time to Complete: 10 minutes

## ! INFO

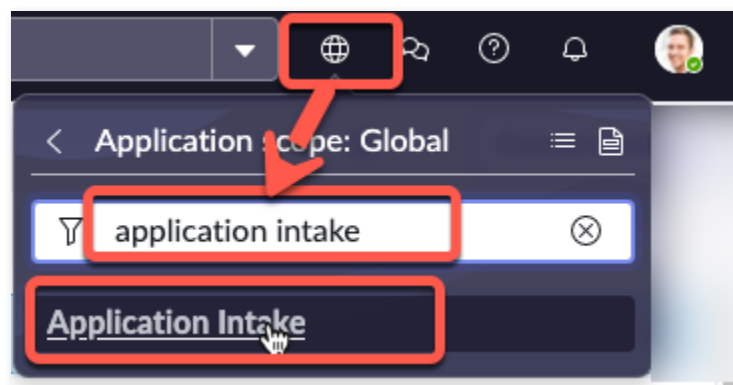
THIS SECTION IS OPTIONAL AND CAN SAFELY BE SKIPPED IF YOU ARE BEHIND. Skipping steps below will have not prevent you from completing the lab.

## Overview

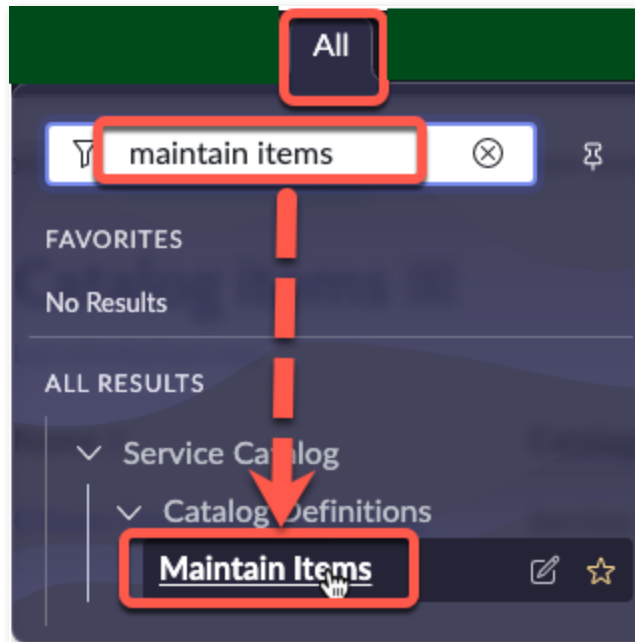
Create a custom app intake experience for your organization by editing the fields and questions on the App Engine Studio (AES) App Intake form in Catalog Builder.

## Instructions

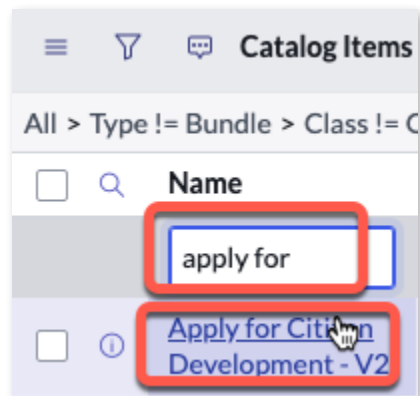
1. Click the Globe in the top right of the page, type "application intake", and click Application Intake.



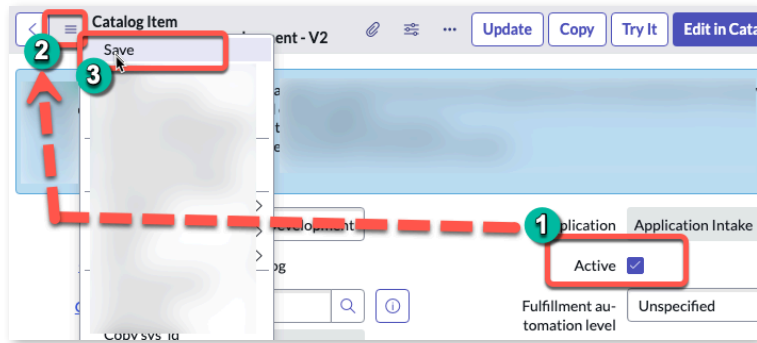
2. Click All, type maintain items, and then click Maintain Items.



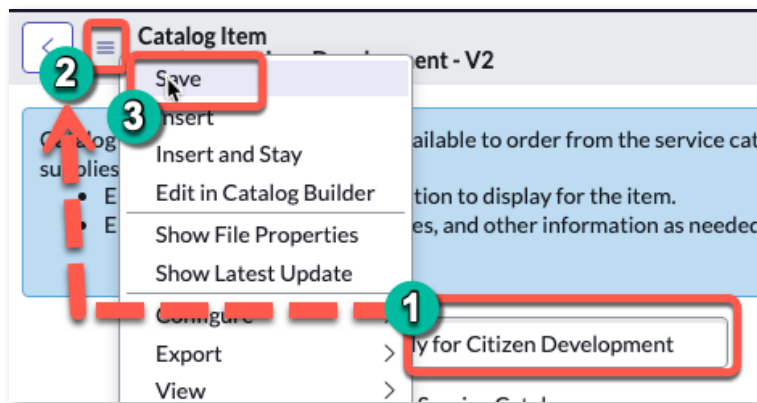
3. In the Name column, type "apply for" and click Apply for Citizen Development - V2.



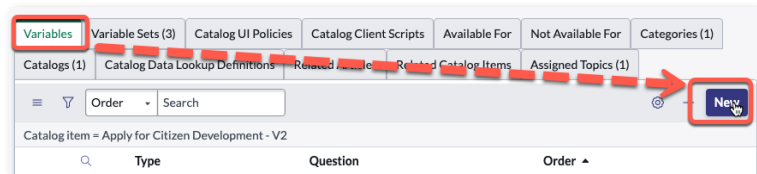
4. Check the Active box, click the hamburger button, then click Save.



5. Remove "- V2" from the Item name field, click the hamburger button, then click Save.



6. Scroll to the bottom of the page, click the Variables tab, and click New.



7. Set the Order field to '150'.



Order	150
-------	-----

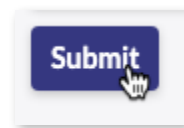
8. Type any question and press TAB. (Example: "Is this part of the lab?")

* Question	Is this part of the lab?
------------	--------------------------

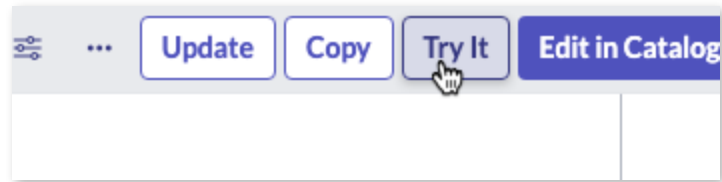
9. After you press TAB, the system should automatically populate the Name field.

* Name	is_this_part_of_the_lab
--------	-------------------------

10. Click Submit.



11. Click Try it to preview the Item.



You should see your new question on the catalog item form.

A screenshot of a catalog item form. At the top, there is a breadcrumb trail: '< Service Catalog > Can We Help You? > Apply for Citizen Development'. Below this is a header section with the text 'This is the first step to empower you to develop and drive the development of your' and an illustration of people working on a large screen. To the right of the illustration, it says 'Fill in the below form with specific details to create the business value you need.' Below the header is a form field labeled '\* Application Name'. Below that, a new question 'Is this part of the lab?' is added, and its corresponding input field is highlighted with a red border.

## Lessons Learned

In this process, we've learned how to customize the App Engine Studio (AES) App Intake form for a tailored experience. By creating and adding our variables, we can modify the form to ask pertinent questions specific to our organizational requirements. Additionally, we've learned the importance of properly ordering these variables to ensure a smooth intake process.

### ! INFO

For more information see

[Product Docs: Customize the App Intake form in Catalog Builder](#) ↗

# Exercise: App Engine Admins

---

Estimated Time to Complete: 5 minutes

## ⓘ INFO

THIS SECTION IS OPTIONAL AND CAN SAFELY BE SKIPPED IF YOU ARE BEHIND.  
Skipping steps below will have not prevent you from completing the lab.

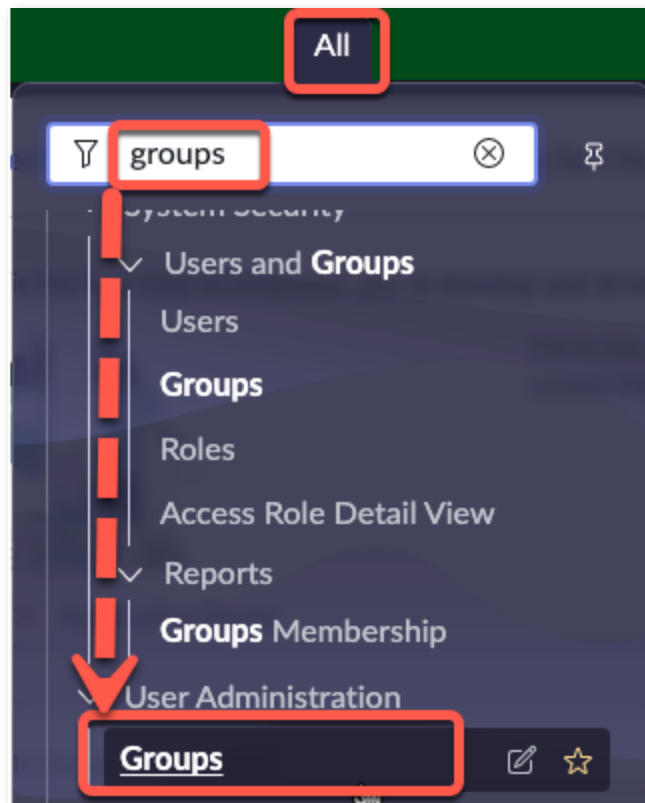
## Overview

Add users to the App Engine Admin group in your production instance to give them administrative rights to App Engine Studio (AES).

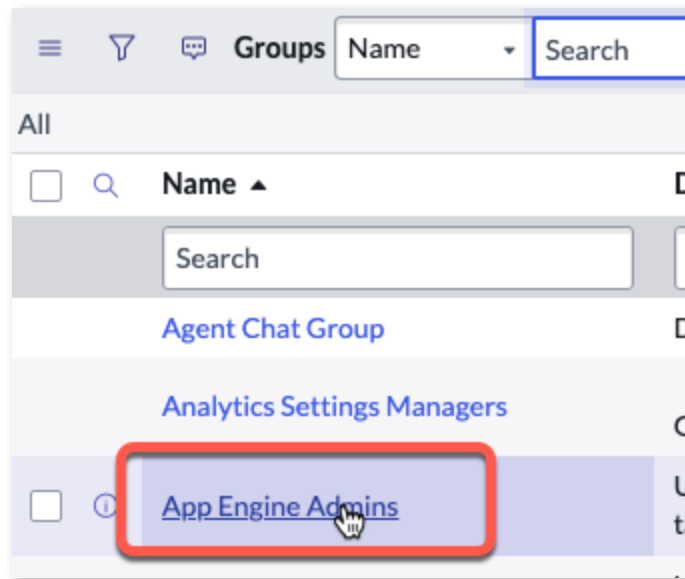
## Instructions

---

1. Click **All**, type **groups**, and then click **Groups** in the **User Administration** section.



2. Open the App Engine Admins group.



3. Select the Group Members related list and select Edit.



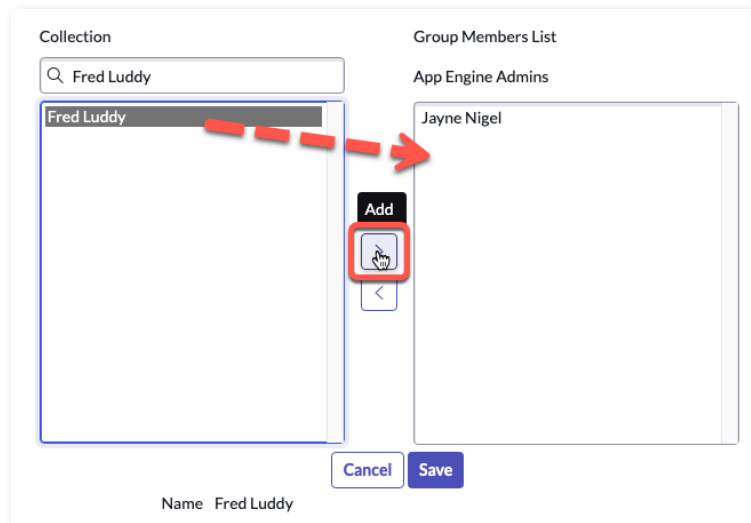
4. Type "Fred Luddy" into the Collection box.



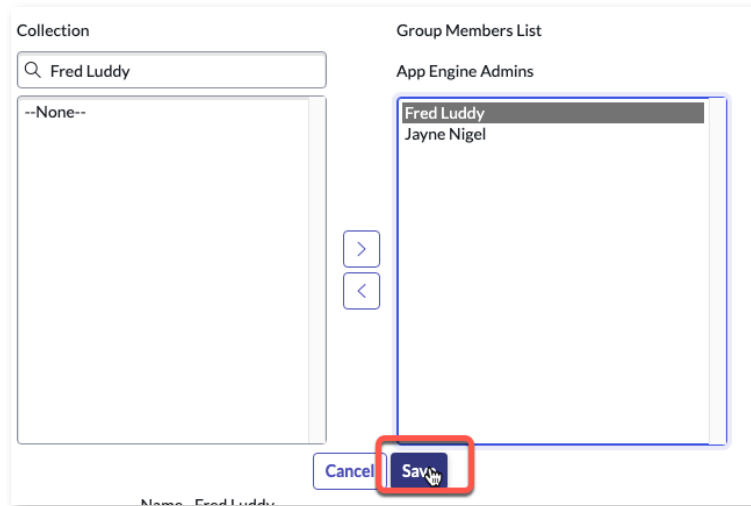
5. Click Fred's name.



6. Click the Add button to move Fred to the right side.



7. Click Save.



## Lessons Learned

You have added users to the **App Engine Admin** group.

They can review app- and deployment-related requests for approval or rejection.

### ! INFO

For more information see

[Product Docs: Add users to the App Engine Admin group](#) ↗

# Prod Complete

---

That completes the Prod section of the Implementation.

Next, you will configure the Dev instance as part of Implementation.



# Exercise: Create Credential Alias

---

Estimated Time to Complete: 10 minutes

## Overview

A [Credential Alias](#) ↗ is a username and password used by ServiceNow to log into a remote system.

In this case, it will be used by App Engine Studio to access the Controller Instance.

You will create the Credential Alias with a user account that has already been created for you.

Before Exercise

After Exercise

## Instructions

### CAUTION

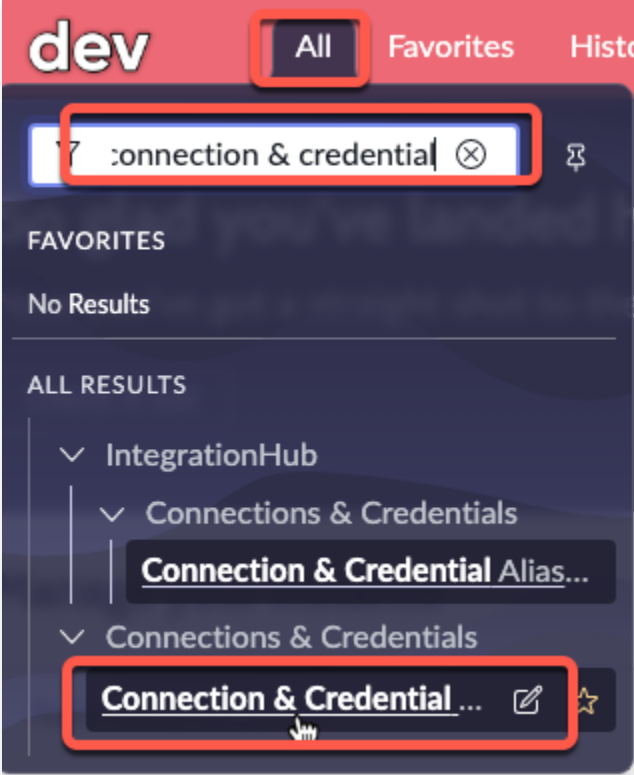
For this exercise, the specific Application Scope is not critical - Global is acceptable. Always check with your Platform Administrator for any company policies around [Connection and Credential Aliases](#) ↗ .

### DANGER

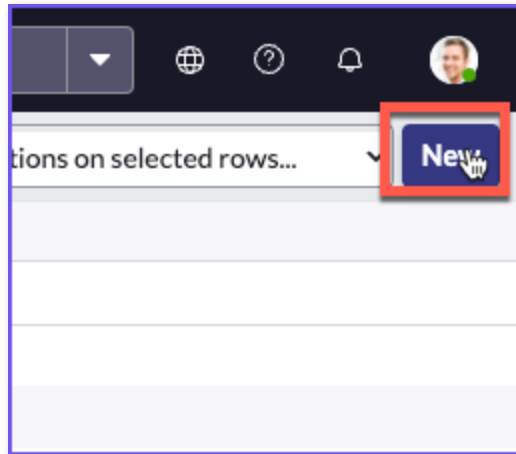
Complete this section in DEV.

1. Log in to DEV.

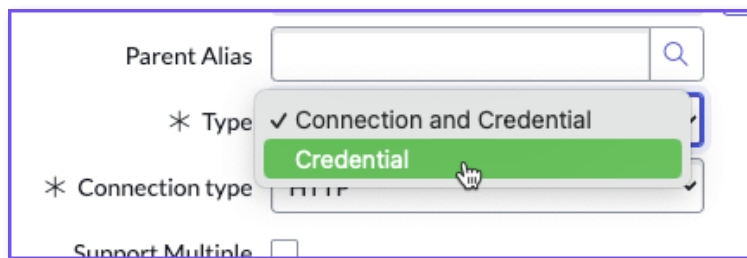
2. Navigate to All >> type Connection & Credential Aliases >> click either Connection & Credential Aliases.



3. Click New in the top-right to create a new Credential Alias record.



4. Set the Type to Credential.



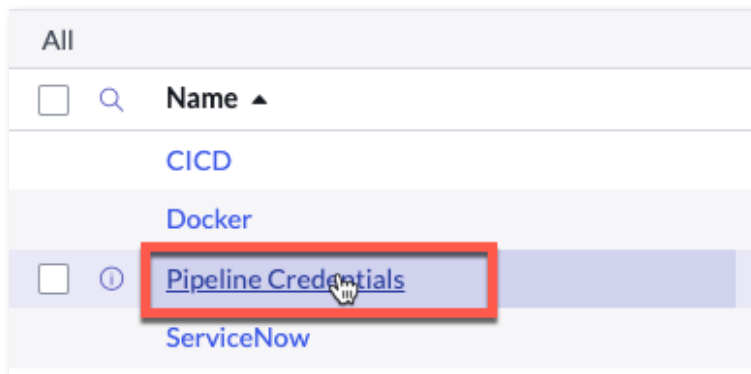
5. Set the Name to 'Pipeline Credentials' and click Submit.

Pipeline Credentials

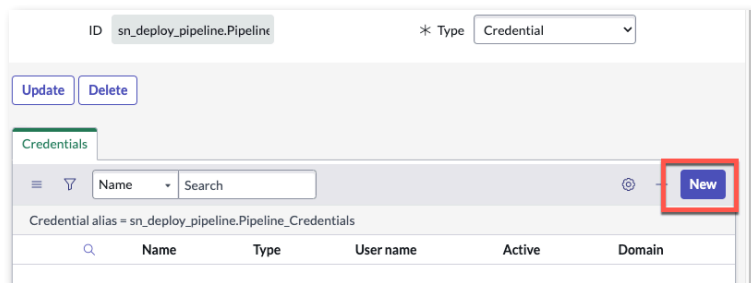
A screenshot of a form. The form has two fields: "\* Name" with a text input field containing the text "Pipeline Credentials", and "ID" with a greyed-out text input field.

Submit

- Click Pipeline Credentials to open the record.



- Navigate to the Credentials Related List and click New to add a credential.



- Select Basic Auth Credentials.

### What type of Credentials would you like to create?

- API Key Credentials
- AWS Credentials
- Azure Enterprise Agreement Credentials
- Azure SAS credentials
- Azure Service Principal
- Basic Auth Credentials**
- CI/CD Credentials

#### CAUTION

You will now input credentials for AES to log into the Controller instance. The password for this remote user account is pre-configured. Errors will occur if you do not use the specified password provided below.

9. On the Basic Auth Credentials form, fill in the following values and click Submit.

#### INFO

Remember to use the copy button to copy the text below.

Name

Pipeline Service Account

Username

svc\_pipeline

Password

ILoveHyperAutomationTimes500!

Name	Pipeline Service Account
User name	svc_pipeline
Password	.....

**i** NOTE

The dots in the password field will be shorter after you save the password. This is normal

## Lessons Learned

In completing this exercise, you have successfully achieved the following milestones:

- Gained experience in creating and managing Connection & Credential Aliases in ServiceNow.
- Familiarized yourself with Basic Authentication Credentials setup.

# Exercise: Create Controller Environment

---

Estimated Time to Complete: 10 minutes

## Overview

Next up, you will add a "Controller" Environment record to your Dev instance.

It will contain the details for Dev to access the Controller Environment which is the Prod instance.

Production is considered the "Controller" Environment as that is where App Engine Management Center is installed and orchestrates activities from.

Before Exercise

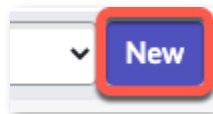
After Exercise

## Instructions

1. Log in DEV.
2. Click All >> type environment >> click Environments




3. Click New in the top-right.



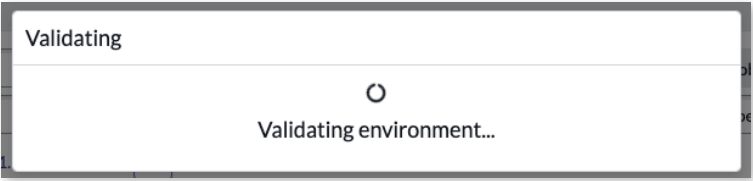
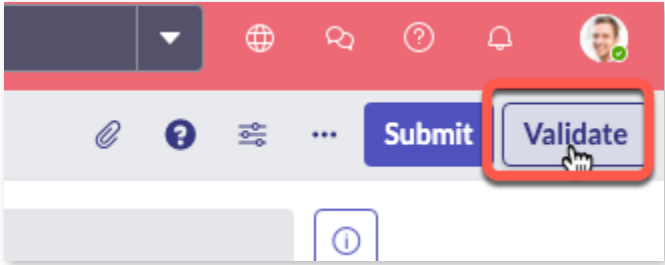
4. Complete the form using the information below.

Field	Value
Name	AEMC Controller
Instance Type	Production

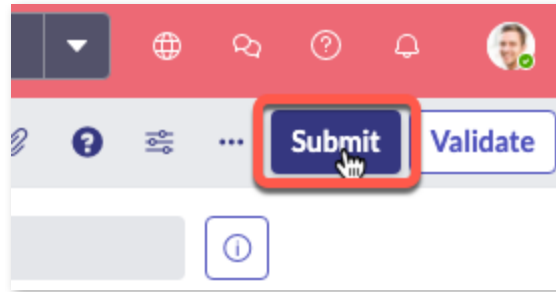


Field	Value
Prod Instance URL	
Instance credential	Pipeline_Credentials
Is Controller?	CHECK THE BOX
Instance Id	This will auto populate after clicking Validate.

5. Click Validate.



6. Click Submit to finish creating the Prod environment record.



## Lessons Learned

### Importance of the Controller Environment

Setting up the Controller Environment is crucial for the operation of the App Engine Management Center. This exercise emphasizes the vital role of the Production instance, as it's where AEMC is installed and from where it orchestrates activities.


# Exercise: Collaboration Descriptors

---

Estimated Time to Complete: 10 minutes

## Overview

People with admin privileges can manage the permissions of developers collaborating on an application.

You can use the predefined [Collaboration Descriptors](#)  that are standard with activation, or create a custom collaboration descriptor.

By using a collaboration descriptor, you can assign, manage, and monitor delegated development permissions.

We will use collaboration descriptors during the Simulation exercises.

Descriptors	Description
Owner	Owner of the application that manages other collaborators and can delete the application.
Editor	Standard descriptor to invite collaborators.
Integration Specialist	Non-standard (custom) descriptor created by the user.

### INFO

Integration Specialist is a custom [Collaboration Descriptor](#)  created for this lab.

## Instructions

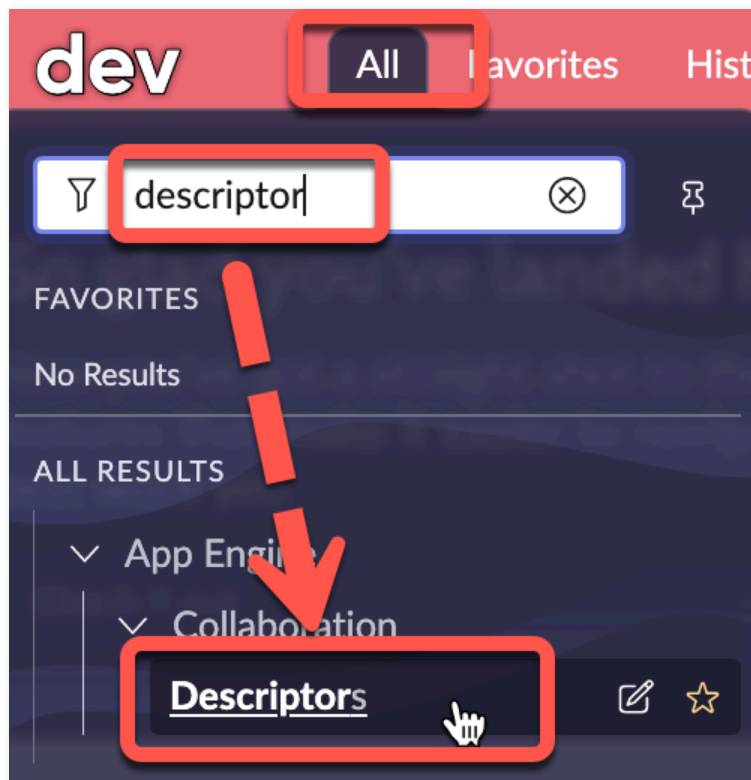
### ⚠ CAUTION

For this exercise, the specific Application Scope is critical - Collaboration Descriptors go in Global.

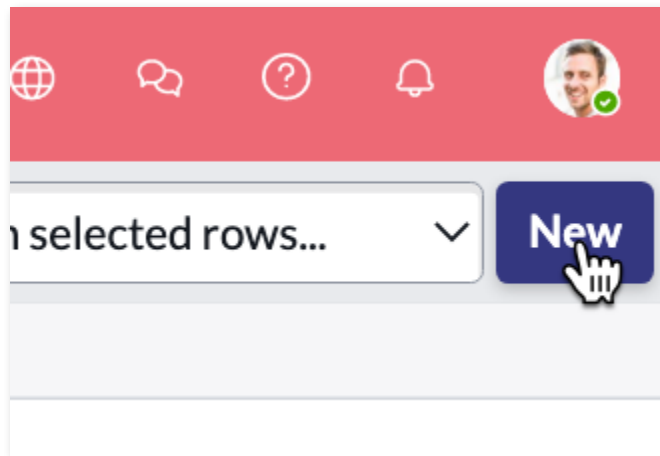
### 🔥 DANGER

Complete this section in DEV.

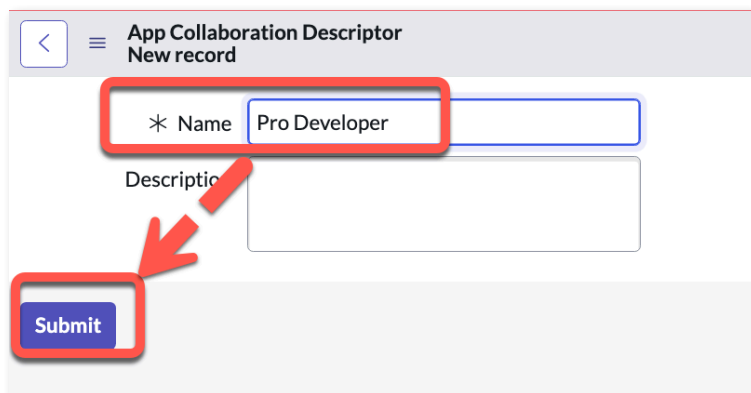
1. Log in DEV.
2. Click All >> type descriptor >> click Descriptors.



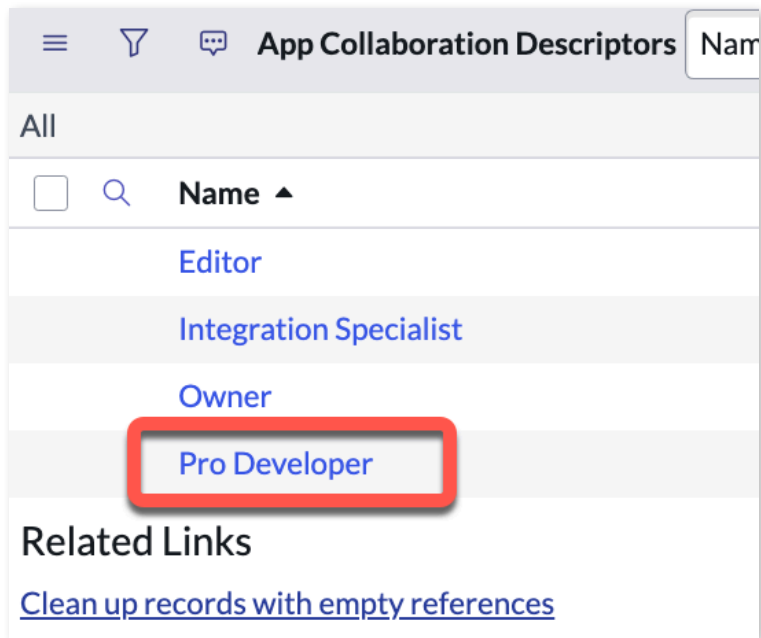
3. Click New.



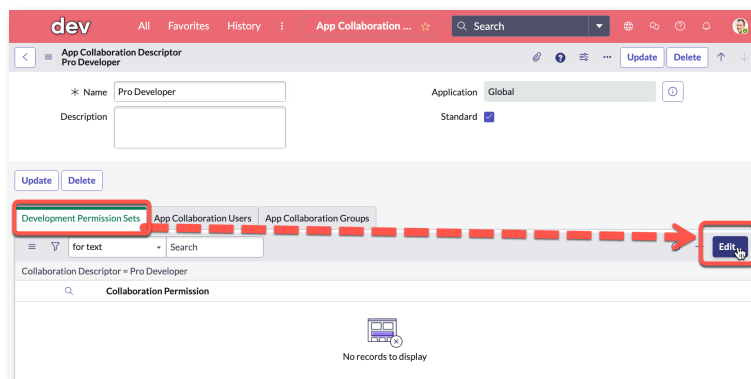
4. In the Name field, type 'Pro Developer', and click Submit



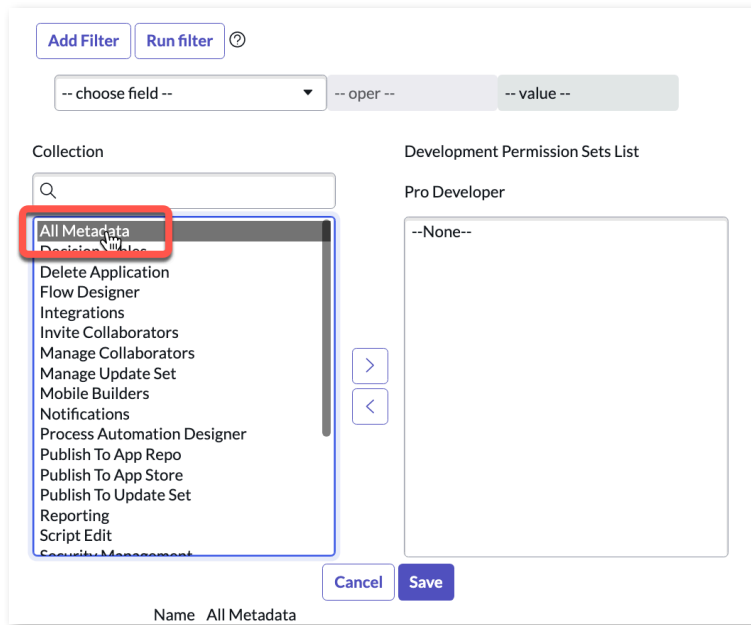
5. Click the Pro Developer record to open it.



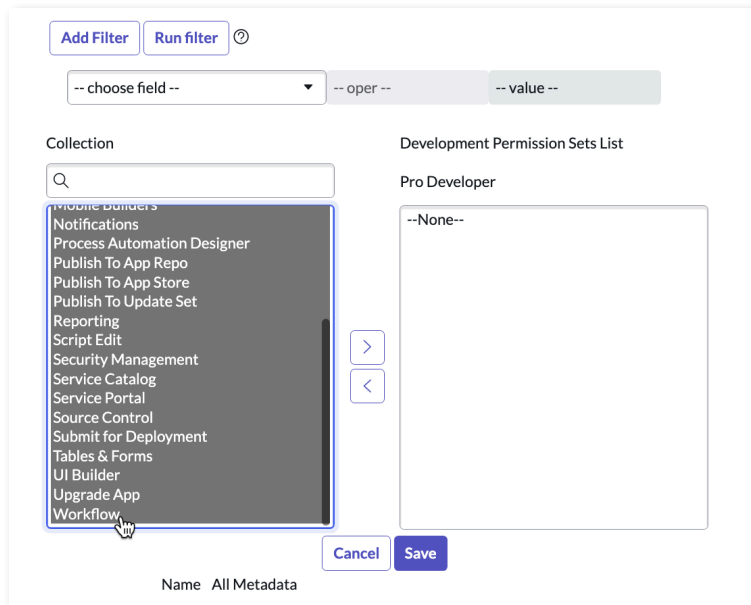
6. Click Edit



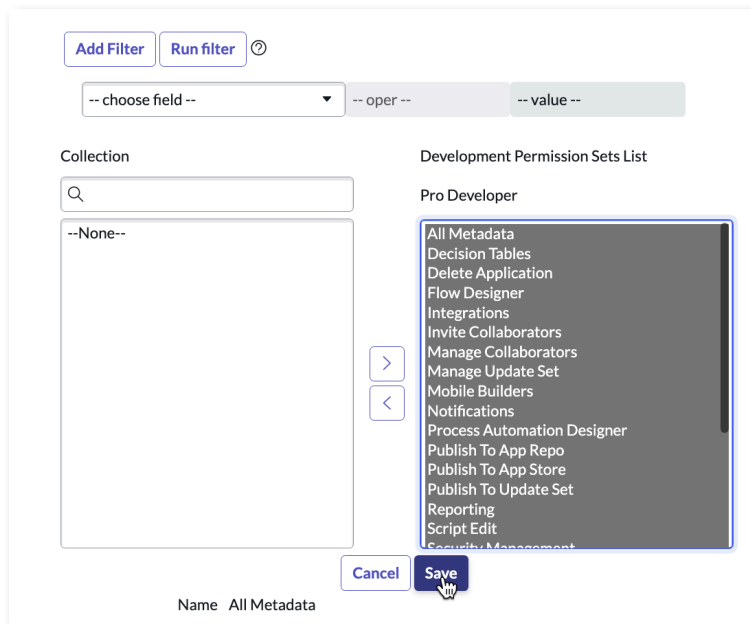
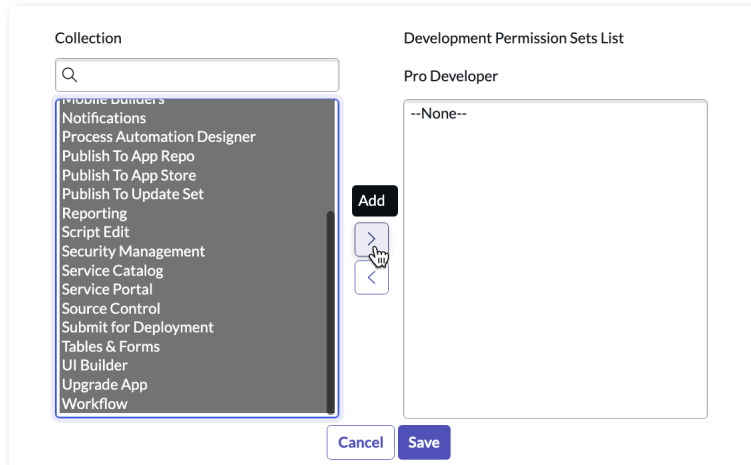
7. Click All Metadata



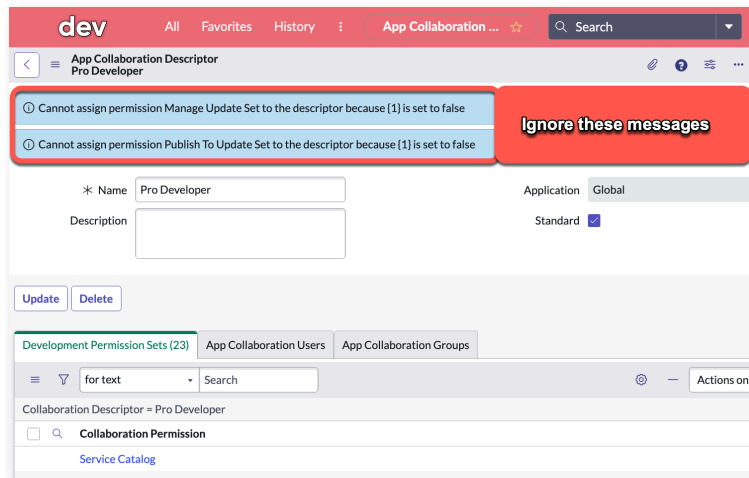
8. Scroll to the bottom, hold SHIFT, and click **Workflow**



9. Click the > button to move the roles to the right side of the list collector and click Save







## Lessons Learned

Here are some important takeaways from this task:

### Role of Collaboration Descriptors

Collaboration descriptors play a critical role in assigning, managing, and monitoring delegated development permissions for each application or uniformly across multiple applications. Therefore, creating appropriate descriptors is essential to ensure efficient delegation of permissions.

### Focus on Global Scope and Standard

It is crucial to create collaboration descriptors in the global scope and set them to standard for them to be visible and usable in AES. Any deviation from this guideline may lead to non-visibility or non-usability of the collaboration descriptors in AES.

### Utility of Custom Collaboration Descriptors

The process allows for the creation of custom collaboration descriptors that cater to specific needs beyond the predefined collaboration descriptors that come with activation. This flexibility allows for more precise permission delegation depending on the roles and requirements of users and groups.

# Implementation Complete

---

That completes the Implementation activities for App Engine.

Next, you will simulate different users as they use App Engine.

# Simulation

---

In these exercises, you will impersonate the different personas in App Engine.

# Exercise: Intake Request


Estimated Time to Complete: 10 minutes

## Overview

### ▶ Citizen Developer [*sit-uh-zuhn dih-vel-uh-per*]

Someone who, despite lacking formal coding training, engages in creating software applications and tools to solve specific business problems or enhance workflows. A citizen developer leverages low-code and no-code platforms to translate their domain knowledge into practical solutions.

In ServiceNow, this could be any user developing in Dev without the 'admin' role.

Sydney Carter	Experienced IT Professional & Citizen Developer
	Sydney Carter, an IT veteran with diverse roles under her belt, aspires to be part of the company's Citizen Developer Program. This section outlines the steps in Sydney's journey to becoming a citizen developer.

## Instructions

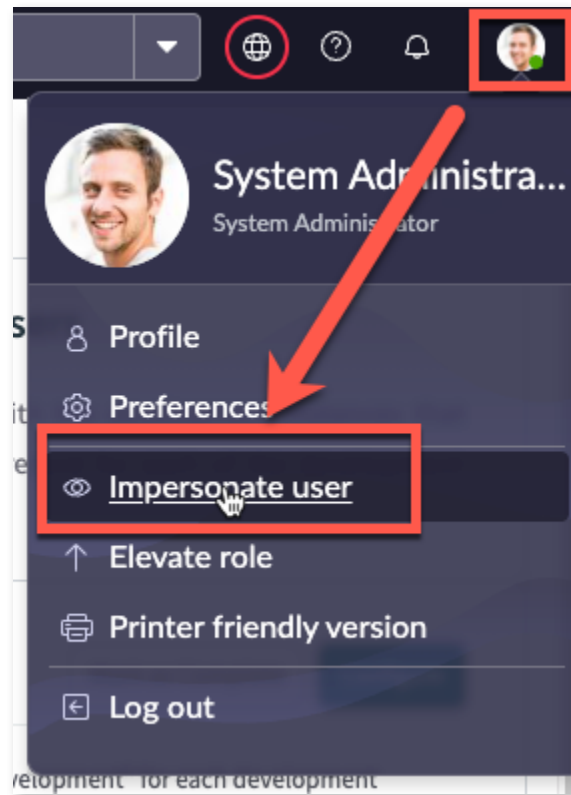


**DANGER**

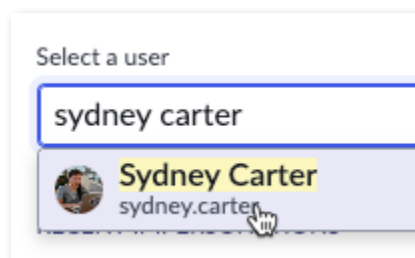
Complete this section in PROD.

1. Log in to Prod

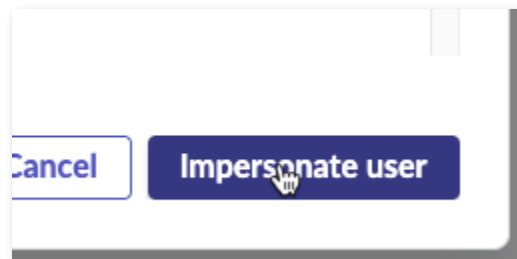
2. In the top right, click the System Administrator avatar >> Impersonate user.



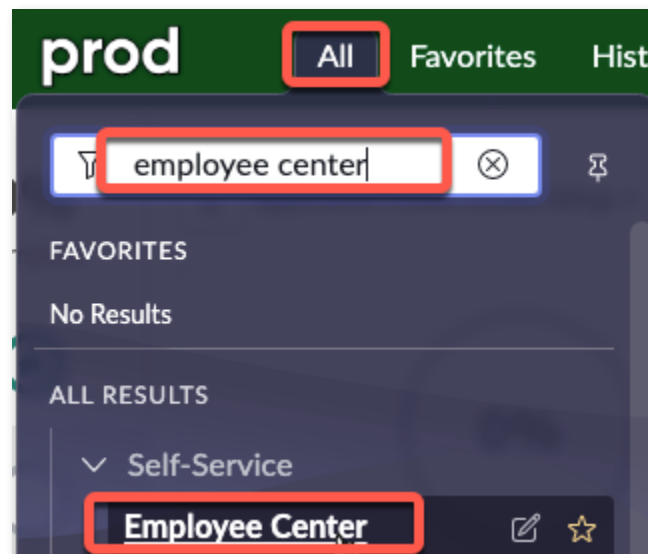
3. Type Sydney Carter and click Sydney Carter in the search results.



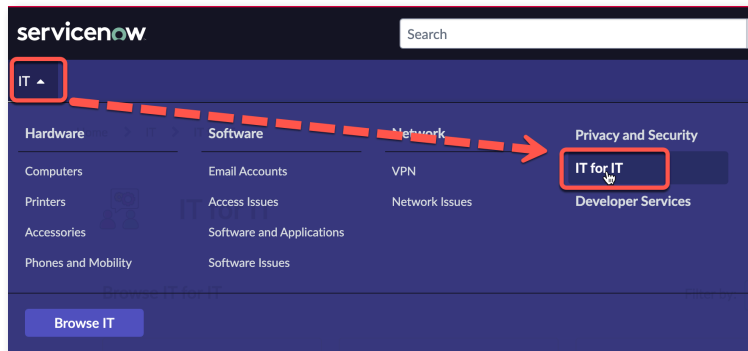
4. Click Impersonate User.



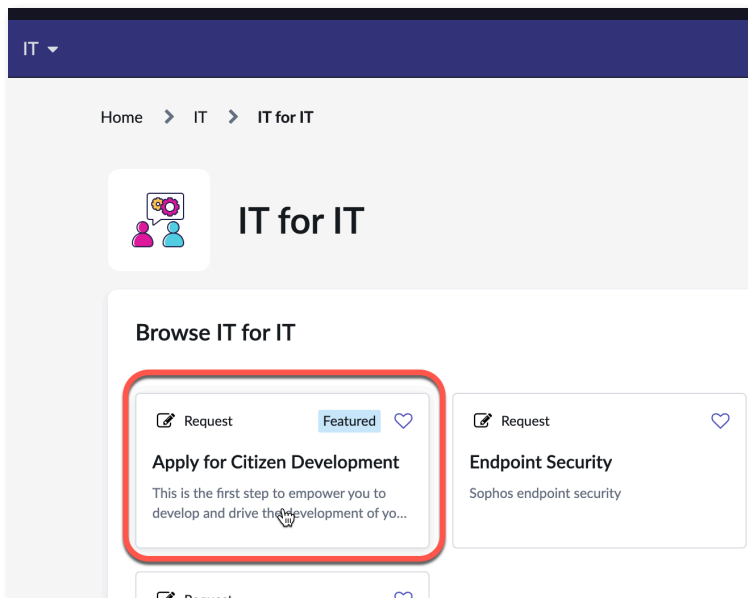
5. Click All >> type employee center >> click Employee Center



6. Click IT in the top left corner, then click on IT for IT



7. Click Apply for Citizen Development



8. Fill out the form with the values below.

**CAUTION**

The answers below will NOT affect the lab. They are only suggestions.

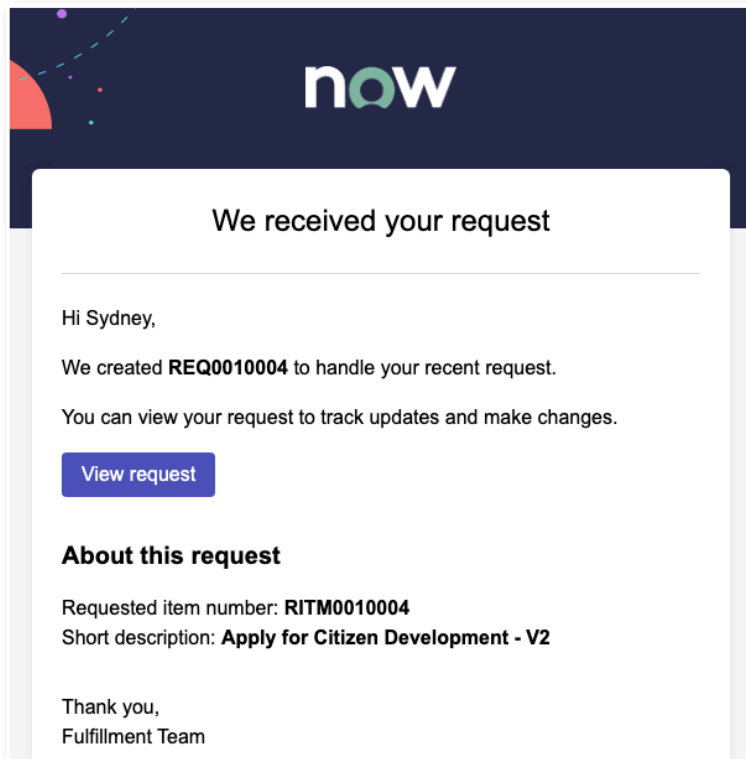
Field	Value
Application Name	Request Time Off
Describe your idea in a few sentences	Allow IT users to request time off with approvals.
Is your process repeatable?	Yes
Do you have an email or excel-based process?	Yes
How many users are involved in this?	>20
Does this involve any sensitive/PII data?	No
Do you need data from other departments?	No
Who are the users that will have access to the data in this application?	Anyone in the IT Department

9. Click Submit.

10. After the request is submitted, close the browser tab and return to the original browser tab for "Prod".

At this point, Sydney will receive an email letting her know the status of her request.





## Lessons Learned

The Intake Request process is an essential step for those who want to participate in the Citizen Developer Program. Properly completing the request form ensures smooth communication and prompt approval.


# Exercise: Intake Approval

Estimated Time to Complete: 10 minutes

## Overview

### ▶ App Engine Admin [*ap en-jin ad-min*]

Someone who has administrative responsibilities for configuring and managing App Engine applications on ServiceNow. An App Engine Admin oversees the deployment, scaling, and maintenance of applications while ensuring optimal performance and availability.

Jayne Nigel	App Engine Admin
	Jayne is familiar with many different software methodologies such as Agile and SAFe. She is familiar with Update Sets from her ServiceNow training, but she does not regularly work with them.

In this section, our App Engine Admin Jayne will review and approve Sydney's App Intake request.

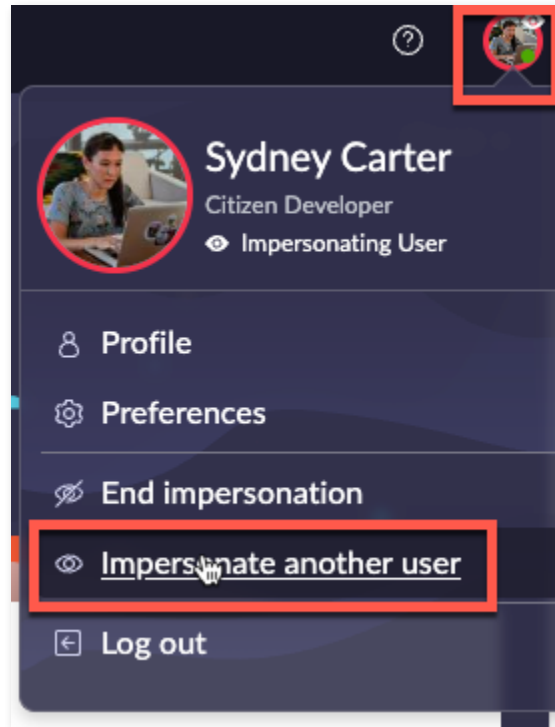
## Instructions

 DANGER

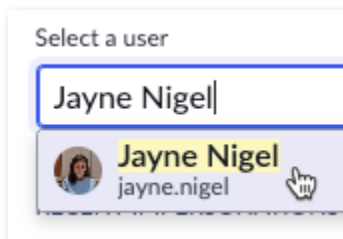
Complete this section in PROD.

1. Log in to Prod

2. Click the avatar in the top-right, then click Impersonate another user.

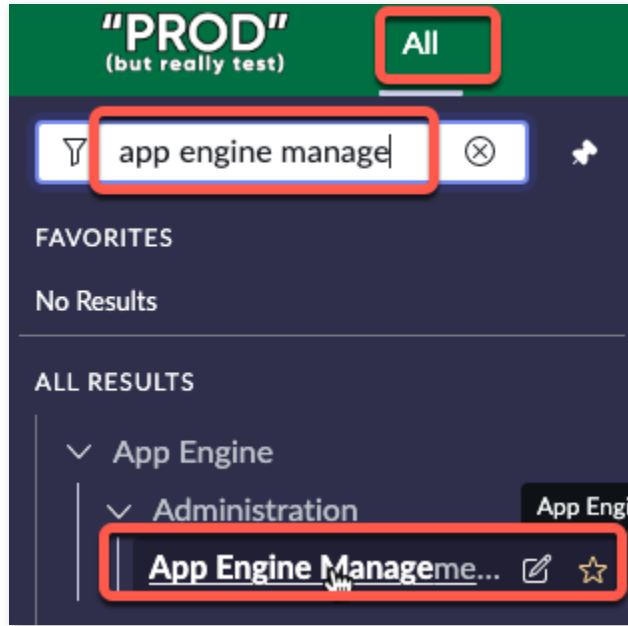


3. Type Jayne Nigel and click Jayne Nigel.



4. Click Impersonate user.

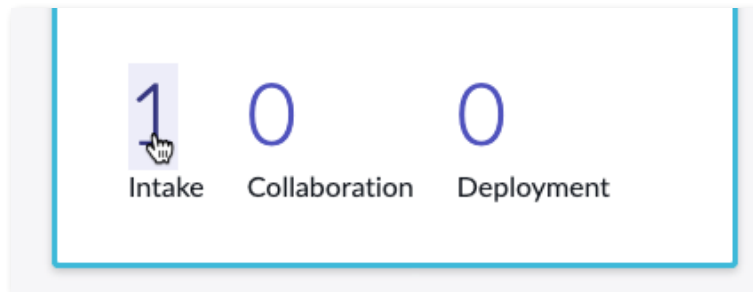
5. Click All >> type app engine management >> click App Engine Management Center.



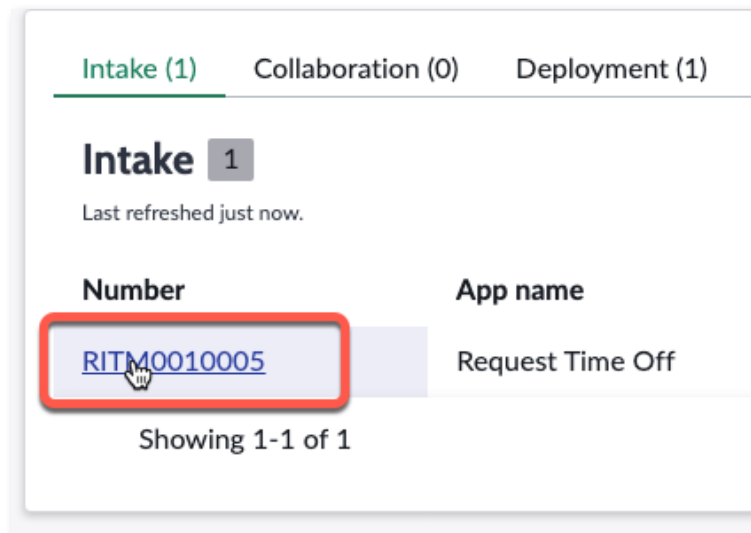
**CAUTION**

This will open AEMC in a new browser tab.

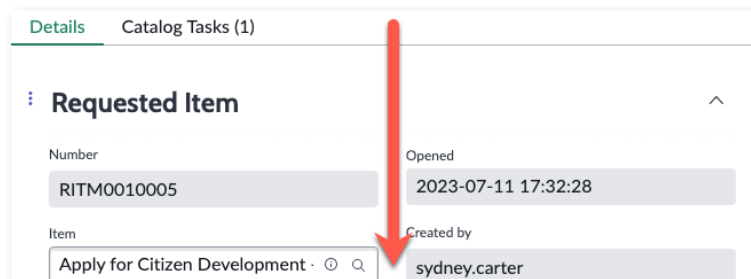
6. In the Pending requests to complete widget, click the number 1 above Intake to open Sydney's request.



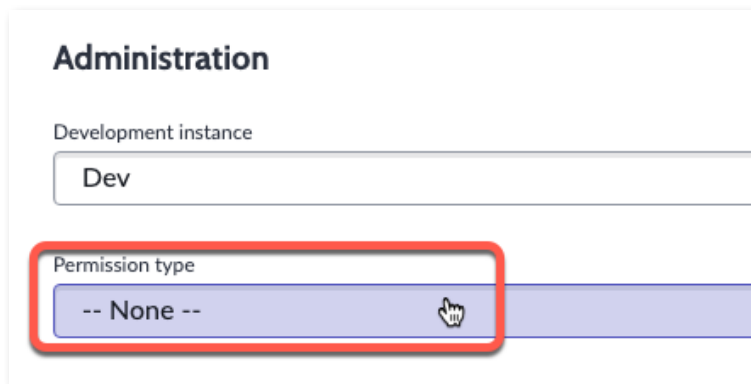
7. Click the RITM number to open Sydney's request.



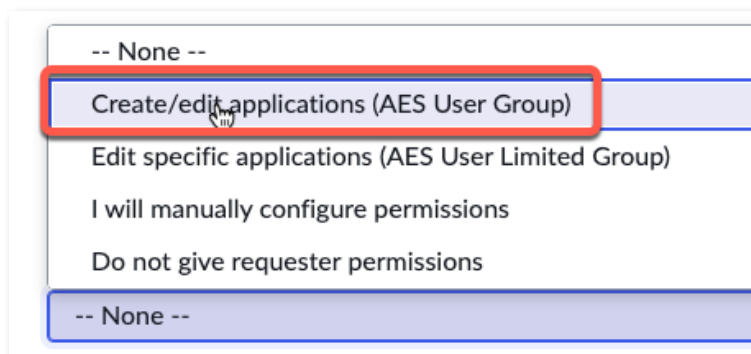
8. Scroll down the left-hand side of the page in the Details section.



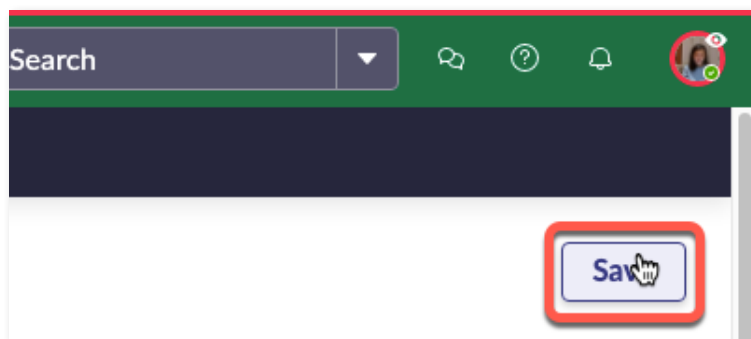
9. In the Administration section of the form, click on the Permission type drop-down field.



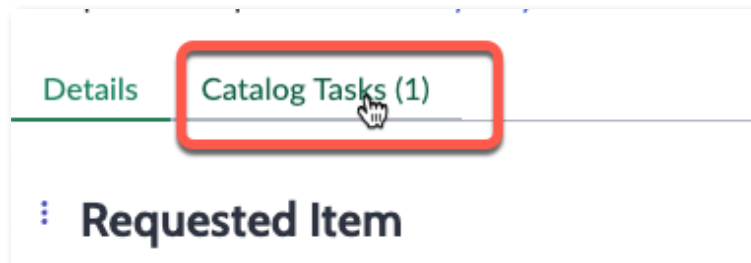
10. Click on Create/edit applications (AES User Group)



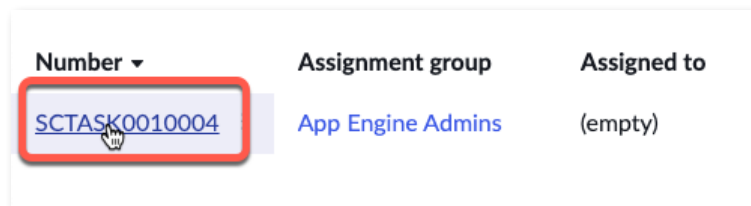
11. Scroll to the top of the page and click Save in the top right.



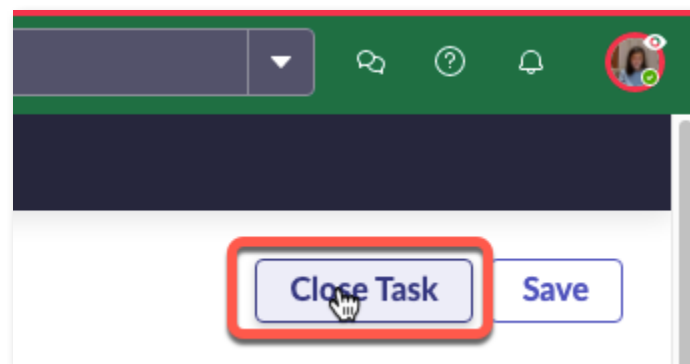
12. Click on the Catalog Tasks (1) related list.



13. Click the SCTASK number to open the Catalog Task.



14. Click Close Task in the top-right.



15. Click the ⓘ icon next to the RITM number to go back to the RITM record.

**Catalog Task**

Number: SCTASK0010004

Approval: Not Yet Requested

Assigned to: [Search]

Priority: 4 - Low

Configuration item: [Search]

State: Closed Complete

Active

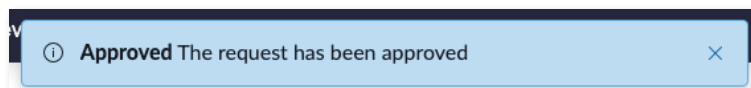
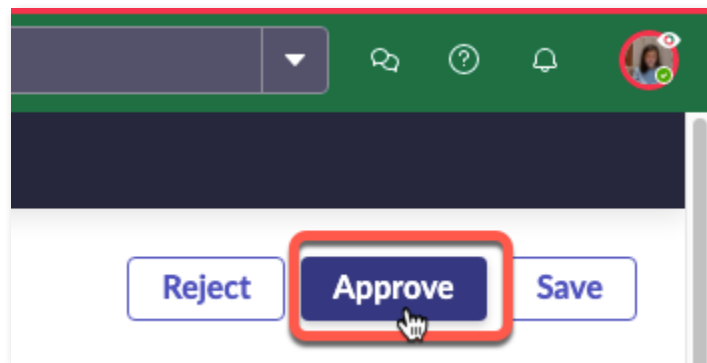
Request item: RITM0010005

Requested for: Sydney Carter

**⚠ CAUTION**

If you do not see the **Approve** button in the next step, you may need to refresh the page.

16. Click **Approve** in the top-right of the page.



The Prod instance will configure Sydney's App Engine User access on the Dev environment.



In the real world, there might be more than one person to do the approval and more time may be taken to review the answers.

You may also want to require the person to undergo additional training before approving their access to build applications.

## Lessons Learned

During our journey through the 'Intake - Approval' process, several insightful takeaways emerged, all of which can prove helpful for future reference:

### Role Clarification

The process shed light on the critical role that Jayne, the App Engine Admin, plays within the system. Her duties, while not labeled 'Platform Administrator' or assigned any 'admin' roles in any environment, are far-reaching. Her approvals initiate a sequence of automated procedures that are crucial for the smooth functioning of the system. This serves as a reminder that all roles, regardless of their titles, have significant responsibilities and impacts.

### Approval Process

The impact of the approval process on subsequent actions (like triggering an automated email to Sydney) was an essential takeaway. This demonstrates the power of automated workflows in saving time and reducing manual errors.

### Segregation of Duties

Successfully completing this process without Jayne needing an 'admin' role showcases the 'Segregation of Duties' principle. It illustrates that high-level access (like the 'admin' role) isn't always required to complete significant tasks. This reduces potential security risks and promotes system integrity.

# Exercise: Build the app from template

---

Estimated Time to Complete: 15 minutes

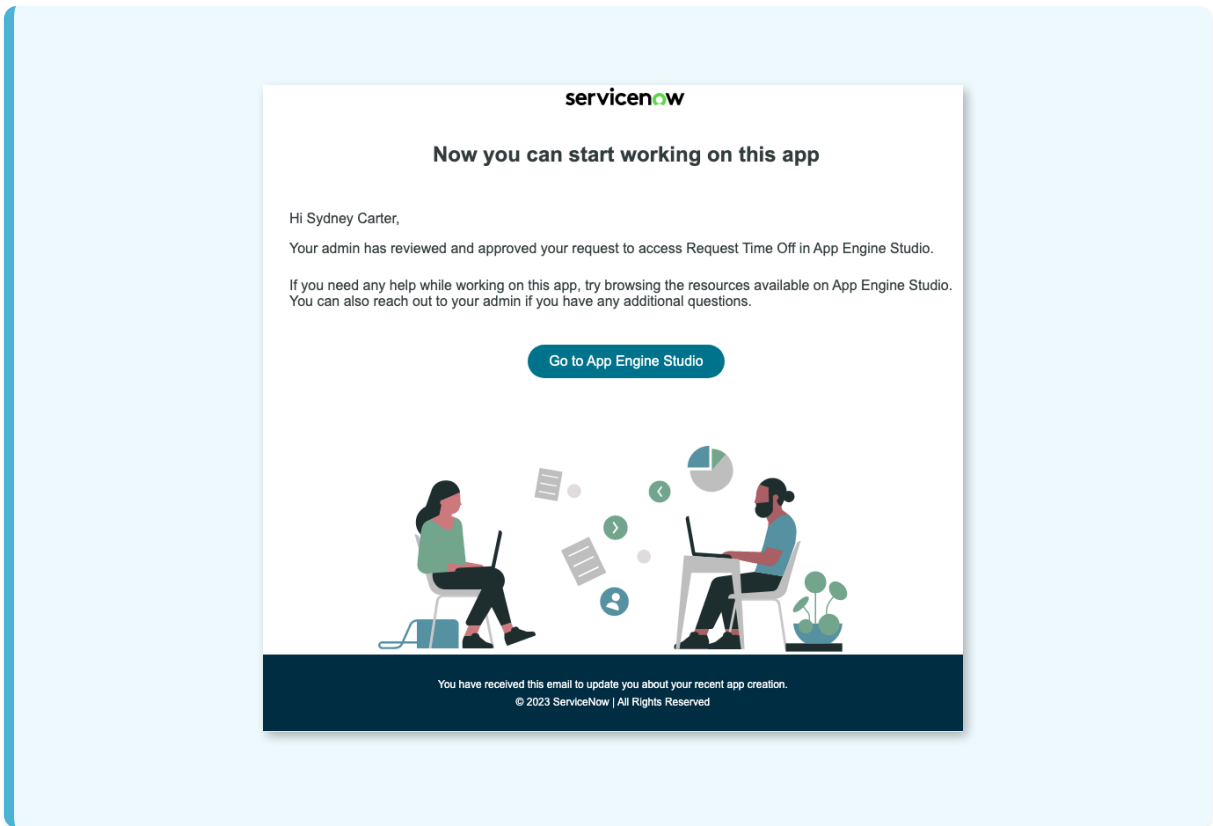
## Overview

In this section, Sydney will build her app using an app template

After Jayne approved her request, she received an email that her app has been approved.

! INFO

SUBJECT: Your app has been approved



She clicks the button in the email to go to App Engine Studio, but we will simulate this.

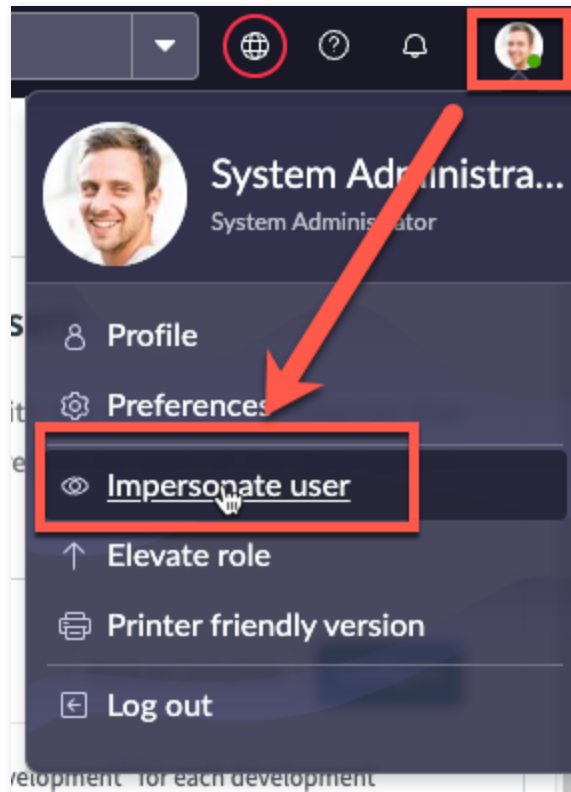
## Instructions



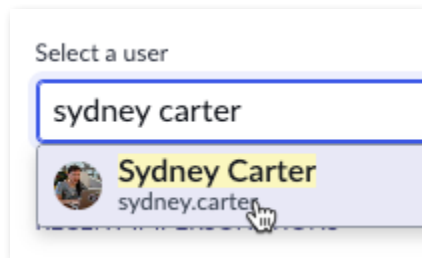
**DANGER**

Complete this section in DEV.

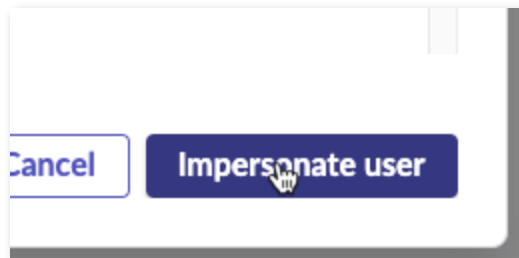
1. Log in to Dev.
2. In the top right, click the System Administrator avatar >> Impersonate user.



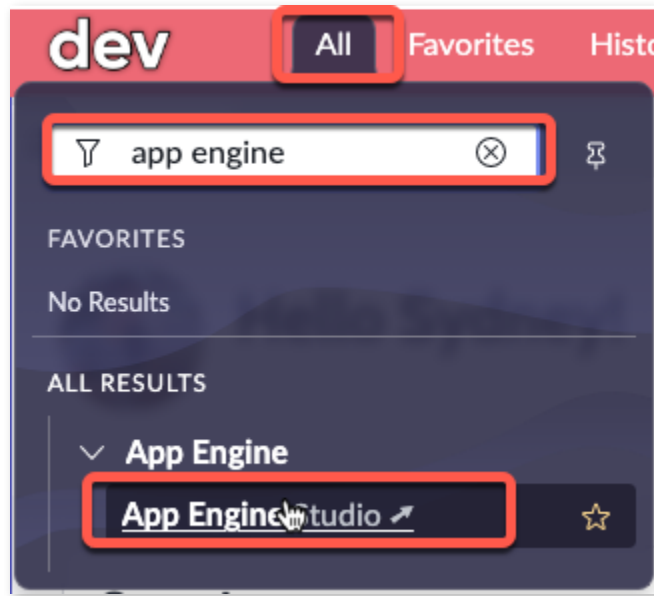
3. Type Sydney Carter and click Sydney Carter in the search results.



4. Click Impersonate User.



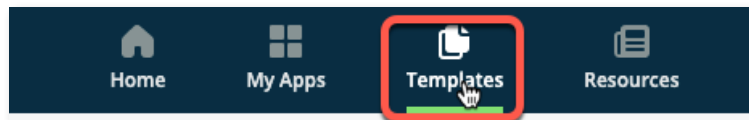
5. Click All » type app engine » click App Engine Studio



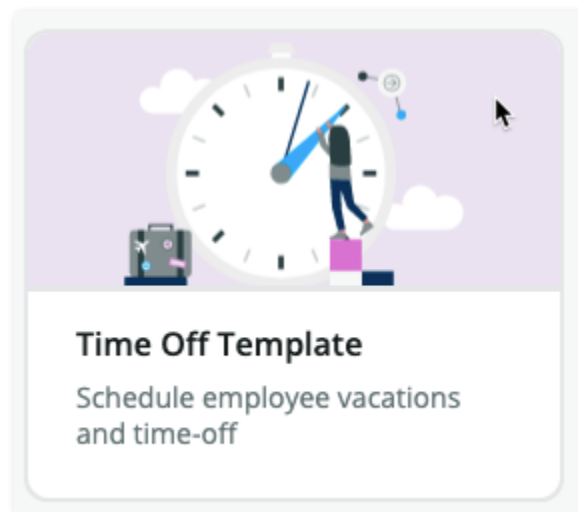
**⚠ CAUTION**

This will open AES in a new browser tab.

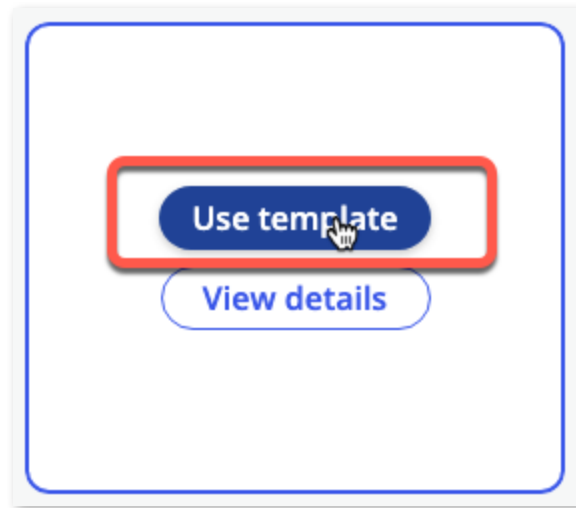
6. At top of page, click Templates.



7. Move your mouse over Time Off Template.



8. Click Use template.



9. Enter the name IT Time Off Request and click Continue



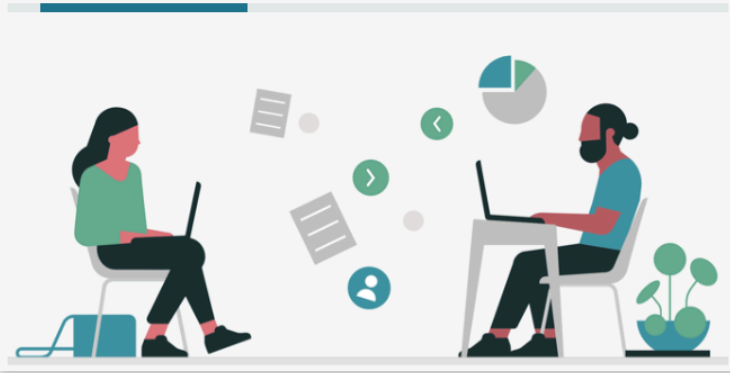
10. Go take a five minute break while the app builds from the template. This is a good time to stretch your legs, check your email, etc.



## CREATE APP

### Sit tight while we create this app

This may take a few minutes...



11. Click Go to app home.

### Great! Let's review what's in your app

Next, you'll be able to review and modify everything that came with the template. Once you're done, you can submit your app to your administrator for review.



[Go to app home](#)

#### CAUTION

In this exercise, we are not exploring App Engine Studio; however, feel free to take a moment and explore the application that was built if you have time.



## Lessons Learned

The template allowed Sydney to quickly build an app. She can now customize it for her needs before requesting to have it deployed to Production.

# Exercise: Collaboration Request

---

Estimated Time to Complete: 5 minutes

## Overview

In this exercise, Sydney needs help building an Integration in her app.

She will submit a [Collaboration Request](#) ↗ for Priya to assist her.

## Instructions

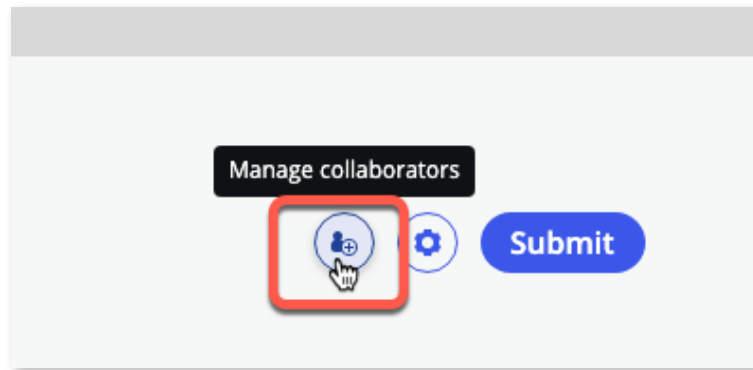
### CAUTION

Complete the next exercise as Sydney in the Dev environment. If you are following the directions from the last exercise, you should still be impersonating her in Dev and still be in App Engine Studio.

### DANGER

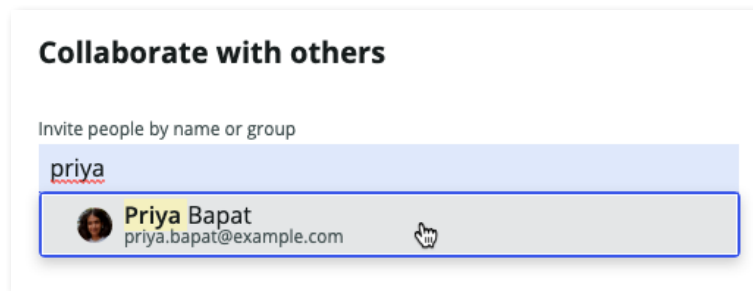
Complete this section in DEV.

1. Log in to Dev.
2. Click the Manage Collaborators button.

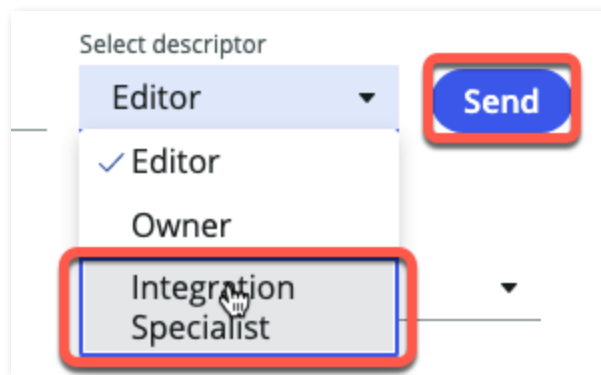


3. Type 'Priya' into the field Invite people by name or group.

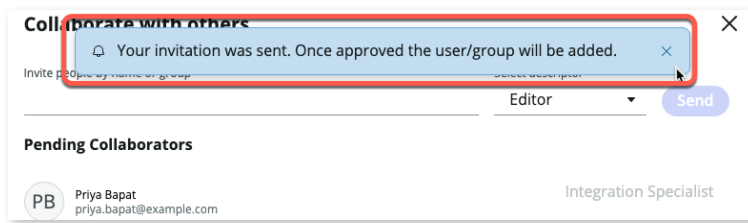
4. Click on her name.



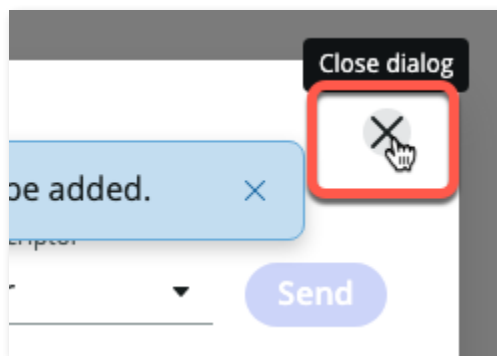
5. Select Integration Specialist in the dropdown and then click Send.



6. You should see a blue message as below.



7. Click the X to close the dialog.



## Lessons Learned

### Simplicity of Collaboration

The process of requesting collaboration within ServiceNow's App Engine Studio is straightforward and user-friendly, facilitating an efficient collaboration process. The simplicity of inviting someone by name or group minimizes confusion and accelerates the collaboration process.

### Role Assignment

The ability to assign specific roles, like "Integration Specialist" in this case, during the invitation process allows for clearer task division and expectations. It reinforces the need to understand different roles within a team and their specific responsibilities.

# Exercise: Collaboration Approval

---

Estimated Time to Complete: 5 minutes

## Overview

Jayne will need to approve Sydney's Collaboration request for Priya to assist her with the integrations in her app.

## Instructions

### DANGER

Complete this section in **PROD**.

### CAUTION

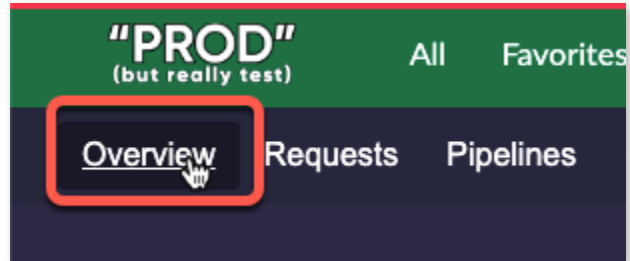
You should still have a browser tab for Prod opened to App Engine Management Center in which you are impersonating Jayne.

If not, follow these quick instructions to be sure:

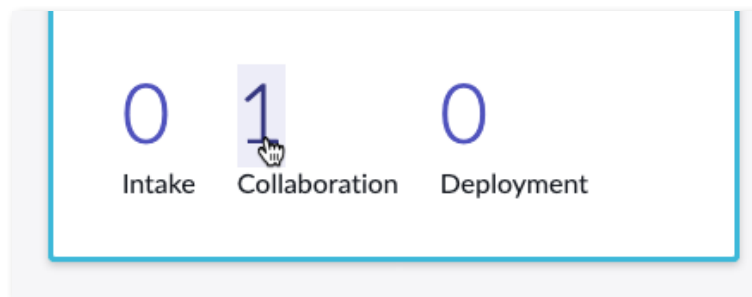
1. Click the avatar in the top-right, then click **Impersonate another user**.
2. Type Jayne Nigel and click Jayne Nigel.
3. Click **Impersonate user**.
4. Click **All >>** type **app engine management >>** click **App Engine Management Center**.

This will open AEMC in a new browser tab.

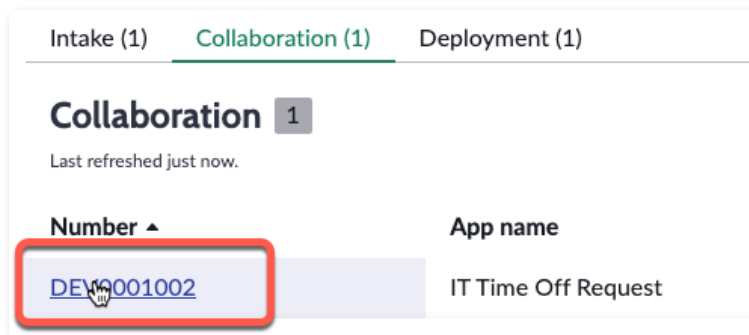
1. Click Overview in App Engine Management Center



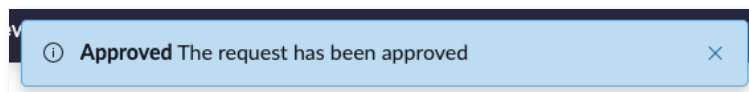
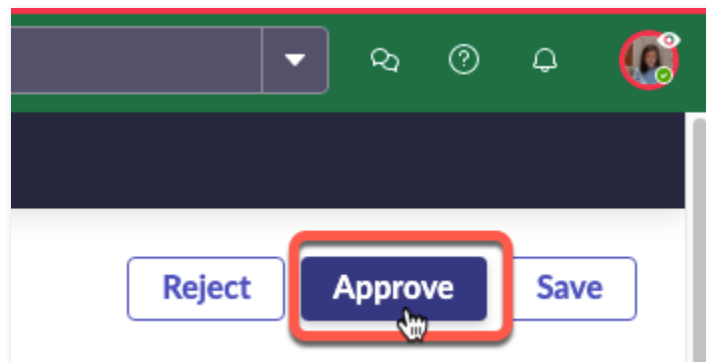
2. Click the 1 above Collaboration



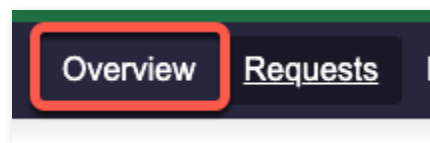
3. Click the DEV ticket to open Sydney's Collaboration request.



4. Click Approve in the top-right of the page.



5. Click Overview to return to the main AEMC page.



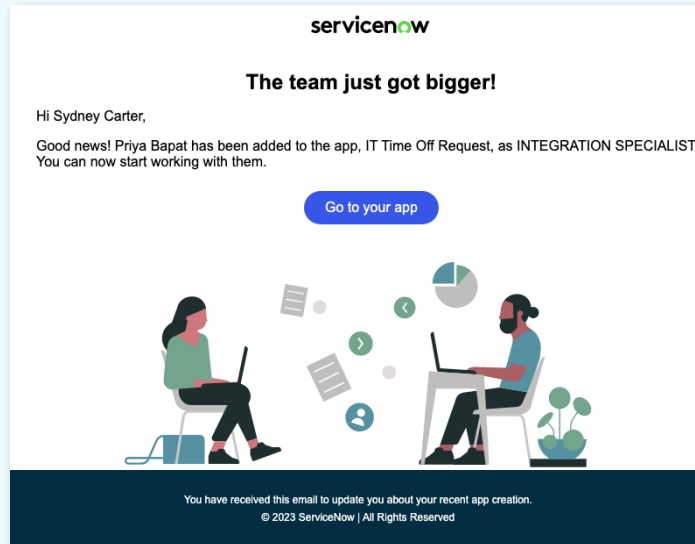
After Jayne approved the Collaboration Request, Sydney received an email to inform



her that the request was approved.

ⓘ INFO

SUBJECT: Your collaborator request was approved



## Lessons Learned

Great!! Approving the request triggered AEMC to automatically provision Priya's access to work on the app with Sydney as an Integration Specialist.

# Exercise: Configure Integration


Estimated Time to Complete: 15 minutes

## ! INFO

THIS SECTION IS OPTIONAL AND CAN SAFELY BE SKIPPED IF YOU ARE BEHIND. Skipping steps below will have not prevent you from completing the lab.

## Overview

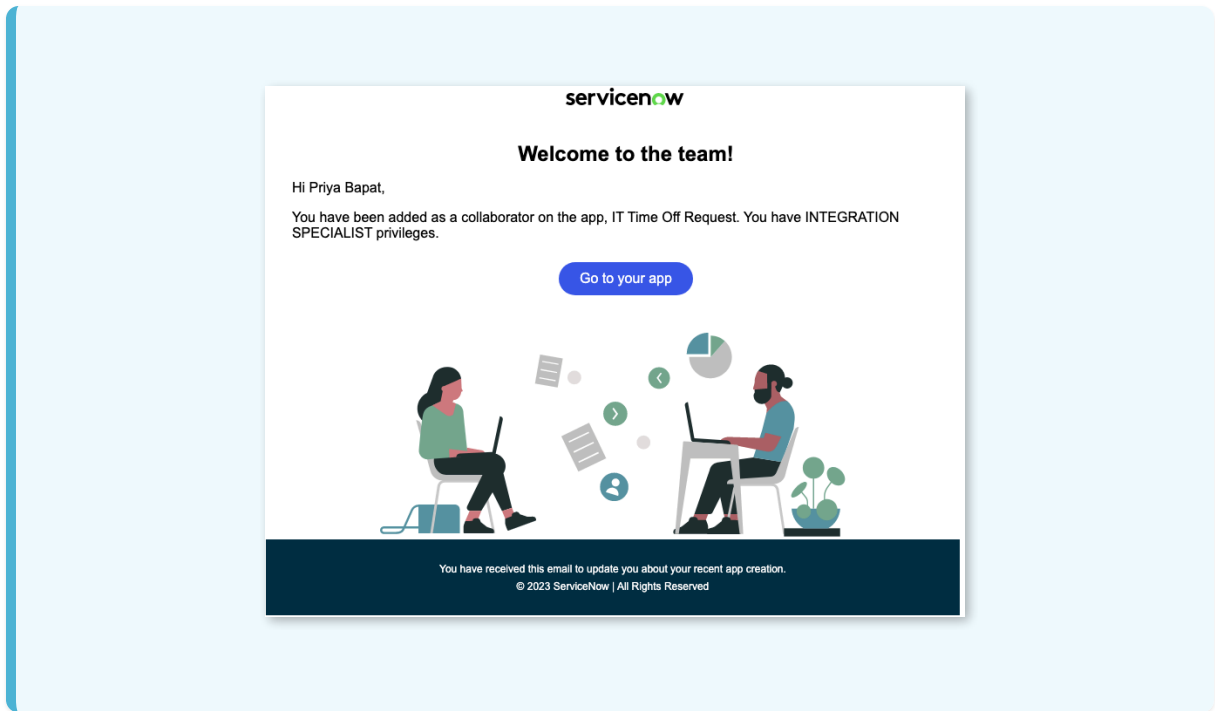
In this section, Priya will go to Dev and work on Sydney's app.

Priya Bapat	Intgration Specialist
	Priya has many years in IT building integrations on various Platforms. She recently completed some Integration training on ServiceNow and is ready to assist Sydney.

After Jayne approved the Collaboration Request, Priya received an email that she is now a collaborator on the app.

## ! INFO

SUBJECT: You are now a collaborator



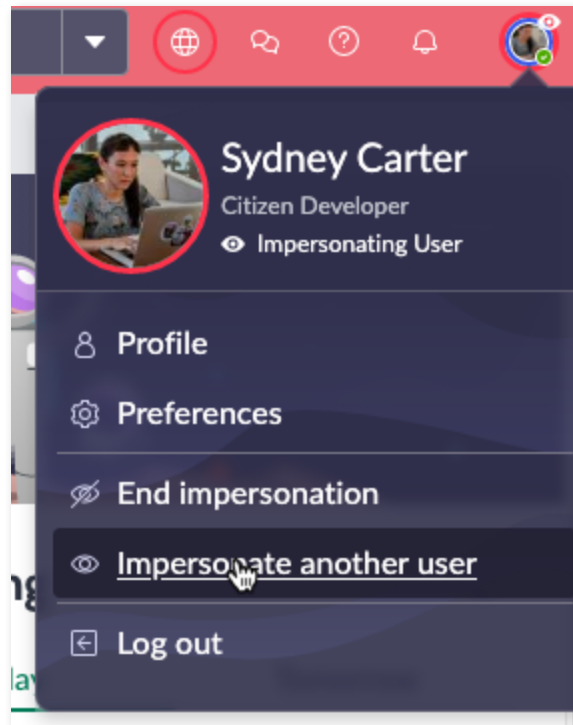
When Priya clicks the button in the email she is taken directly to App Engine Studio in Dev; however, we will simulate this.

## Instructions

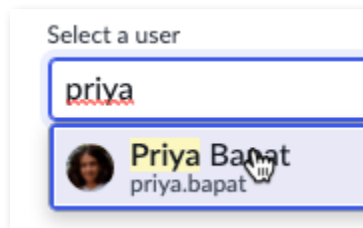
 DANGER

Complete this section in DEV.

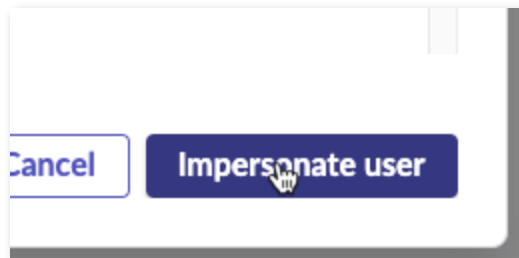
1. Log in to Dev.
2. In the top right, click the user avatar icon » Impersonate another user.



3. Type Priya and click Priya Bapat in the search results.



4. Click Impersonate User.



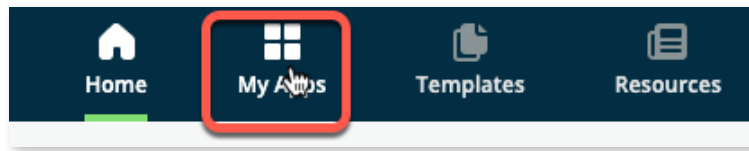
5. Click All » type `app engine` » click App Engine Studio



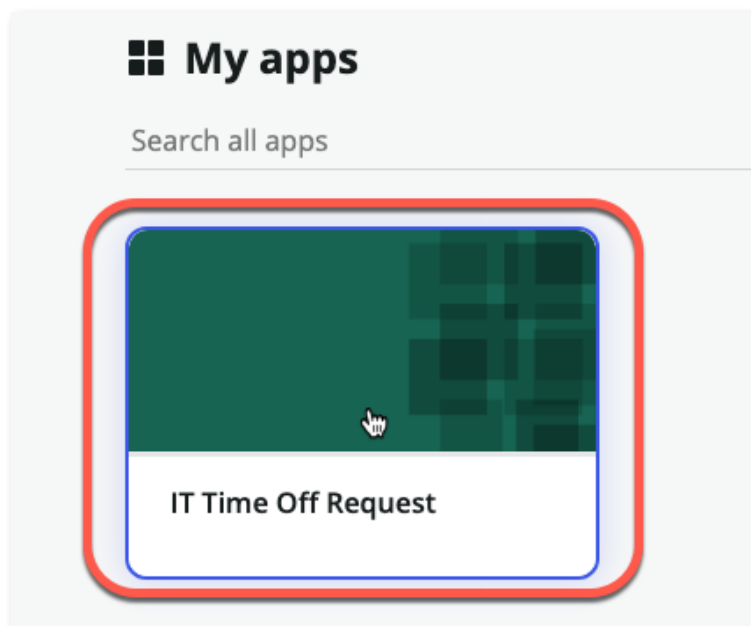
**⚠ CAUTION**

This will open AES in a new browser tab.

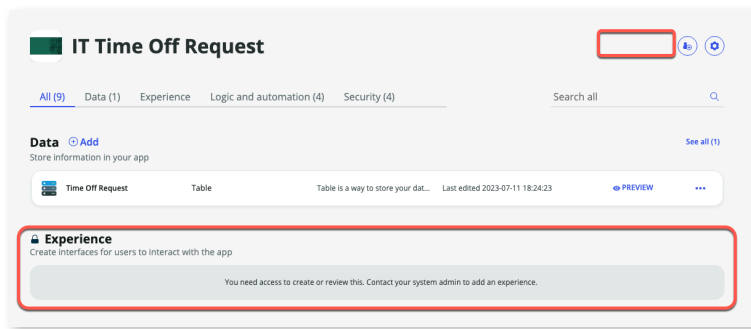
6. Click My Apps



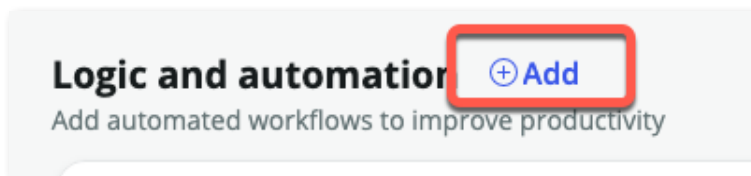
7. Click on the IT Time Off Request app



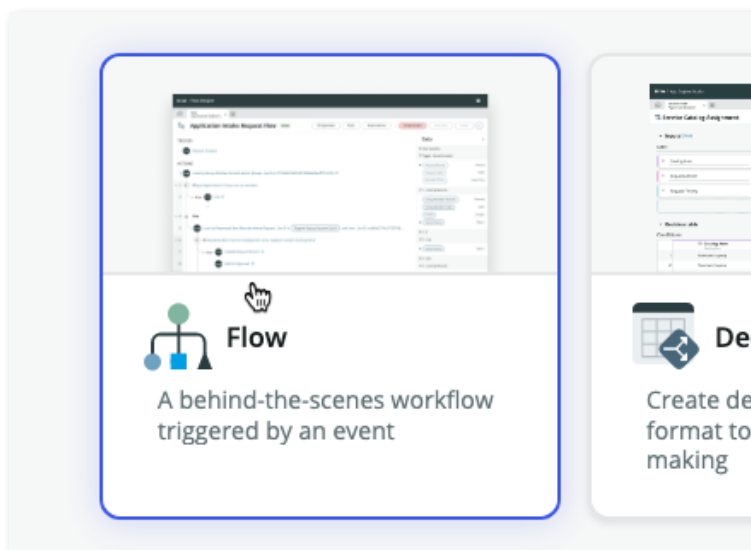
8. Note some of the differences in Priya's experience versus Sydney's. She does not have access to the **Submit** button to request to deploy the app. She also can not see or modify any of the **Experiences** in the app.



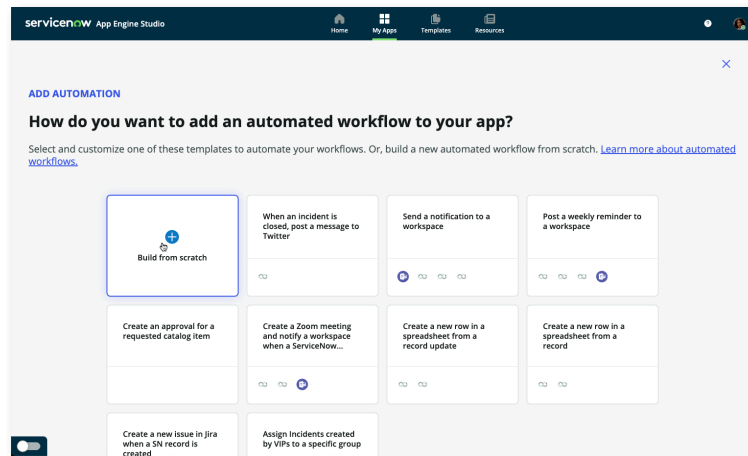
9. Scroll down and click +Add next to Logic and Automation.



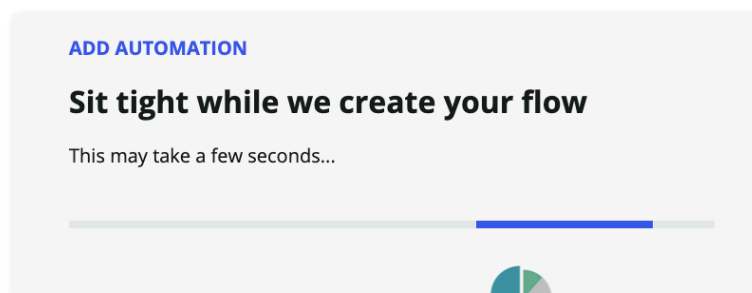
10. Click Flow



11. Click Build from scratch

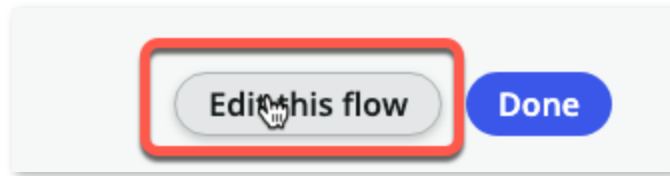


12. Enter **Send data to Workday** in the Name field and click Continue

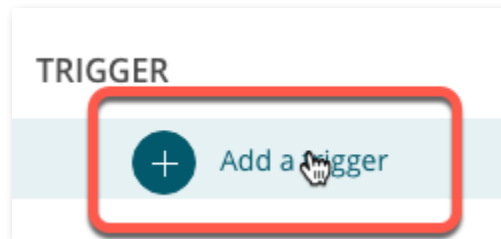


13. Click Edit this flow

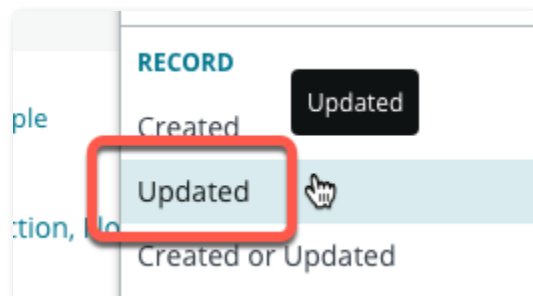




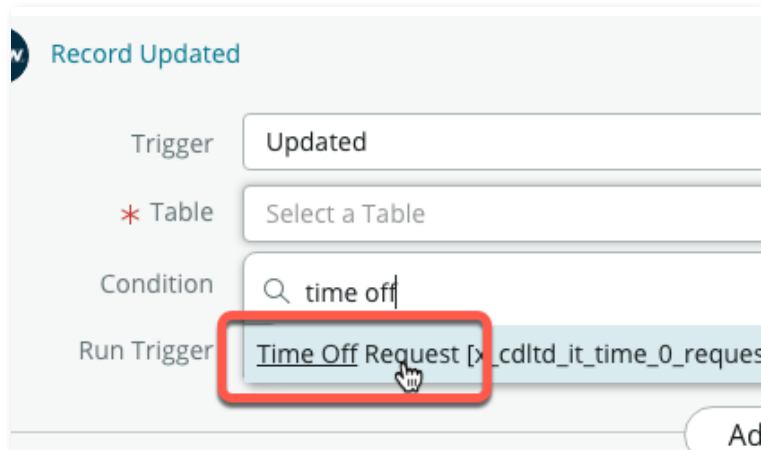
14. Click Add a trigger



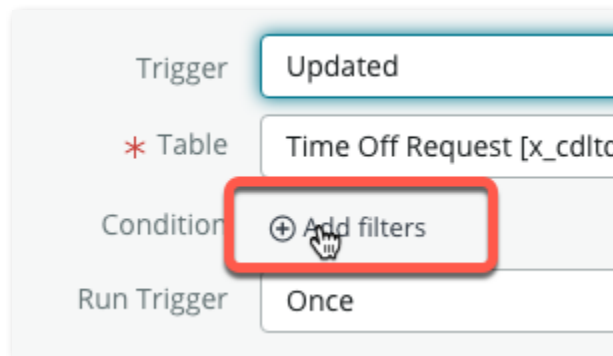
15. Click Updated



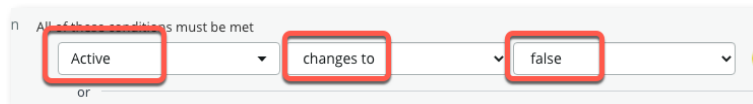
16. In the Table field, type `time off` and click the Time Off Request table.



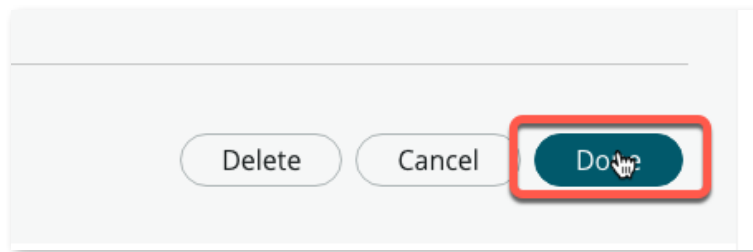
17. Click Add filters next to Condition



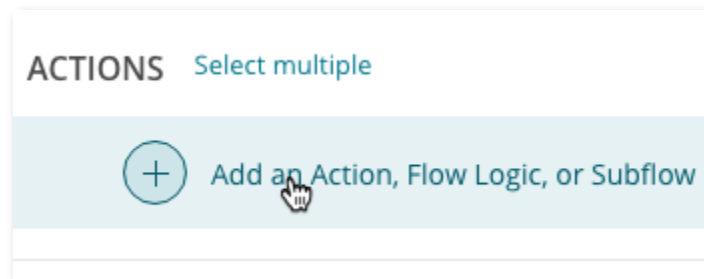
18. Configure a condition of Active changes to false



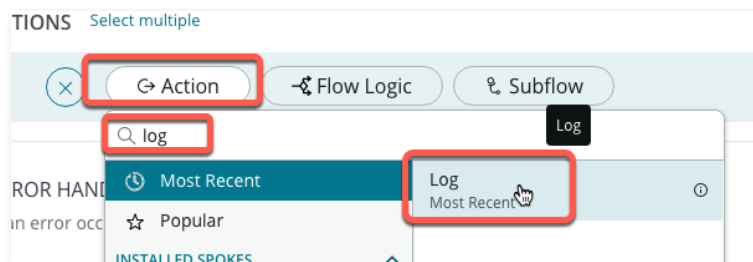
19. Click Done



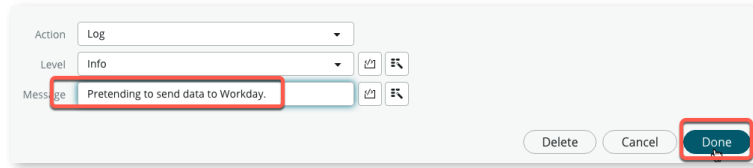
20. Click Add an Action, Flow Logic, or Subflow



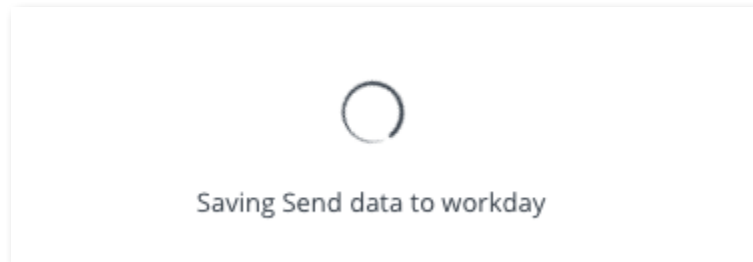
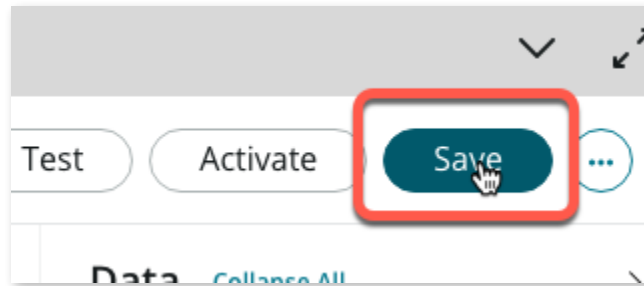
21. Click Action >> type `log` >> click Log



22. Type something in the Message field and click Done

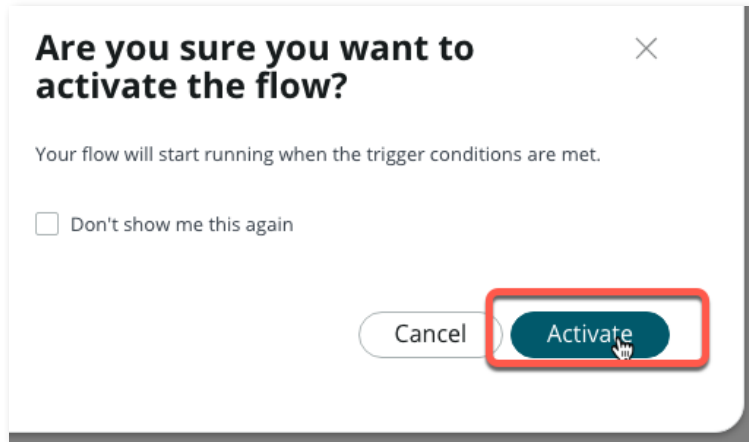


23. Click Save in the top-right

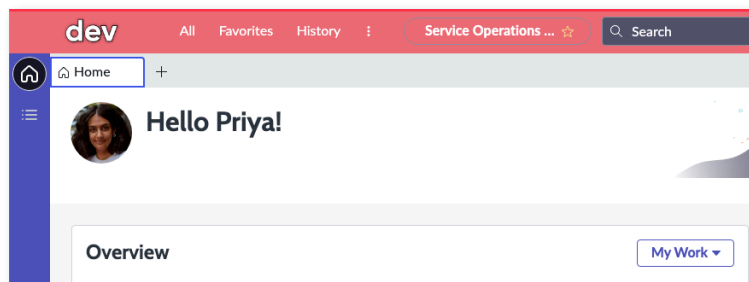


24. Click Activate >> Activate





25. Click the App Engine Studio browser tab and return to the Dev platform view.



## Lessons Learned

Priya has built an integration in the app. It is simplistic for the lab, but in a real world scenario she would build a more complex integration for Sydney.

The app is now ready for Sydney to request it to be deployed.

# Exercise: Request Deployment

---

Estimated Time to Complete: 5 minutes

## Overview

Sydney will click the **Submit** button in AES which will trigger a Deployment request to get created in AEMC on the Prod instance.

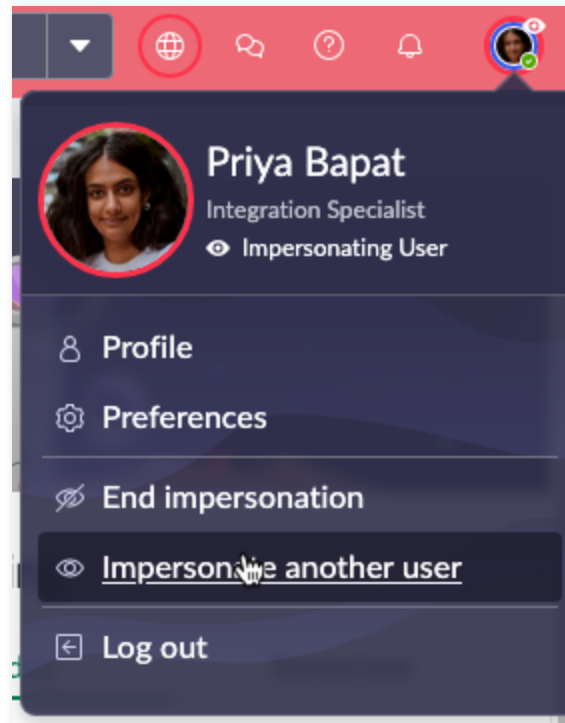
## Instructions



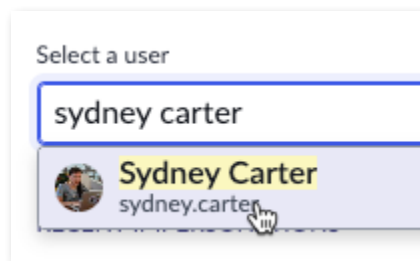
**DANGER**

Complete this section in DEV.

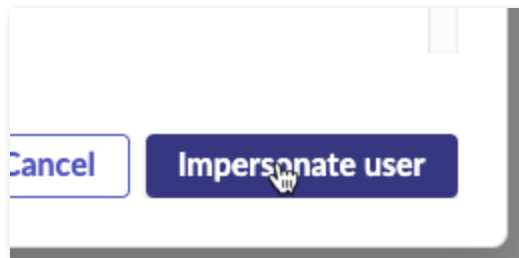
1. Log in to Dev.
2. In the top right, click the user avatar icon » Impersonate another user.



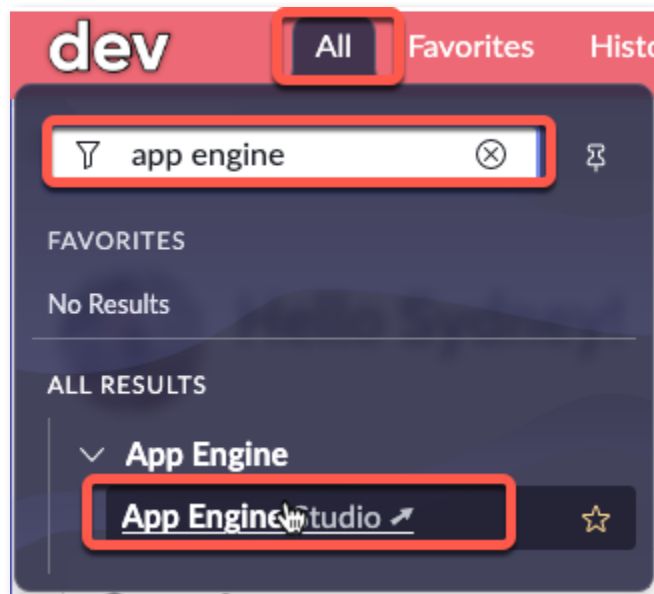
3. Type Sydney Carter and click Sydney Carter in the search results.



4. Click Impersonate User.



5. Click All » type `app engine` » click App Engine Studio

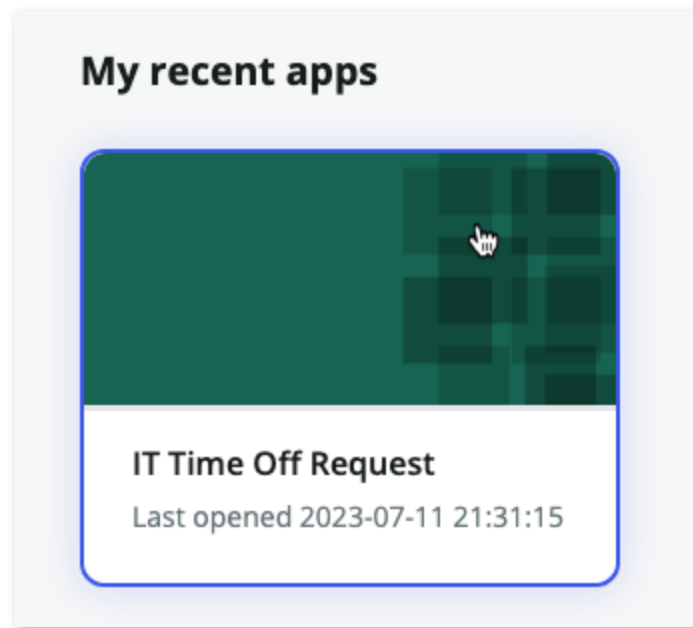


**⚠ CAUTION**

This will open AES in a new browser tab.

6. Click the IT Time Off Request tile.

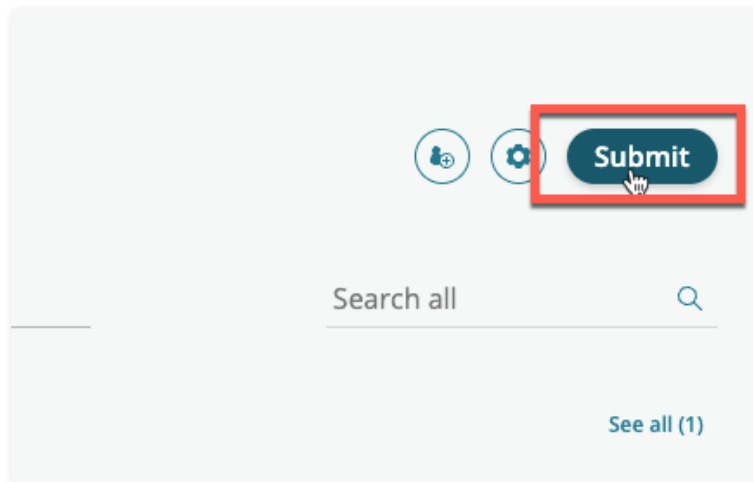




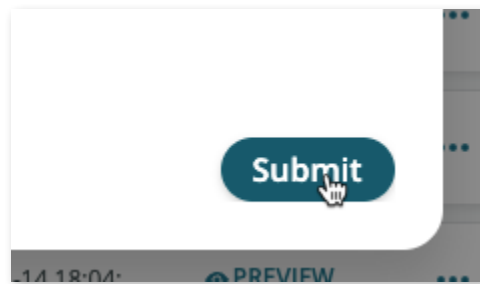
**⚠ CAUTION**

If the Submit button is grayed out in the next step, try refreshing the page.

7. Click Submit.

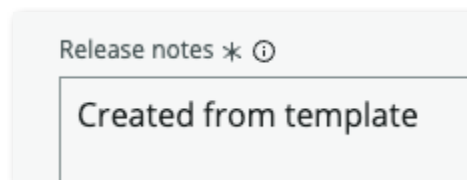


8. Click **Submit** in the pop-up.

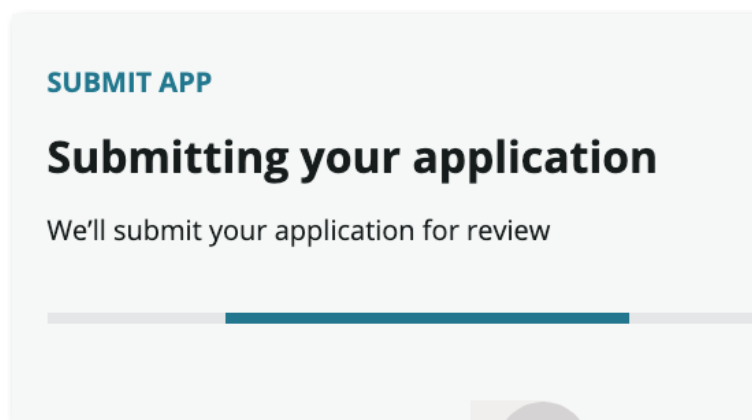


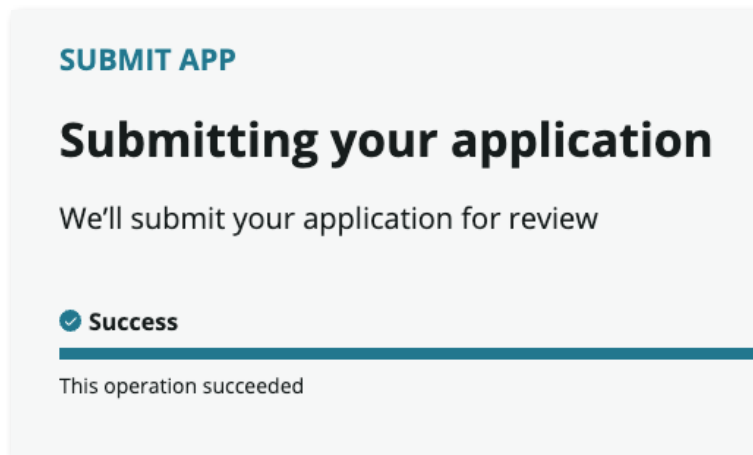
9. Scroll down.

10. In the Release notes field type **Created from template**.

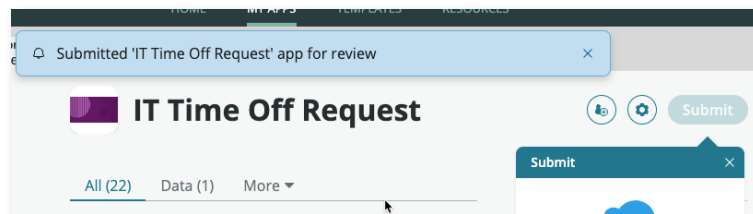
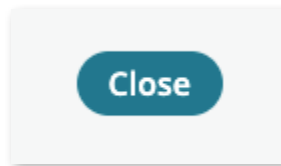


11. Click **Continue**.





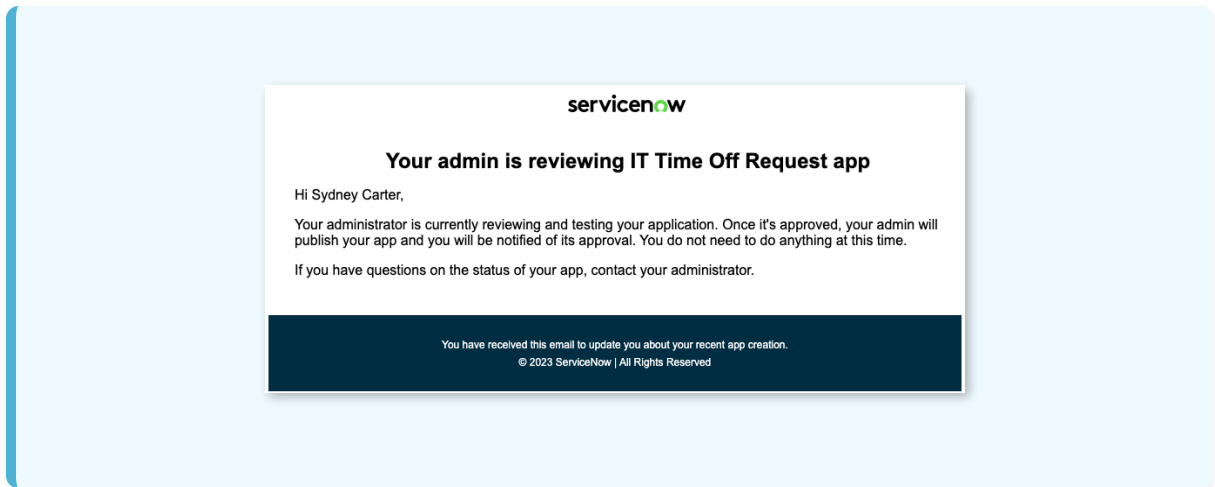
12. Click Close in the bottom-right corner.



Sydney receives an email letting her know that her Deployment Request was submitted.

 INFO

SUBJECT: IT Time Off Request is being reviewed by your admin



## Lessons Learned

Sydney was able to build an app and submit a request to deploy it. She did not need to learn about Update Sets.

She can continue to edit the app, but she will not be able to submit a new deployment request until the current one is resolved.

# Exercise: Approve and Deploy

## Sydney's App

---

Estimated Time to Complete: 15 minutes

### Overview

When Sydney submitted the Deployment Request in App Engine Studio, it triggered an email to Jayne prompting her to take action.

#### ! INFO

SUBJECT: Deployment Request DEP0010002 Approval Request

Short Description: Deploy 'IT Time Off Request 1.0.1' to 'Prod' for Sydney Carter  
Priority: 4 - Low  
Category:

Comments:

[Click here to approve DEP0010002](#)

[Click here to reject DEP0010002](#)

Approval Activity:  
You are the sole approver.

Click here to view Approval Request: [LINK](#)  
Click here to view Deployment Request: [LINK](#)

#### ! CAUTION

You should still have a browser tab for Prod opened to App Engine Management Center in which you are impersonating Jayne.

If not, follow these quick instructions to be sure:

1. Click the avatar in the top-right, then click **Impersonate another user**.
2. Type Jayne Nigel and click Jayne Nigel.
3. Click **Impersonate user**.
4. Click **All >> type app engine management >> click App Engine Management Center**.

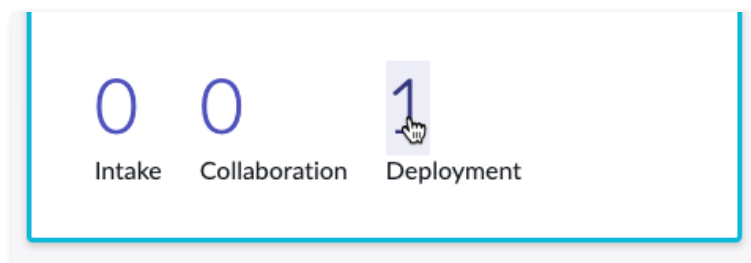
This will open AEMC in a new browser tab.

## Instructions

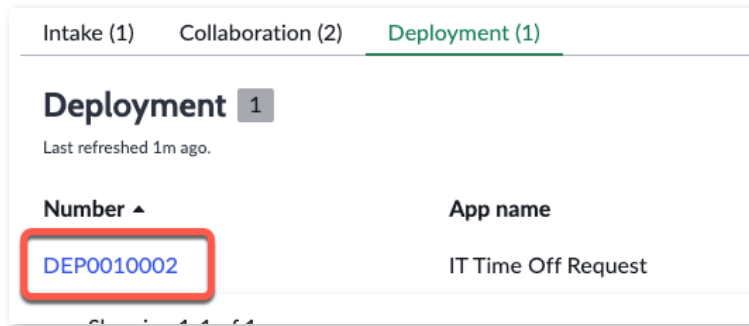
### DANGER

This section is to be completed in your **Production** instance where AEMC is running. That is your controller instance.

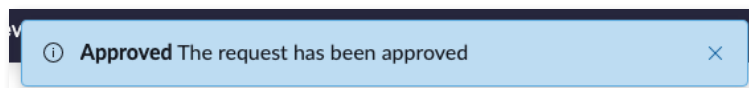
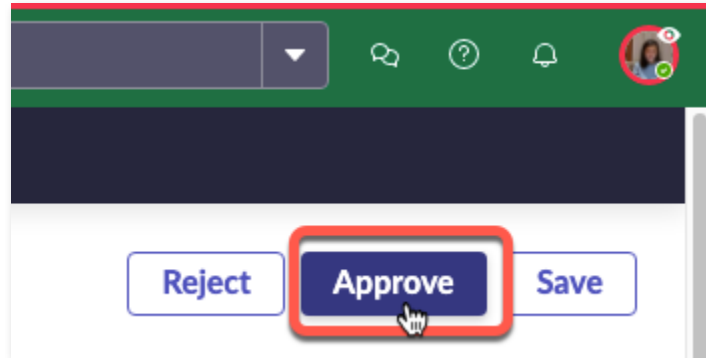
1. Click the 1 above **Deployment**.



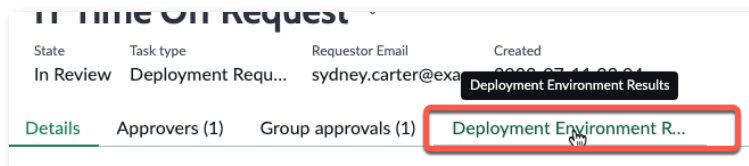
2. Click the DEP ticket to open Sydney's Deployment Request.



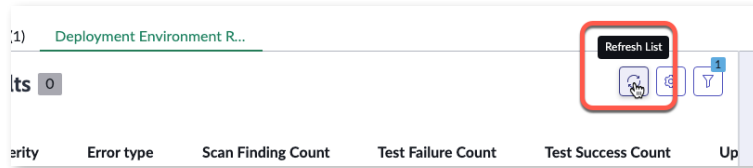
3. Click Approve in the top-right of the page.



4. To check the status of the Deployment... Click the related list Deployment Environment R..



- Occasionally click the **Refresh** button to check on the progress. Once the app is deployed to the Prod environment a record will appear here.



**⚠ CAUTION**

This is a good point to go take a five minute break or stretch. It can take several minutes or more for the app to deploy to the next environment.

- Eventually, you will see a record appear for your deployment.

The screenshot shows a table titled 'Deployment Environment Results' with a '1' in a grey box next to the title. Below the title, it says 'Last refreshed just now.' The table has columns for 'Environment', 'Has Error', 'Error severity', and 'Error type'. A record is shown with 'Prod' in the 'Environment' column and 'false' in the 'Has Error' column. The record is highlighted with a red box.

Environment	Has Error	Error severity	Error type
Prod	false		

If Has Error is false, then the app deployed successfully to the next environment.

*How can Jayne confirm if the app actually deployed?*

**⚠ CAUTION**

As part of the automated setup for your lab instance, Jayne was granted the role `sn_appclient.app_client_company_installer`.

This is a base Platform role that has existed for many years that allows a non-admin user to view apps in My Company Applications.

For more information see



## Product Docs: Instance-specific deployment user roles

17. Click All >> type 'My Company Applications' >> Click My Company Applications.

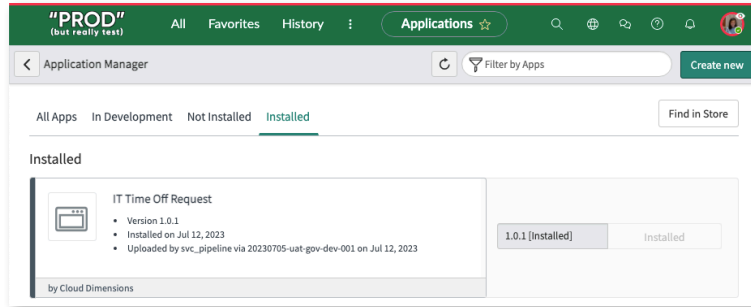
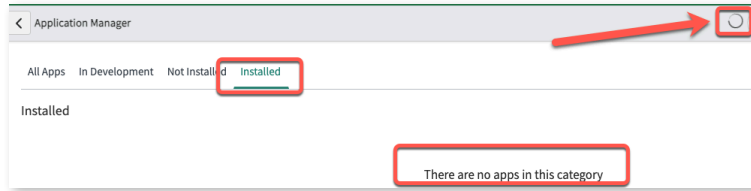


### CAUTION

My Company Applications shows you the applications that have been published to your company's App Repo as well as what is installed on your current instance.

For more information see [Product docs: ServiceNow application repository](#)

20. Click the Installed tab. *If the app does not appear in the list, then give it a few minutes to refresh. There is an icon in the top-right that will stop spinning when the page is done loading.*



## Lessons Learned

You have made it!! You deployed an app with the click of a button!

That completes the App Engine Governance Lab!

# Lab Complete

---

Got feedback?

Please consider taking the post-workshop survey, thanks!

[Post Lab Survey ↗](#)

Want to learn more?

Feel free to explore the Additional Content on the side of the page.

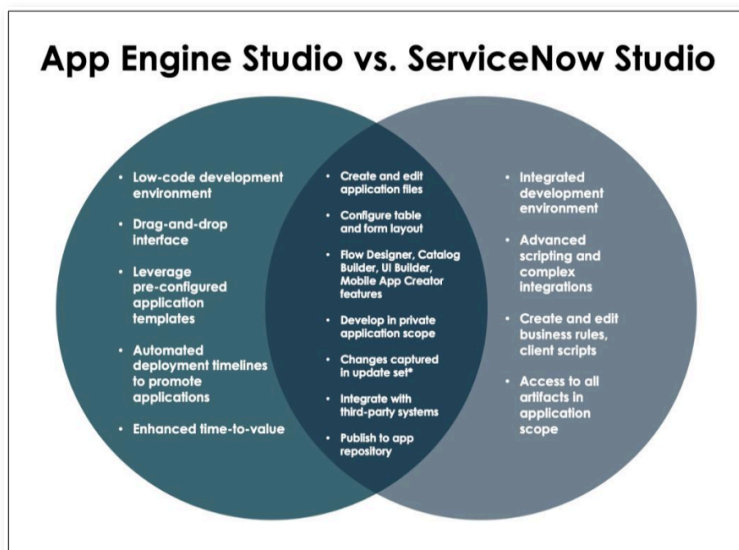
# App Engine Studio vs. ServiceNow Studio

## ⚠ CAUTION

This part of the lab is for information purposes only. No activity required.

**App Engine Studio** is the ideal ServiceNow development environment for no-code and low-code application development for developers of all skill levels.

Pro-code developers can use App Engine Studio as a starting point for application development and easily transfer their applications to **ServiceNow Studio** when advanced configurations are necessary.



Since development is supported between the two developer environments, low-code and pro-code developers can easily collaborate to deliver custom applications.

All configurations to an application, whether performed in **App Engine Studio** or **ServiceNow Studio**, can be captured in the same private application scope.

Application artifacts configured in **App Engine Studio** can also be managed in **ServiceNow Studio**.

# Cloning Tips

---

## How to Preserve Development Data During System Clone

[NowCreate: App Engine - Data Preservation During System Clone ↗](#)

An informational, authoritative white paper on different scenarios to consider when cloning Production to Subproduction environments in regards to preserving application development data.

# FAQs

---

## Are App Engine Studio applications built in a private application scope?

Yes, each application is developed in its own scope.

## Can applications be accessed from App Engine Studio to ServiceNow Studio, and vice-versa?

Yes. Applications built in App Engine Studio are accessible from ServiceNow Studio.

All application configurations are captured in the application scope, regardless of the development studio the configurations are performed in.

There are several capabilities available in ServiceNow Studio that are not accessible from App Engine Studio, including business rules, client scripts, and notifications.

## Can App Engine Studio applications be integrated with other applications outside of ServiceNow?

Yes, App Engine Studios can interact with third-party systems by leveraging IntegrationHub spokes / actions exposed through Flow Designer.

## Can I integrate App Engine Studio with Source Control?

Yes, application developers can integrate with a Git Source Control repository to save and manage multiple versions of an application from a sub-production instance.

 NOTE

For more information see

[Product Documentation:Source control operations in App Engine Studio](#)

## Can I turn off the automatic execution of ATF tests and Instance Scan definitions?

If the ATF system properties are not enabled in the testing instance as part of the guided setup, you will receive a warning during deployment, however the flow will continue.

## Can I change the instance where ATF tests and Instance Scan definitions are executed?

Yes, this can be accomplished by updating the 'Instance Type' value for an Environment record to 'Testing'. Alternatively, the decision table can be modified to make the decision for 'Instance Type = Staging' (for example) point to the Testing sub-flow.

## Does App Engine Studio support domain separation?

Domain separation is not supported for App Engine Studio. The domain field may exist on data tables, but there is no business logic to manage the data.

## How can I safely expose third-party integrations in App Engine Studio?

Consider pre-configuring authorized sub-flows for application developers to use when sharing data with third-party systems instead of allowing App Engine Studio developers to access IntegrationHub actions via Flow Designer.

This way, developers are not directly accessing any third-party systems, and data is shared and presented in a consistent manner across all applications built in App Engine Studio.



## What credential types does App Engine Studio support?

For Utah and previous versions, only Basic Auth is supported.

In Vancouver, OAuth credential support was added for Pipelines and Deployments.

For more information, please visit

[Product Docs: Configure OAuth credentials for use in Pipelines and Deployments](#) .

# Install Requirements

---

## How do I install App Engine Studio?

App Engine Management Center is only available on ServiceNow instances that have purchased stand-alone App Engine licensing.

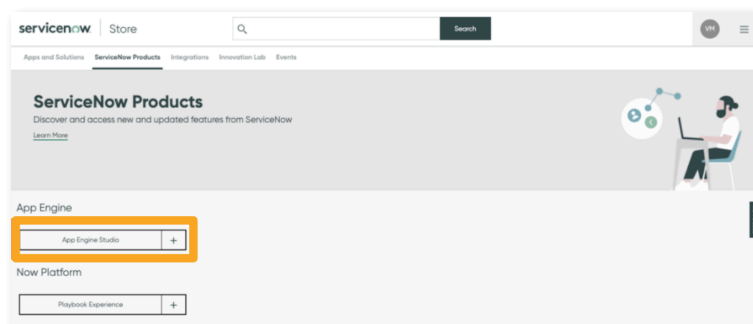
This includes both Production and Subproduction instances, but does not extend to Personal Developer Instances (PDIs).

Thus, this Lab is ideally suited for exploratory training and learning purposes.

## Opt-in and manage entitlements

Before installing the application from the ServiceNow Store, verify the instance has valid ServiceNow entitlements.

In the ServiceNow Store, use the search criteria to find App Engine Studio.



Click **Opt-in** and agree to the ServiceNow terms and conditions to verify entitlements.

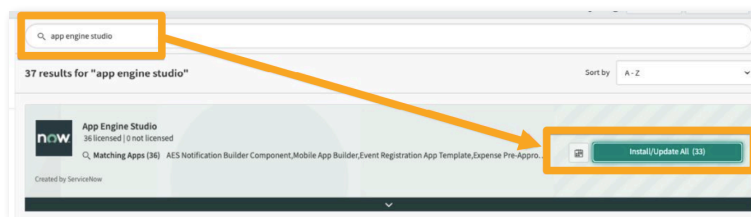
Click **Manage Entitlements** and set the 'Entitlement Type' value to **Entitle all Instances** (if you prefer to manually select which instances which will be affected, select **Entitle**

Selected Instances).

## Install the App Engine Studio application

To install the App Engine Studio application (*com.snc.app-engine-studio*), login to your development instance and navigate to System Applications >> All Available Applications >> All.

Use the search criteria to find the App Engine Studio application. Click Install / Update All.



The App Engine Studio bundle will be installed in the development instance – including the App Engine Studio application and all dependent applications.

Repeat this process on all instances for cloning purposes.

### **i** NOTE

For more information see

- [Product Documentation: Install App Engine Studio](#)
- [ServiceNow Store: Install a ServiceNow Product](#)

# App Engine Studio Optional Setup

## Tasks to Consider

---

 CAUTION

This part of the lab is for information purposes only. No activity required.

As part of your App Engine Studio setup, you will want to potentially explore and configure the following:

### Connect spokes in Dev Environment

 CAUTION

This step is not required as part of the initial App Engine Studio setup.

A spoke is a scoped application that contains Flow Designer or Integration Hub actions or Subflows.

System Administrators can connect IntegrationHub spokes to App Engine Studio, allowing developers to integrate custom applications with third-party systems. For example – connecting the Slack spoke allows developers to post a message containing Incident details to a specific Slack channel each time a high-priority Incident is created.

Many Integration Hub spokes are available, but not all need to be connected to App Engine Studio. Review some of the common spokes below and install them based on organizational needs.

Integration Hub Spoke	Installing this spoke provides developers
Microsoft Teams	Actions to support cross-functional communications and collaborations in Microsoft Teams.
Slack	Actions to automate the management of Slack channels, users, and software subscriptions.
Jira	Actions to automate tasks in Jira for managing users, issues, projects, and software development lifecycle. Synchronize data in ServiceNow with Jira to increase collaboration between ServiceNow users and DevOps teams.
Gmail	Actions to automate email and label management in Google Gmail.
Azure AD	Actions to automate Microsoft Azure Active Directory tasks for user management, authentication, and group membership.
Twitter	Actions to automate posting messages and media content to a corporate Twitter news feed.
MS 365 Excel	Actions to automate worksheet, table, and cell management in Microsoft Excel.

As your organization's citizen development program matures and scales, additional spokes can be installed and connected based on application demand and use cases.

 NOTE

For more information on integrations with third-party systems, see

[Product Documentation: Integration Hub](#)

## Review Flow Designer access settings

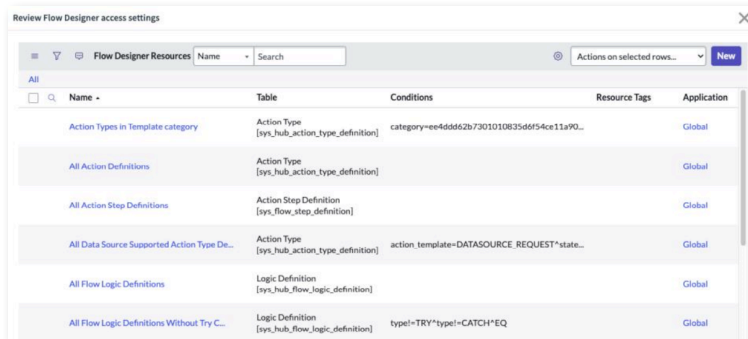
### CAUTION

This step is not required as part of the initial App Engine Studio setup.

Review and update App Engine Studio developer access settings in the development environment to Flow Designer Resources and update as necessary.

Developers can leverage Flow Designer capabilities while creating logic and automation for custom applications.

Consider restricting developer access to Flow Designer Resources using content filtering for Flow Designer. This allows administrators to manage access to Flow Designer resources and specify which features App Engine Studio users can leverage while building applications.



Name	Table	Conditions	Resource Tags	Application
Action Types in Template category	Action Type [sys_hub_action_type_definition]	category=ee4ddd62b7301010835d6f54ce11a90...	Global	
All Action Definitions	Action Type [sys_hub_action_type_definition]		Global	
All Action Step Definitions	Action Step Definition [sys_flow_step_definition]		Global	
All Data Source Supported Action Type De...	Action Type [sys_hub_action_type_definition]	action_template=DATASOURCE_REQUEST*state...	Global	
All Flow Logic Definitions	Logic Definition [sys_hub_flow_logic_definition]		Global	
All Flow Logic Definitions Without Try C...	Logic Definition [sys_hub_flow_logic_definition]	type!=TRY*type!=CATCH*EQ	Global	

Flow Designer access from App Engine Studio can be updated later to provide developers with the editing capabilities that best suit their experience and needs.

### NOTE

For more information on Flow Designer resources, see

[Product Documentation: Content Filtering for Flow Designer](#)

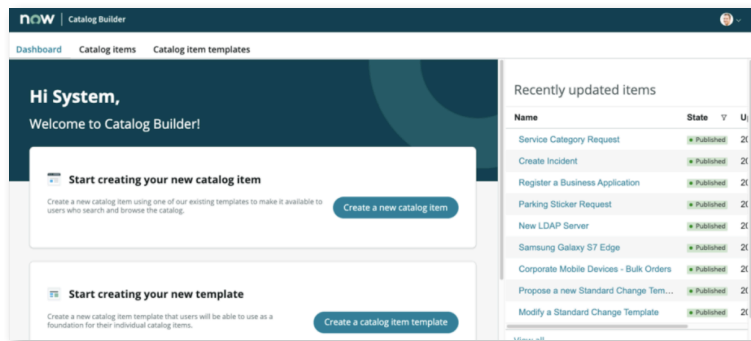
## Review Service Catalog access settings

### CAUTION

This step is not required as part of the initial App Engine Studio setup.

Review App Engine Studio developer access in the development environment to the Catalog Builder tool's catalog item templates and catalogs/categories and update access as necessary.

By default, App Engine Studio developers can leverage catalog templates to quickly create record producers or catalog items. Developers can also publish catalog items to any catalog. If you wish to limit access to templates or restrict publishing access to catalogs or categories, update the access accordingly in Catalog Builder.



Catalog access from App Engine Studio can be updated at a later point to modify developer access to App Engine Studio catalogs, categories, and catalog templates.

For more information on creating or editing catalog items, see

[Product Documentation: Catalog Builder](#)

## Configure Instance Scan

### CAUTION

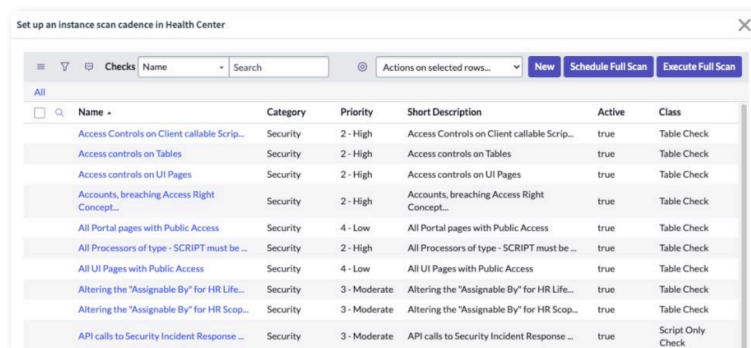
This step is not required as part of the initial App Engine Studio setup.

Deploy custom applications with confidence by setting up Instance Scan definitions to monitor instance health throughout the deployment process. Instance scans interrogate your instance for configurations and health issues, allowing administrators an opportunity to address best practices and avoid similar configuration issues in the future.

Instance Scan definitions are executed automatically when App Engine Studio applications are promoted to the **Testing** instance. Instance Scan results will be logged in the Notes section of the Deployment Request record.

The App Engine Studio application does not ship with any out-of-box Instance Scan definitions (however, a few Instance Scan definitions are installed with the Deployment Pipeline plugin to run basic performance checks).

Work with professional ServiceNow developers to configure Instance Scan definitions and enforce best practices in your environments.



Name	Category	Priority	Short Description	Active	Class
Access Controls on Client callable Scripts	Security	2 - High	Access Controls on Client callable Scripts	true	Table Check
Access controls on Tables	Security	2 - High	Access controls on Tables	true	Table Check
Access controls on UI Pages	Security	2 - High	Access controls on UI Pages	true	Table Check
Accounts, breaching Access Right Concept...	Security	2 - High	Accounts, breaching Access Right Concept...	true	Table Check
All Portal pages with Public Access	Security	4 - Low	All Portal pages with Public Access	true	Table Check
All Processors of type - SCRIPT must be ...	Security	2 - High	All Processors of type - SCRIPT must be ...	true	Table Check
All UI Pages with Public Access	Security	4 - Low	All UI Pages with Public Access	true	Table Check
Altering the "Assignable By" for HR Life...	Security	3 - Moderate	Altering the "Assignable By" for HR Life...	true	Table Check
Altering the "Assignable By" for HR Scop...	Security	3 - Moderate	Altering the "Assignable By" for HR Scop...	true	Table Check
API calls to Security Incident Response ...	Security	3 - Moderate	API calls to Security Incident Response ...	true	Script Only Check

Enable and configure Instance Scan properties in your production instance if you plan to clone!

**i** NOTE

For more information on managing instance health scans, see

[Product Documentation: Instance Scan](#)



# Personas and Roles

---

## CAUTION

This part of the lab is for information purposes only. No activity required.

## System Administrator

---

### Responsibilities

- Install and configure App Engine Studio and dependencies
- Define environments and configure pipelines
- Provision App Engine Studio administrator group access
- Upgrade the App Engine Studio application, when applicable
- Define guardrails for which App Engine Studio Administrators can approve or reject application intake requests
- Collaborate with professional ServiceNow developers to create Instance Scan definitions for the platform
- Collaborate with App Engine Studio Administrator(s) to resolve application conflicts that arise on the platform

### Role required:

- admin

# App Engine Studio Administrator

---

## Responsibilities

- Manage App Engine Studio processes and properties
- Review and approve / reject submitted application requests, based on intake guardrails defined by system administrator
- Provision App Engine Studio user access
- Review submitted App Engine Studio applications
- Manage deployment requests
- Manage promotion of App Engine Studio applications across instances
- Execute ATF tests / suites in testing instance
- Ensure Instance Scan is run with proper definitions
- Collaborate with system administrator to resolve application conflicts that arise on the platform

## Role(s) required:

- sn\_app\_eng\_studio.admin
- atf\_test\_designer
- scan\_admin

# Professional ServiceNow Developer

---

## Responsibilities

- Build and test applications in ServiceNow Studio and App Engine Studio
- Collaborate with citizen developers, on an as-needed basis, to deliver applications in App Engine Studio, including:

- Ensure application development and organizational best practices are followed by citizen developers
- Assist in review and testing of submitted App Engine Studio applications
- Build complex configurations involving UI Builder, Mobile App Builder, Flow Designer, or other builder tools
- Create ATF tests / suites for applications
- Collaborate with system administrator to create Instance Scan definitions for the platform

Role(s) required:

- delegated\_developer
- sn\_app\_eng\_studio.user
- atf\_test\_admin
- scan\_admin

## Low-Code / Citizen Developer

---

### Responsibilities

- Submit requests for new custom applications to build in App Engine Studio
- Understand ServiceNow and application development best practices
- Build and test applications in App Engine Studio
- Submit developed App Engine Studio applications for IT review
- Maintain and modify the application during its lifecycle if determined during the intake process

Role required:

- sn\_app\_eng\_studio.user



In the Utah release, the 'App Engine Studio Users Limited' group was added to allow AES administrators to give developers assigned to the group the ability to collaborate on existing applications without being able to create new applications or access templates.

# Prepare Environments

---

Before installing App Engine Studio, review all platform requirements and define an organizational instance strategy to prepare for successful installation and configuration.

## Review platform requirements

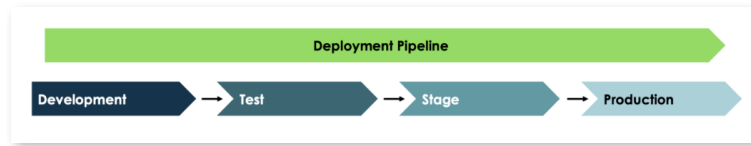
- App Engine license required
  - *Contact your ServiceNow Account Manager for additional information on App Engine, or see [ServiceNow App Engine](#)*
- Instances must be on the same release (Ex. Utah)
- The admin role is required in all instances to install App Engine Studio and dependent applications from the ServiceNow Store

## Define instance strategy

When defining the instance strategy for App Engine Studio, it is recommended to leverage one production instance and at least two sub-production instances – however, App Engine Studio can support any number of sub-production instances as part of an instance strategy.

Applications are deployed to the production instance once developed and successfully tested in sub-production instances. One sub-production instance will serve as the development environment and the other as the test environment.

If your organization uses sandbox or staging environments in addition to test and development, they can be incorporated into the instance strategy accordingly based on organizational needs.



*Example instance strategy with one production instance and three sub-production instances*

Sub-production instances that are most similarly configured to your production instance are the best candidates for test and stage environments. This way, administrators can more accurately find issues that may arise if the application is deployed to production.

**i** NOTE

For more information see

[Product Docs: Instance Strategy for App Engine Studio](#)

# Product Documentation

---

[Docs: Building applications ↗](#)

Learn how to become an application developer using Now Platform low-code tools. Start with what you know and use a library of reusable components and published applications to modernize your legacy processes.

- [Learning about developing on the ServiceNow Now Platform ↗](#)
- [Planning your application ↗](#)
- [Developing your application ↗](#)
- [Testing and debugging applications ↗](#)
- [Deploying applications ↗](#)
- [Maintaining your application ↗](#)

[Docs: Application Repository ↗](#)

After you develop and test a custom application, you can make the application available to company instances by publishing it to the ServiceNow application repository...

# App Engine System Administrator Guide

---

[NowCreate: App Engine Studio - System Administrator Guide on ↗](#)

41-page guide that provides system administrators guidance and insights in the installation, configuration, and management of the App Engine Studio (AES) application and processes.

It also contains information about cloning configurations and considerations for App Engine.

Please consider taking the post-workshop survey, thanks!

[Post Lab Survey ↗](#)



# Download Lab Guide PDF

---

[Click here to download Lab Guide PDF](#)