

Lab Guide: Introduction to AI Governance in watsonx

Author: Elena Lowery, elowery@us.ibm.com



Contents

Overview 3

Required software, access, and files 3

AI Governance and watsonx.governance..... 5

 Watsonx.governance..... 6

Part 1: Create a Model Inventory..... 7

Part 2: Prompt Development and Evaluation 14

Conclusion 43

Overview

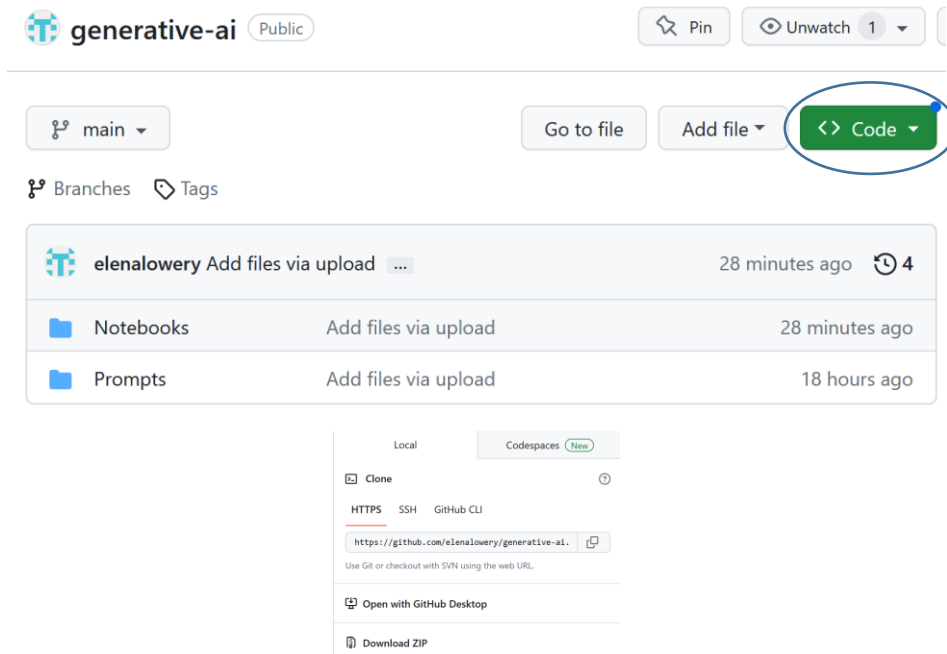
In this lab you will learn how to track and evaluate large language model (LLM) prompts in *watsonx.governance*. We will complete the following tasks:

- Create a model inventory
- Create a use case
- Test and evaluate a prompt
- Promote the prompt to production
- Use the prompt from a client application.

Required software, access, and files

Note: if you have completed these steps in the previous labs, you don't need to repeat them.

- To complete this lab, you will need access to *watsonx.ai* and *watsonx.governance*. You can get access by signing up for an [IBM Cloud account](#) and provisioning *watsonx.ai* and *watsonx.governance* services.
- Recommended, but not required: A Python IDE with Python 3.10 or 3.11 environment (*Visual Studio Code* or *PyCharm*). If you don't have the Python IDE, you can test integration with LLMs in *watsonx.ai* Notebooks.
- You will also need to download and unzip this GitHub repository:
<https://github.com/elenalowery/generative-ai>
- Click the **Code** button and select **Download ZIP**





Unzip the downloaded zip file. In the lab, we will refer to this folder as the *git repo* folder.

Important note: Some screenshots in the lab may be slightly different from the product. If you have questions, please ask your workshop instructor.

AI Governance and watsonx.governance

AI Governance is a discipline that includes business process and implementation of best practices for *creating, deploying, managing, and monitoring* AI artifacts, such as Machine Learning (ML) models and Large Language Models (LLMs).

In the past, components of *AI Governance* were implemented by companies in highly regulated industries or for specific use cases. For example, companies in the financial industry must have Model Risk Governance (MRG) practice to comply with regulatory requirements that were first [published](#) in 2012, and even as early as 2000. However, in some cases, the implementation of AI Governance was simply keeping track of models in a spreadsheet or a document.

During the past few years, various government organizations proposed and approved laws for governing AI. Here is an example of a law related to the use of AI:

New York

Enacted

In December 2021, New York City passed the first law ([Local Law 144](#)), in the United States [requiring employers to conduct bias audits of AI-enabled tools used for employment decisions](#). The law imposes notice and reporting obligations.

Specifically, employers who utilize automated employment decision tools (AEDTs) must:

1. [Subject AEDTs to a bias audit](#), conducted by an independent auditor, within one year of their use;
2. Ensure that the date of the most recent bias audit and a "summary of the results", along with the distribution date of the AEDT, are publicly available on the career or jobs section of the employer's or employee agency's website;
3. Provide each resident of NYC who has applied for a position (internal or external) with a notice that discloses that their application will be subject to an automated tool, identifies the specific job qualifications and characteristics that the tool will use in making its assessment, and informs candidates of their right to request an alternative selection process or accommodation (the notice shall be issued on an individual basis at least 10 business days before the use of a tool); and
4. Allow candidates or employees to request alternative evaluation processes as an accommodation.

[Source](#)

Regulatory landscape for AI continues to evolve, with important regulations such as the United States [AI Bill of Rights](#) and [EU AI Act](#) will likely accelerate development of guidelines and laws.

One of the best ways to prepare for upcoming regulations is to infuse AI Governance in several steps of the ML model and LLM lifecycle. *watsonx* includes multiple features that

will streamline not only the governance of models but also improve operationalization and simplify maintenance.

In this lab, we will focus on the *operational* aspects of implementing AI Governance. We will cover *Model Risk Governance (MRG)* and the automatic integration of MRG with the model lifecycle in a separate lab.

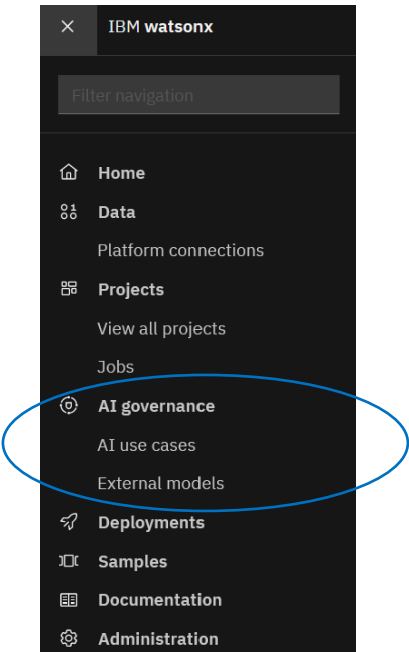
Watsonx.governance

Many software offerings today are packaged as individual services that include a set of features. Services provide granular capabilities that can be included in multiple offerings. IBM's *watsonx.governance* includes two services:

- **AI Factsheets:** used to capture model and prompt metadata as well as track model lifecycle
- **OpenScale:** used to monitor model performance, accuracy, provide explainability, and alert for bias.

It is common in microservice software design to have prerequisite services. In the case of *watsonx.governance*, the prerequisite services are *watsonx.ai* and *Watson Machine Learning*.

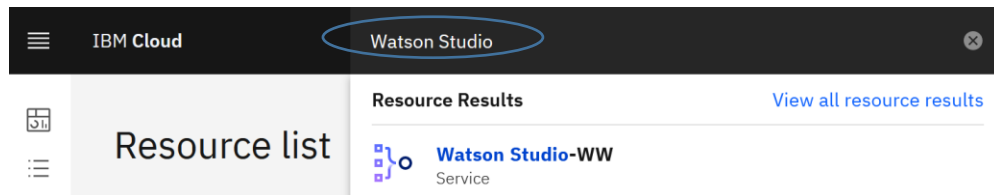
From the end-user perspective, all services are seamlessly integrated into *watsonx* UI. As additional services are provisioned, new actions (menus/buttons) become available in various components of *watsonx*. In other words, end users don't need to know the names of the services, but this information may be useful to administrators. For example, if a user doesn't see a certain action, the administrator can check if the required service was provisioned.



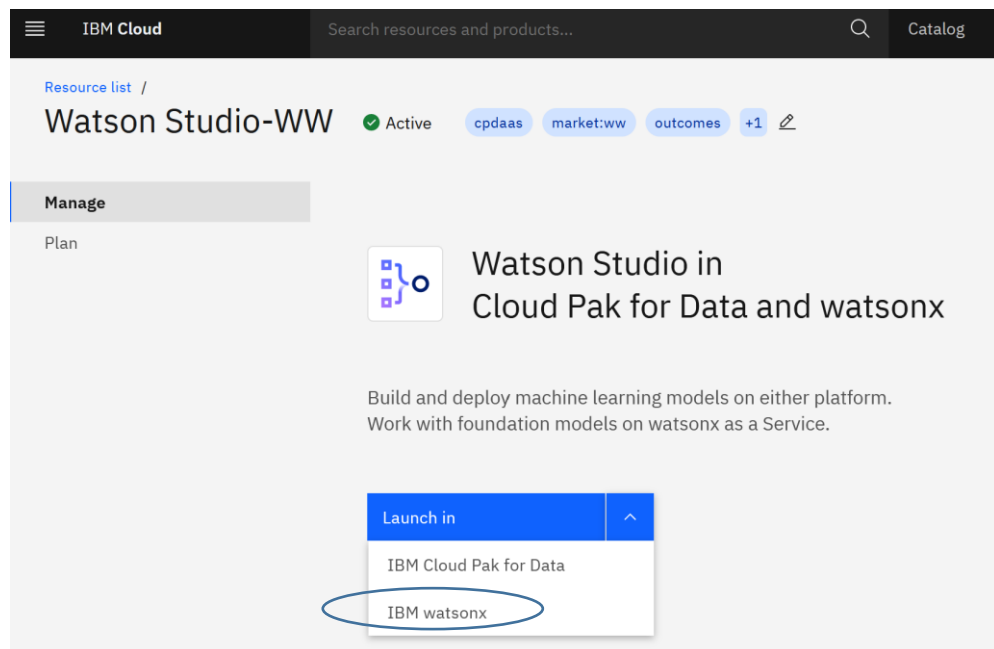
Part 1: Create a Model Inventory

Creating a *Model Inventory* is the first step in better organization of AI artifacts.

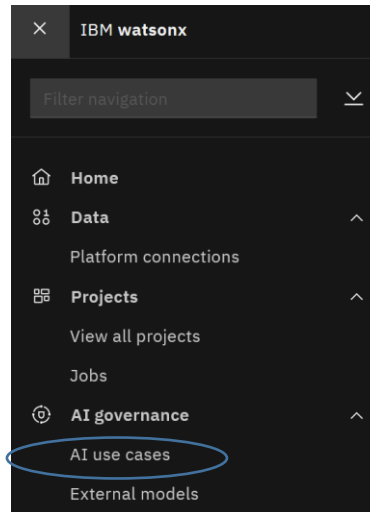
1. Log in to your **IBM Cloud account**. Your lab instructor will provide the URL and userid/password.
2. If you're not already in watsonx.ai service, you can navigate to it from the **IBM Cloud Resources** page.
 - Search for *Watson Studio* in the search box on top of the page.



- On the *Watson Studio* service page, make sure to select *Launch in IBM watsonx*.

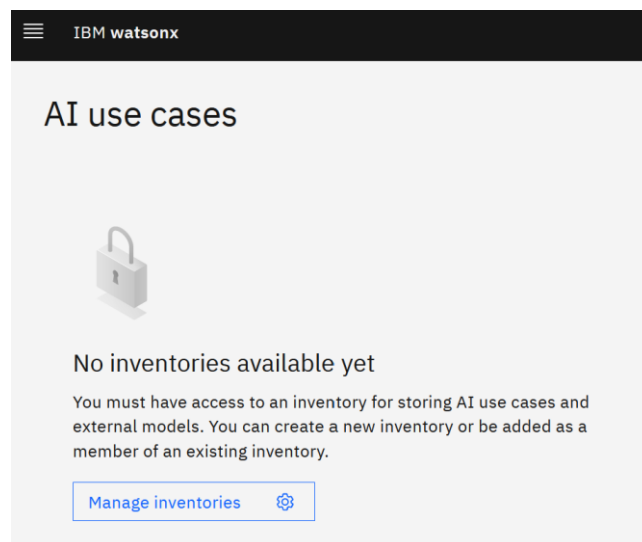


3. From the main menu in the top left corner select **AI Use Cases**



Note: If you don't see this menu, check that OpenScale service was provisioned in your IBM Cloud account.

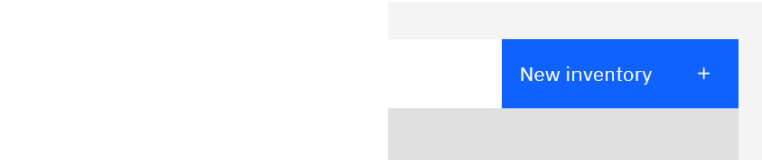
4. If you have not previously created a Model Inventory, you will see an option to create it. Click **Manage Inventories**.



5. Click on the **New Inventory** button, then provide a unique name for your inventory, for example, *LLM Insurance Use Cases <your initials>*. Make sure to select *Cloud Object Storage* associated with your account id.

Click **Create**.

Optionally, add collaborators and close the screen.



New Inventory

Name

LLM Insurance Use Cases EL

Description (optional)

Inventory for tracking LLM use cases for the insurance industry

☒ Add collaborators after creation

An inventory without collaborators will be visible to only you. Add members to collaborate on AI use cases.

IBM Cloud Object Storage

This service stores the files associated with assets in the inventory.

Object storage instance

Cloud Object Storage-ur

Or [create a new Cloud Object Storage instance](#)

Your inventory will look similar to the following screenshot:

Inventories

Add or modify an inventory to manage access and storage for a collection of AI use cases and external models.

Find an inventory

Name	Date created	Creator	Your role
LLM Insurance Use Cases EL	1 minute ago	<div>EL</div> Elena Lowery	Admin

Next, we will create an AI use case.

6. Navigate to the **AI Use case** menu again and click the **New AI use case** button.

AI use cases

Find a AI use case

New AI use case

Name	Status	Owner	Inventory	Tags	Risk level	Alerts in
------	--------	-------	-----------	------	------------	-----------

7. Review metadata options available on this screen.

First, we provide use case information, such as *name* and *description*, as well as *risk level* and *inventory name*. Risk level designation can be based on industry regulations or an organization’s evaluation of risk. It can also help with prioritization of work if multiple models perform below specified thresholds.

New AI use case

Create a use case to define a business problem, request a model, and specify details such as risk level and status.

General information

Name (Required)

Description

An LLM will be used to extract entities (car model, location, time) from a text description of an incident.

Risk level

Medium

Custom risk label

Inventory (Required)

LLM Insurance Use Cases EL (Elena Lowery's Account)

Details

On the right side, we can capture some aspects of model lifecycle process, specifically the status, which we decided to set to *Development in Progress*. Adding tags to the use case entry will help us easily find use cases in the inventory.

Owner

Elena Lowery

elowery@us.ibm.com

Status

Development in progress

Tags

Search tags

llm entity extraction

Create a use case record using the sample values in the screenshot or provide your own.

- Name: *Insurance Claim Entity Extraction <Your Initials>*
- Risk level: *Medium*
- Status: *Development in progress*

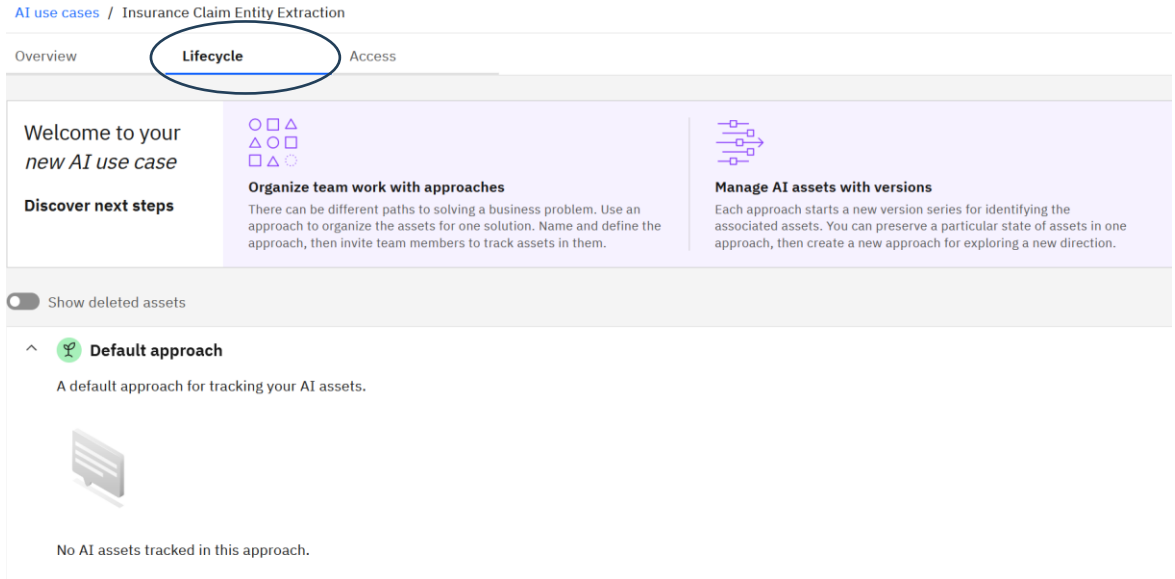
The detailed *Use Case* view provides the summary of details we entered.

The screenshot shows the IBM Watsonx interface for managing AI use cases. The top navigation bar includes 'Overview', 'Lifecycle', and 'Access' tabs. The main content area displays the 'Overview' tab for a use case named 'Insurance Claim Entity Extraction'. It includes a 'Welcome to your new AI use case' message, a 'Discover next steps' section, and a 'Set your AI use case status' section. The 'General information' section lists details such as Name, Description, Owner, Status, Risk level, Inventory, and Tags. The 'Details' section is also visible at the bottom.

Notice that we can add different types of supporting documentation to the use case, which is useful for both transparency and auditing of AI use case implementation.

The screenshot shows the 'Details' section of the IBM Watsonx interface. It includes a 'Purpose' field with a placeholder text 'Please enter business purpose of the requested model'. There is a 'Supporting documentation' field with a placeholder text 'Please enter a URL pointing to external documentation'. Below these fields is a 'Risk level' field with a placeholder text 'Enter Risk level'. At the bottom, there is an 'Attachments' section with a placeholder text 'Add attachments, such as images, that support the AI use case.' and a button labeled 'Add attachments +'.

If you click on the *Lifecycle* tab, you will notice that we don't have any information yet because we have not added project assets that we want to track.



8. Create a second use case entry for the *Summarization* use case. You can use the values in the following screenshot or provide your own.

- Name: *Insurance Claim Summarization <Your Initials>*
- Risk level: *Low*
- Status: *Development in progress*

New AI use case

Create a use case to define a business problem, request a model, and specify details such as risk level and status.

General information

Name (Required)

Insurance Claim Summarization

Description

Summarization of an submitted insurance claim using an LLM

Risk level

Low

Custom risk label

Inventory (Required)

LLM Insurance Use Cases EL (Elena Lowery's Account)

When finished, you AI use case view should look similar to this screenshot.

AI use cases

Find a AI use case

New AI use case +

Name	Status	Owner	Inventory	Tags	Risk level	Alerts in
Insurance Claim Summarization	Development in progress	<div>EL</div> Elena Lowery	LLM Insurance Use Cases EL	llm summarization	<div></div> Low	<div></div> none
Insurance Claim Entity Extraction	Development in progress	<div>EL</div> Elena Lowery	LLM Insurance Use Cases EL	llm entity extraction	<div></div> Medium	<div></div> none

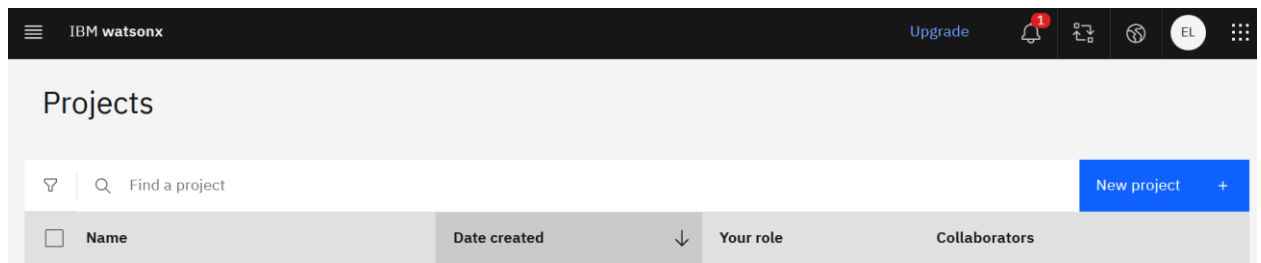
We have completed setting up a *Model Inventory* and creating metadata for AI use cases that we will implement with LLMs. In just a few minutes, we provided a better organization for auditing, tracking, and monitoring AI assets that will be used in various applications.

Part 2: Prompt Development and Evaluation

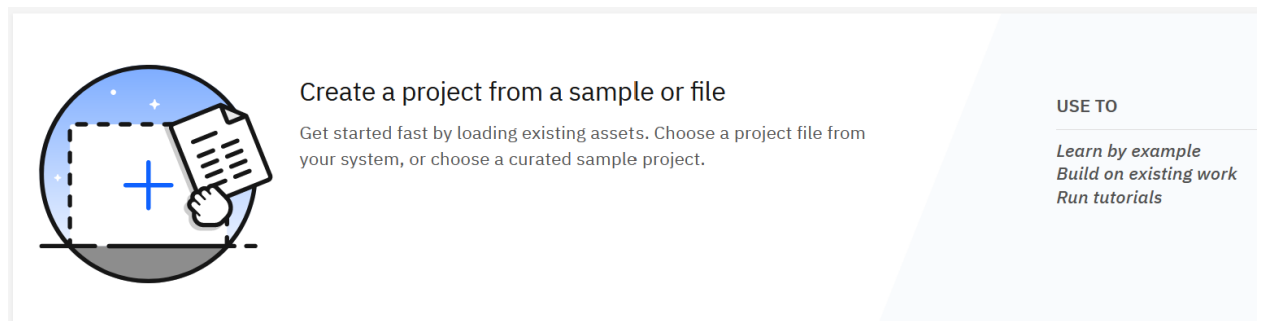
In this section, we will use provided prompts for the defined use cases to walk through the lifecycle of deploying a prompt/LLM for use by production applications. We will not cover the iterative process of prompt engineering and prompt tuning. Since we are focusing on governance, we are making the assumption that the development process for the prompts has been completed.

First, we will import the project that contains prompts into watsonx.

1. In *watsonx* navigate to the **Projects** view (from the main menu in the top left corner) and click the **New Project** button.



2. Click the **From file** option and navigate to the downloaded lab repo */Cloud Projects* folder to select *Insurance_LLM_Use_Cases.zip*.



Provide a unique project name and click **Create**.

IBM watsonx

Upgrade

1

Create a project from sample or file

From a file

From sample

Select file ✓

Choose a .ZIP file that contains an exported Watson Studio project.

Insurance_LLM_Use_Cases.zip ✕

Name

Insurance LLM Use Cases EL

Description (optional)

A collection of prompts, data, and notebooks for insurance use cases implemented with LLMs

Controls

Cancel

Back

Create

Storage

Project includes integration with [Cloud Object Storage](#) for storing project assets.

Cloud Object Storage-ur

- Switch to the **Manage** tab, then select **Services and Integrations** tab. Click **Associate Service**.

IBM watsonx

Upgrade

1

EL

Projects / Insurance LLM Use Cases EL

▼

Launch IDE ▼

Overview

Assets

Jobs

Manage

Project

General

Access control

Environments

Resource usage

Services & integrations

Services & integrations

IBM services

Third-party integrations

Find services

Associate service +

<input type="checkbox"/>	Name	Service type

- Select the displayed **Machine Learning** service and click **Associate**.

Projects / Insurance LLM Use Cases EL

↕

⌵

👤

Launch IDE ⌵

ⓘ

🔗

🕒

💬

8

Overview

Assets

Jobs

Manage

Project

⚙️ General

🔑 Access control

🌐 Environments

📊 Resource usage

⚙️ Services & integrations

Services & integrations

IBM services (1)

Third-party integrations

🔍 Find services

Associate service +

<input type="checkbox"/> Name	Service type
<input type="checkbox"/> Machine Learning-fz	Watson Machine Learning

5. Switch to the **Assets** tab and expand **Prompts**.

Projects / Insurance LLM Use Cases EL

↕

⌵

👤

Launch IDE ⌵

ⓘ

🔗

🕒

💬

8

Overview

Assets

Jobs

Manage

🔍 Find assets

7 assets

📁 All assets

Asset types

> 📁 Data 4

> 📁 Prompts 3

Prompts

Name	Last modified
<div>🔗</div> <div>Insurance claim suggested next steps</div> <div>Prompt template</div>	<div>2 minutes ago</div> <div>Modified by Service</div>
<div>🔗</div> <div>Insurance claim key information extraction</div> <div>Prompt template</div>	<div>2 minutes ago</div> <div>Modified by Service</div>
<div>🔗</div> <div>Insurance claim summarization</div> <div>Prompt template</div>	<div>2 minutes ago</div> <div>Modified by Service</div>

Let's review prompts for our AI use cases.

6. Click on the *Insurance claim key information extraction* prompt. We are making an assumption that you are familiar with prompt structure, but let's point out a few things that may be unique to prompt format in the *Prompt Lab*.

Note: when prompted, check all the boxes to agree to terms and click *skip tour*.

Welcome to Prompt Lab

By using any foundation model provided with this Cloud Service, you acknowledge and understand that:

- ☒ Some models included in the Cloud Service are Non-IBM Products. Review the applicable model details on the third party provider and license terms that apply.
- ☒ Models may generate outputs that contain misinformation, obscene or offensive language, or discriminatory content. Client should review the outputs for such information or content prior to re-use. Users should review and validate the outputs generated.
- ☒ The output generated by all models is provided to augment, not replace, human decision-making by the Client.

[Skip tour](#)
[Start tour](#)

When you open the prompt, you can choose to open in preview mode or edit mode. Edit mode will lock the prompt and allow you to make changes. Choose *Edit* for now.

Edit this prompt template?

If you open this prompt template in edit mode, you lock it for other users. You can preview the prompt template without locking it.

☐ Don't show this message again.

[Go to project](#)
[Preview](#)
[Edit](#)

The prompt is displayed in either *Structured* or *Freeform* view. Switch to the *Freeform* view.

Insurance claim key information extraction

[Evaluate](#)
[New prompt +](#)
[Save work](#)

Prompt: Autosaved 1:26 PM

Structured Freeform

Model: flan-ul2-20b (x) </> ⌵

Read this Insurance claim description and extract the Car make and model, Location of the incident like street and date time if there is any mentioned. If you don't find these details in the description, please fill it as Not Found.

A car accident occurred on Jan 1st, 2023 at 5pm at the intersection of woodbridge. The insured vehicle, a Honda Civic, was hit by another vehicle that ran a red light. The insured driver, John, was driving within the speed limit and following all traffic rules. The accident resulted in significant damage to the insured vehicle, including a broken bumper and damaged front fender. There were no injuries reported. The insured is filing a claim for the repairs and any necessary medical expenses.
Car Details: Honda Civic;Location: Woodbridge;Date: Jan 1st, 2023;Time of Incident: 5pm

The insured vehicle, a Ford RAM, was stolen from Boston on Dec 2nd 2022. The vehicle was parked in a secure parking lot, and all necessary precautions were taken, such as locking the doors and activating the alarm system. The insured immediately reported the theft to the police and obtained a police report. The vehicle had comprehensive insurance coverage, and the insured is filing a claim for the stolen vehicle, including its estimated value, accessories, and personal belongings that were inside the vehicle at the time of theft.
Car Details: Ford RAM;Location: Boston;Date and time: Dec 2nd 2022;Time of Incident: Not Found

The insured vehicle, a Tesla model X, was vandalized on march 23rd while parked in front of the insured's residence on Magador Street. The vandalism included scratched paint, broken windows, and damage to the side mirrors. The insured promptly reported the theft to the police and obtained a police report. The insured is filing a claim for the repairs and any necessary replacement parts. The estimated cost of repairs has been assessed by a reputable auto repair shop.
Car Details: Tesla Model X;Location: Magador Street;Date: march 23rd;Time of Incident: Not Found

The insured vehicle, was parked outside during a severe hailstorm. As a result, the vehicle suffered extensive hail damage, including dents on the roof, hood, and trunk. The insured promptly reported the incident and is filing a claim for the necessary repairs. The estimated cost of repairs has been assessed by an authorized auto repair shop.
Car Details: Not Found;Location: Parked outside;Date: Not Found;Time of Incident: Not Found

While driving on Anthony Street on 1st June, the insured vehicle, a BMW Q1, collided with a large animal (e.g., deer) that suddenly crossed the road. The accident resulted in damage to the front bumper, grille, and headlights. The insured promptly reported the incident and is filing a claim for the repairs. Additionally, the insured sought medical attention for any potential injuries resulting from the collision.
Car Details: BMW Q1;Location: Anthony Street;Date: 1st June;Time of Incident: Not Found

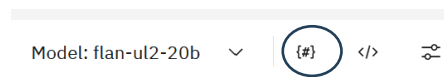
The insured vehicle, caught fire on april 1st due to a mechanical malfunction. The fire resulted in significant damage to the vehicle, including damage to the engine, interior, and exterior. The insured immediately contacted the fire department, and the incident was reported to the police. The insured is filing a claim for the repairs and is providing the fire department report as evidence of the fire incident.
Car Details: Not Found;Location: Not Found;Date: April 1st, 2023;Time of Incident: Not Found

In the beginning of the prompt, we see the instruction for the LLM: *Read this Insurance claim description and extract the Car make and model, Location of the incident like street and date time if there is any mentioned. If you don't find these details in the description, please fill it as Not Found.*

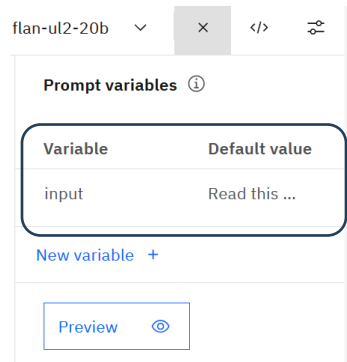
After that we have a few examples with text and extracted entities, which means that this is a *few-shot prompt*.

At the end of the prompt we have an `{input}` variable. This variable will be passed in during inference – it will contain the text (claim summary) from which the LLM will extract entities.

In the top right corner of the *Prompt Lab* we see that this prompt was created for the *flan-ul2-20b* model.



Next to the model we have the parameters button, where we can review and add parameters for the prompt. Notice that our parameter name is *input* (referenced in the prompt as `{input}`)



Finally, we can view model parameters by clicking on the *Parameters* icon.

an-ul2-20b
▼
{#}
</>
×

Model parameters

Decoding

Greedy ☒ Sampling ⓘ

Repetition penalty

1
●
2
1

Stopping criteria ⓘ

Stop sequences

+

Min tokens
Max tokens

0
200

Enter up to 6 sequences to stop output after the minimum number of tokens is reached.

If you wish, test the prompt by clicking the **Generate** button. We get the results because the `{input}` variable has a default value (claim text).

`{input}`Car Details: Tesla Model S; Location: Parked outside; Date: April 15th, 2023

- Return to the project view by clicking on the project name on the top of the screen.

Projects / Insurance_LLM_Use_Cases / Insurance claim key information ex [...]]

- Expand **Prompts** on the **Assets** tab. Click on the vertical ellipses menu and select **View AI factsheet**.

Projects / Insurance_LLM_Use_Cases

Overview

Assets

Jobs

Manage

Find assets

Import assets

New asset +

3 assets

All assets

Asset types

Prompts 3

Name	Last modified	
Insurance claim key information extraction Prompt template	10 minutes ago Modified by you	⋮
Insurance claim suggested next steps Prompt template	19 minutes ago Modified by Service	⋮
Insurance claim summarization Prompt template	19 minutes ago Modified by Service	⋮

Evaluate

View AI factsheet

Promote to space View AI factsheet

Delete

Notice that the factsheet captures the details (prompt, model parameters) that you have used in the **Prompt Lab**. At this time, if you make a change in the prompt template using the **Prompt Lab**, it will be automatically reflected in the factsheet.

If you wish, in the Prompt Lab change the *max number of tokens* model parameter and verify that the changes are reflected in the factsheet.

Notice the comment on top of the factsheet: we can make changes to the prompt template until we start tracking it.

Governance

This prompt template is not tracked.

To track a prompt template, add it to an AI use case. Tracking captures details about the asset for governance purposes.

Important:

Once you start tracking a prompt template in a use case, you can no longer edit it. Wait until the prompt template is stable to start tracking.

Track in AI use case

Since we made the assumption that our template is final, let’s start tracking.

- Click the **Track in AI use** case button and select the entity extraction use case you created earlier.

Track in AI use case

Track an asset to collect details about the asset in factsheets as part of a governance strategy.

Define AI use case

Define approach

Assign model versions

Define AI use case

Choose an existing AI use case or create a new one for tracking facts about an asset

Title	Inventory
<div>▼</div> <div>○</div> <div>Insurance Claim Summarization</div>	LLM Insurance Use Cases EL
<div>▼</div> <div>●</div> <div>Insurance Claim Entity Extraction</div>	LLM Insurance Use Cases EL

Click **Next** on **Define Approach** screen, then select **Stable** on the **Assign model version** screen. Click **Next**.

Note: we will review approaches later in the lab.

Track in AI use case

Track an asset to collect details about the asset in factsheets as part of a governance strategy.

Define AI use case

Define approach

Assign model versions

Assign model version

Approach: **Default approach** | Use case: **Insurance Claim Entity Extraction**

Choose the starting point for this approach.

🧪

Experimental

Use this as a starting point if your model is just starting in development and its input and output structure will likely change in the near future.

0.0.1

📌

Stable

Use this as a starting point if your model is in a production state and you won't expect any major changes in its input and output structure soon.

1.0.0

🔧

Custom

Define your own starting version if you already tracked this model in a versioning context before.

Version number

✖

Comment (optional)

Click the **Track asset** button to finish this task.

Typically, a data scientist will perform testing prior to releasing the template for production. The *evaluation* process that we will complete in the next few steps is performed primarily for the purpose of documenting test results.

Evaluation metrics provided in watsonx are standard evaluation metrics that are used for various LLM tasks such as classification, summarization, and extractions. You can learn more about evaluation metrics in [documentation](#).

Evaluation of LLMs requires “reference data”, which is the “expected output” for the LLM. Reference data can be created manually or generated using various approaches. For generation use cases such as summarization and content generation, it is important to provide high quality reference data, which means that it may need to be created by experts.

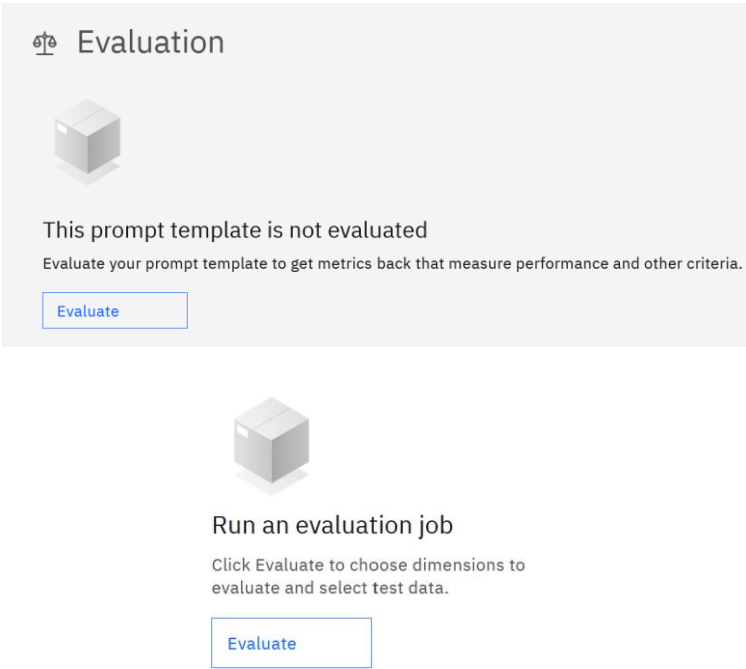
Example of reference data for information extraction use case:

Input Data	Reference Data
A car accident occurred on Jan 1st, 2023 at 5pm at the intersection of woodbridge. The insured vehicle, a Honda Civic, was hit by another vehicle that ran a red light. The insured driver, John, was driving within the speed limit and following all traffic rules. The accident resulted in significant damage to the insured vehicle, including a broken bumper and damaged front fender. There were no injuries reported. The insured is filing a claim for the repairs and any necessary medical expenses.	Car Details: <i>Honda Civic</i> ; Location: <i>Woodbridge</i> ; Date: <i>Jan 1st, 2023</i> ; Time of Incident: <i>5pm</i>

You can find reference data for our use cases in the *lab repo/Test Data* folder.

Evaluation can be done either from the Factsheet view (bottom of the page) or the ellipses menu next to the prompt in the project view.

- Click **Evaluate** in the factsheet, then **Evaluate** on the **Run an evaluation job** screen.



Notice that evaluation metrics relevant to our use case, *extraction*, were automatically selected.

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Select dimensions to evaluate

These dimensions are based on the prompt template task type. [Learn more](#)

☒ Dimension
 Description

☒ Generative AI Quality
 The Generative AI Quality monitor calculates a variety of metrics based on prompt template task type. Some metrics compare model output to the reference output you provide. Other metrics analyze model input and output and do not require reference output.

ROUGE

ROUGE, or Recall-Oriented Understudy for Gisting Evaluation, is a set of metrics and a software package used for evaluating automatic summarization and machine translation software in natural language processing. Generative text quality monitor calculates rouge1, rouge2, rougeL, and rougeLSum to compare an automatically produced summary or translation against a reference or a set of references (human-produced) summary or translation. The metric values are in the range of 0 to 1, higher value is better.

Multi-label/class metrics

The Multi-label/class metrics calculate a list of quality metrics on each class.

Exact match

A given predicted string's exact match score is 1 if it is the exact same as its reference string, and is 0 otherwise.

Advanced settings

Cancel

Back

Next

As a reminder, the task type is captured in the factsheet.

Note: The task type is specified when we save a prompt. Since we imported a prompt, we did not need to complete this step.

AI Factsheet

Evaluate

Governance

Foundation model

Prompt template

Prompt parameters

Evaluation

Attachments

Other attachments

[...] Prompt template

Prompt template name

Insurance claim key information extraction

e5e02a45-c0f5-4299-99eb-aa26285d4b11

Task type

Extraction

Click **Next** on the *Evaluate Prompt Template* screen.

- On the select data screen, navigate to the *lab repo/Test Data* folder and select the *text_extraction_claims.csv* file.

If you wish, open this file to review its content. As we discussed earlier, this file provides “expected LLM output” for the instruction that we are asking it to perform.

Select *Claims text* and the input and *Extracted Key Facts* as reference output.

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Map prompt variables to columns

For each prompt variable, select the associated column. [Learn more](#)

Field separation

Select delimiter

Comma (,)

Input

input

Claims text

Reference output

Reference output

Extracted Key Facts

Click **Next**.

12. Click **Evaluate** to run evaluation.

Projects

Insurance LLM Use Cases EL

Insurance claim key information ex

Open in Prompt Lab

AI Factsheet

Evaluate

Deployment details

Test data set

test_extraction_claims.csv

Test details

1

Tests run

Tests passed

1

Tests failed

0

Model health

Records

16

Records

Token count

15,578

Total input token count

Latency (record)

2,541

ms Median record latency

466

Total output token count

Generative AI Quality - Entity extraction

No alerts

Alerts

0

Feedback

Metric	Score	Violation
ROUGE-1	0.97	none
ROUGE-2	0.96	none

Notice that the test result is *passed*, which means that the calculated evaluation metrics are above specified thresholds. The *passed/failed* metric helps data scientists and auditors make sure that only models that meet accuracy requirements are deployed into production.

To view the thresholds, click on the **Configuration** icon.

Compare the *rouge* metric thresholds defined on this screen with the values from our test run.

Generative AI Quality - Entity extraction

No alerts

Alerts
 0

Metric	Score	Violation
^ Rouge		
ROUGE-1	0.97	none
ROUGE-2	0.96	none
ROUGE-L	0.97	none
ROUGE-Lsum	0.97	none

ROUGE

Lower thresholds

ROUGE-1

0.8

ROUGE-2

0.8

ROUGE-L

0.8

ROUGE-Lsum

0.8

Our test is marked as passed because all values are above the thresholds.

Next, we will create another prompt template to understand prompt format that is required for *watsonx.governance* implementation.

13. Locate the sample prompt *Extract_info_insurance_claim_llama* in the *lab repo/Prompts* folder.
14. Navigate to your project and create a new prompt. In the **Prompt Lab** use the *Freeform* view to paste the sample prompt.

Make sure to select the *llama-2-70b-chat* model and set max tokens to 200.

Projects / AI gov lab / Prompt Lab

Unsaved [icon] New prompt + [icon] AI guardrails on [icon]

Structured Freeform

Model: llama-2-70b-chat [icon] [icon] [icon]

[[INST]] [[SYS]]

You are a model that extracts entities from insurance claims. You specialize in finding car make and model, location, date and time of an incident.

Read the description below and extract the car make and model, location of the incident like street and date time if there is any mentioned. If you don't find these details in the description, please fill it as Not Found. Format extracted values as a list separated by semicolon. Example: Car Details: car make and model; location; date and time. Always start response with "Car Details:" Do not include any other information. The following is an example:

Description: While driving on Anthony Street on 1st June, the insured vehicle, a BMW Q1, collided with a large animal (e.g., deer) that suddenly crossed the road. The accident resulted in damage to the front bumper, grille, and headlights. The insured promptly reported the incident and is filing a claim for the repairs. Additionally, the insured sought medical attention for any potential injuries resulting from the collision.

Car Details: BMW Q1;Location: Anthony Street;Date: 1st June;Time of Incident: Not Found

[[SYS]]

Description: A car accident occurred on Jan 1st, 2023 at 5pm at the intersection of woodbridge. The insured vehicle, a Honda Civic, was hit by another vehicle that ran a red light. The insured driver, John, was driving within the speed limit and following all traffic rules. The accident resulted in significant damage to the insured vehicle, including a broken bumper and damaged front fender. There were no injuries reported. The insured is filing a claim for the repairs and any necessary medical expenses. [[INST]]

Stop reason: End of sequence token encountered
Tokens: 406 input + 39 generated = 445 out of 4096
Time: 2.4 seconds

Generate →

Model parameters

Decoding

Greedy ☒ Sampling [icon]

Repetition penalty

1 ● 2 1

Stopping criteria [icon]

Stop sequences

+

Min tokens Max tokens

0 200

Enter up to 6 sequences to stop output after the minimum number of tokens is reached.

Reset to default [icon]

Notice that this prompt has been formatted specifically for *llama* (with *INST* and *SYS* tags). It is also a one-shot prompt, which, compared to the first example, will utilize less tokens for inference.

15. Click the **Generate** button to test the prompt.

Notice that the **Evaluate** button (or icon) is disabled for this prompt. In order to evaluate, track, and deploy this prompt, we need to add a *parameter* – the input that will be passed to the prompt.

Note: You will also not be able to track or deploy prompts without parameters.

New prompt + [icon] ☐ AI guardrails off

16. In the **Prompt variables** view of the **Prompt Lab**, add the *claim_desc* parameter.

Model: llama-2-70b-chat [icon] [icon] [icon]

Prompt variables [icon]

[icon]

Model: llama-2-13b-chat

Model, ...

date ...

al (e.g., ...)

the

Prompt variables ⓘ

Variable	Default value
claim_desc	The insured ...

New variable +

Preview ⓘ

You can use the following value as the default value:

A car accident occurred on Jan 31st, 2023 at 5pm at the intersection of woodbridge. The insured vehicle, a Tesla Model Y, was hit by another vehicle that ran a red light. The insured driver, John, was driving within the speed limit and following all traffic rules. The accident resulted in significant damage to the insured vehicle, including a broken bumper and damaged front fender. There were no injuries reported. The insured is filing a claim for the repairs and any necessary medical expenses.

If you wish, you can provide a different claim description from the test file we used earlier.

Next, add the word *Description* and the *parameter* enclosed in curly brackets to the prompt. Notice that now the **Evaluate** button is enabled.

rojects / Insurance LLM Use Cases EL / Prompt Lab

Unsaved

New prompt +

AI guardrails on

Chat

Structured

Freeform

AI

Model: llama-2-70b-chat

<s>[INST] <<SYS>>

You are a model that extracts entities from insurance claims. You specialize in finding car make and model, location, date and time of an incident.

<</SYS>>

Description: Read the description below and extract the car make and model, Location of the incident like street and date time if there is any mentioned. If you don't find these details in the description, please fill it as Not Found. Format extracted values as a list separated by semicolon. Example: Car Details: car make and model; location; date and time. Always start response with "Car Details:" Do not include any other information.

While driving on Anthony Street on 1st June, the insured vehicle, a BMW Q1, collided with a large animal (e.g., deer) that suddenly crossed the road. The accident resulted in damage to the front bumper, grille, and headlights. The insured promptly reported the incident and is filing a claim for the repairs. Additionally, the insured sought medical attention for any potential injuries resulting from the collision.

Car Details: BMW Q1;Location: Anthony Street;Date: 1st June;Time of Incident: Not Found

Description:{claim_desc}

[/INST]

Prompt variables ⓘ

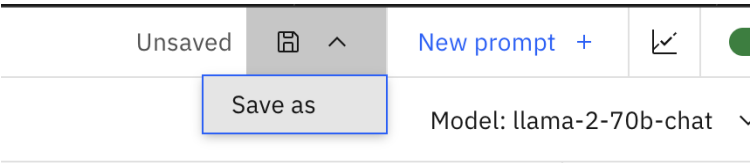
Variable	Default value
claim_desc	A car accident ..

New variable +

Remove special characters

Preview ⓘ

17. Save the prompt template by clicking the save icon then *save as*. Make sure to select *Extraction* as the task type.



Important Note: At this time the task type will automatically determine which evaluation tests will be run. You will not be able to change the task type after saving the template.

Save your work

Specify how to save your work by selecting an asset type and defining details.

Asset type

Prompt template

Save the current prompt only, without its history.

Prompt session

Save history and data from the current session.

Notebook

Save the current prompt as a notebook.

Define details

Name

Info_extraction_llama_70B

Task

Extraction

Description (optional)

What's the purpose of this prompt asset?

☐ View in project after saving

18. Click the **Evaluate** icon to run evaluation for this prompt using the same dataset and values as in the previous prompt evaluation.

Test dataset: *text_extraction_claims.csv* file in the *lab_repo/Test Data* folder

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Map prompt variables to columns

For each prompt variable, select the associated column. [Learn more](#)

Field separation

Select delimiter

Comma (,)

Input

input

Claims text

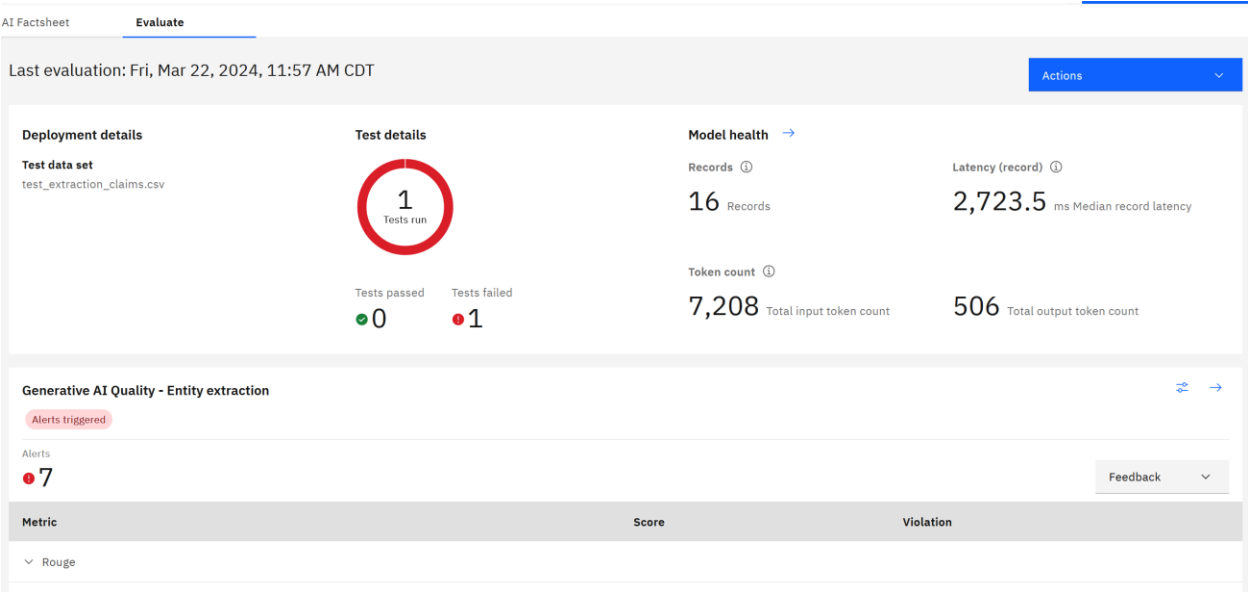
Reference output

Reference output

Extracted Key Facts

With this template, some of the test results are below the thresholds. If you wish, investigate individual metrics to understand which ones have failed the test.

In general, the prompt template doesn't have to pass every test to be ready for production. Usually, a data scientist can decide which tests are applicable for their use case and set the appropriate thresholds.



Next, we will enable tracking for this template following the same steps as you have done for the other prompt.

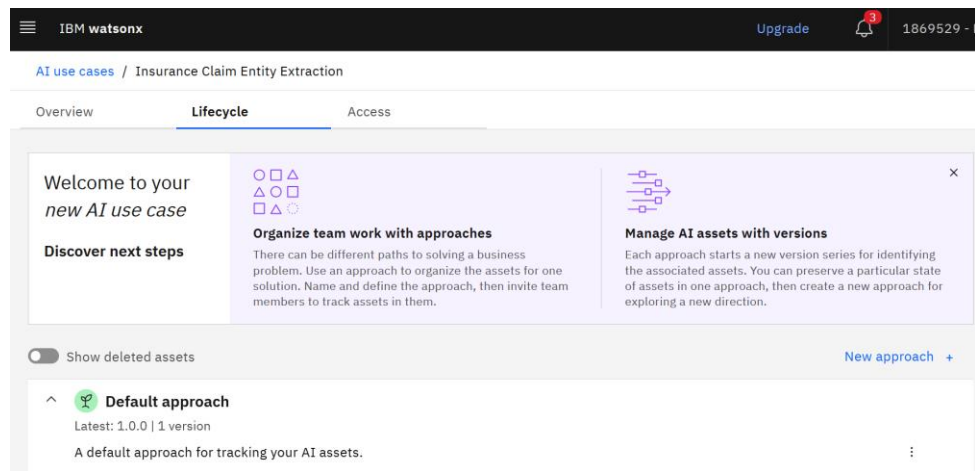
Before we do that, we will create a new *approach* for our use case. Approaches help us organize different *implementations* for the same use case. For example, some companies may do A/B testing to find out which prompt/model delivers the best performance.

19. Navigate to the *AI use cases* view and select your use case.

AI use cases

Name	Status	Owner	Inventory	Tags
Insurance Claim Entity Extraction	Development in progress	Elena Lowery	LLM Insurance Use Cases EL	llm entity extraction

20. In the **Lifecycle** view click **New Approach**.



21. Create the new approach using values similar to the ones shown in the screenshot and click **Create**.

New approach

An approach defines one path for solving the goal of the use case. For example, an approach might be a variation on a machine learning model, or a challenger model. Each approach can include multiple versions.

The screenshot shows the 'New approach' form. It has a header section with 'Icon' (a package icon) and 'Color' (a dropdown menu set to 'Gray'). Below this is the 'Title' field with the text 'One shot extraction with llama'. Underneath is the 'Description (optional)' field with the text 'Prompt that uses one-shot example and llama 70B model'. At the bottom right, there are two buttons: 'Cancel' and 'Create'.

At this time, we do not have any assets tracked for this approach.

Show deleted assets

^

One shot extraction with llama

Prompt next that uses one-shot example and llama 70B model

No AI assets tracked in this approach.

22. Navigate to the **Factsheet** view of your *llama* prompt (from the project **Assets** view).

All assets

Name	Last modified	
<div>Info_extraction_llama_70B</div> <div>Prompt template</div>	Now	Modified by Service
<div>Insurance claim key information extraction</div> <div>Prompt template</div>	37 minutes ago	Modified by you
<div>Insurance claim summarization</div> <div>Prompt template</div>	59 minutes ago	Modified by Service
<div>Insurance claim suggested next steps</div> <div>Prompt template</div>	59 minutes ago	Modified by Service

Evaluate

View AI factsheet

Promote to space

Delete

23. Click **Track in AI use case**. Select your use case and the newly created approach.

Governance

This prompt template is not tracked.

To track a prompt template, add it to an AI use case. Tracking captures details about the asset for governance purposes.

Important:

Once you start tracking a prompt template in a use case, you can no longer edit it. Wait until the prompt template is stable to start tracking.

Track in AI use case

Lab: AI Governance in watsonx

31

Info_extraction_llama_70B

Track in AI use case

Track an asset to collect details about the asset in factsheets as part of a governance strategy.

Define AI use case

Define approach

Assign model versions

Define approach

Use case: **Insurance Claim Entity Extraction**

An approach defines one path for solving the goal of the use case. Ex. an approach might be a variant. Each approach can include multiple versions.

[New approach +](#)

One shot extraction with llama

Prompt that uses one-shot example and llama 70B model

Default approach

A default approach for tracking your AI assets.

Latest: 1.0.0 | 1 Version

Select the *Stable* model version, click **Next**, then click **Track asset**.

Assign model version

Approach: **One shot extraction with llama** | Use case: **Insurance Claim Entity Extraction**

Choose the starting point for this approach.

Experimental

Use this as a starting point if your model is just starting in development and its input and output structure will likely change in the near future.

0.0.1

Stable

Use this as a starting point if your model is in a production state and you won't expect any major changes in its input and output structure soon.

1.0.0

Custom

Define your own starting version if you already tracked this model in a versioning context before.

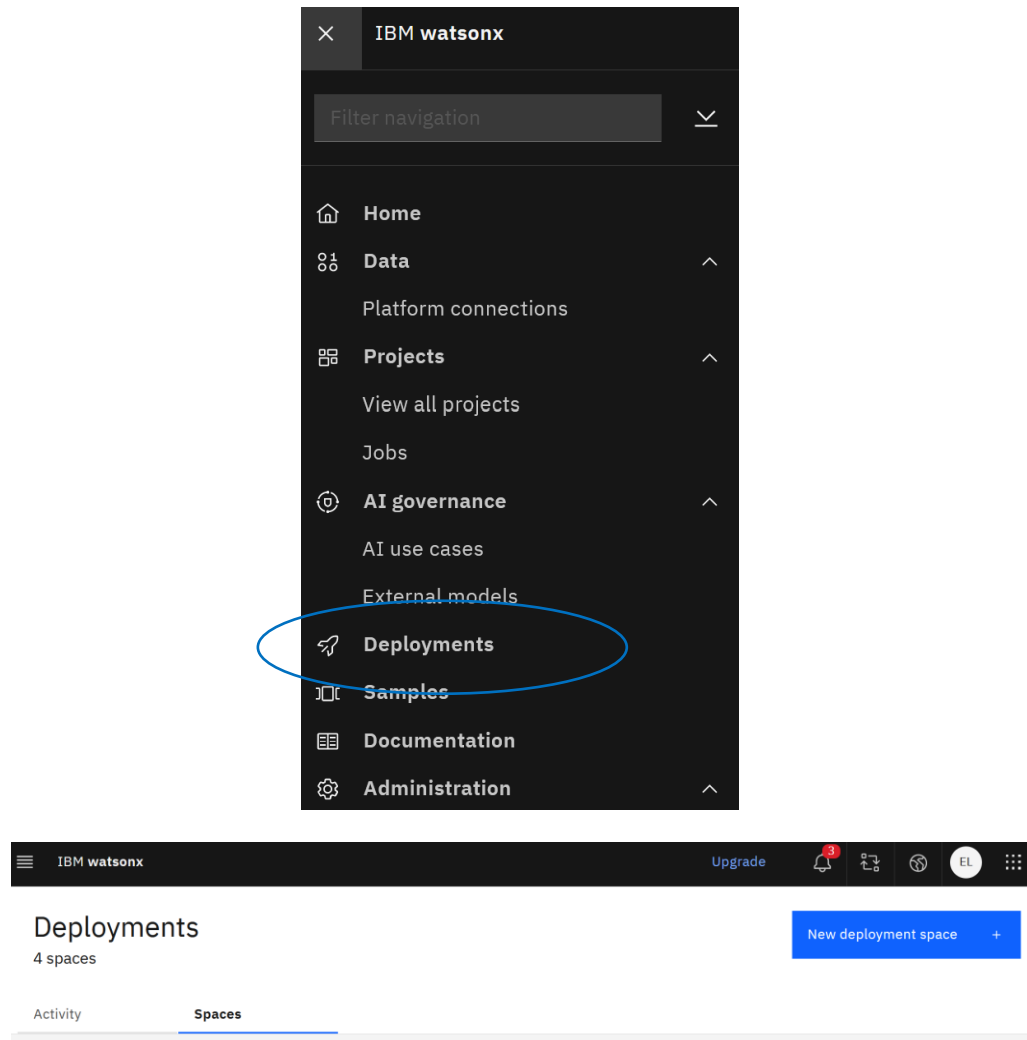
If you wish, come back to the **AI use case** view, and verify that both templates are now tracked – one under the *Default* approach, and one under the custom one that we created.

Next, we will promote both prompt templates to production. In this lab we will skip the validation step. Validation is similar to model evaluation that we have done in this lab, but it is usually done by a different team in a different *watsonx* project. If you would like to learn more about the validation step, you can review [documentation](#).

24. From the main menu navigate to **Deployments** and create a new deployment space using values similar to the following screenshot.

Notes:

- *Make your deployment space name unique by adding your initials.*
- *Your machine learning service name will be different.*



Create a deployment space

Use a space to collect assets in one place to create, run, and manage deployments

Define details

Name

Insurance Use Cases EL - Production

Description (Optional)

Deployment space description

Deployment stage ⓘ

Production

Deployment space tags (optional) ⓘ

Add a tag

Select services

Select storage service ⓘ

Cloud Object Storage-ur

Select machine learning service (optional) ⓘ

Machine Learning-fz

Upload space assets (optional)

Populate your space with assets exported from a project or space to a .zip file. You can add more assets after the space is created.

Drop .zip file here or browse your files to upload

Cancel

Create

25. Navigate back to your projects. From the **Assets** view of the project, select **Promote to Space**.

26. Select the space that you previously created and click **Promote**.

All assets

Name	Last modified	
<div><div></div><div>Info_extraction_llama_70B</div><div>Prompt template</div></div>	9 minutes ago Modified by Service	
<div><div></div><div>Insurance claim key information extraction</div><div>Prompt template</div></div>	46 minutes ago Modified by you	<div>Evaluate</div> <div>View AI factsheet</div> <div>Promote to space</div>
<div><div></div><div>Insurance claim summarization</div><div>Prompt template</div></div>	1 hour ago Modified by Service	
<div><div></div><div>Insurance claim suggested next steps</div><div></div></div>	1 hour ago	

Promote to space

Use a deployment space to organize supporting resources such as input data and environments; deploy models or functions to generate predictions or solutions; and view or edit deployment details.

Target space

Insurance Claims EL - production

×

▼

Why don't I see all of my spaces?

ⓘ

☐
Go to the space after promoting the prompt template

Selected assets (1)

Name	Format
Info_extraction_llama_70B	Prompt template

Select version

ⓘ

Promoting a version of an asset to a space creates a new asset in the space, with a new asset ID.

Cancel

Promote

- Navigate to the **Deployment space**. Next, we will deploy both templates.
- In the **Assets** view of the space, click on **Deploy** next to the template.

Insurance Claims EL - production

Overview

Assets

Deployments

Jobs

Manage

Find assets

Import assets

2 assets

All assets

2

Asset types

Assets

Name	Last modified
<div> <div>...</div> <div>Info_extraction_llama_70B</div> <div>Prompt template</div> </div>	5 minutes ago Service

Deploy

Delete

⋮

Provide a deployment name and click **Create**.

Create a deployment

Deployment type

Online

Run the prompt template on data in real-time, as data is received by a web service.

Name

Extract_claim_info_llama

Serving name ⓘ

Deployment serving name

Description

Deployment description

Cancel

Create

29. Click on the created deployment and explore the various tabs.

[Deployments](#) / [Insurance Claims EL - production](#) / [Info_extraction_llama_70B](#) /

Extract_claim_info_llama

Deployed Online

API reference

Test

Evaluations

AI Factsheet

Direct link

Private endpoint

Text endpoint

<https://private.us-south.ml.cloud.ibm.com/ml/v1-beta/deployments/ee>

Stream endpoint

<https://private.us-south.ml.cloud.ibm.com/ml/v1-beta/deployments/ee>

Public endpoint

Text endpoint

<https://us-south.ml.cloud.ibm.com/ml/v1-beta/deployments/eeaceab3~4>

Stream endpoint

<https://us-south.ml.cloud.ibm.com/ml/v1-beta/deployments/eeaceab3~4>

[Learn more](#) about the 2021-05-01 version query parameter

On the **Test** tab we can interactively test the prompt.

If you wish, you can copy one of the claim description values from the provided csv file that was used for evaluations and test the LLM.

Extract_claim_info_llama ✓ Deployed Online

API reference
Test
Evaluations
AI Factsheet

Enter input data

Text	Stream	JSON
-------------	--------	------

These variables were specified in the Prompt Lab to provide values for the prompt template. You can edit the value of these variables before generating text.

claim_desc

The insured vehicle, a Tesla model X, was vandalized on march 23rd while parked in front of the insured's i ↻

The **AI Factsheet** tab shows the same information as the factsheet in the project.

30. If you wish, repeat the same steps to deploy the 2nd template.

Next, we will walk through the process of setting up *Evaluations*.

The process of evaluation, whether it is done manually in a project, manually in the *Deployment Space*, or automatically is always the same: generated output is compared with the reference data. The only difference is how generated and reference data are provided.

In **Deployment Spaces**, we can set up automatic payload logging (payload data is the data that is passed in to the LLM). You can find more information about setting up payload logging in [documentation](#).

Payload logging is not in the scope of this lab. Since we do not have payload logging configured, we can go through the setup process and run manual evaluation.

You can run evaluation for one or both of the deployed templates.

31. On the *Evaluations* tab click **Activate**.

Deployments / Insurance Claims EL - production / Info_extraction_llama_70B /

Extract_claim_info_llama ✔ Deployed Online

API reference Test **Evaluations** AI Factsheet



Activate monitoring

Click Activate to choose dimensions to evaluate.

Activate

32. Click on **Next** on the **Select dimensions** screen (accept defaults).

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Select dimensions to evaluate

These dimensions are based on the prompt template task type. [Learn more](#)

<input checked="" type="checkbox"/> Dimension	Description
<input checked="" type="checkbox"/> Generative AI Quality	The Generative AI Quality monitor calculates a variety of metrics b reference output you provide. Other metrics analyze model input a

33. On the **Select test data** screen browse to the *lab repo/Test Data* folder and select the *test_extraction_claims.csv* file.

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate



Drop a file here or browse for a file to upload

Add a CSV file that include input and output examples. Maximum s records is 10.

Browse

34. Similar to the evaluation configuration in the project, select the *Input* and *Reference* output fields as showing the following screenshot.

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Map prompt variables to columns

For each prompt variable, select the associated column. [Learn more](#)

Field separation ⓘ

Select delimiter

Comma (,)

Input

input

Claims text

Reference output

Reference output

Extracted Key Facts

35. Click **Next**, then click **Evaluate**.

Evaluate prompt template

Choose the evaluation dimensions and select the test data. [Learn more](#)

Select dimensions

Select test data

Review and evaluate

Review

Task:

Entity extraction

Test data:

test_extraction_claims.csv

Evaluations:

Generative AI Quality

Note:

Evaluation can take a several minutes to complete. You can continue to work on other things while your evaluation is in progress.

Review evaluation results.

Get_key_info_prompt Deployed Online

API reference Test **Evaluations** AI Factsheet

Last evaluation: Sun, Jan 21, 2024, 4:01 PM CST

Actions

Deployment details

Test data set

test_extraction_claims.csv

Test details



Tests passed

0

Tests failed

1

Model health [→](#)

Records ⓘ

16 Records

Latency (record) ⓘ

2,743.5 ms Median record latency

Token count ⓘ

15,578 Total input token count

320 Total output token count

While this evaluation process is similar to the process you completed in the project, it has a different purpose. *Deployment Space* is a production environment, and here the evaluations capture measurements for prompts that are currently in production, which is important for understanding model performance, transparency, and auditing.

Next, we will invoke the deployed template from a client application. If you would like to do a quick test, you can use a notebook that is running in *watsonx*. However, in production deployment, the client application will be running outside of *watsonx*. We will explain where to find and how to modify the REST call for invoking the template, and you can decide how you would like to test it.

36. In your *Deployment Space* project click on the **API reference** tab.

Deployments / Insurance claims - production / Insurance claim key information ... /

Get_key_info_prompt Deployed Online

API reference Test Evaluations AI Factsheet

Direct link

Private endpoint

Text endpoint

Stream endpoint

Bearer <token>

TAM

Public endpoint

Text endpoint

Stream endpoint

Learn more about the 2021-05-01 version query parameter

Code snippets

cURL

NOTE: you must set \$API_KEY below using information retrieved from your IBM Cloud account (https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/ml-authentication.html)

```
curl --insecure -X POST --header "Content-Type: application/x-www-form-urlencoded" --header "Accept: \
application/json" --data-urlencode "grant_type=urn:ibm:params:oauth:grant-type:apikey" \
--data-urlencode "apikey=$API_KEY" "https://iam.cloud.ibm.com/identity/token"
```

the above CURL request will return an auth token that you will use as \$IAM_TOKEN in the scoring request below

TODO: manually define and pass values to be scored below

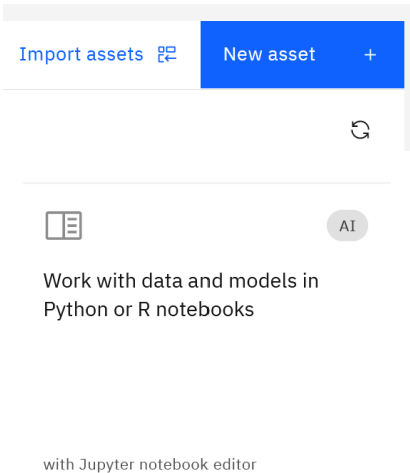
```
curl -X POST --header "Content-Type: application/json" --header "Accept: application/json" --header "Authorization: \
Bearer $IAM_TOKEN" -d '{ "parameters": { "prompt_variables": { "input": "Read this Insurance claim description and extract the Car make and model, Location of the incident like street and date time if there is any mentioned. "
```


If you would like to do a quick test with a notebook, complete the next two steps. You can also skip these steps and go the step of testing prompt template invocation with a Python script.

Note: if you typically don't work with the REST API, we recommend that you skip the next 2 steps.

37. Create a new notebook in your watsonx project (open a new browser tab).

- In the **Project** view select **New Asset -> Work with data and models in Python or R notebooks**



- Select the default Python environment.

Define details

Name

Description (optional)

Define configuration

Select runtime

Runtime 23.1 on Python 3.10 XS (2 vCPU 8 G

The selected runtime has 2 vCPU and 8 GB RAM. It consumes 1 capacity unit per hour. [Learn more](#) about capacity unit hours and Watson Studio pricing plans.

Language

☒ Python 3.10

38. Copy the code from the *Code snippets* into the notebook.

After the initial copy, break up the code for getting the token and invoking the prompt into 2 cells. We need to this to get the token value that we will paste into the invocation cURL command.



Update the first cell with your API key, run the cell, then copy the token into the cURL command in the 2nd cell.

Note that the sample code in deployment points to the streaming text URL. If you wish, you can replace it with the text endpoint URL, which you can find on the same **API reference** tab.

```
# NOTE: you must set $API_KEY below using information retrieved from your IBM Cloud account (https://dataplane)

!curl --insecure -X POST --header "Content-Type: application/x-www-form-urlencoded" --header "Accept: application/json" --data-urlencode "grant_type=urn:ibm:params:oauth:grant-type:apikey" \
--data-urlencode "apikey=*****" "https://iam.cloud.ibm.com/identity/token" Cell 1
```

[illegible]

Test invoking the prompt with the single text output

```
# The curl command in the cell above above curl request will return an auth token that you will use as $IAM_TOKEN in the scoring request b
# Include the token without quotes or any other characters, for example Bearer abc123
# Replace the last line of the request with the following line:

# Note this code is testing the private URL with text generation (last line of the request), and not streaming

curl -X POST --header "Content-Type: application/json" --header "Accept: application/json" --header "Authorization: Bearer $IAM_TOKEN" \
-d '{ "parameters": { "prompt_variables": { "input": "The insured vehicle, a Tesla model X, was vandalized on March 23rd while \
parked in front of the insured residence on Magador Street. The vandalism included scratched paint, broken windows, and damage to the side \
The insured promptly reported the incident to the police and obtained a police report. The insured is filing a claim for the repairs and an \
replacement parts. The estimated cost of repairs has been assessed by a reputable auto repair shop." } } }' \
"https://private.us-south.ml.cloud.ibm.com/ml/v1-beta/deployments/23c8e34f-44ea-4ca9-b256-8bb9865b62f5/generation/text?version=2021-05-01"
```

In the next step you will run a Python client script that invokes the template. This code invokes the same deployed template as the notebook, but we refactored the code to make it easier to understand and maintain.

To complete this step, you will need a Python IDE environment (VS Code, PyCharm, etc.). If you don't have a Python IDE, then you can watch a video in the *lab repo/ Reference folder*.

39. Find the following Python scripts in the downloaded lab repo */Scripts* folder:

- *demo_invoke_template.py*

Let's update and review the script.

On top of the script replace variables with your IBM cloud API key and the public URL of your deployed template.

```
import requests, json

# Replace with your IBM Cloud API key
cloud_api_key = ''

# In most cases the URL for authentication should be this value.
# If you get an authentication error, check the URL in IBM Cloud
auth_url = 'https://iam.cloud.ibm.com/identity/token'
# Make sure to provide public, text URL (not private and not straming)
prompt_url = ''
```

The script has the following functions:

- `get_credentials()`: generates the authentication token
- `invoke_prompt()`: invokes the prompt
- `demo_invoke_prompt()`: invokes all other functions for testing

40. Run the script. The output will be shown in Python terminal.

```
C:\ProgramData\anaconda3\envs\Python310\python.exe C:\Users\1A3030897\PycharmProjects\LLM_Workshc
C:\Users\1A3030897\AppData\Roaming\Python\Python310\site-packages\urllib3\connectionpool.py:1056:
  warnings.warn(
The access token is: eyJraWQiOiIyMDI0MDEwNjA4MzciLCJhbGciOiJSUzI1NiJ9.eyJpYW1faWQiOiJJJk1pZC0xMTA
The generated text is: Car Details: Tesla Model X;Location: Magador Street;Date:

Process finished with exit code 0
```

You have finished testing the deployed prompt template.

Conclusion

You have finished the **AI Governance in watsonx** lab. In this lab you learned:

- Best practices for organizing AI use cases
- Tracking prompt template lifecycle in watsonx
- Invoking the deployed prompt template from a client application.