Workato Genie Lab: Step-by-Step Instructions

Welcome

Welcome to the Workato Agentic AI Lab. In this lesson, you'll learn how to build and test your own AI-powered Genie using the Workato platform. Let's get started.

---

1. Create Your Project and Folder

On Screen:

- Let's start by creating a new project for this activity.
- Click the **Create Project** button.
- In the **Project Name** field, enter:
  `Workato Agentic AI Lab`
- Click **Create Project**.

On Screen:

- Within your new project, click the **Create Folder** button.
- In the **Folder Name** field, enter:
  `Genie Lab`
- Ensure the folder location is set to the **Workato Agentic AI Lab project**.
- Click **Create Folder**.

---

2. Add End Users and Groups

Now, let's set up end users and user groups for Genie access.

On Screen:

- From the left menu, select **Workspace Admin**.
- Under **Solutions Access**, click **End Users**.

- Click **Add User Manually**.
- Enter your full name and email address.
- Click **Invite**.
- Next, click **End-User Groups**.
- Click **Create End-User Group**.
- Enter `Genie User` as the group name.
- Select yourself as the user.
- Click **Create**.

Check your email for the invitation, click **Get Started**, and sign in.

---

3. Create and Configure Your Genie

Now let's build your Genie!

On Screen:

- In Workato, click **AI Hub** from the left menu.
- Click **Create Genie**.
- In the **Name** field, enter:
  `AniTech Genie Lab`
- Save the Genie to the **Genie Lab** folder.
- Click **Start Building**.
- These elements allow us to determine our Genie's behaviors and responses. There's the Job Description, Skills, and Knowledge Base, which we will customize momentarily.
- For now, let's begin with the chat interface. This is where we will select the interface we will use to communicate with our Genie.
- Click **Select Chat Interface**.
- We have a couple of options, but for this activity, choose **Workato Go**.
- Click **Connect Interface**.
- Click **Exit** (top right).
- Go to the **End User Access** tab.
- Click **Add User Groups**.
- Select the **Genie User** group.
- Click **Add**.
- Click **Start Genie**.

---

4. Test Genie Access via Workato Go

Now we're going to Test Genie Access via Workato Go. Let's test your Genie from the end-user perspective.

On Screen:

- In **AI Hub**, go to the **Workato Go** tab.
- Click **End User Access**.
- Click **Add End User Groups** and select **Genie User Group**.
- Click **Add**.
- Go to the **Subdomain** tab.
- Copy the subdomain URL and open it in a new browser tab. Be sure to use this exact subdomain URL; otherwise, you may encounter an error message.
- Complete the URL by adding `.workato.ai` to access Workato Go.
- From the **Ask** dropdown, select **AniTech Genie**.
- Let's ask:

  ```
  What can you help me with?
  ```

Take a look at the Genie's response. Keep in mind this response is based on its default profile.

---

5. Customize the Genie's Job Description

Now, let's personalize your Genie's job description.

On Screen:

- Return to the Workato platform and **AI Hub**.
- Select the **AnniTech Genie**.
- Click **Edit**.
- Click **Edit** next to the Genie's Job Description. This prompt defines the genie's overall purpose, objectives, and the use case it is meant to address.
- The job description acts as a high-level summary and is auto-filled as the starting point for the genie's instructions.
- You'll notice the editor is using Markdown for structure. Markdown formatting and maintaining consistency in Genie Job Descriptions ensure clarity, improve Genie's performance, and make prompt management easier for everyone involved.
- **Delete** the default text and **paste** in the provided **Job Description** for this lab:

  ```
  # AniTech Helpdesk Assistant



  ## Role
  ```

```
You are a friendly helpdesk assistant for AniTech employees.
Handle technical issues, policy questions, and IT inquiries.
Business purposes only - conversations may be logged.


## Communication Rules

- One question at a time - wait for responses before asking more

- Be clear, concise, and professional but friendly

- Politely redirect non-business requests


## Workflow

1. Initial Contact: Greet and ask user to describe their issue

2. Troubleshooting: Gather details, summarize, offer solutions

3. Policy Questions: Use AniTech context if possible; otherwise,
state limitations

4. Escalation & Tickets: Create Jira issues using the user's own
credentials when needed. Users may be prompted to connect their
Jira account the first time they use these features.
```

- Let's click the **Preview** button to see how the markdown formatting looks.
- Now click **Save**.

---

6. Test the Customized Genie

Time to test our Genie. Let's see how the Genie responds with its new job description.

On Screen:

- Go back to **Workato Go**.
- Click **AI Genies** from the left menu.
- Click **Chat with Genie**.

- Type:
  `What can you help me with?`
- Press Enter.

Now review the Genie's updated response. Notice how it has changed since we customized the job description.

---

7. Build a Knowledge Base and Recipe

Next, we'll build a knowledge base recipe to give our Genie access to helpful documentation.

On Screen:

- In the Genie editor, you'll find the Knowledge Base section.
- Click **Add**. Then "**New Knowledge Base**".
- Click **Create Knowledge Base**.
- Name it:
  `AniTech Help Desk`
- Ensure it's in the **Genie Lab** folder.
- Under **Sync Knowledge Usage**, select **Knowledge Recipes**.
- Click **Create Knowledge Base**.
- Click **Create Knowledge Recipe**.
- Name it:
  `FileStorage Knowledge Recipe`
- Ensure it's in the **Genie Lab** folder.
- Click **Start Building**.
- For the **Trigger Event**, select **Workato FileStorage** as the app.
- For the **Trigger**, select **New File**.
- Set the trigger time to **12 AM the day before today**. Setting the trigger back in time will allow the recipe to pick up the trigger when we start it, as opposed to setting the trigger for any point in the future, when we would need to wait for that time to trigger the recipe.
- For **Directory path to monitor**, select `it-knowledgebase` from the Root Directory dropdown.
- Toggle the **Set Trigger Condition** button.
- In **Trigger Data**, insert the **File Name Step 1** data pill.
- Set the condition to **Contains** and value to `.md`, to align with the markdown file extension of the example files we uploaded in the setup steps.

Now, let's configure the action.

On Screen:

- Set **Knowledge Base** to `AniTech Help Desk`.
- For **Document ID**, insert **File Path Step 1** data pill.
- For **Document Title**, insert **File Name Step 1** data pill.
- For **Document Body**, insert **File Contents Step 1** data pill.
- Under **Document Metadata**, click **Show**, then **Add Parameter**.
- We'll get a better result if we specify the language, so we'll do that here…
- Within **Key** we'll enter the abbreviation for language: `lang`
- And within **Value** we'll enter the abbreviation for English: `en`
- For **Updated At**, insert the **Job Created At Properties data pill**. We add this data pill to ensure that if the recipe is run again, it's treated as an update. This helps guarantee the recipe behaves as intended in this example.
- Click **Save**.
- Click **Test Recipe** and review the successful test job.
- Here you can see the recipe has run successfully.
- From the Output tab of the Trigger, we can see the File name and File path.
- From the input tab of the Action, we can see the contents of the document from our Knowledge Base, including the metadata we configured in the recipe setup.
- **Exit** the Recipe editor and **Start the recipe.**

---

8. Add a Genie Skill for Jira Issue Creation

Now, let's add a skill that will empower our Genie to create Jira issues.

On Screen:

- In **AI Hub**, select the **AniTech Genie** and click **Edit**.
- In the **Skills** section, click **Add** > **New Skill** > **Create New Skill**.
- Name it:
  `Create Jira Issue`
- Make sure it is saved to your **Genie Lab** folder.
- Click **Start Building**.
- In "When should your genie run this skill?", type:
  `When the user asks to create an issue or when the genie decides to escalate because the issue is not resolved. Example: I'd like to submit a ticket`
- Under "What inputs will your genie require to run this skill?", click **Add Fields Manually** and add:
  - **Label:** `summary`
    Hint: `A short summary of the problem—like the title of the issue`
  - Click **Add Field**

    - Now let's add some more.
    - Within **Label** enter**:** `description`
      Within **Hint** enter**:** `Provide a summary of the problem and any troubleshooting steps already taken.`
    - For the third skill, within **Label** enter**:** `priority`
      For the **Hint:** `Priority of the issue—can be High, Medium, or Low`
    - Finally, for the last skill, within **Label:** `label`
      For the **Hint:** `Provide a comma-separated list of category labels for the issue. No spaces. Example: "Team-Impact,Single-User-Impact,Security-Issue"`
    - Click **Add Field**
- Under "What should be returned to the genie after this skill is run?",
- Click **Add fields manually**:
    - For the **Label** enter**:** `result`
    - Click **Add Field**

Now, let's configure the first action in this recipe.

On Screen:

- Select **Jira** as the app.
- Select **Create Issue** as the action.
- Ensure **End User's connection** is selected.
- Create your Jira connection using **OAuth 2.0**.
- For **Host Name**, copy your Jira subdomain (from your Jira account URL) and paste it in.
- Click **Connect** and **authorize your account**.
- For **Project Issue Type**, select `Genie Lab : Task`.
- Click **Optional Fields**, select **Labels**, and click **Apply Changes**.
- Map the following fields:
    - Within **Summary:** the **Summary Step 1** data pill
    - Within **Description:** the **Description Step 1** data pill
    - Within **Priority Name:** select **Use Priority Name** from the dropdown menu, then insert **Priority Step 1** data pill
    - Within **Labels:** the **Label Step 1** data pill
- For the **Result** field, switch to **formula mode** and insert the **Step 2 output** data pill. Using Formula mode here will format the output before returning it to our Genie.
- **Save the Recipe** and **Start the Recipe**.

---

9. Test Genie's Jira Skill in Workato Go

Now let's test our Genie's Jira Skill in Workato Go and see our Genie in action.

On Screen:

- Type:
  `I'd like to submit a ticket`
- The Genie will ask for more details.
- Respond with:
  `My computer is locked and displaying a bitcoin address for payment.`

The Genie recognizes a ransomware issue and offers guidance. It also asks if you'd like to create a ticket.

- Confirm to create the ticket.

On Screen:

- Notice you'll be prompted to **connect your Jira account** securely. This is the runtime user connection.
- **Select your account** and click **Accept**.
- Return to the Genie chat.

The Genie has confirmed Jira ticket creation.

- Now, if we **navigate to our Jira account**, we can see the new ticket with all the details.

---

10. Try Another Scenario

Let's try one more issue.

On Screen:

- In Genie chat, type:
  `Coffee spilled on my computer and now it won't turn on.`

The Genie provides troubleshooting steps and again offers to create a ticket.

- **Confirm ticket creation**, and in Jira, we see the new Jira issue created with the correct description, priority, and labels.

Congratulations!

You’ve successfully built, customized, and tested a Genie with knowledge base integration and Jira escalation.