Date : 20-10-2023

WORKOUT PLANNER (1.0)

# Overview

Get workout plans based on time duration, target muscle, fitness level, fitness goals, and equipment available.

# Getting Started

## Prerequisites

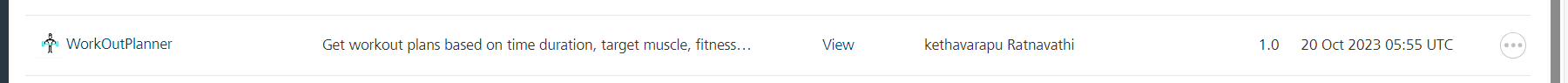
* Volt Foundry

## Importing the adapter

**To import the Data Adapter to Volt Foundry, do the following:**

1. Sign in to the  [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Custom Data Adapters**.  
   
4. Click **IMPORT** to import a custom data adapter.  
   
5. On the Import Data Adapter dialog box, click browser to import.  
   
6. Select Workoutplanner zip file and click **IMPORT**.

After you import the data adapter, Volt Foundry opens a window that shows the metadata of the data adapter.



After you import the data adapter, you can view it on the Custom Data Adapters page and use it to create services on Volt Foundry.

A screenshot of a computer

Description automatically generated

## [Creating an Integration service](javascript:void(0);)

After you import the data adapter into Volt Foundry, you can use it to create an Integration Service.

Follow the given steps to create an Integration service using the <Custom> Adapter.

1. Sign in to the [HCL Foundry](https://manage.hclvoltmx.com/).
2. From the left navigation menu, select **API Management**.
3. In **API Management**, select **Integration**.  
   
4. To create a new service, click the **+** button or the **CONFIGURE NEW** button.  
   
5. On the Service Definition tab, select the service type as Workout planner, and click **SAVE**.  
   A screenshot of a computer

   Description automatically generated

Alternatively, you can also create a Foundry app and create an Integration service inside it.

E. [Creating and Executing operations](javascript:void(0);)



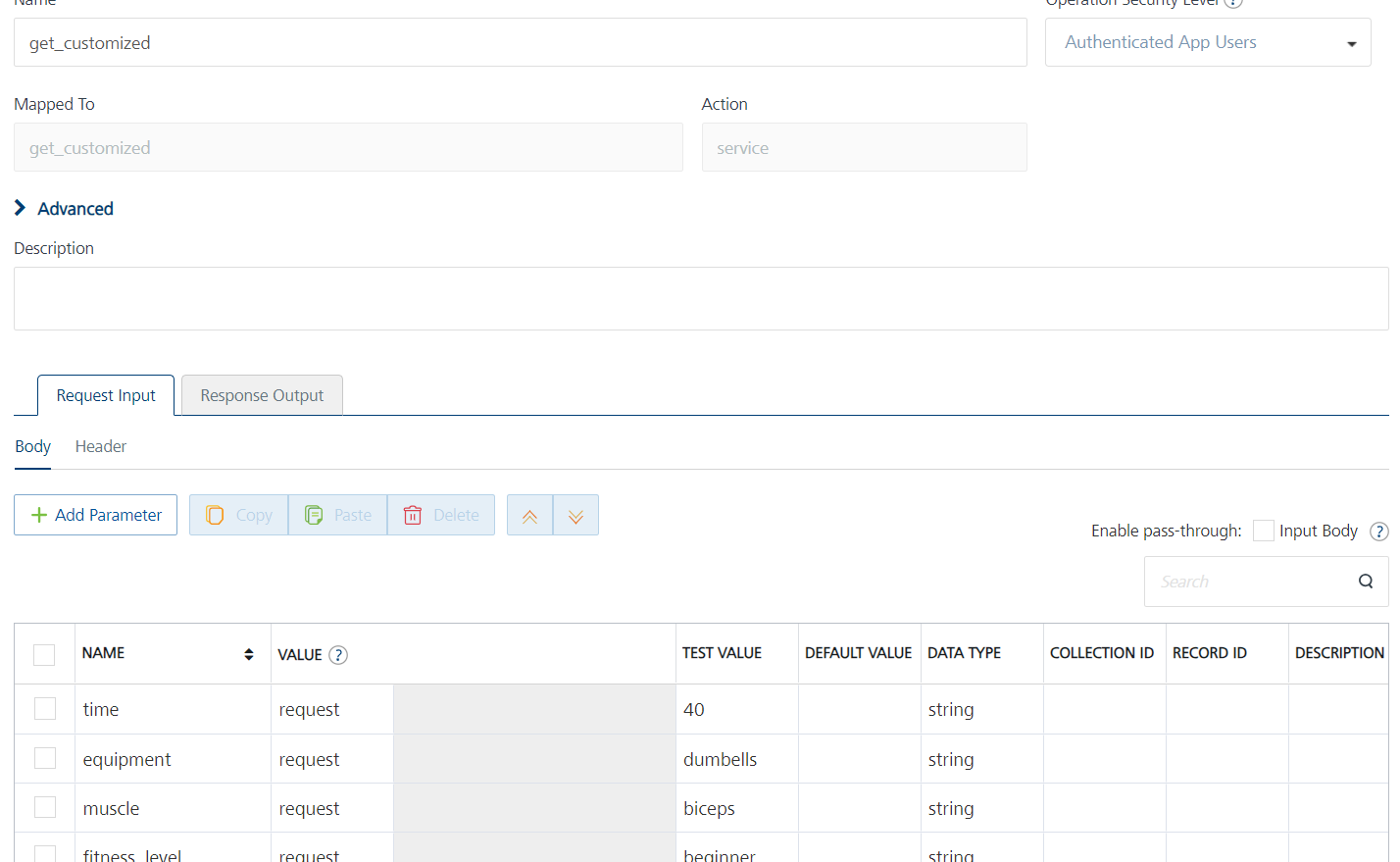
After you create an integration service, you can create and execute operations using the service.

#### Creating an Operation

* In **API Management/Foundry app you created**, in the **Integration** section, select the service that you created.
* After you select the service, navigate to the **Operation List** tab.  
  
* From the drop down list, select an operation that you want to execute, and click **ADD OPERATION**.  
  A screenshot of a computer

  Description automatically generatedExecuting an Operation
* From the **Operations List** tab, in the **Configured Operations** section, select the operation you want to execute.



* On the Operation Page, in the Request Input tab, enter a TEST VALUE for all the fields.  
  

And in the header provide the api-key from the rapid-api

* Select a run-time environment and click **Save and Fetch Response** to get a response based on your inputs.  
  

## [Publishing your application](javascript:void(0);)

If you want to use the services in client applications, you need to publish an app to a run-time environment. You can create the service (as described above) in an application or import the service into an application and publish the application.

# References

## Endpoint Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO | Endpoint Name | Endpoint | Description |
| 1 | Get Customised Plan | https://workout-planner1.p.rapidapi.com/customized | The /customized endpoint provides a way for users to generate personalized workout plans based on their preferences. It takes various parameters into account to create a workout plan tailored to the user's needs. |
| 2 | Get Workout plan | https://workout-planner1.p.rapidapi.com/ | Get Workout Plans  This api request can be used to get Workout plan based on:   * Time duration * Target Muscle * Location * Equipment   Hence api call takes 4 query parameters:   * time * muscle * location... |

## For more details about Dta adapter and endpint related query please check the [Workout Planner API DocumentationRapidAPI](https://rapidapi.com/nabeeldev1340/api/workout-planner1)

# Revision History

Adapter version 1.0.0:

## Known Issues

-List known issues

## Limitations

-List limitations