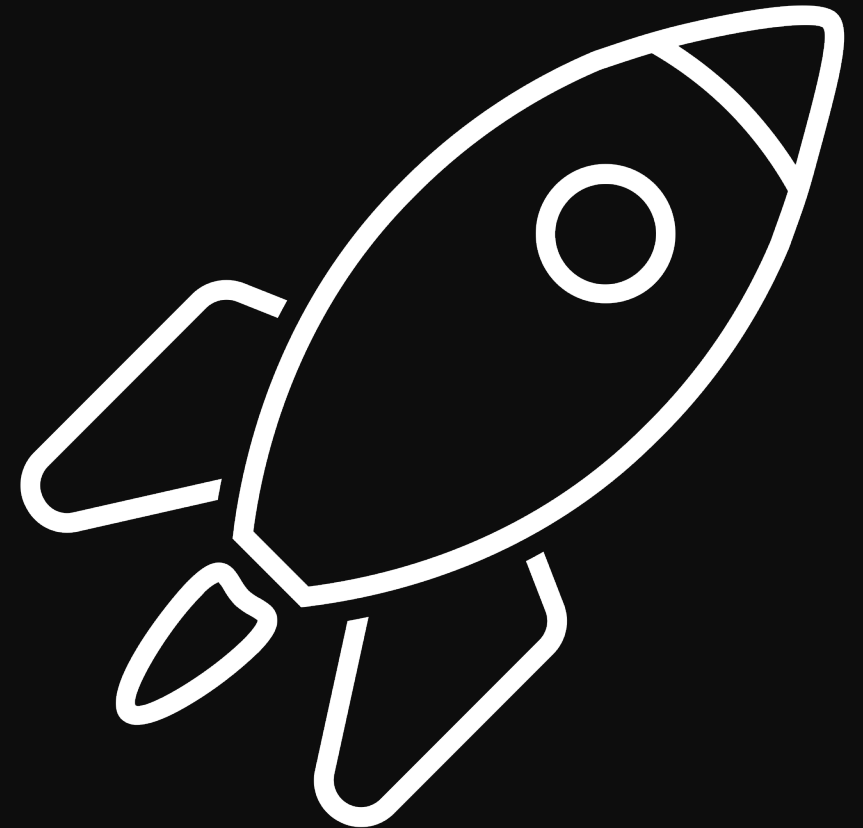


# Watson Assistant with External API

---

Build Lab, IBM Ecosystem



# Introduction

---

IBM **Watson Assistant** is an AI-capable chatbot that can allow you to harness the power of AI with minimal technical knowledge across any messaging platform, application, device or channel. This self-help Watson Assistant service provides users with swift, consistent and accurate means of communication along with an ability to improvise and learn from user conversation.

Out of the stack, we will provide the tools to accelerate the use of **Web Functions** to perform the API calls, return data to the chatbot, and then display the data to augment conversations.

**Web Functions**, also known as FaaS (Function-as-a-Service), are a type of cloud-computing service that allows you to execute code in response to events without the complex infrastructure typically associated with building and launching micro-services applications.

# Why Web Functions?

---

Seamless integration with External application/ API

Pay only for the resources you use, when you use them

Scale up or down automatically

Focus more on code, not infrastructure

# Value proposition of the solution

---

## Natural conversation with Intelligent reasoning

- Provides swift, consistent and accurate means of communication along with an ability to improvise and learn from user conversation

## Easy of use

- Low code solution which is easy to master and requires minimal technical knowledge

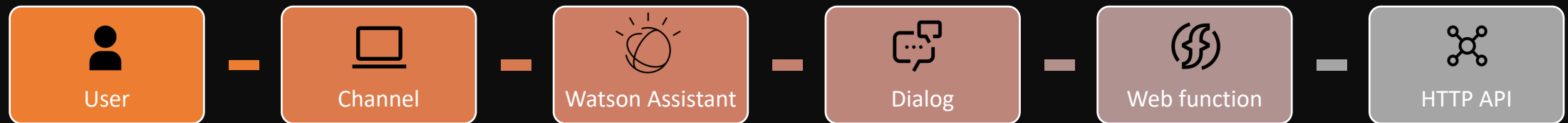
## Omnichannel Integrations

- Connect it to any messaging platform, application, device or channel

# How it works

---

The webhook mechanism allow program to call an external program. When used in a dialog skill, a webhook gets triggered at the moment Watson Assistant processes a node for which webhook is enabled. The webhook collects the specified data or the data that the program collects from the user during the conversation and stores in context variables. It sends the data as part of a HTTP POST request to the URL you specify as part of your webhook definition. The URL that receives the webhook is the listener. It performs a predefined action on the information passed and returns a desired response.



# Example Business Use-cases

---

## 1. Machine Learning:

An intelligent mortgage advisor that is linked with a machine learning model. It collects user data and suggest properties, potential lenders, improvement criteria based on user habits and finances.

## 2. Order Management System:

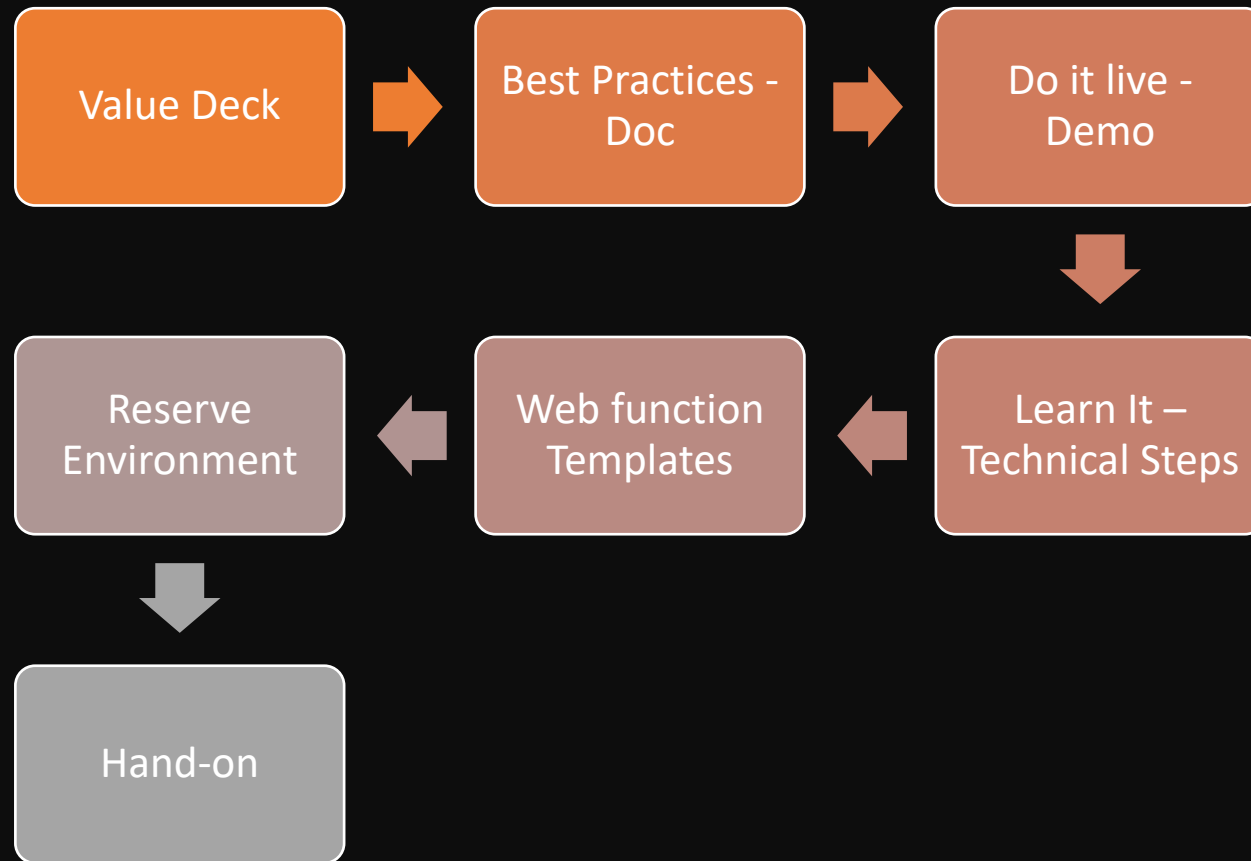
A shopping assistant for an eCommerce store linked with the order management system with an ability to provide product information, add to cart functionality, order status track and other related functionalities. This could further be enhanced with speech to text and text to speech functionality for hands-free environment.

## 3. Project Management :

A smart assistant for project management linked with a backend data. It can be used to fetch the data regarding the projects, hours utilization, unallocated resources etc.

# Journey

---



# Next Step

---

Best Practices – review this before taking the demo

Demo - Business Use-case