Chatbot deplyment with ibm cloud watson assistant

AI Chatbot Project Implementation: A Step-by-Step Guide for Managers

## Table of contents

* [Stage 1: Identifying Customer Needs](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Stage-1:-Identifying-Customer-Needs)
* [Stage 2: Process Analysis](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Stage-2:-Process-Analysis)
* [Stage 3: Chatbot Implementation on the Platform and Tests](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Stage-3:-Chatbot-Implementation-on-the-Platform-and-Tests)
* [Stage 4: Pilot Production Deployment](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Stage-4:-Pilot-Production-Deployment)
* [Stage 5: Chatbot Maintenance and Development](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Stage-5:-Chatbot-Maintenance-and-Development)
* [Work with SentiOne – Your Bot-Building Partner](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Work-with-SentiOne-%E2%80%93-Your-Bot-Building-Partner)
* [Article Summary](https://sentione.com/blog/chatbot-project-implementation-guide-for-managers#Article-Summary)

So, you’ve chosen a [technological partner](https://sentione.com/blog/how-to-choose-chatbot-voice-bot-provider) that will help you build your [chatbot](https://sentione.com/blog/what-is-ai-chatbot) or [voice bot](https://sentione.com/blog/what-is-voice-bot) – now what? There are five essential stages of any bot-building project, and you should know and follow them in order to ensure smooth cooperation with your partner and the delivery of a successful bot that your customers want to talk to. And that’s what we want to discuss in this post.

The main stages of a bot-building project that we will discuss in this article are as follows:

1. Identifying customer needs
2. Process analysis
3. Chatbot implementation and testing
4. Pilot production deployment
5. Chatbot maintenance and development

Following this process ensures you get a virtual assistant that’s not only correctly designed from a technical standpoint but also helpful when it comes to supporting your team with repetitive everyday tasks.

Without further ado, let’s get down to the details.

## Stage 1: Identifying Customer Needs

[Your future chatbot has to address a certain use case and bring tangible results](https://sentione.com/blog/maximising-roi-with-ai-chatbots-voice-bots) – it should support your customers in dealing with everyday tasks such as booking an appointment, asking for order status, and more. This stage is essential as it shapes the whole **[chatbot design process](https://sentione.com/blog/how-to-design-effective-chatbot)**. That’s why everyone involved in direct work with customers should be included.

### Stakeholders

Identifying customer needs should happen with the participation of:

* A **project sponsor** (typically, that’s management or a manager/director who’s responsible for customer service)
* A [**project manager**](https://sentione.com/blog/how-to-manage-ai-chatbot-project) (a person responsible for the oversight of the entire design process; in smaller organisations, a project sponsor and a project manager can be the same person)
* **An analysts** (their job is to provide your company with the necessary data and trends concerning customer service, e.g., what are the typical problems customers have)
* A **Contact centre team**(naturally, you don’t have to include all the consultants you have on your team; captains and supervisors usually are sufficient)

### What you need to do

Your [bot provider](https://sentione.com/blog/how-to-choose-chatbot-voice-bot-provider) will most likely schedule [a chatbot workshop](https://sentione.com/blog/ai-and-change-management-how-to-introduce-chatbot-to-company). This part is important, especially when that’s the first AI chatbot project for your company. You want to make sure that everyone involved in the chatbot design process understands [**conversational AI basics**](https://sentione.com/features/automate/conversational-ai-chatbots-platform), [**NLU basics**](https://sentione.com/features/automate/advanced-natural-language-understanding), all the specific terms, and how a chatbot can be made using an online platform.

The next thing to do is a design thinking workshop. During such a workshop, your team should discuss the chatbot’s goals (what it should do), identify the target audience (who it should serve), [KPIs (how to know your chatbot is useful)](https://sentione.com/blog/chatbot-kpis), and a **[chatbot persona](https://sentione.com/blog/how-to-develop-a-bot-persona)** (how your chatbot should look and behave).

At this point, it’s worth discussing a chatbot prototype – what features it should have so that your company can assess the profitability of the project without investing too much money.

## Prerequisites

Before you get started, make sure that you have a Python version available that works for this ChatterBot project. What version of Python you need depends on your operating system:

* [Windows](https://realpython.com/build-a-chatbot-python-chatterbot/#windows-1)
* [Linux](https://realpython.com/build-a-chatbot-python-chatterbot/#linux-1)
* [macOS](https://realpython.com/build-a-chatbot-python-chatterbot/#macos-1)

You need to use a Python version below 3.8 to successfully work with the recommended version of ChatterBot in this tutorial. You can [install Python 3.7.9 using pyenv-win](https://realpython.com/python-coding-setup-windows/#installing-python-with-pyenv-for-windows).

If you’ve installed the right Python version for your operating system, then you’re ready to get started. You’ll touch on a handful of Python concepts while working through the tutorial:

* [Conditional statements](https://realpython.com/python-conditional-statements/)
* [while](https://realpython.com/python-while-loop/) loops for iteration
* [Lists and tuples](https://realpython.com/python-lists-tuples/)
* [Python functions](https://realpython.com/defining-your-own-python-function/)
* [Substring checks](https://realpython.com/python-string-contains-substring/) and [substring replacement](https://realpython.com/replace-string-python/)
* [File input/output](https://realpython.com/read-write-files-python/)
* [Python comprehensions](https://realpython.com/list-comprehension-python/) and [generator expressions](https://realpython.com/introduction-to-python-generators/)
* [Regular expressions (regex)](https://realpython.com/regex-python/) using re

If you’re comfortable with these concepts, then you’ll probably be comfortable writing the code for this tutorial. If you don’t have all of the prerequisite knowledge before starting this tutorial, that’s okay! In fact, you might learn more by going ahead and getting started. You can always stWhat is chatbot in IOT

Chatbots use a dialog system to have a conversation with a human. There are a few steps a chatbot goes through to process human information. The first step is converting human input into an understandable context for the chatbot.

op and review the resources linked here if you get stuck.

## 

PROGRAM

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

<link rel="stylesheet" href="/static/style.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

</head>

<body>

<h1 class="jumbotron text-center">AI Chatbot with Python</h1>

<div class="container">

<div class="row">

<div class="col-sm-6 offset-sm-3">

<div id="chatbox" class="border border-success">

<p class="botText"><span>Hi! I'm Chatterbot</span></p>

</div>

<div id="userInput">

<input id="textInput" class="form-control" type="text" name="msg" placeholder="Type Your Message Here">

<input id="buttonInput" class="btn btn-success form-control" type="submit" value="Send">

</div>

</div>

</div>

<script>

function getResponse() {

let userText = $("#textInput").val();

let userHtml = '<p class="userText"><span>' + userText + '</span></p>';

$("#textInput").val("");

$("#chatbox").append(userHtml);

document.getElementById('userInput').scrollIntoView({block: 'start', behavior: 'smooth'});

$.get("/get", { msg: userText }).done(function(data) {

var botHtml = '<p class="botText"><span>' + data + '</span></p>';

$("#chatbox").append(botHtml);

document.getElementById('userInput').scrollIntoView({block: 'start', behavior: 'smooth'});

});

}

$("#textInput").keypress(function(e) {

//if enter key is pressed

if(e.which == 13) {

getResponse();

}

});

$("#buttonInput").click(function() {

getResponse();

});

</script>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>

</div>

</body>

</html>

THANK YOU