

Deployment Guide

Download or clone this repo

```
git clone https://github.com/ronikurnia1/CustomQABot.git
```

Change **languageProjectName** and **appName** parameters with a unique short name (initial) in the **~/CustomQABot/Bicep/main.parameters.json** file

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "languageProjectName": {
      "value": "[your-unique-language-project-name]"
    },
    "appName": {
      "value": "[your-unique-app-name]"
    },
    "languageSku": {
      "value": "F0"
    },
    "searchSku": {
      "value": "free"
    },
    "appServicePlanSku": {
      "value": "F1"
    },
    "azureBotSku": {
      "value": "F0"
    }
  }
}
```

Use Azure CLI to deploy the resources

```
az login --tenant "[your-tenant-id]"
```

```
$rgName="[your-resource-group-name]"
```

```
$location="[azure-location]"
```

```
az group create --location $location --name $rgName
```

```
az deployment group create --resource-group $rgName `
--template-file main.bicep --parameters main.parameters.jsonv --yes
```

Wait for about 5-10 minutes for the deployment of azure resources and app to complete

When it completed you can try to access the app at **[https://\[your-unique-app-name\]-webapp.azurewebsites.net](https://[your-unique-app-name]-webapp.azurewebsites.net)**, see if it's working

CustomQA Bot

Your bot is ready!

You can test your bot in the Bot Framework Emulator by connecting to <http://localhost:3978/api/messages>.

[Download the Emulator](#)

Visit [Azure Bot Service](#) to register your bot and add it to various channels. The bot's endpoint URL typically looks like this:

https://your_bots_hostname/api/messages

HOW TO BUILD A BOT



Plan:

Review the bot [design guidelines](#)



Build:

Create a bot from [Azure](#) or [locally](#)
Download [Command-line tools](#)
Add services such as [Language Understanding](#),
[QnA Maker](#) and [Dispatch](#)



Test:

Test using the [Emulator](#)
Test online in [Web Chat](#)



Publish:

Publish [directly to Azure](#) or
Use [Continuous Deployment](#)