

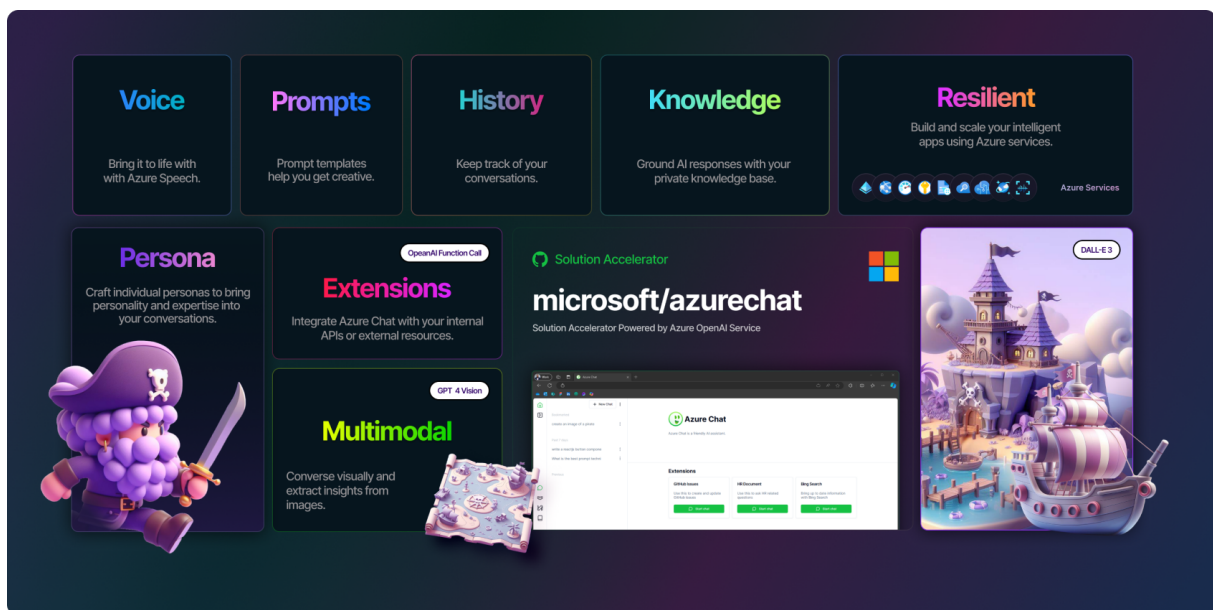
---

## Unleash the Power of Azure Open AI

1. Introduction
2. Solution Overview
3. Deploy to Azure
4. Run from your local machine
5. Deploy to Azure with GitHub Actions
6. Add identity provider
7. Chatting with your file
8. Persona
9. Extensions
10. Environment variables
11. Migration considerations

## Introduction

*Azure Chat Solution Accelerator powered by Azure Open AI Service*



*Azure Chat Solution Accelerator powered by Azure Open AI Service* is a solution accelerator that allows organisations to deploy a private chat tenant in their Azure Subscription, with a familiar user experience and the added capabilities of chatting over your data and files.

Benefits are:

- 
1. Private: Deployed in your Azure tenancy, allowing you to isolate it to your Azure tenant.
  2. Controlled: Network traffic can be fully isolated to your network and other enterprise grade authentication security features are built in.
  3. Value: Deliver added business value with your own internal data sources (plug and play) or integrate with your internal services (e.g., ServiceNow, etc).

## Deploy to Azure

You can provision Azure resources for the solution accelerator using either the Azure Developer CLI or the Deploy to Azure button below. Regardless of the method you chose you will still need set up an identity provider and specify an admin user

### Deployment Options

You can deploy the application using one of the following options:

- 1. Azure Developer CLI
- 2. Azure Portal Deployment

#### 1. Azure Developer CLI

[!IMPORTANT] This section will create Azure resources and deploy the solution from your local environment using the Azure Developer CLI. Note that you do not need to clone this repo to complete these steps.

1. Download the Azure Developer CLI
2. If you have not cloned this repo, run `azd init -t microsoft/azurechat`. If you have cloned this repo, just run 'azd init' from the repo root directory.
3. Run `azd up` to provision and deploy the application

```
1 azd init -t microsoft/azurechat
2 azd up
3
4 # if you are wanting to see logs run with debug flag
5 azd up --debug
```

---

## 2. Azure Portal Deployment

[!WARNING] This button will only create Azure resources. You will still need to deploy the application by following the deploy to Azure section to build and deploy the application using GitHub actions.

Click on the Deploy to Azure button to deploy the Azure resources for the application.



[!IMPORTANT] The application is protected by an identity provider and follow the steps in Add an identity provider section for adding authentication to your app.

Next

## Contributing

This project welcomes contributions and suggestions. Most contributions require you to agree to a Contributor License Agreement (CLA) declaring that you have the right to, and actually do, grant us the rights to use your contribution. For details, visit <https://cla.opensource.microsoft.com>.

When you submit a pull request, a CLA bot will automatically determine whether you need to provide a CLA and decorate the PR appropriately (e.g., status check, comment). Simply follow the instructions provided by the bot. You will only need to do this once across all repos using our CLA.

This project has adopted the Microsoft Open Source Code of Conduct. For more information see the Code of Conduct FAQ or contact [opencode@microsoft.com](mailto:opencode@microsoft.com) with any additional questions or comments.

## Trademarks

This project may contain trademarks or logos for projects, products, or services. Authorized use of Microsoft trademarks or logos is subject to and must follow Microsoft's Trademark & Brand Guidelines. Use of Microsoft trademarks or logos in modified versions of this project must not cause confusion or imply Microsoft sponsorship. Any use of third-party trademarks or logos are subject to those third-party's policies.