

Creating a webchat testsite for testing dialogue flows

First create a storage account in Azure

The screenshot shows the 'Create a storage account' page in the Azure portal. The browser address bar shows 'portal.azure.com/#create/Microsoft.StorageAccount'. The page has a blue header with the 'Microsoft Azure' logo. Below the header, there's a breadcrumb 'Home > Storage accounts >'. The main heading is 'Create a storage account' followed by a three-dot menu. There are tabs for 'Basics', 'Advanced', 'Networking', 'Data protection', 'Tags', and 'Review + create'. The 'Basics' tab is selected. A paragraph describes Azure Storage as a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. It lists various storage options like Azure Blobs, Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. Below this is the 'Project details' section, which asks to select a subscription and a resource group. The 'Subscription' dropdown is set to 'Microsoft Azure (rio365test): #1227352'. The 'Resource group' dropdown is set to 'rio-extraservices', with a 'Create new' link below it. The 'Instance details' section follows, with a link for legacy storage account types. It includes fields for 'Storage account name', 'Region' (set to '(Europe) West Europe'), 'Performance' (with 'Standard' selected and 'Premium' as an option), and 'Redundancy' (set to 'Locally-redundant storage (LRS)'). At the bottom, there are three buttons: 'Review + create' (highlighted in blue), '< Previous', and 'Next: Advanced >'.

portal.azure.com/#create/Microsoft.StorageAccount

Microsoft Azure

Home > Storage accounts >

Create a storage account

Basics Advanced Networking Data protection Tags Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more about Azure storage accounts](#)

Project details

Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.

Subscription * Microsoft Azure (rio365test): #1227352

Resource group * rio-extraservices
[Create new](#)

Instance details

If you need to create a legacy storage account type, please click [here](#).

Storage account name *

Region * (Europe) West Europe

Performance *
☒ Standard: Recommended for most scenarios (general-purpose v2 account)
☐ Premium: Recommended for scenarios that require low latency.

Redundancy * Locally-redundant storage (LRS)

[Review + create](#) < Previous Next: Advanced >

The storage account name should be global unique. I think the best way to do this is to make a company prefix and then choose what you want.

You can use the standard option with LRS, because this is the cheapest (it just test, no need for redundancy). The rest of the settings we accept the defaults azure gives.

After creating of the storage account we have to configure the static website option.

Storage account

Search (Ctrl+ /)

Overview

- Activity log
- Tags
- Diagnose and solve problems
- Access Control (IAM)
- Data migration
- Events
- Storage browser (preview)

Data storage

- Containers
- File shares
- Queues
- Tables

Security + networking

- Networking
- Azure CDN
- Access keys
- Shared access signature
- Encryption
- Security

Data management

- Geo-replication
- Data protection
- Object replication
- Blob inventory
- Static website**
- Lifecycle management
- Azure search

Open in Explorer Delete Move

Microsoft recommends upgrading to the new

Essentials

Resource group (move) : rio-extraservices

Location : West Europe

Subscription (move) : Microsoft Azure (rio36)

Subscription ID : 4b09faaf-e71a-4aff-ae

Disk state : Available

Tags (edit) :

Properties Monitoring Capabilities (7)

Blob service

- Hierarchical namespace
- Default access tier
- Blob public access
- Blob soft delete
- Container soft delete
- Versioning
- Change feed
- NFS v3
- Allow cross-tenant replication

File service

- Large file share
- Active Directory
- Soft delete
- Share capacity

Queue service

- CMK support

Table service

- CMK support

The static website option is under the data management header shown in the screenshot.

<  Save  Discard

Enabling static websites on the blob service allows you to host static content. Webpages may inc
[more](#)

Static website

☐ Disabled ☒ Enabled

Index document name ⓘ

Error document path ⓘ

Set the toggle to enabled and for the index document name enter index.html and for the error document path enter 404.html.

Static website ...

 Save  Discard

Enabling static websites on the blob service allows you to host static content. Wel
[more](#)

Static website

☐ Disabled ☒ Enabled

An Azure Storage container has been created to host your static website.
[\\$web](#)

Primary endpoint ⓘ

Index document name ⓘ

Error document path ⓘ

After saving the page an URL is created for you. This URL we use later in the ROGER365.io portal for connecting the endpoint. Also there is also a \$web link created. This is where we have to upload the two documents we have to create.

For creating the documents, you can just use notepad copy the contents provided below and save as the document name.

For index.html the content is:

```
<!DOCTYPE html>
<html>
  <head>
    <title>ROGER365.io Testwebchat</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

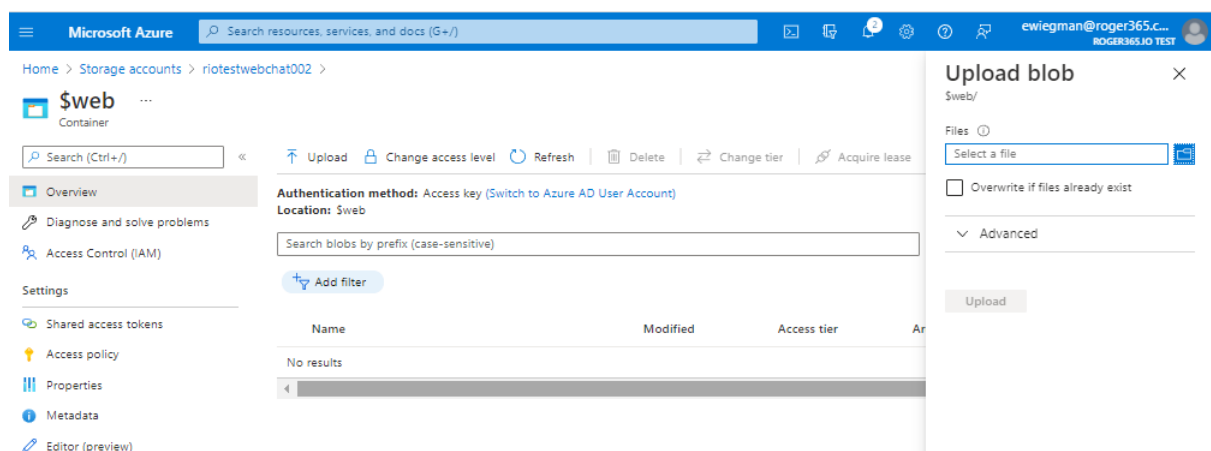
    <link rel="stylesheet" type="text/css" href="https://api.roger365.io/v1.0/webchat/files/startup.v1.css">
    <link rel="stylesheet" type="text/css" href="https://api.roger365.io/v1.0/webchat/files/webchat.v1.css">
    <link rel="stylesheet" type="text/css" href="https://api.roger365.io/v1.0/webchat/files/spinner.v1.css">

    <script src="https://api.roger365.io/v1.0/webchat/files/defaultWebchatOptions.js"></script>
    <script src="https://api.roger365.io/v1.0/webchat/files/webchat.js"></script>
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

For the 404.html the content is:

```
<!DOCTYPE html>
<html>
  <body>
    <h1>404</h1>
  </body>
</html>
```

You should not have two files which have to be uploaded to the \$web container. If you press the \$web link, you will go to the right container, where you can upload the files.



Just upload both the files and after that your website is ready to be added to ROGER365.io as an touchpoint ! You can use the URL which is shown under the static website option.

Link to tutorial to create a webchat touchpoint

<https://doc.roger365.io/documentation/configuration/social-messaging/webchat-touchpoint/>