

DOCUMENTATION TECHNIQUE



1 - Créer et déployer votre application vers Azure Container Apps

2 - Création d'une Web App dans Azure

3 – Activation du déploiement continu sur l'application

1 - Créer et déployer votre application vers Azure Container Apps

Installez et mettez à jour l'extension Azure Container Apps pour l'interface CLI avec la commande :

```
az extension add --name containerapp --upgrade
```

```
fabio@LAPTOP-61HM44QQ MINGW64 ~ (main)
$ az extension add --name containerapp --upgrade
The installed extension 'containerapp' is in preview.
```

Inscrivez l'espace des noms Microsoft.App et le fournisseur Microsoft.OperationalInsights :

```
az provider register --namespace Microsoft.App
```

```
az provider register --namespace Microsoft.OperationalInsights
```

```
fabio@LAPTOP-61HM44QQ MINGW64 ~ (main)
$ az provider register --namespace Microsoft.App

fabio@LAPTOP-61HM44QQ MINGW64 ~ (main)
$ az provider register --namespace Microsoft.OperationalInsights
```

Renseignez les variables pour plus de praticité

```
NOM="EntrezVotreNom"
```

```
RESOURCE_GROUP="tosios"$NOM
```

```
LOCATION="francecentral"
```

```
ENVIRONMENT="env-tosios-containerapps"$NOM
```

```
API_NAME="tosios-api"$NOM
```

```
FRONTEND_NAME="tosios-ui"$NOM
```

```
GITHUB_USERNAME="EntrezVotreNomDeCompte"
```

```
ACR_NAME="acatosios"$NOM
```

Dans votre navigateur, ouvrez le lien <https://github.com/kiwphi/pop-shoot> puis sélectionnez le bouton **Fork** en haut de la page pour activer le référentiel dans votre compte puis créez le.

The image shows two screenshots from a GitHub browser interface. The top screenshot displays the repository page for `halftheopposite/TOSIOS`, which is a public repository. It shows navigation tabs for Code, Issues (3), Pull requests (1), Actions, Projects, Security, and Insights. Below these, there are buttons for 'Go to file', 'Add file', and 'Code'. A commit by `dependabot[bot]` is visible, along with a commit by `8a62ebd` on Feb 5 with 178 commits. The 'About' section describes 'The Open-Source IO Shooter' as an open-source multiplayer game in the browser. The bottom screenshot shows the 'Create a new fork' form. It includes fields for 'Owner' (MPFabio) and 'Repository name' (TOSIOS). A description field contains the text 'The Open-Source IO Shooter is an open-source multiplayer game in the browser'. There is a checkbox for 'Copy the master branch only' which is checked. At the bottom, there is a 'Create fork' button.

halftheopposite/TOSIOS Public

Watch 15 Fork 62 Star 315

<> Code Issues 3 Pull requests 1 Actions Projects Security Insights

master 3 branches 23 tags

Go to file Add file <> Code About

dependabot[bot] Bump json5 from 2.2.0 to 2.2.3 (#63) 8a62ebd on Feb 5 178 commits

The Open-Source IO Shooter is an open-source multiplayer game in the browser

halftheopposite/TOSIOS Public

Watch 15 Fork 62 Star 315

<> Code Issues 3 Pull requests 1 Actions Projects Security Insights

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Owner * Repository name *

MPFabio / TOSIOS

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

The Open-Source IO Shooter is an open-source multiplayer game in the browser

☒ Copy the master branch only

Contribute back to halftheopposite/TOSIOS by adding your own branch. [Learn more.](#)

ⓘ You are creating a fork in your personal account.

Create fork

Clonez votre repository à l'aide de la commande git clone

git clone [https://github.com/\\$GITHUB_USERNAME/TOSIOS.git](https://github.com/$GITHUB_USERNAME/TOSIOS.git)

```
fabio@LAPTOP-61HM44QQ MINGW64 ~ (main)
$ git clone https://github.com/$GITHUB_USERNAME/TOSIOS.git
Cloning into 'TOSIOS'...
remote: Enumerating objects: 2157, done.
remote: Counting objects: 100% (269/269), done.
remote: Compressing objects: 100% (199/199), done.
remote: Total 2157 (delta 109), reused 191 (delta 62), pack-reused 1888
Receiving objects: 100% (2157/2157), 3.63 MiB | 12.44 MiB/s, done.
Resolving deltas: 100% (1195/1195), done.
```

Créez un groupe de ressource pour organiser les services qui y seront déployées :

```
az group create --name $RESOURCE_GROUP --location "$LOCATION"
```

```
fabio [ ~ ]$ az group create --name $RESOURCE_GROUP --location "$LOCATION"
{
  "id": "/subscriptions/393e3de3-0900-4b72-8f1b-fb3b1d6b97f1/resourceGroups/tosiosfabio",
  "location": "francecentral",
  "managedBy": null,
  "name": "tosiosfabio",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
```

Créez ensuite une instance Azure Container Registry (ACR) dans votre groupe de ressources pour stocker l'image conteneur de l'API album une fois qu'elle est générée.

Le fait de spécifier "true" à l'option admin-enabled permet d'activer l'accès Docker sur ACR

```
az acr create --resource-group $RESOURCE_GROUP --name $ACR_NAME --sku Basic --admin-enabled true
```

```
fabio [ ~ ]$ az acr create --resource-group $RESOURCE_GROUP --name $ACR_NAME --sku Basic --admin-enabled true
{
  "adminUserEnabled": true,
  "anonymousPullEnabled": false,
  "creationDate": "2023-04-25T13:23:02.256148+00:00",
  "dataEndpointEnabled": false,
  "dataEndpointHostNames": [],
  "encryption": {
    "keyVaultProperties": null,
    "status": "disabled"
  }
}
```

Rendez-vous à la racine de votre référentiel cloné, où se trouve le Dockerfile à l'aide de la commande cd

```
cd Tosios
```

Editez votre fichier Dockerfile avec l'éditeur de texte de votre choix et éditez la version du node en renseignant alpine comme suit pour définir la dernière version.

Nano Dockerfile

```
GNU nano 6.0
FROM node:alpine

WORKDIR /usr/src/app

ARG REACT_APP_GA_TRACKING_ID

# Dependencies
COPY ./package.json .
COPY ./yarn.lock .
COPY ./packages/client/package.json ./packages/client/
COPY ./packages/common/package.json ./packages/common/
COPY ./packages/server/package.json ./packages/server/
RUN yarn

# Files
COPY . .

# Build
RUN BUILD_MODE=production yarn build
```

Générez l'image avec la commande ACR build :

az acr build --registry \$ACR_NAME --image \$API_NAME .

```
fabio [ ~/TOSIOS ]$ az acr build --registry $ACR_NAME --image $API_NAME .
Packing source code into tar to upload...
Excluding '.git' based on default ignore rules
Excluding '.gitignore' based on default ignore rules
Uploading archived source code from '/tmp/build_archive_c58487d4bbc44a17bba26f5cfd3225ed.tar.gz'...
Sending context (1.586 MiB) to registry: acatosiosfabio...
Queued a build with ID: dd1
Waiting for an agent...
```

Créez un environnement Container Apps, celui-ci agit comme une limite sécurisée autour d'un groupe d'applications de conteneur.

```
az containerapp env create --name $ENVIRONMENT --resource-group $RESOURCE_GROUP --location "$LOCATION"
```

```
fabio [ ~/TOSIOS ]$ az containerapp env create --name $ENVIRONMENT --resource-group $RESOURCE_GROUP --location "$LOCATION"
No Log Analytics workspace provided.
Generating a Log Analytics workspace with name "workspace-tosiosfabiomMS3"

Container Apps environment created. To deploy a container app, use: az containerapp create --help
```

À présent que vous avez créé un environnement, vous pouvez créer et déployer votre application de conteneur

```
az containerapp create --name $API_NAME --resource-group $RESOURCE_GROUP --environment $ENVIRONMENT --image $ACR_NAME.azurecr.io/$API_NAME --target-port 3001 --ingress 'external' --registry-server $ACR_NAME.azurecr.io --query properties.configuration.ingress.fqdn
```

```
fabio [ ~/TOSIOS ]$ az containerapp create --name $API_NAME --resource-group $RESOURCE_GROUP --environment $ENVIRONMENT --image $ACR_NAME.azurecr.io/$API_NAME --target-port 3001 --ingress 'external' --registry-server $ACR_NAME.azurecr.io --query properties.configuration.ingress.fqdn
No credential was provided to access Azure Container Registry. Trying to look up credentials...
Adding registry password as a secret with name "acatosiosfabioazurecrio-acatosiosfabio"
| Running ..
```

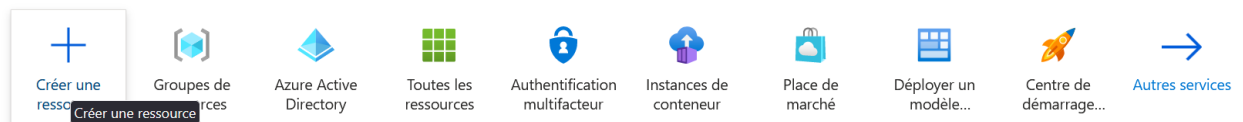
La commande query renvoie le nom de domaine complet de l'application, vous pouvez grâce à ce lien accéder directement à celle-ci.

```
Container app created. Access your app at https://tosios-apifabio.purpleplant-cef57593.francecentral.azurecontainerapps.io/
"tosios-apifabio.purpleplant-cef57593.francecentral.azurecontainerapps.io"
```

2 - Création d'une Web App dans Azure

Sur la page d'accueil du portail Azure, cliquez sur créer une ressource

Services Azure



Sélectionnez Web App puis créer



Web App

Microsoft

Azure Service

Bénéficiez d'options sécurisées et adaptées pour le développement, le déploiement et la mise à l'échelle de votre application web

Créer ▼



Choisissez votre abonnement puis renseignez :

- Votre groupe de ressource
- Le nom de l'instance
- Choisissez conteneur docker pour la publication
- Choisissez Linux comme OS
- Choisissez France Central comme région
- Créez un plan Linux
- Définissez votre plan de tarification préférentiel

[Accueil](#) > [Créer une ressource](#) >

Créer une application web

Sélectionnez un abonnement pour gérer les coûts et les ressources déployées. Utilisez les groupes de ressources comme des dossiers pour organiser et gérer toutes vos ressources.

Abonnement * ⓘ

Groupe de ressources * ⓘ
[Créer nouveau](#)

Détails de l'instance

Vous avez besoin d'une base de données ? [Essayez la nouvelle expérience web + base de données.](#)

Nom * [.azurewebsites.net](#)

Publier * ☐ Code ☒ Conteneur Docker ☐ Application web statique

Système d'exploitation * ☒ Linux ☐ Windows

Région *
Vous ne trouvez pas votre plan App Service ? Essayez une autre région ou sélectionnez votre environnement App Service Environment.

Plans de tarification

Le niveau tarifaire du plan App Service détermine l'emplacement, les fonctionnalités, le coût et les ressources de calcul associés à votre application. [En savoir plus](#)

Plan Linux (France Central) * ⓘ
[Créer](#)

Plan de tarification
[Découvrir les plans de tarification](#)

Redondance de zone

Vous pouvez déployer un plan App Service en tant que service redondant interzone dans des régions le prenant en charge. Vous devez décider de cela au moment du déploiement car vous ne pouvez plus rendre une zone plan App

Vérifier + créer

< Précédent

Suivant : Docker >

Dans la partie Docker :

- Sélectionnez la source de l'image
- Sélectionnez votre registre
- Sélectionnez votre image

De base Docker Réseau Surveillance Balises Vérifier + créer

Extrayez des images conteneurs du registre de conteneurs Azure, de Docker Hub ou d'un dépôt Docker privé. App Service déploie en production l'application conteneurisée avec vos dépendances préférées en quelques secondes.

Options	Conteneur unique
Source d'image	Registre de conteneurs Azure
Options de registre de conteneurs Azure	
Registre *	acatosiosfabio
Image *	tosios-apifabio
Balise *	latest
Commande de démarrage ⓘ	

Cliquez sur Vérifier + créer puis sur Créer

Créer une application web ...

De base Docker Réseau Surveillance Balises Vérifier + créer

Récapitulatif



Application web
par Microsoft

Détails

Abonnement	393e3de3-0900-4b72-8f1b-fb3b1d6b97f1
Groupe de ressources	tosiosfabio
Nom	Fabio-Tosy
Publier	Conteneur Docker
Image:Étiquette	acatosiosfabio.azurecr.io/tosios-apifabio:latest
URL du serveur	https://acatosiosfabio.azurecr.io

Plan App Service

Nom	ASP-abdelnodeondockergroup-b0a8
Système d'exploitation	Linux
Région	France Central
Référence	Gratuit
ACU	Infrastructure partagée
Mémoire	1 Go de mémoire

Surveillance

Application Insights	Non activé
----------------------	------------

Déploiement

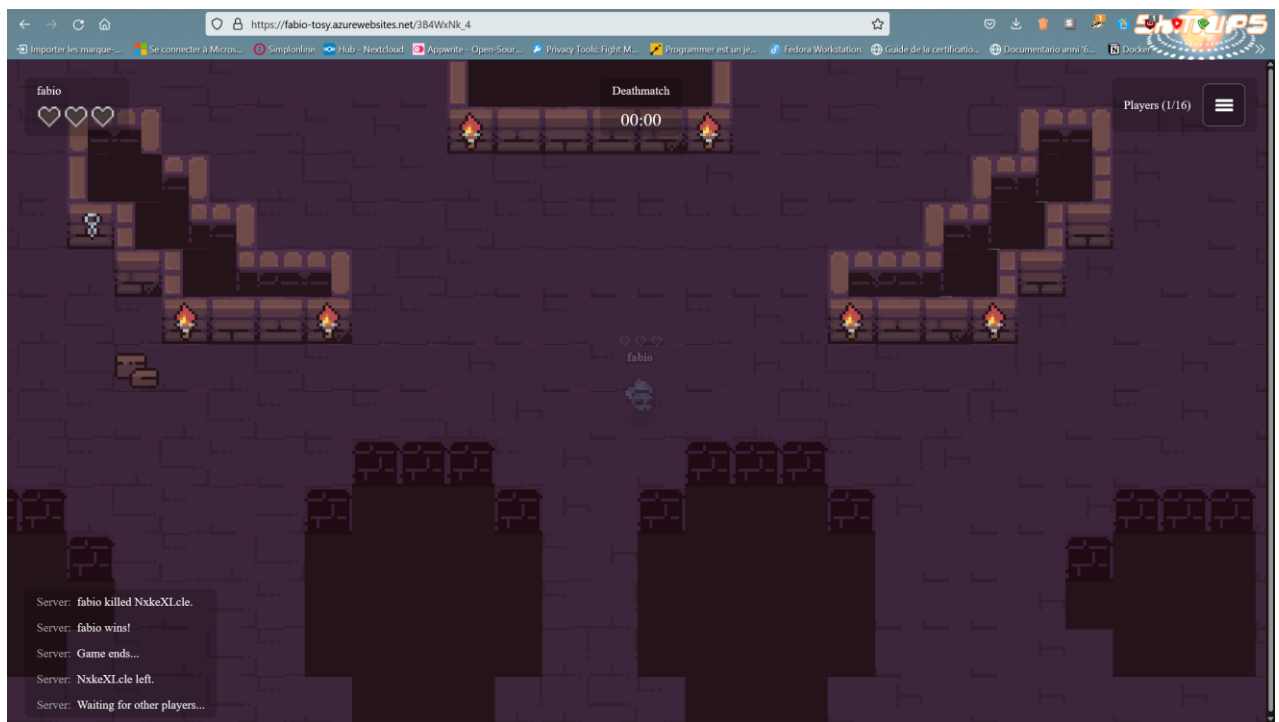
Déploiement continu	Non activé/configuré après la création de l'application
---------------------	---

Créer

< Précédent

Suivant >

[Télécharger un modèle pour automation](#)



3 – Activation du déploiement continu sur l'application

Dans le menu de votre application web, sélectionnez centre de déploiement sur la gauche

Choisissez GitHub Actions comme source puis renseignez votre organisation, dépôt et branche une fois la liaison validée.

[Enregistrer](#) [Abandonner](#) [Parcourir](#) [Gérer le profil de publication](#) [Laisser un commentaire](#)

[Paramètres *](#) [Journaux](#) [Informations d'identification FTPS](#)

Utilisez ces paramètres pour configurer votre modèle et votre registre de déploiement d'application conteneur. Nous recommandons GitHub Actions pour une plus grande efficacité opérationnelle. [En savoir plus](#)

Source *

☐ Registre de conteneurs: Configurez votre application pour extraire l'image conteneur d'un registre.

☒ GitHub Actions: Créez, déployez et gérez automatiquement votre application conteneur avec GitHub Actions.

☐ Azure Pipelines: Configurez un pipeline de déploiement robuste pour votre application à l'aide d'Azure Pipelines, qui fait partie des services Azure DevOps (anciennement VSTS).

GitHub Actions

App Service placera un flux de travail GitHub Actions dans le dépôt choisi pour générer et déployer votre application chaque fois qu'une validation est effectuée sur la branche choisie. Si vous ne trouvez pas d'organisation ou de dépôt, vous devrez peut-être activer des autorisations supplémentaires sur GitHub. Vous devez avoir un accès en écriture au dépôt GitHub choisi pour effectuer un déploiement avec GitHub Actions. [En savoir plus](#)

Connecté comme MPFabio [Changer de compte](#) ⓘ

Organisation *

Dépôt *

Branche *

Option de workflow *

☒ Ajouter un workflow : ajouter un nouveau fichier de workflow 'master_Fabio-Tosy.yml' dans le dépôt et la branche sélectionnés.

☐ Utiliser le workflow disponible : utiliser l'un des fichiers de workflow disponibles dans le dépôt et la branche sélectionnés.



Sign in to **GitHub**
to continue to **Azure App Service**

Username or email address

MPFabio

Password

[Forgot password?](#)

.....

Sign in

New to GitHub? [Create an account.](#)



Authorize Azure App Service



Azure App Service by [Azure App Services](#)
wants to access your MPFabio account



Repository webhooks and services
Admin access



Repositories
Public and **private**



Workflow
Update GitHub Action Workflow files.



Cancel

Authorize AzureAppService

