

DevOps avec Bluemix

Adoptez une approche DevOps en utilisant le Continuous Delivery, qui inclut une chaîne d'outils ouverte automatisant la génération et le déploiement d'applications. Vous pouvez commencer en créant une chaîne d'outils de déploiement simple qui prend en charge les tâches de développement, de déploiement et les opérations.

Vous utiliserez l'application Todo avec les outils DevOps de Bluemix.



Fork me on GitHub

Type a new todo

☐

subscribe to Bluemix

☐

deploy to Bluemix

☐

create a tutorial

☐

Select All

Clear selected

Objectif

Dans l'exercice suivant, vous allez apprendre à :

- Mettre en place un dépôt pour votre code source afin de collaborer
- Gérer l'intégration continue et le déploiement continu

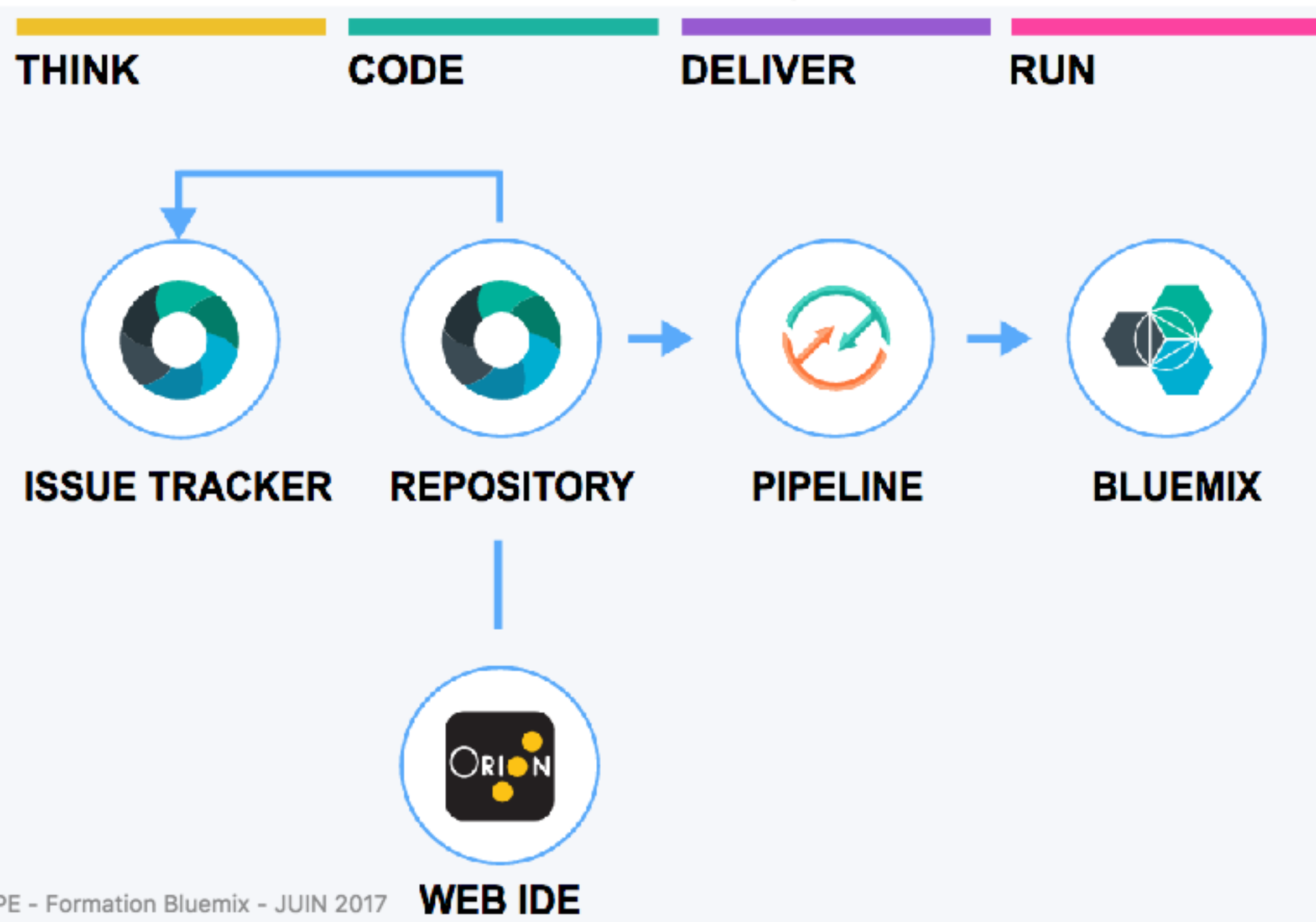
Prérequis

- Avoir un [Bluemix IBM id](#), ou utiliser son compte existant.
- Installer le [Bluemix CLI](#)
- Installer un [Git client](#)
- Installer [Node.js](#)

Etapes

1. Activer le déploiement continu
2. Soumettre votre changement et le voir se déployer automatiquement

- soit par IBM et fonctionne avec GitLab Community Edition en mode privé.



THINK

CODE

DELIVER

RUN



ISSUE TRACKER
Github



REPOSITORY
Github



PIPELINE



BLUEMIX



1. Depuis la page **Overview** de votre application, recherchez **Continuous Delivery** et cliquez sur le bouton **Enable**.
2. Une nouvelle fenêtre s'ouvre pour configurer la chaîne d'outil, Toolchain.
La Toolchain contient un dépôt Git, un Pipeline de déploiement et un IDE web.

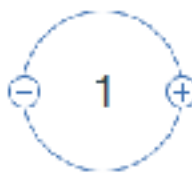
- Getting started
- Overview
- Runtime
- Connections
- Logs
- Monitoring
- API Management

webapp-jd Running [Visit App URL](#)

Routes   



BUILDPACK
SDK for Node.js™



INSTANCES
All instances are running
Health is 100%



MB MEMORY PER INSTANCE



TOTAL MB ALLOCATION
213.75 GB still available

Connections (1)



webapp-cloudant-jd

Connect new

Connect existing

Runtime cost

0,00 €

Current charges for billing period

0,00 €

Estimated total for billing period
(Jun 1, 2017 - Jun 30, 2017)

Current and estimated cost excludes [connected services](#).

[View full usage details](#)

Activity feed

No activity available

Continuous delivery

Continuous delivery is not enabled for this app.
Enable continuous delivery to automate builds, tests, and deployments through the Delivery Pipeline, GitHub, and more.

activer le déploiement continu

Enable

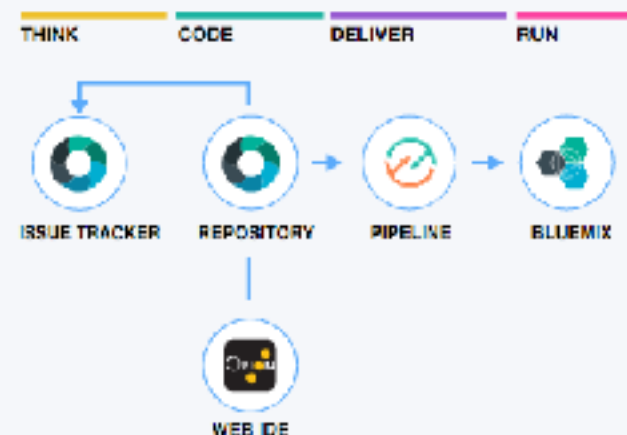
1. La toolchain propose un nom par défaut qui est modifiable. Dans **Configurable Integrations** en bas, sélectionner **Git Repos et Issue Tracking**.
2. Garder les options par défaut **Clone** pour cloner le code source de l'application "Hello World!" dans votre compte GitLab ou GitHub.

Continuous Delivery Toolchain

This toolchain includes tools to develop and deploy your app. Depending on your app, when you create the toolchain, the Git repository will either be empty or will contain source code from your app.

This toolchain uses tools that are part of the Continuous Delivery service. If an instance of that service isn't already in your organization, when you click **Create**, it is automatically added along with you. For more information and links, see the Bluemix catalog.

To get started, click **Create**.



Organization

CDO

Toolchain Name

hello-jd

Tool Integrations



Git Repository and Issue Tracking



Eclipse Orion Web IDE



Delivery Pipeline

Create

1. La toolchain a été configuré avec succès. Un nouveau dépôt Git a été créé, ainsi qu'un Pipeline qui pourra déployer votre application automatiquement à chaque commit.

IBM Bluemix DevOps

OPEN Groupe | US South | ODO | dev

Catalog Support Manage

Toolschain

Overview

Connections

Manage

webapp-jd

View app

Add a Tool

THINK

CODE

DELIVER

Issues webapp-jd

Git webapp-jd

Delivery Pipeline webapp-jd

Eclipse Orion Web IDE

✓ Configured

✓ Configured

✓ Configured

✓ Configured

Add a Calendar...

1. Ouvrez le dépôt Git et notez son URL.
2. Ouvrir un terminal ou une invite de commande afin de cloner le repository git

```
git clone <URL-OF-YOUR-GIT-REPO>
```

3. Cette commande crée un répertoire de votre projet localement sur votre disque dur.

Dans le chapitre précédent, vous poussiez le changement manuellement. Ici, vous bénéficierez des fonctionnalités du dépôt Git et du pipeline de déploiement automatique.

Etape 2 - Soumettre votre changement et le voir se déployer automatiquement

1. Depuis la console Bluemix, aller dans l'onglet **Overview** de votre application.
2. Cliquer sur le bouton **View Toolchain** dans la section Continuous Delivery.

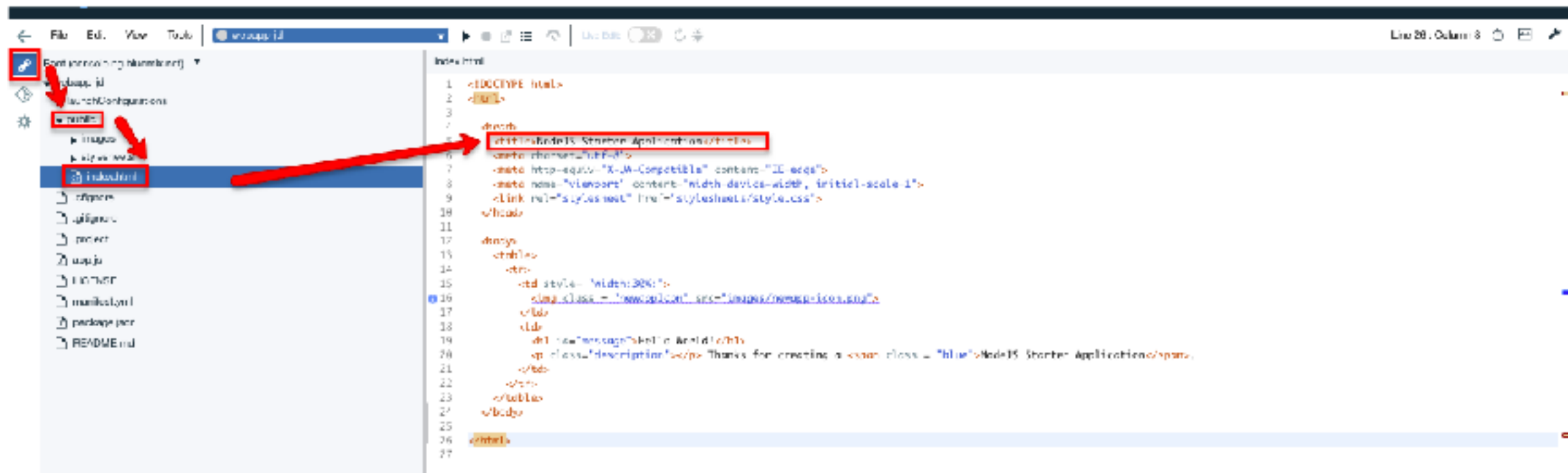
1. Cliquer sur ORION, la Web IDE

The screenshot shows the IBM Bluemix DevOps Toolschain interface for an application named 'webapp-jd'. The top navigation bar includes a user profile, 'OPEN Groups', 'US South', 'COD', 'dev', and a user icon. A yellow callout bubble points to the 'Add a Calendar...' button. The left sidebar contains 'Toolschains', 'Overview', 'Connections', and 'Manage'. The main area displays four tool cards under the heading 'webapp-jd':

- THINK**: Issues (webapp-jd) - Configured
- CODE**: Git (webapp-jd) - Configured
- DELIVER**: Delivery Pipeline (webapp-jd) - Configured
- Eclipse Orion Web IDE** - Configured (highlighted with a red box)

An 'Add a Tool' button is visible in the top right corner of the main area.

1. Une nouvelle fenêtre s'ouvre, vous permettant d'accéder en édition à votre code source.



1. Ouvrir le fichier **public/index.html**.
2. Changer le titre de la page à la ligne 5.
3. Sauvegarder le changement.

1. Soumettre le changement via l'icône Git.

The screenshot shows the IBM Bluemix IDE interface with several red annotations and arrows indicating the steps to submit a change via Git:

- se placer dans le contexte Git**: Points to the Git icon in the top left toolbar.
- nommer le changement**: Points to the "update [n]" input field in the "Viewing Directory Changes" panel.
- commit du changement**: Points to the "Commit" button in the top right corner.
- pousser le changement**: Points to the "Push" button in the "Viewing Directory Changes" panel.
- visualiser la modification**: Points to the code editor showing the changes.

The code editor displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <title>NodeJS Starter Application</title>
6   <title>Mon application #2</title>
7   <meta charset="utf-8">
8   <meta http-equiv="X-UA-Compatible" content="IE=edge">
9   <meta name="viewport" content="width=device-width, initial-scale=1">
10   <link rel="stylesheet" href="stylesheets/style.css">
11 </head>
12
13 <body>
14   <div>
15     <div style="width:300px">
16       
17     </div>
18     <div>
19       <div id="message">Hello World</div>
20       <div class="description"></div> Thanks for creating a <span class = "blue">NodeJS Starter Application</span>.
21     </div>
22   </div>
23 </body>
24 </html>
25
```

1. Revenir sur **Delivery Pipeline**.

2. Regarder la prise en compte de votre changement par le Delivery pipeline qui redéploie l'application.

← Toolchain
webapp-jd | Delivery Pipeline



The screenshot displays the Bluemix Delivery Pipeline interface for a toolchain named 'webapp-jd'. It shows two stages: 'Build Stage' and 'Deploy Stage'. The 'Build Stage' has a 'LAST INPUT' section with a commit link and a 'LAST EXECUTION RESULT' section showing a successful build. The 'Deploy Stage' has a 'LAST INPUT' section showing the build as input and a 'LAST EXECUTION RESULT' section showing a successful deployment of the application. Red arrows indicate the flow from the build stage to the deploy stage. A blue 'Add Stage' button is visible on the right.

Build Stage

READY FOR BUILD

LAST INPUT: Last commit by joranku@users. [View commit](#)

JOB: Build Passed 2m 40s

LAST EXECUTION RESULT: Build 1

Deploy Stage

READY FOR BUILD

LAST INPUT: Stage Build Stage / Job Build

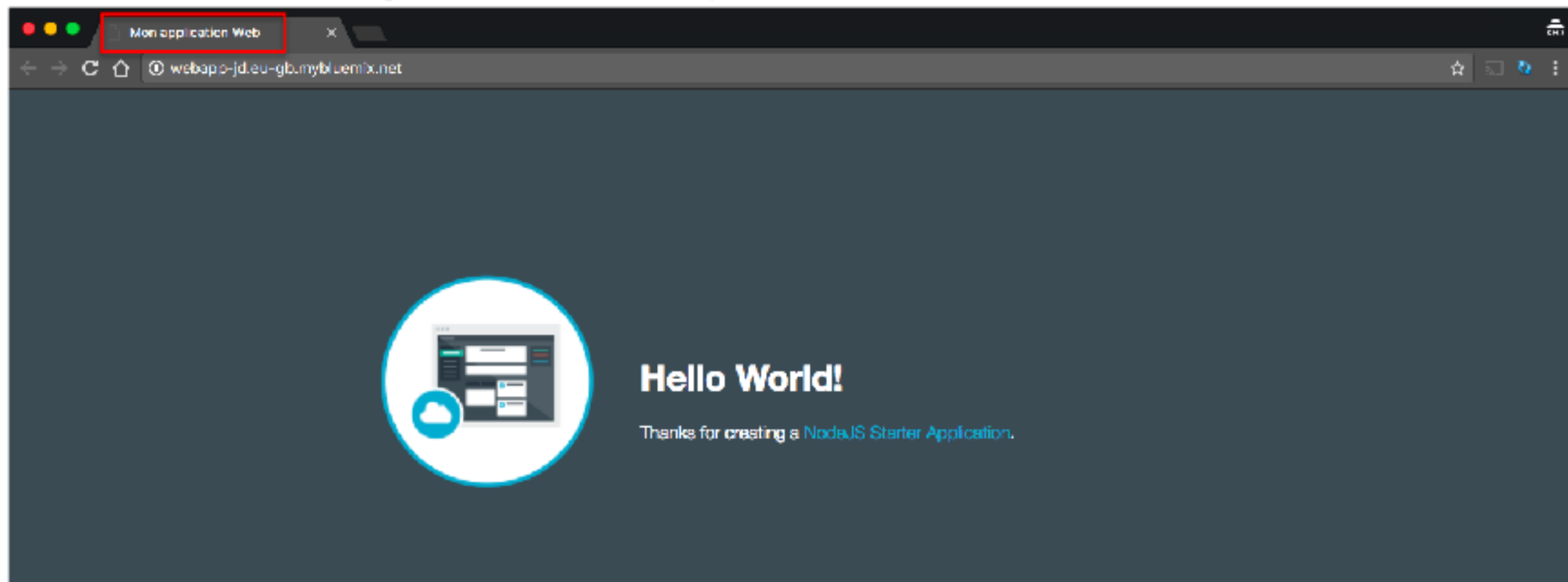
JOB: Deploy Passed now

LAST EXECUTION RESULT: webapp-jd [View commit](#) [View runtime log](#)

Build 1

Add Stage

1. Quand la commande est terminée, accéder à l'application s'exécutant dans le cloud pour confirmer que le changement a été déployé.



1. Il est maintenant nécessaire de copier le code source de l'application Todo vers ce nouveau répertoire local créé par la commande git clone.
2. Ajouter les nouveaux fichiers à Git.

```
git add .
```

3. Soumettre le changement.

```
git commit -a -m "full solution"
```

Note: Vous devrez peut-être configurer pour la première fois:

```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

4. Pousser votre changement.

```
git push
```

1. Revenir sur **Delivery Pipeline**.

2. Regarder la prise en compte de votre changement par le Delivery pipeline qui redéploie l'application.

← Toolchain webapp-jd | Delivery Pipeline

Build Stage

READY FOR BUILD

LAST INPUT [View Build](#)

Last commit by jason.choua [View Build](#)

JOB [View Jobs](#)

Build Passed 100%

LAST EXECUTION RESULT

Build [View Build](#)

>

Deploy Stage

READY FOR DEPLOY

LAST INPUT [View Build Stage / Job Build](#)

Build [View Build](#)

JOB [View Jobs](#)

Deploy Passed 100%

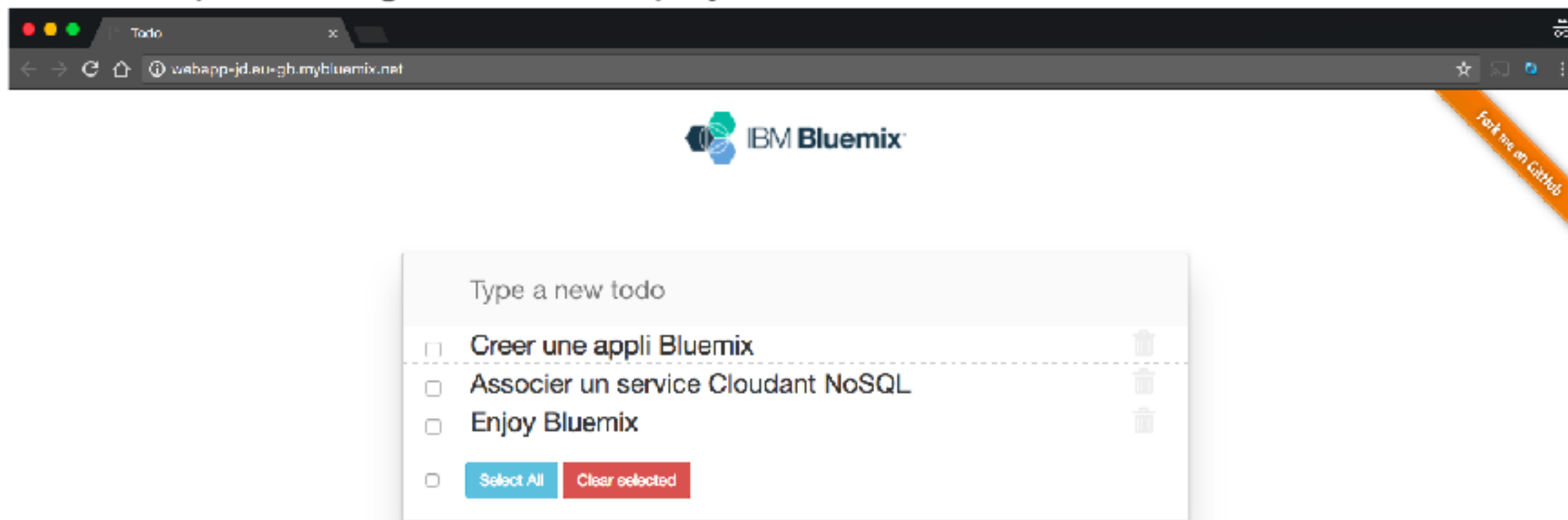
LAST EXECUTION RESULT

webapp-jd [View Build](#)

Build [View Build](#)

Add Stage

1. Quand la commande est terminée, accéder à l'application s'exécutant dans le cloud pour confirmer que le changement a été déployé.



Suivre le chapitre suivant [Les containers avec Bluemix.](#)

Enjoy Bluemix ! 