eZComptes

Ce document est destiné aux développeurs qui souhaitent mettre en place un environnement de développement eZComptes.

Dossier d’installation de l’environnement de développement v1.0

# Table des illustrations

[Figure 1 Installation du SDK Tools d’Androïd 5](#_Toc375773626)

[Figure 2 Installation des packages Androïd 5](#_Toc375773627)

[Figure 3 Configuration du plugin ADT 7](#_Toc375773628)

Sommaire

[Table des illustrations 1](#_Toc503626225)

[Suivi des modifications 4](#_Toc503626226)

[Caracteristiques techniques 4](#_Toc503626227)

[1. Installation 5](#_Toc503626228)

[Pré-requis 5](#_Toc503626229)

[Comptes utilisateurs 5](#_Toc503626230)

[Docker 5](#_Toc503626231)

[2. Installer Java 5](#_Toc503626232)

[1. Installer Maven 6](#_Toc503626233)

[2. Installer Gradle 6](#_Toc503626234)

[3. Installer Eclipse 6](#_Toc503626235)

[4. Installer GIT 6](#_Toc503626236)

[5. Installer JHipster 6](#_Toc503626237)

[6. Installer les plugiciels Eclipse 7](#_Toc503626238)

[EGit 7](#_Toc503626239)

[Checkstyle 8](#_Toc503626240)

[EclEmma 8](#_Toc503626241)

[FindBugs 8](#_Toc503626242)

[PMD 8](#_Toc503626243)

[CodePro AnalytiX de Google 9](#_Toc503626244)

[Usus 9](#_Toc503626245)

[Metrics 9](#_Toc503626246)

[JDepend 9](#_Toc503626247)

[Plugin Google 9](#_Toc503626248)

[7. Configurer Eclipse 9](#_Toc503626249)

[Configurer Java 10](#_Toc503626250)

[Configurer Gradle 10](#_Toc503626251)

[8. Master de développement 10](#_Toc503626252)

[9. Checkout des sources GitHub 10](#_Toc503626253)

[10. Démarrer l’application 10](#_Toc503626254)

[10.1. Démarrer l’application jhIPSTER-registry 10](#_Toc503626255)

[10.2. Démarrer l’application ezcomptes-microservice-application 11](#_Toc503626256)

# Suivi des modifications

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Description | Modifié par | Version |
| 25/12/2013 | Initialisation du document | TGI | v1.0 |
|  |  |  |  |
|  |  |  |  |

# Caracteristiques techniques

Le poste de développement doit répondre au minimum aux caractéristiques suivantes :

|  |  |
| --- | --- |
| Caractéristiques | |
| Système d’exploitation | Windows |
| Type du système | Système d’exploitation 64 bits |
| Processeur | Intel® Core 2 Duo CPU P8700 @ 2.53 GHz 2.53 GHz |
| Mémoire installée (RAM) | 4,00 Go |

# Installation

## Pré-requis

Création d’une machine virtuelle avec Hyper-V

|  |  |
| --- | --- |
| Caractéristiques | |
| Système d’exploitation | Centos 7 Linux |
| Type du système | Système d’exploitation 64 bits |
| Processeur | Intel® Core 2 Duo CPU P8700 @ 2.53 GHz 2.53 GHz |
| Espace disque | 10 Go |
| Mémoire installée (RAM) | 4 Go |

## Comptes utilisateurs

|  |  |  |  |
| --- | --- | --- | --- |
| Utilisateur | Mot de passe | UID | GID |
| root | tg82843d |  |  |
| docker | tg82843d | 1000 | 1000 |

## Docker

Installation de Docker

JHipster

Ansible

Docker

Eureka

Zuul

ZooKeeper

Spring boot

Spring Configuration

Spring cloud

Spring data

Angular 5

HTML5/CSS3

ELK

# Installer Java

1. Télécharger le Java SE Development Kit (JDK) 8.0
2. Installer le JDK dans le répertoire par défaut C:\Program Files\Java\jdk1.8.0\_151\
3. Ajouter les variables d’environnement :
   * Créer la variable d’environnement JAVA\_HOME avec la valeur C:\Program Files\Java\jdk1.8.0\_151
   * Ajouter %JAVA\_HOME\bin; au début de la variable PATH de Windows
   * Ajouter %JAVA\_HOME\lib\tools.jar; au début de la variable CLASSPATH de Windows
4. Dans la console, faites java –version pour vérifier l’installation de Java

Nom du fichier: jdk-8u151-windows-x64.exe

Lien de téléchargement: [http://www.oracle.com/technetwork/java/javase/downloads/](http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase6-419409.html)

# Installer Maven

1. Télécharger Maven 3.5.2
2. Extraire le contenu de l’archive dans le répertoire « C:\01-outils\ »
3. Ajouter Maven dans la variable d’environnement PATH :
   * Ajouter C:\01-outils\apache-maven-3.5.2\bin; au début de la variable PATH de Windows
4. Dans la console, faites mvn –version pour vérifier l’installation de Maven

Nom de l’archive: apache-maven-3.5.2-bin.zip

Lien de téléchargement:  [https://maven.apache.org/download.cgi](http://maven.apache.org/download.html)

# Installer Gradle

1. Télécharger Gradle 4.4.1
2. Extraire le contenu de l’archive dans le répertoire « C:\01-outils »
3. Ajouter Gradle dans la variable d’environnement PATH :
   * Ajouter C:\01-outils\gradle-4.4.1\bin; au début de la variable PATH de Windows
4. Dans la console, faites gradle –version pour vérifier l’installation de Gradle

Nom de l’archive: gradle-4.4.1-bin.zip

Lien de téléchargement:  [https://gradle.org/install/](http://maven.apache.org/download.html)

# Installer Eclipse

1. Télécharger Eclipse Oxygen (Eclipse IDE **for Java EE Developpers**)
2. Extraire le contenu de l’archive dans le répertoire « C:\01-outils\eclipse-jee-oxygen-2-win32-x86\_64 »

Nom de l’archive: eclipse-jee-oxygen-2-win32-x86\_64.zip

Lien de téléchargement: <http://www.eclipse.org/downloads/>

# Installer GIT

1. Télécharger Git 2.15.1.2
2. Installer Git dans le répertoire « C:\01-outils\Git-2.15.1.2-x64
3. Ajouter Git dans la variable d’environnement PATH :
   * Ajouter C:\01-outils\Git-2.15.1.2-x64\cmd; au début de la variable PATH de Windows
4. Dans la console, faites git version pour vérifier l’installation de Git

Nom du fichier: Git-2.15.1.2-64-bit.exe

Lien de téléchargement: <http://git-scm.com/download/win>

# Installer JHipster

1. Installer Node.js (préférer une version LTS) : <https://nodejs.org/en/>
   * Installer Node.js dans le répertoire « C:\01-outils\node-v8.9.4-x64\ »
2. Installer Yarn : <https://yarnpkg.com/en/docs/install>
   * Installer Yarn dans le répertoire « C:\01-outils\yarn-1.3.2\ »
3. Installer Python : <https://www.python.org/downloads/>
   * Installer Python dans le répertoire « C:\01-outils\python-3.6.4\ »
4. Pour installer le Marketplace JHipster, installer Yeoman: yarn global add yo
5. Installer JHipster : yarn global add generator-jhipster

Lien : <http://www.jhipster.tech/installation/>

# Installer les plugiciels Eclipse

1. Aller dans le menu « Help > Install New Software… »
2. Dans la fenêtre, sélectionner ou ajouter l’item « Work with » indiqué pour chacune des sections ci-dessous.
3. Cocher les composantes indiquées ci-dessous. Installer les composantes.
4. Redémarrer Eclipse.

## EGit

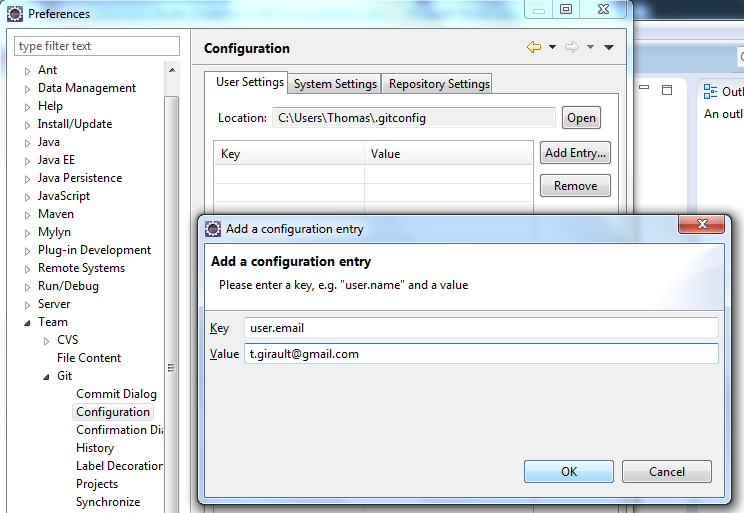
Work with: http://download.eclipse.org/egit/updates

*Composantes:*

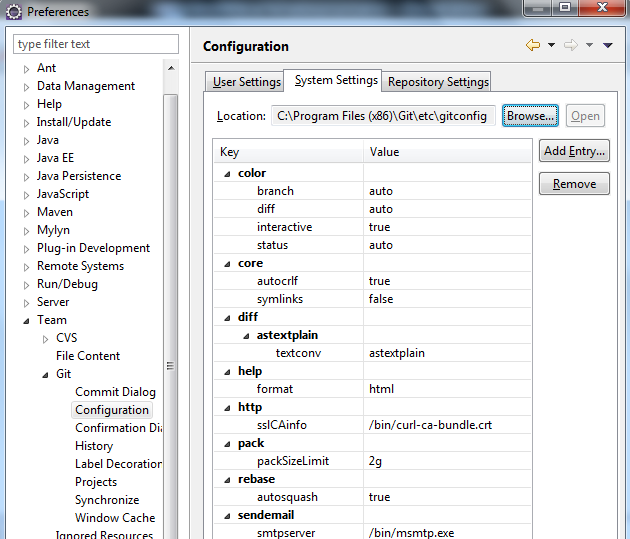
* + …

Après redémarrage d'eclipse, se déplacer dans : Windows > Preferences > Team > Git > Configuration

* Définir les paramètres utilisateur :
  1. Onglet "User Settings"
  2. Cliquer sur "Add Entry..."
  3. Ajouter les clés user.name et user.email



* Définir le chemin vers le répertoire d'installation de Git
  1. Onglet "System Settings"
  2. Cliquer sur "Browse..." et se déplacer dans le répertoire d'installation de Git



* Configurer le chemin du repository local

Ajouter la variable d'environnement HOME

* 1. Sur Window 10, taper "environnement" dans le menu démarrer
  2. Sélectionner "Modifier les variables d'environnement pour votre compte"
  3. Cliquer sur "Nouvelle..."
  4. Saisir "HOME" comme nom de variable
  5. Saisir "%USERPROFILE%" comme valeur de la variable

Lien utile: [http://wiki.eclipse.org/EGit/User\_Guide](http://maven.apache.org/download.html)

## Checkstyle

Work with: <http://eclipse-cs.sf.net/update/>

*Composantes:*

* Checkstyle
  + Eclipse Checkstyle Plug-in

## EclEmma

Work with: <http://update.eclemma.org/>

*Composantes:*

* EclEmma (tous)

## FindBugs

Work with: <http://findbugs.cs.umd.edu/eclipse>

*Composantes:*

* FindBugs (tous)

## PMD

Work with: <http://pmd.sourceforge.net/eclipse>

*Composantes:*

* PMD for Eclipse 3 (tous)

## CodePro AnalytiX de Google

Work with : <http://dl.google.com/eclipse/inst/codepro/latest/3.7>

*Composantes:*

* CodeCoverage (tous)
* CodePro (tous)
* Infrastructure (tous)

## Usus

Work with:<http://projectusus.googlecode.com/svn/updates/>

*Composantes:*

* Project Usus (tous)

## Metrics

Work with: <http://metrics.sourceforge.net/update>

*Composantes:*

* Uncategorized
  + Metrics plugin for Eclipse (tous)

## JDepend

Work with: <http://andrei.gmxhome.de/eclipse/>

*Composantes:*

* Eclipse 3.5 - 4.2 plugins
  + DataHierarchy
  + JDepend4Eclipse

## Plugin Google

Work with: https://dl.google.com/eclipse/plugin/4.3

*Composantes:*

* Developper Tools
* Google App Engine Tools for Android (requires ADT)
* Google Plugin for Eclipse (required)

Après le redémarrage d’eclipse, configurer le plugin Google.

Ajouter le chemin vers le SDK Google App Engine

1. Dans le menu « Window > Preferences »
2. Sélectionnez l’onglet « Google > App Engine »
3. Ajouter le chemin du SDK précédemment installée (Add…)
   * Exemple : « C:\Outils\appengine-java-sdk-1.8.8 »

# Configurer Eclipse

## Configurer Java

Vérifier qu’Eclipse utilise le bon JDK

1. Menu « Help > About Eclipse »
2. « Installation Details » puis onglet « Configuration »
3. Rechercher la ligne « java.home = »
   * Elle devrait contenir le chemin vers votre JDK suivi de « \jre » (ex : java.home = C:\Program Files\Java\jdk1.8.0\_151\jre)
   * Si elle pointe vers une JRE seule (ex : java.home = C:\Program Files\Java\jre8) alors vous devez corriger la situation

Ajouter une JDK

1. Dans le menu « Window > Preferences »
2. Sélectionnez l’onglet « Java > Installed JREs »
3. Ajouter le chemin de la JDK précédemment installée (Add…)
   * Exemple : « C:\Program Files\Java\jdk1.8.0\_151 »

## Configurer Gradle

Ajouter une installation externe de Gradle (à défaut de la version Gradle « Embedded »)

1. Dans le menu « Window > Preferences »
2. Sélectionnez l’onglet « Gradle »
3. Sélectionner « Local installation directory » et renseigner le chemin d’installation de Gradle
   * Exemple : « C:\01-outils\gradle-4.4.1 »
4. Dans le champ « Gradle User Home », renseigner le chemin du repository local Gradle
   * Exemple : C:\02-mes-projects\ezcomptes\gradle-repository

# Master de développement

TODO

# sources

## Gestion de version Git

### Pré-requis :

* Disposer d’un compte GitHub : <https://github.com/tgirault>
* Git est installé sur le poste de développement

### Créer un nouveau projet

1. Créer un nouveau repository sur GitHub.

Pour éviter des erreurs, ne pas initialiser un nouveau repository avec un README, une license, ou des fichiers gitignore. Ces fichiers pourrons être ajoutés après que le projet soit poussé vers GitHub

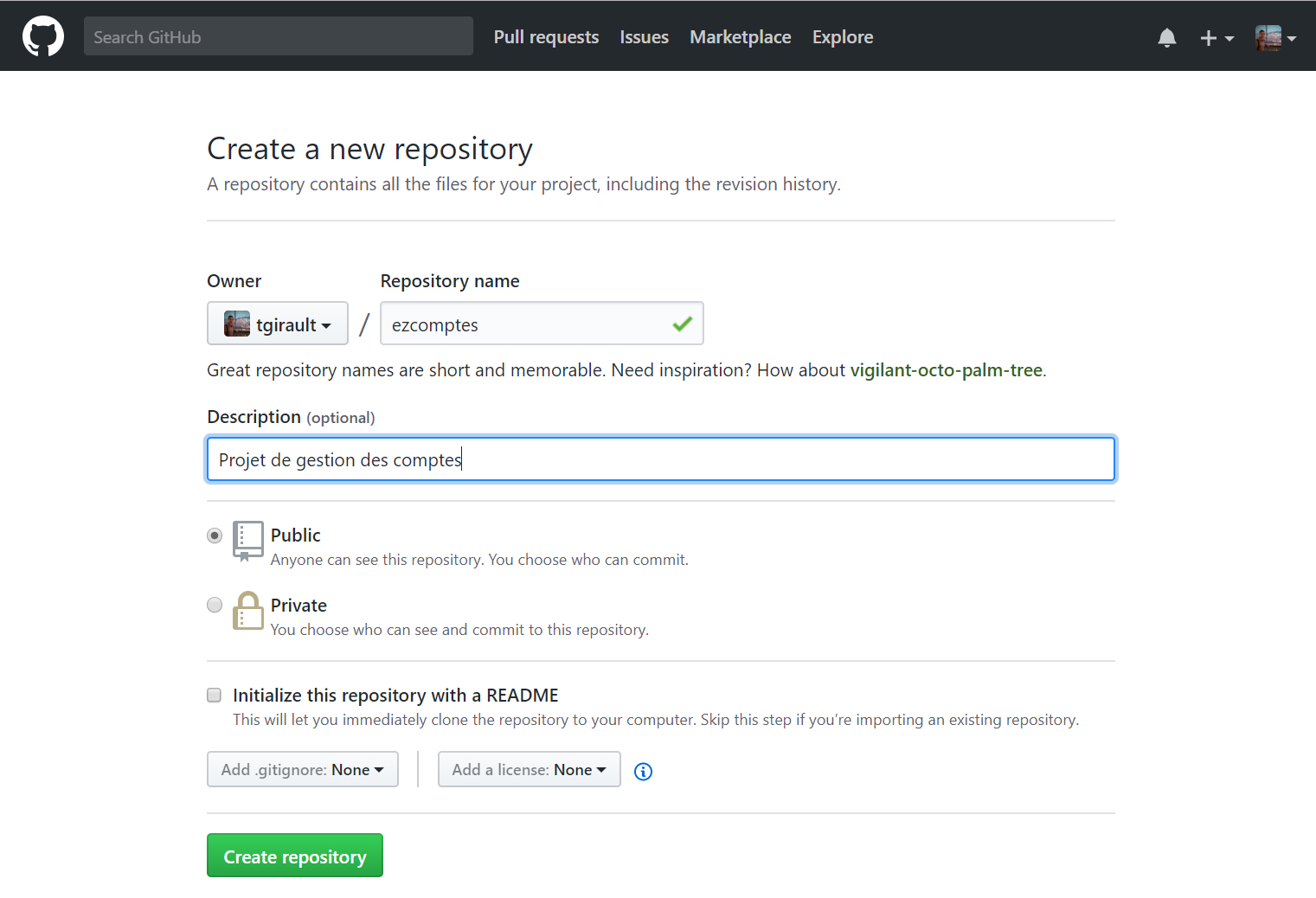


Figure Création d'un repository GitHub

Nouveau repository : https://github.com/tgirault/ezcomptes.git

1. Ouvrir un invité de commandes dans le répertoire du projet local
2. Initialiser le répertoire local en tant que repository Git

git init

1. Ajouter les fichiers dans le nouveau repository local

git add .

# Adds the files in the local repository and stages them for commit. To unstage a file, use 'git reset HEAD YOUR-FILE'.

1. Commit les fichiers sélectionnés dans le repository local

git commit -m "First commit"

# Commits the tracked changes and prepares them to be pushed to a remote repository. To remove this commit and modify the file, use 'git reset --soft HEAD~1' and commit and add the file again.

1. Copier l’URL du repository distant en haut de la page du repository GitHub
   * https://github.com/tgirault/ezcomptes.git

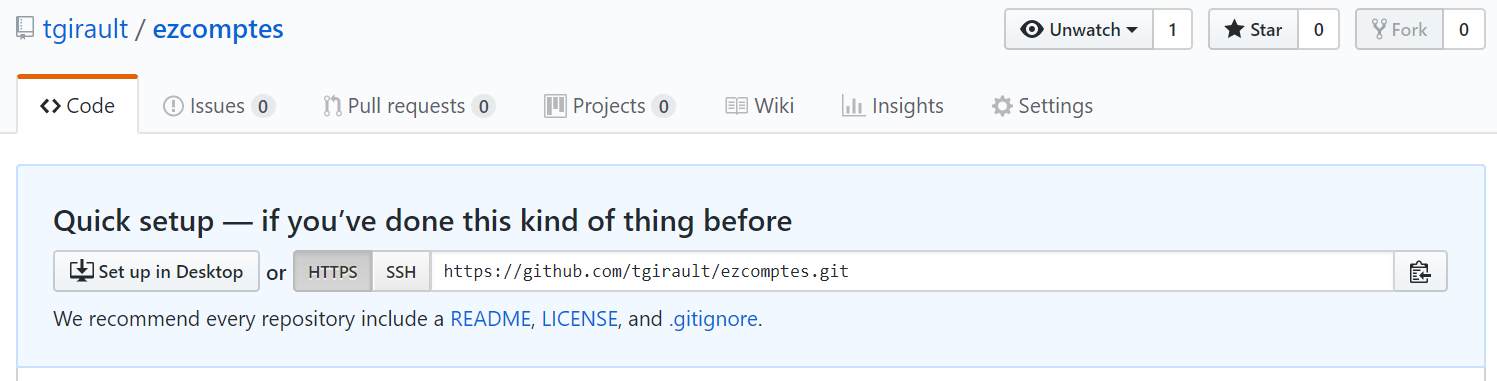


Figure URL du repository GitHub

1. Dans l’invité de commandes, ajouter l’URL du repository distant vers lequel seront push les fichiers du repository local

git remote add origin <remote repository URL>

# Sets the new remote

git remote -v

# Verifies the new remote URL

1. Pousser les fichiers présents dans le repository local vers le repository GitHub

git push origin master

# Pushes the changes in your local repository up to the remote repository you specified as the origin

Source : <https://help.github.com/articles/adding-an-existing-project-to-github-using-the-command-line/>

### Récupérer les sources d’un projet GitHub

# Démarrer l’application

## Démarrer l’application jhIPSTER-registry

Se déplacer dans le répertoire du projet :

* C:\02-mes-projects\ezcomptes\01-repository\ezcomptes-backend\jhipster-registry

1. Lancer la commande d’installation de Sass :

yarn add node-sass

1. Lancer la commande d’installation de webpack et webpack-dev-server :

yarn add --dev webpack webpack-dev-server

1. Démarrer le serveur :

.\mvnw –Pdev –Pwebpack

L’application est accessible à l’URL : <http://localhost:8761>

## Démarrer l’application ezcomptes-microservice-application

Lancer la tâche gradle :

L’application est accessible à l’URL : <http://localhost:8080>