



RAPPORT DU TP1 : JAVA EE 7

Septembre 2020

Réalisé par :
AMANAR Abderrahim

Master Spécialisé Ingénierie de Données et Développement Logiciel
ANNEE ACADEMIQUE : 2019/2020

Table des matières

1 Java EE 7 3

1.1	Objectif	2
1.2	Installation de NetBeans	2
1.3	Configuration de NetBeans pour le développement Java EE	08
1.3.1	L'intégration de NetBeans avec un serveur d'application	08
1.3.2	L'intégration d'un SGBDR avec NetBeans	10
1.3.3	La Connexion à un JDBC	11
1.4	Déploiement d'une première application	11

TP1: Java EE 7

1.1 Objectif

L'objectif de ce TP est la prise en main de l'environnement de développement Java EE 7. Plus précisément, la manipulation consiste à effectuer l'installation et la configuration de NetBeans 8.2

1.2 Installation de NetBeans

La plate forme NetBeans est un environnement de développement intégré (*Integrated Development Environment IDE*) supporte plusieurs langages de programmation et technologies comme Java, JavaFX, C, C++ et PHP.

NetBeans peut être téléchargé à partir du site <https://netbeans.org/downloads/old/8.2/>

Le tableau suivant résume les différentes fonctionnalités et technologies prises en charge par les différentes versions de NetBeans. (Voir Fig 1)

NetBeans IDE 8.2 Download 8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly ☒ NetBeans can contact me at this address

IDE Language: English Platform: Windows

Note: Greyed out technologies are not supported for this platform.

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download Download Download x86 Download x86 Download x86 Download x64 Download x64 Download x64 Download

Free, 95 MB Free, 197 MB Free, 108 - 112 MB Free, 108 - 112 MB Free, 107 - 110 MB Free, 221 MB

Figure 1 – Les différentes versions offertes par NetBeans

NetBeans 8.2 nécessite l'installation d'un *Java Development Kit* (JDK) version 8.0 ou plus récente (la version choisie ici est 8.0u241).

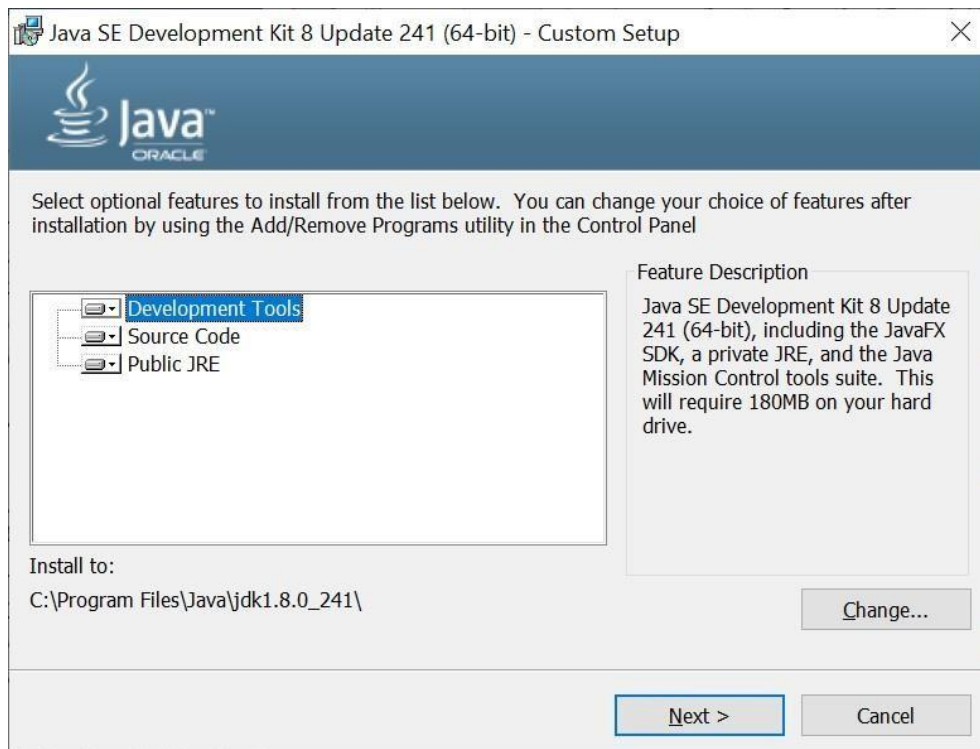


Figure 2 – Assistant d'installation JDK 8.0u241

Après l'exécution du fichier d'installation de NetBeans 8.2, une fenêtre similaire à celle illustrée à la Figure 2 apparaît :

Pour continuer l'installation de NetBeans, cochez les deux serveurs et cliquez sur le bouton **NEXT**

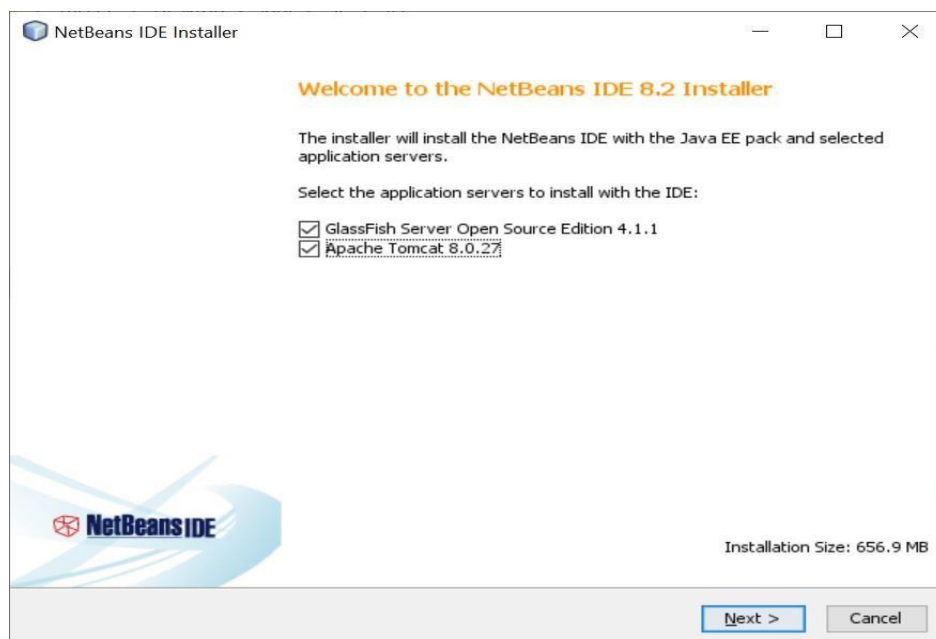


Figure 3 – Installateur Netbeans : Les serveurs d'applications à installer avec l'IDE L'installateur va maintenant nous inviter à spécifier un répertoire d'installation pour NetBeans, et celui du JDK à utiliser avec NetBeans.

(Dans notre cas nous avons choisi la version 8.0_241 du JDK)

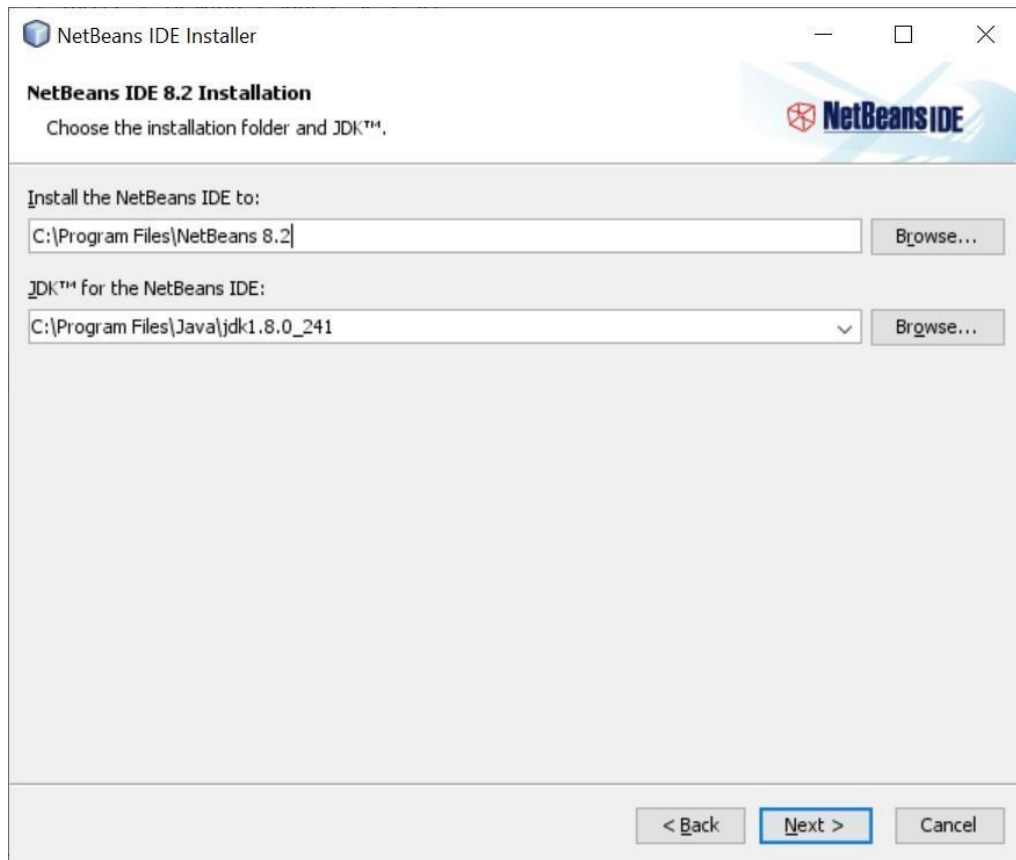


Figure 4 – Installateur Netbeans : Répertoire d'installation pour l'IDE NetBeans

Une fois les répertoires d'installation appropriés sont choisis, cliquez sur le bouton **Next** pour continuer L'installation.

A cette étape l'installateur va nous inviter à choisir un répertoire d'installation pour le serveur GlassFish 4.1.1, et celui du JDK à utiliser ce serveur. On peut soit choisir de nouvelles valeurs ou prendre celles fournies par défaut.

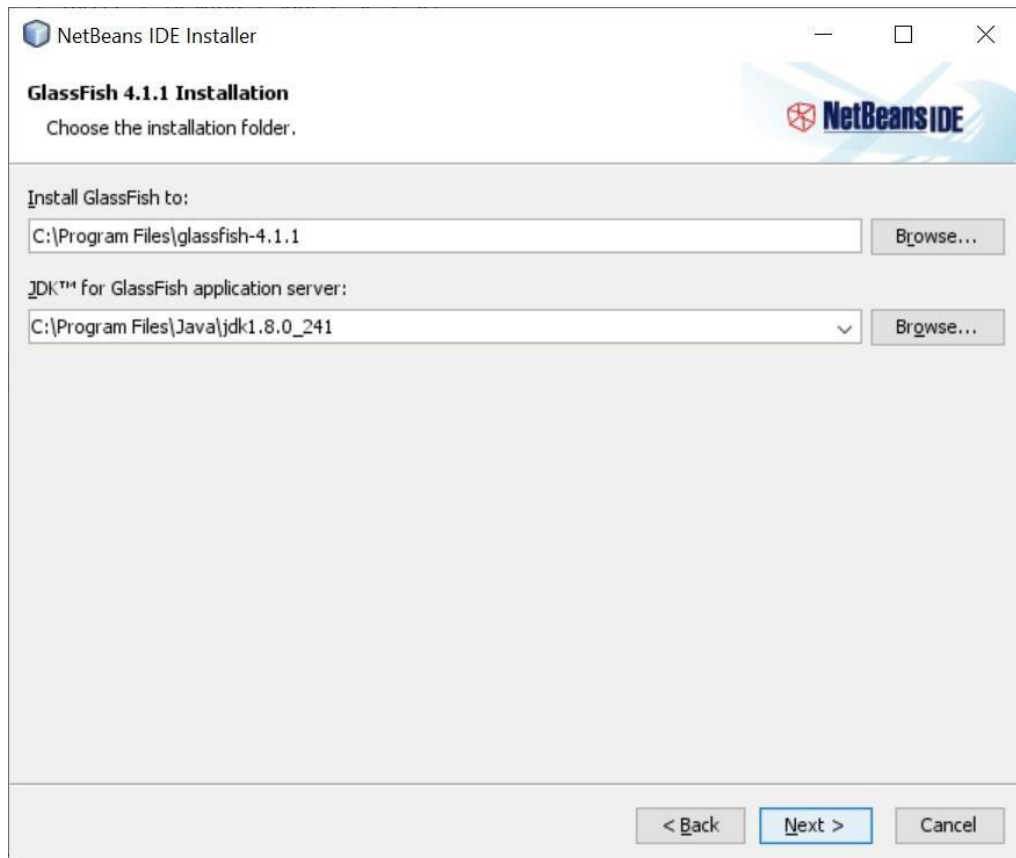


Figure 5 – Installateur Netbeans : Répertoire d'installation pour le serveur GlassFish 4.1.1

Une fois les répertoires d'installation appropriés sont choisis, cliquez sur le bouton **Next** pour continuer l'installation.

Dans l'étape suivante de l'assistant, l'installateur va nous inviter à choisir un répertoire d'installation pour Tomcat, un conteneur de servlets très populaire, qui est livré avec NetBeans.

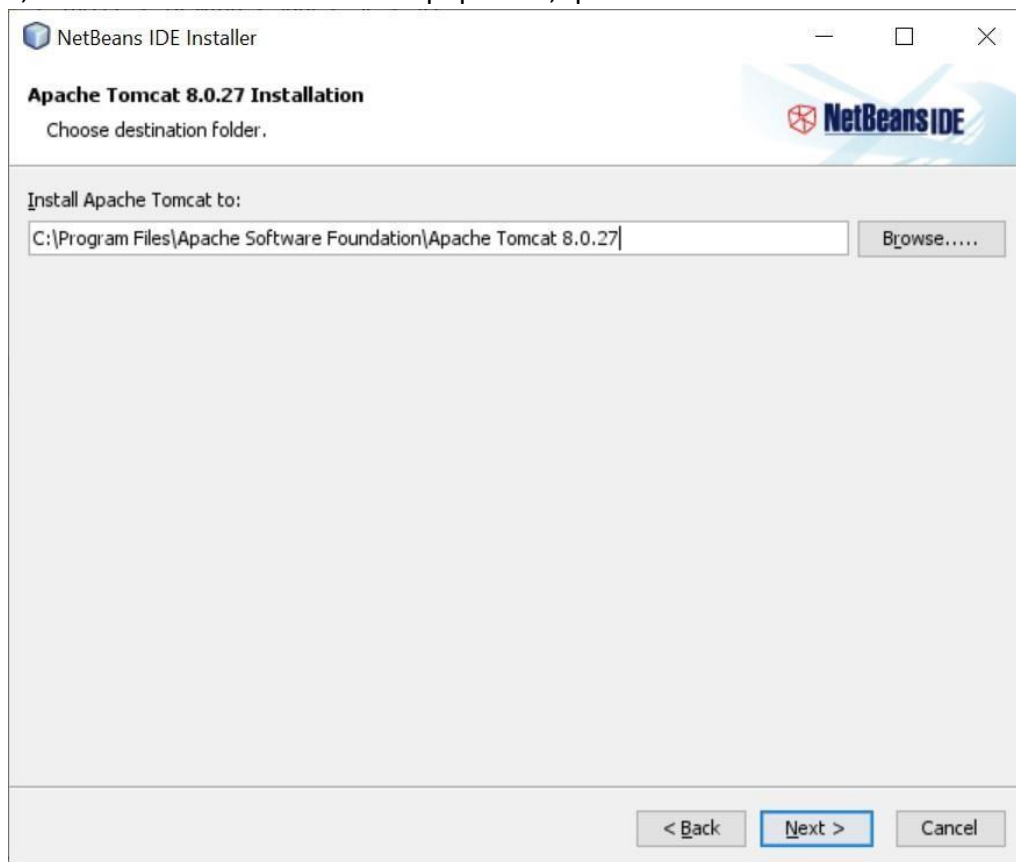


Figure 6 – Installateur Netbeans : Répertoire d'installation pour le serveur Tomcat 8.0.27

L'installateur affiche alors un résumé des choix effectués.

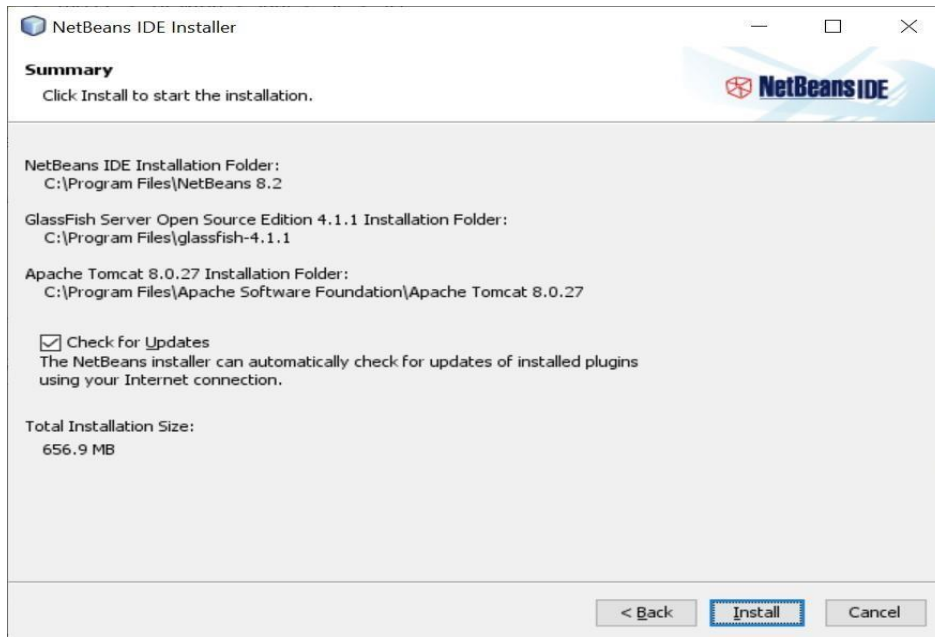


Figure 7 – Installateur Netbeans : Résumé des choix effectués

Pour Terminer l'installation de l'IDE NetBeans, cliquez sur le bouton **Install**
Ensuite, le programme d'installation affiche une barre de progression de l'installation.

Après avoir installé tous les composants, l'assistant d'installation nous indique que l'installation a bien réussie.

1.3 Configuration de NetBeans pour le développement Java EE

NetBeans 8.2 est préconfiguré avec le serveur d'application GlassFish 4.1.1, et avec le SGBDR JavaDB. Si nous souhaitons utiliser GlassFish et JavaDB, il n'y a rien à configurer.

1.3.1 L'intégration de NetBeans avec un serveur d'application

L'intégration de NetBeans avec un serveur d'application est très simple, elle se fait en suivant les étapes suivantes :

1. Cliquez sur **WINDOW** -> **SERVICES**
2. Ensuite, on doit faire un clic droit sur le nœud **Servers** dans la Fenêtre Services, sélectionner **Add Server** ... dans le menu contextuel

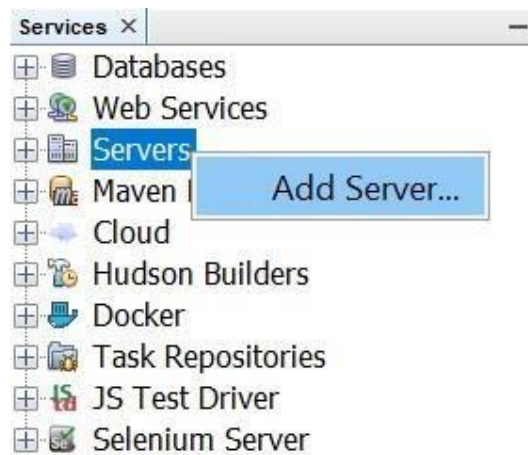


Figure 8 – Ajout d'une instance de serveur (1)

3. Ensuite, on doit sélectionner le serveur à installer dans la liste déroulante, puis cliquer sur le bouton **Next**

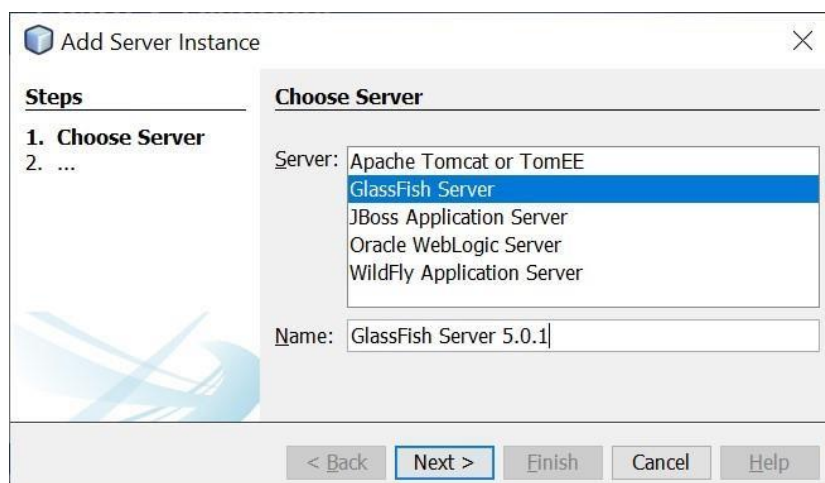


Figure 9 – Ajout d’une instance de serveur (2)

4. On a besoin alors de saisir l’emplacement d’installation du serveur souhaité et cliquer sur **next**

The screenshot shows the 'Add Server Instance' dialog box with the 'Server Location' tab selected. The 'Steps' list on the left indicates the current step is '2. Server Location'. The 'Installation Location' text box contains the path 'C:\Users\dell\glassfish-5.0.1\glassfish5', with a 'Browse...' button to its right. Below this, there are radio buttons for 'Local Domain' (selected) and 'Remote Domain'. A 'Download Now...' button is also present. A checkbox for 'I have read and accept the license agreement... (click)' is unchecked. An information message at the bottom states: 'Detected a GlassFish Server 5.0 install. Click Next to register remote or custom local domains.' The bottom of the dialog features buttons for '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

Figure 10 – Ajout d’une instance de serveur (3)

5. Enfin, on doit sélectionner un domaine, l’hôte et le port pour notre serveur, puis cliquer sur le bouton **finish**

The screenshot shows the 'Add Server Instance' dialog box with the 'Domain Location' tab selected. The 'Steps' list on the left indicates the current step is '3. Domain Name/Location'. The 'Domain' dropdown menu is set to 'domain2'. The 'Host' dropdown menu is set to 'localhost', with a checked 'Loopback' checkbox. The 'DAS Port' is '4848' and the 'HTTP Port' is '8080', both with checked 'Default' checkboxes. There are empty text boxes for 'Target', 'User Name', and 'Password'. An information message at the bottom states: 'Create new embedded domain: domain2'. The bottom of the dialog features buttons for '< Back', 'Next >', 'Finish' (highlighted in blue), 'Cancel', and 'Help'.

Figure 11 – Ajout d’une instance de serveur (4)

1.3.2 L'intégration d'un SGBDR avec NetBeans

Le système JavaDB est intégré avec NetBeans. En outre, il est livré avec les pilotes JDBC pour les systèmes MySQL et PostgreSQL. Il est également livré avec le pilote JDBC-ODBC qui permet de se connecter à la plupart des systèmes SGBDR sans avoir à obtenir un pilote JDBC. Pour ajouter un pilote JDBC à NetBeans :

1. Cliquez sur le nœud **Databases**, ensuite on doit faire un clic droit sur le nœud **Drivers** dans la Fenêtre Services, et sélectionner New Driver ... dans le menu contextuel.

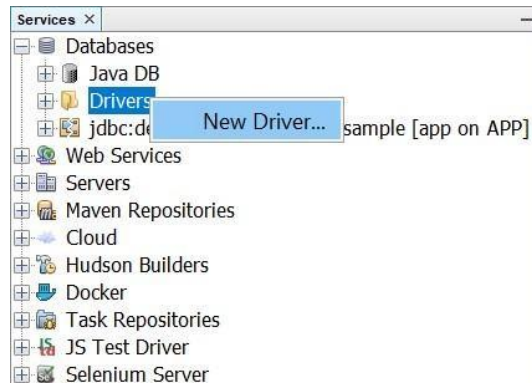


Figure 12 – Ajout d'un pilote JDBC (2)

2. Sélectionnez le fichier JAR contenant le pilote JDBC. Si l'opération est réussie le nouveau pilote JDBC sera affiché dans la liste des pilotes.

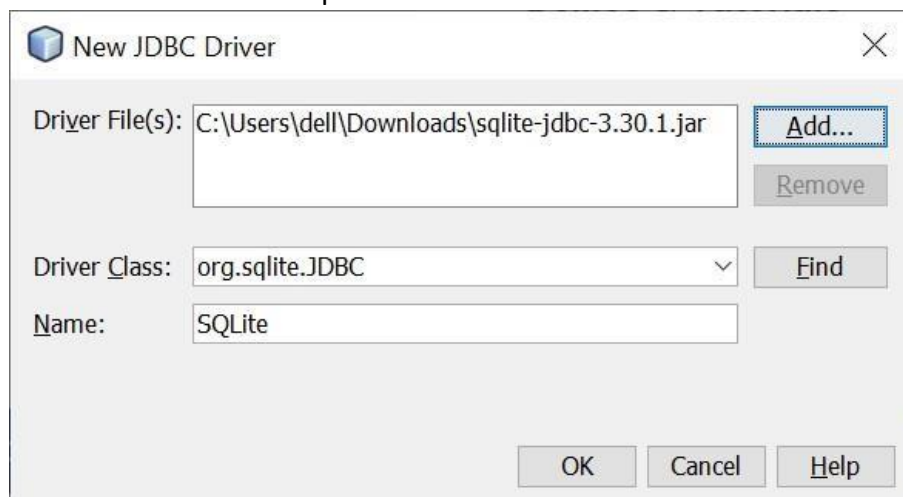
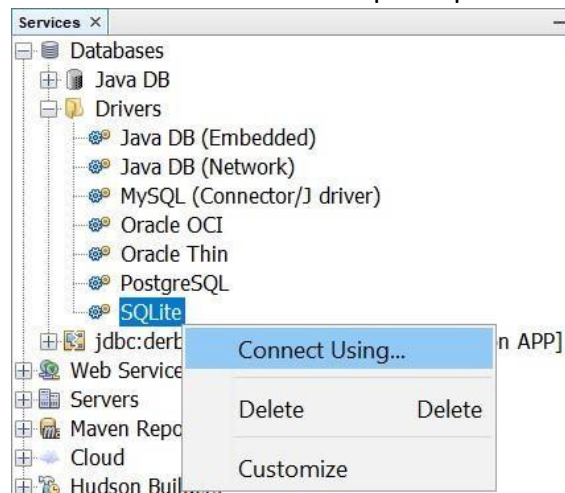


Figure 13 – Ajout d'un pilote JDBC (2)

1.3.3 La Connexion à un JDBC

Pour nous connecter à un SGBDR, il faut cliquer droit sur son pilote dans la fenêtre Services, puis cliquer sur **Connect Using...**

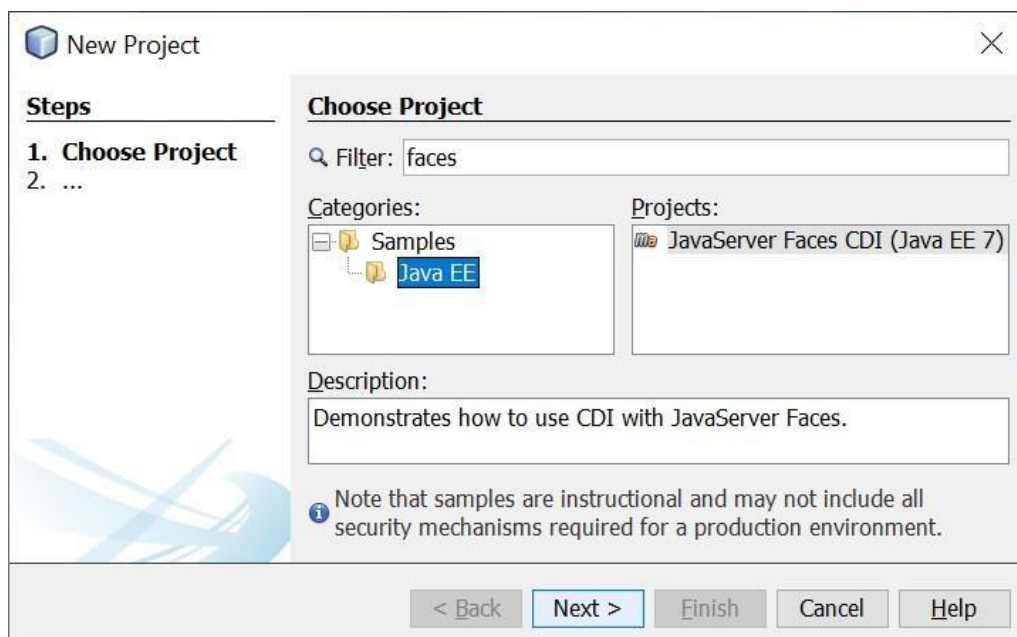
Ensuite, on saisit l'URL JDBC, nom d'utilisateur et mot de passe pour notre base de données.



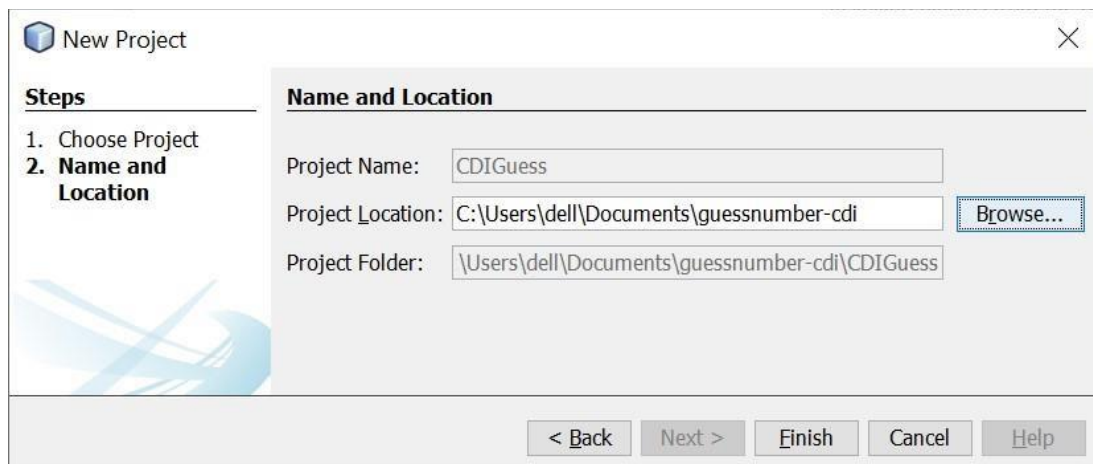
1.4 Déploiement d'une première application

NetBeans est livré avec un certain nombre d'exemples d'applications.

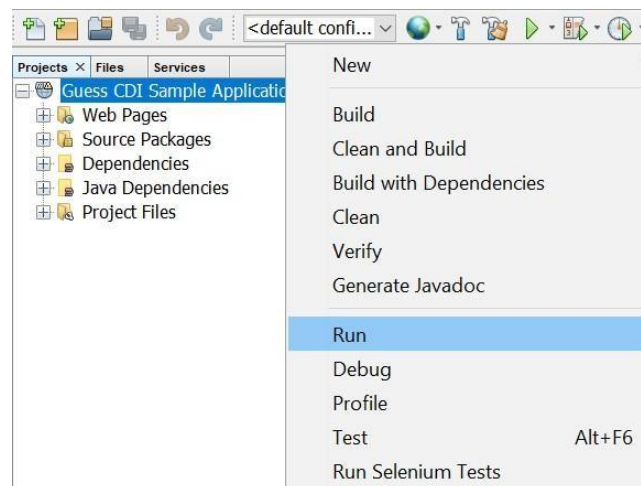
Pour ouvrir un projet exemple, allez dans **File -> New Project** puis sélectionner **Samples -> Java EE** dans la liste des catégories puis le projet **JavaServer Faces CDI** (un projet Java EE 7 impliquant à la fois les JSF et les contextes et l'injection des dépendances).



Après avoir cliqué sur le bouton **NEXT** , on est invité à entrer la localisation du projet (les valeurs par défaut répondent au besoin).



Lorsqu'on clique sur le bouton **Finish** , notre nouveau projet est affiché dans la fenêtre des projets. On peut compiler, empaqueter et déployer notre projet en un seul coup en cliquant sur le bouton droit et en sélectionnant Run (ou sur F6)



Lors de la première exécution, la sortie de la console affiche l'erreur suivante :

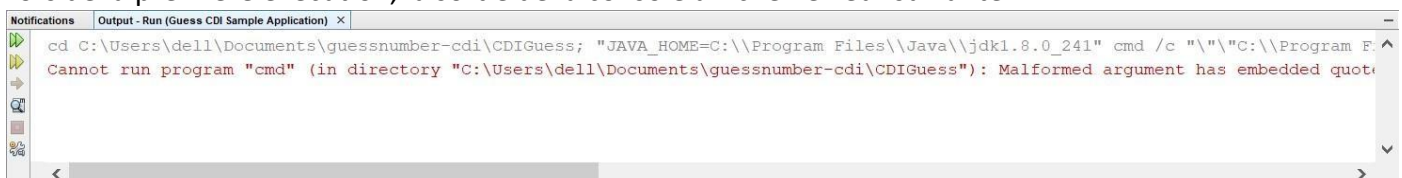


Figure 14 – La console de sortie affiche le journal d'exécution

Ce comportement (Fig 14) est une régression à partir d'un correctif de sécurité pour JDK8221858 (non public)[2].

Suivez ce lien pour une description complète :

<https://www.oracle.com/technetwork/java/javase/13-0-1-relnotes-5592797.html#JDK-8221858>

Pour résoudre ce problème, il faut ajouter l'option `-J-Djdk.lang.Process.allowAmbiguousCommands = true` à **netbeans_default_options** dans "`<netbeans-dir>\etc\netbeans.conf`"

```
c: > Program Files > NetBeans 8.2 > etc > netbeans.conf

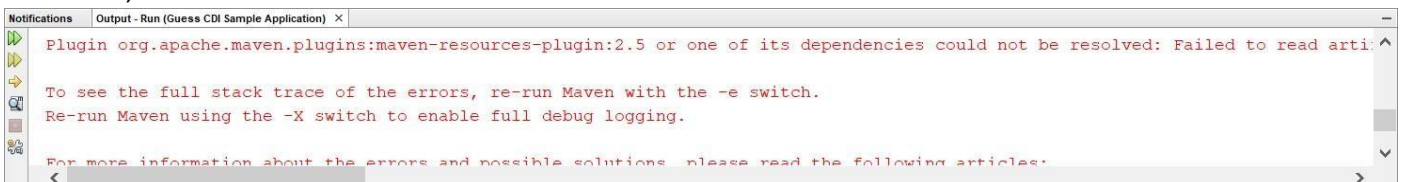
34 # Options used by NetBeans Launcher by default:
35 # (can be overridden by explicit command line switches)
36 #
37 # Note that default -Xmx is selected for you automatically.
38 # You can find these values in var/log/messages.log file in your userdir.
39 # The automatically selected value can be overridden by specifying -J-Xmx
40 # here or on the command line.
41 #
42 # If you specify the heap size explicitly, you may also want to enable
43 # Concurrent Mark & Sweep garbage collector.
44 # (see http://wiki.netbeans.org/FaqGCPauses)
45 #
46 netbeans_default_options="-J-Djdk.lang.Process.allowAmbiguousCommands=true -J-client -J-Xss2m -J-Xms32m"
```

Figure 15 – Application du patch pour résoudre le problème d'exécution

Après cela, nous devons enregistrer les modifications et redémarrer l'IDE NetBeans.

Nous allons re-exécuter le projet.

Cette fois, la console de sortie affiche une autre erreur :



The screenshot shows the NetBeans console with the following error message: "Plugin org.apache.maven.plugins:maven-resources-plugin:2.5 or one of its dependencies could not be resolved: Failed to read artifact metadata for org.apache.maven.plugins:maven-resources-plugin:2.5". Below the error message, there is a suggestion to "To see the full stack trace of the errors, re-run Maven with the -e switch. Re-run Maven using the -X switch to enable full debug logging." and a link to "For more information about the errors and possible solutions, please read the following articles:".

Figure 16 – Console de sortie affiche le nouveau journal d'exécution

En effet, en analysant le journal d'exécution, on se rend compte qu'une des dépendances du projet n'a pas pu être résolue, en d'autres termes le gestionnaire maven n'arrive pas à télécharger les plugins depuis le central (le référentiel officiel de maven) <http://repo.maven.apache.org/maven2>

Le journal d'exécution nous indique également le code retour renvoyé `Return code is: 501, ReasonPhrase:HTTPS Required`.

Cela signifie que le site n'est plus disponible en HTTP, et qu'une connexion sécurisée est obligatoire pour accéder aux ressources.

Pour résoudre ce problème, nous allons mettre à jour maven :

1. Téléchargez la dernière version de Maven sur le site <https://maven.apache.org/download.cgi>

Files

Maven is distributed in several formats for your convenience. Simply pick a ready-made binary distribution archive and follow the [installation instructions](#). Use a source archive if you intend to build Maven yourself.

In order to guard against corrupted downloads/installations, it is highly recommended to [verify the signature](#) of the release bundles against the public [KEYS](#) used by the Apache Maven developers.

	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.6.3-bin.tar.gz	apache-maven-3.6.3-bin.tar.gz.sha512	apache-maven-3.6.3-bin.tar.gz.asc
Binary zip archive	apache-maven-3.6.3-bin.zip	apache-maven-3.6.3-bin.zip.sha512	apache-maven-3.6.3-bin.zip.asc
Source tar.gz archive	apache-maven-3.6.3-src.tar.gz	apache-maven-3.6.3-src.tar.gz.sha512	apache-maven-3.6.3-src.tar.gz.asc
Source zip archive	apache-maven-3.6.3-src.zip	apache-maven-3.6.3-src.zip.sha512	apache-maven-3.6.3-src.zip.asc

Figure 17 – Site de téléchargement Maven

2. Extraire l'archive de distribution dans n'importe quel répertoire

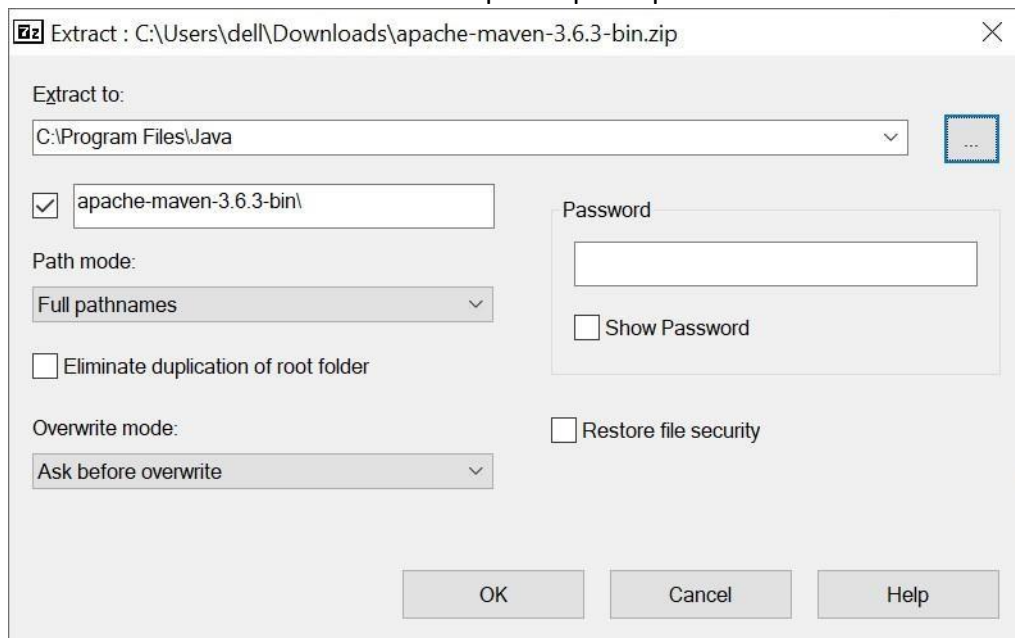


Figure 18 – Décompression du fichier apache-maven-3.6.3-bin.zip

3. Lier NetBeans à la dernière version téléchargée de Maven. Pour cela allez à Tools -> Options -> Java -> Maven et changer la valeur de Maven Home par le répertoire d'installation de la nouvelle distribution Maven.

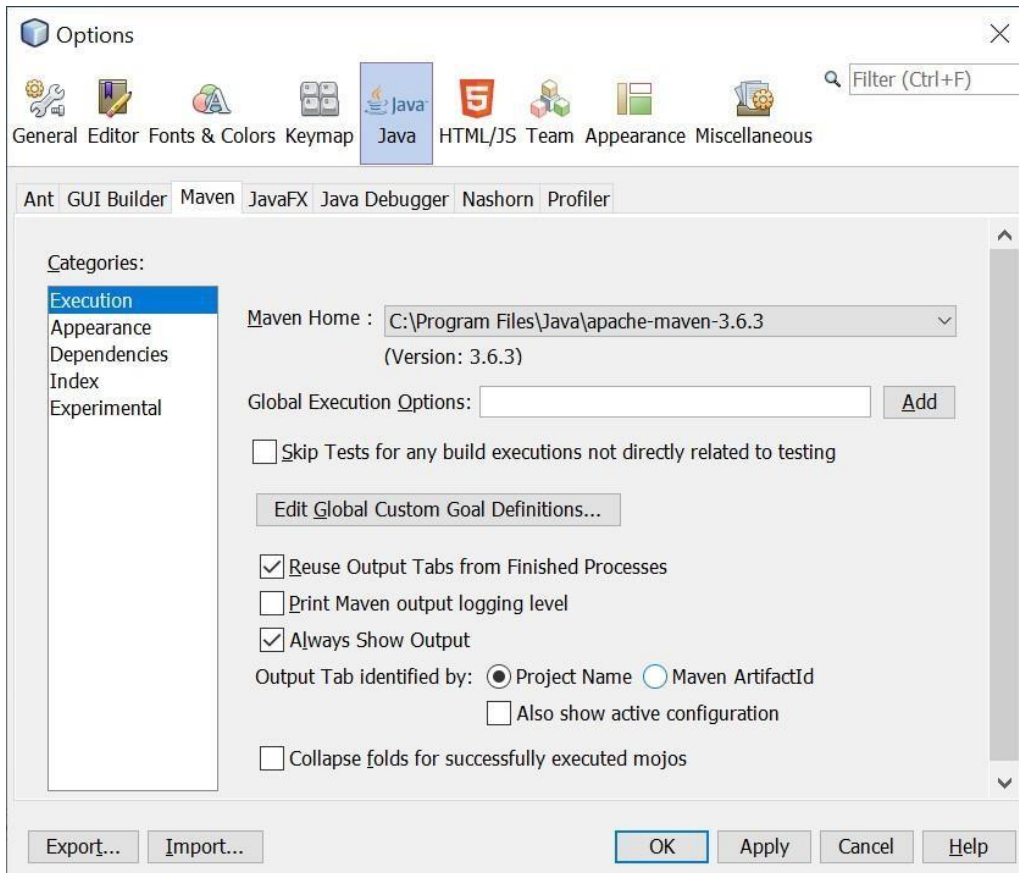


Figure 19 – Configuration de la distribution Maven sur NetBeans

Maintenant que nous allons terminer la configuration de Maven, Nous allons re-exécuter le projet.

Cette fois, la console de sortie du serveur GlassFish lève une exception :



Figure 20 – Console de sortie du serveur GlassFish

Après avoir inspecté le journal d'exécution du serveur GlassFish (Fig 1.20), nous réalisons que la source des exceptions est le fichier beans.xml.

(Ce bug est relatif à la version 8.2 de NetBeans)

Le fichier beans.xml est utilisé par CDI pour indiquer au serveur d'exécution Java EE que le projet est un module contenant des beans CDI [3, 1].

Pour résoudre ce problème, supprimez simplement ce fichier du projet.

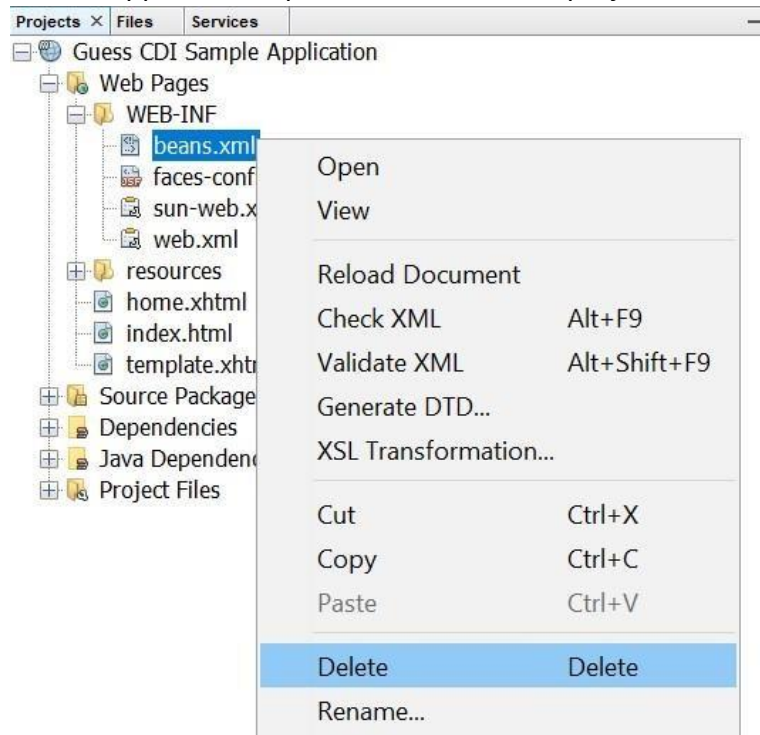


Figure 21 – Suppression du fichier beans.xml

Maintenant, exécutons le projet une autre fois.



Figure 22 – Console de sortie du serveur GlassFish

Le journal d'exécution nous indique que l'application a été déployé avec succès en 1 194 millisecondes (Fig 22), et par conséquent une nouvelle fenêtre du navigateur démarre automatiquement et affiche la page par défaut pour notre application (Fig 23).

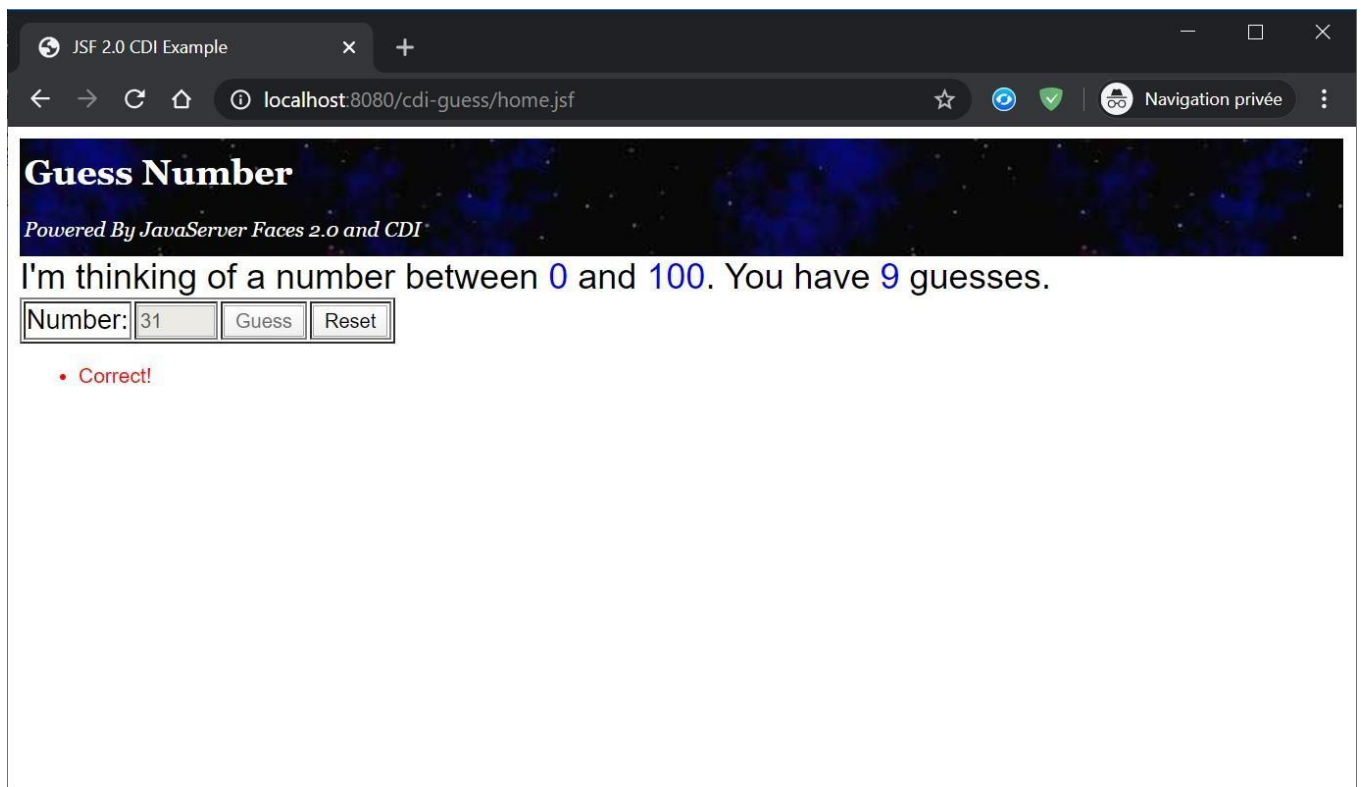


Figure 23 – Page d'application "JSF 2.0 CDI Example"

Bibliographie

- [1] Getting started with contexts and dependency injection and jsf 2.x.
<https://netbeans.org/kb/docs/javaee/cdi-intro.html>. Accessed : 2020-03-29.
- [2] Java with maven wouldn't build : Cannot run program "cmd" ...
<https://stackoverflow.com/a/58484737/7134209>. Accessed : 2020-03-24.
- [3] Set context and dependency injection on netbeans (after project creation).
<https://stackoverflow.com/a/7604524/7134209>. Accessed : 2020-03-25.