↩ Retour à Proc#136 - Installation d'un poste de travail collaborateur	
Document technique	Rattaché à Proc#136. Non soumis à approbation (Proc#163) : tenu à jour par l'équipe de développement.

Document technique — Installation du poste de développement

Ce document décrit les étapes à suivre pour préparer un poste de développeur : installation de Sourcetree, accès à GitLab, récupération des dépôts, puis installation des outils nécessaires pour compiler et lancer les projets `logeas-lib`, `logeas-web` et `logeas-assistance`. Il permet à un nouveau développeur de rendre son poste opérationnel avant de commencer son travail.

Avant de commencer : le poste doit déjà respecter les règles communes de Proc#136 (session locale, disques C et D chiffrés, logiciels de base). Les dépôts sont récupérés dans `D:\Developpements`, comme le prévoit cette procédure.

Étape 1 : Installation de Sourcetree et des outils Git

1. Téléchargez le programme d'installation sur le site officiel de Sourcetree.
2. Lancez l'installation. Lors de l'étape **Registration**, vous pouvez ignorer la création de compte Atlassian (ou vous connecter si vous le souhaitez).
3. À l'étape **Pick tools**, cochez **Git**. Laissez Sourcetree télécharger et installer sa version intégrée de Git (*Embedded Git*), ce qui garantit une compatibilité parfaite.
4. À l'étape **Preferences**, renseignez le **Nom** et l'**Adresse e-mail** qui apparaîtront sur vos commits (généralement les mêmes que votre compte GitLab).

Étape 2 : Création du jeton d'accès (PAT) sur GitLab

Pour des raisons de sécurité, GitLab n'accepte plus votre mot de passe habituel pour les opérations Git en HTTPS.

1. Connectez-vous sur GitLab.
2. Cliquez sur votre avatar en haut à gauche, puis allez dans **Preferences** (Préférences).
3. Dans le menu de gauche, cliquez sur **Access Tokens** (Jetons d'accès).
4. Cliquez sur **Add new token** :
   - Nommez le jeton (ex : `Sourcetree PC Fixe`).
   - Cochez obligatoirement la case **api** (indispensable pour que Sourcetree communique correctement).
   - Optionnel : cochez également **read_repository** et **write_repository**.
5. Cliquez sur **Create personal access token**.
6. Copiez immédiatement le jeton généré et conservez-le précieusement.

> Le jeton n'est affiché qu'une seule fois. S'il est perdu, il faudra en créer un nouveau.

Étape 3 : Authentification globale dans Sourcetree

Ajouter votre compte au gestionnaire global permet à Sourcetree de lister vos projets.

1. Ouvrez Sourcetree et allez dans le menu supérieur : **Tools > Options**.
2. Cliquez sur l'onglet **Authentication**.
3. Cliquez sur **Add** (Ajouter) :
    - **Hosting Service** : sélectionnez **GitLab**.
    - **Host URL** : laissez `https://gitlab.com` (ou l'URL de votre serveur privé).
    - **Preferred Protocol** : choisissez **HTTPS**.
    - **Authentication** : choisissez **Personal Access Token**.
    - **Username** : entrez votre identifiant exact GitLab (votre pseudo/handle, pas votre adresse e-mail).
4. Cliquez sur **Refresh Password** (ou **Regenerate Token**), puis collez votre jeton d'accès précédemment copié.
5. Validez par **OK**. Votre compte apparaît désormais avec un statut connecté.

Étape 4 : Récupération (clonage) du premier dépôt

1. Sur GitLab, rendez-vous sur la page de votre projet.
2. Cliquez sur le bouton bleu **Clone** et copiez l'adresse sous **Clone with HTTPS**.
3. Dans Sourcetree, cliquez sur l'onglet **New** (ou **File > Clone / New**).
4. Sélectionnez l'option **Clone**.
5. Dans le champ **Source Path / URL**, collez l'URL copiée.
6. Cliquez dans le champ suivant (**Destination Path**).
7. Sourcetree valide l'accès en tâche de fond. Le message vert *This is a Git repository* apparaît.
8. Choisissez le dossier de destination sur votre disque dur et cliquez sur **Clone**.

Votre projet est désormais copié sur votre ordinateur. Répétez cette étape pour chaque dépôt dont vous avez besoin.

Les dépôts à récupérer se trouvent dans le groupe GitLab `logeas-informatique/logeas/newinterfaces-group` :

Dépôt	Rôle	Dossier de destination
`logeas-lib`	Bibliothèques communes (à récupérer en premier)	`D:\Developpements\Sources\NewInterfaces\logeas-lib`
`logeas-web`	Application LoGeAs Web	`D:\Developpements\Sources\NewInterfaces\logeas-web`
`logeas-assistance`	Application Assistance	`D:\Developpements\Sources\NewInterfaces\logeas-assistance`

> Respectez exactement ces dossiers de destination :
> * les applications référencent la bibliothèque par le chemin relatif `../logeas-lib` : les dépôts doivent donc être **côte à côte** dans le même dossier ;
> * les scripts de compilation (`scripts_bat\Compile.bat`, `BuildVersion.ps1`) contiennent le chemin `D:\Developpements\Sources\NewInterfaces\` en dur.

Étape 5 : Installation de Node.js et npm

Les projets Angular ont besoin de **Node.js** pour être compilés et lancés. **npm** (le gestionnaire de paquets) est installé automatiquement avec Node.js : il n'y a rien à installer séparément.

1. Déterminez la version à installer (voir la note ci-dessous).
2. Téléchargez cette version (branche **LTS**) sur le site officiel de Node.js.
3. Lancez l'installation en conservant les options par défaut.
4. Ouvrez un **nouveau** terminal (PowerShell) et vérifiez l'installation :

```
node -v
npm -v
```

Les deux commandes doivent afficher un numéro de version.

> **Versions à installer** : elles évoluent avec les projets et ne sont donc pas figées dans cette procédure.
> * La version d'Angular de chaque projet est indiquée dans son fichier `package.json`, à la ligne `@angular/cli`.
> * La version de Node.js à installer en découle : consultez le tableau de compatibilité d'Angular en prenant la version d'Angular la plus récente parmi les projets.
> * En cas de doute, référez-vous au chef de projet.

Étape 6 : Installation d'Angular CLI

Angular CLI fournit la commande `ng` utilisée pour lancer, compiler et tester les projets. Installez la version majeure relevée dans les `package.json` (remplacez XX par ce numéro) :

```
npm install -g @angular/cli@XX
ng version
```

Chaque projet embarque sa propre version d'Angular CLI : lancée depuis le dossier d'un projet, la commande `ng` utilise automatiquement la version de ce projet.

> Si PowerShell refuse d'exécuter `ng` avec un message du type *l'exécution de scripts est désactivée sur ce système*, lancez une fois la commande suivante puis rouvrez le terminal :
> `Set-ExecutionPolicy -Scope CurrentUser -ExecutionPolicy RemoteSigned`

Étape 7 : Installation des dépendances du projet

Les dépôts clonés ne contiennent ni les bibliothèques externes (dossier `node_modules`), ni les bibliothèques LoGeAs compilées (dossier `dist` de `logeas-lib`). Il faut les produire une première fois, **dans l'ordre ci-dessous**.

7.1 Compiler logeas-lib

`logeas-web` et `logeas-assistance` utilisent trois bibliothèques issues de `logeas-lib` (`ngx-logeascom-ui`, `ngx-logeasweb-ui`, `ngx-interface-ui`), installées sous forme de fichiers `.tgz`. Ces fichiers ne sont pas dans Git : il faut les générer avant toute autre installation.

```
cd D:\Developpements\Sources\NewInterfaces\logeas-lib
npm install
.\scripts_bat\Compile.bat
```

Le script se termine par le message `COMPILATION OK` et produit les trois fichiers `.tgz` dans `logeas-lib\dist`.

7.2 Récupérer les fichiers de licence

Les licences DevExtreme et Stimulsoft ne sont pas dans Git. Demandez-les à un membre de l'équipe et copiez-les dans `logeas-web\src\assets\licence\` :

- `DevExpress_License.txt`
- `stimulsoft-license.key`

7.3 Installer les dépendances des applications

```
cd D:\Developpements\Sources\NewInterfaces\logeas-web
npm install

cd D:\Developpements\Sources\NewInterfaces\logeas-assistance
npm install
```

Cette opération peut prendre plusieurs minutes. Elle est à refaire chaque fois que le fichier `package.json` est modifié (par exemple après avoir récupéré les modifications d'un collègue).

> Après une modification de `logeas-lib`, relancez `Compile.bat` puis, depuis le dossier de l'application, `..\logeas-lib\scripts_bat\MajBiblio.bat <version>` pour réinstaller les bibliothèques (la version est celle du `package.json` de `logeas-lib`).

Étape 8 : Premier lancement du projet

1. Dans le dossier de l'application, lancez :

```
npm start
```

1. Attendez la fin de la compilation, puis ouvrez l'adresse correspondante dans votre navigateur :

Application	Commande	Adresse
`logeas-web`	`npm start`	http://localhost:4200
`logeas-web` (serveurs locaux)	`npm run start:local`	http://localhost:4200
`logeas-assistance`	`npm start`	http://localhost:112

1. Si l'application s'affiche, votre poste est opérationnel.

Pour arrêter le serveur de développement : **Ctrl + C** dans le terminal.

Outils complémentaires (selon les besoins)

- **SDK .NET** : uniquement pour travailler sur le serveur de formulaires de `logeas-web` (`npm run forms:watch`).
- **Playwright** : uniquement pour les tests d'intégration de `logeas-web` (`npx playwright install`, puis `npm run test-integration`).
- **Tests unitaires** : `npm test` (Jest) dans chaque projet, sans installation supplémentaire.

Étape 9 : Avant de commencer à développer

- **Éditeur de code** : installez Visual Studio Code et l'extension **Angular Language Service**.
- **Branche de travail** : ne travaillez pas directement sur la branche principale. Dans Sourcetree, cliquez sur **Branch** pour créer votre propre branche avant vos premières modifications.
- **Mise à jour** : avant chaque session de travail, cliquez sur **Pull** dans Sourcetree pour récupérer les dernières modifications.

Récapitulatif

Outil	Rôle	Vérification
Sourcetree / Git	Récupérer et envoyer le code	Le dépôt apparaît dans Sourcetree
Node.js	Exécuter les outils de compilation	`node -v`
npm	Télécharger les dépendances	`npm -v`
Angular CLI	Lancer et compiler le projet	`ng version`
Bibliothèques `logeas-lib`	Composants communs aux applications	Trois fichiers `.tgz` dans `logeas-lib\dist`
Fichiers de licence	DevExtreme et Stimulsoft	Présents dans `logeas-web\src\assets\licence`
Dépendances du projet	Bibliothèques utilisées par le code	Dossier `node_modules` présent

From:
https://wiki-logeas.fr/certif/ - **dokuwiki-certif**

Permanent link:
https://wiki-logeas.fr/certif/doku.php?id=certif:procedure:develop:installationpostedetravail

Last update: **2026/09/30 14:50**