

Consignes pour le rendu des TPs via Git

Pour les ressources « R3.05 - Programmation Système » et « R4.A.11 - Développement pour applications mobiles », le rendu des TPs se fait via le système de gestion de version Git.

1 Consignes générales

- Un seul dépôt par matière doit être créé.
- Chaque TP aura un répertoire dans ce dépôt.
- Les dépôts doivent impérativement être **privés**¹.

2 Nomenclature

Le nom de votre dépôt doit IMPÉRATIVEMENT être de la forme :

- **OS_Nom1_Nom2** pour le cours de système d'exploitation
- **Android_Nom1_Nom2** pour le cours d'android

Le nom du répertoire pour un TP est précisé au début de chaque sujet.

3 Procédure

3.1 Avant le premier TP

Les étapes à suivre avant le premier TP :

1. Vérifier que vous avez bien un compte GitLab sur le serveur de l'IUT à l'adresse : gitlab.iut-valence.fr
2. Créer un nouveau projet pour la matière. Veillez à respecter la nomenclature ci-dessus.
3. Assurez vous que votre projet est **privé**¹.
4. Partagez ce projet avec votre encadrant de TP. Il doit avoir les droits **Developer** ou **Maintainer** pour pouvoir accéder au code. Vous pouvez en profiter pour le partager avec votre binôme.
5. Avant de *cloner* ce projet sur votre compte, créez et ajoutez une **clef ssh** à votre compte GitLab IUT. Pour créer la paire de clefs :

```
$ ssh-keygen
```

Cela génère par défaut une paire de clef (`id_rsa` et `id_rsa.pub`) dans le répertoire `~/.ssh/` de votre compte.
6. Ajoutez la clef **publique** (fichier `.pub`) à votre compte GitLab : section "Clefs SSH" de votre profil GitLab. Il faut copier le texte complet de la clef dans le champ de texte de GitLab.
7. Clonez ensuite le projet GitLab sur votre compte avec par exemple :

```
$ git clone git@gitlab.iut-valence.fr :<login>/OS_UnNom_UnAutreNom.git
```

```
$ cd OS_UnNom_UnAutreNom
```

1. Sinon des indélicats peuvent venir piocher dans vos TPs et diviser votre note par deux ou vous attirer un zéro.

Une fois votre projet correspondant à la matière Systèmes d'exploitation cloné, placez vous dans le répertoire résultant du clone si ce n'est pas déjà fait :

```
$ cd OS_UnNom_UnAutreNom
```

Ajoutez ensuite les fichiers ".o" à la liste des fichiers ignorés :

```
$ echo '*.o' >> .gitignore
```

Si vous avez une erreur concernant la clef au moment du `clone` ou du `push`, vérifiez que votre clef ssh est bien prise en compte par votre `ssh-agent`. Votre clef doit apparaître lorsque vous faites :

```
$ ssh-add -l
```

Si ce n'est pas le cas, ajoutez votre clef à votre `ssh-agent` :

```
$ ssh-add ~/.ssh/nom_de_votre_clef
```

3.2 À chaque TP

Une fois votre projet GitLab créé et cloné, pour chaque sujet de TP :

1. Créez un répertoire pour le sujet de TP :

```
$ mkdir NomDuTP
```

2. Ajoutez un fichier `.gitignore` dans ce nouveau répertoire qui contiendra les fichiers inutiles spécifiques à ce sujet.

3. Ajoutez le tout dans votre dépôt local, commitez dans votre dépôt local puis poussez sur GitLab :

```
$ git add NomDuTP
```

```
$ git commit -a -m "Ajout initial"
```

```
$ git push -u origin main
```

4. Assurez vous que votre répertoire apparaît bien dans votre projet sur GitLab.

Ajoutez ensuite le nom de vos exécutables dans le fichier `.gitignore`.

3.3 Git Ignore

Le fichier `.gitignore` qui précise les fichiers qui doivent être ignorés par Git est un moyen de garder vos dépôt "propres". Vous pouvez ajouter un `.gitignore` à la racine de votre projet et également dans chaque dossier ou sous-dossier de votre projet. Vous trouverez la [documentation](#) et des [exemples](#) sur internet. Pour vos projets C, vous devez a minima mettre vos .o en ajout une ligne "*.o" dans ce fichier.

Dans ce fichier, chaque nom de fichier ou expression (*.o) doit être sur une ligne différente.

4 Évaluation

Votre encadrant effectuera un clone de votre projet global (OS_UnNom_UnAutreNom) à la date qu'il a fixé. En l'absence de code dans votre dépôt git **sur GitLab** (c'est-à-dire ce que vous *pushez*), il ne sera pas en mesure de vous noter (et devra donc vous attribuer 0). Pensez donc bien à effectuer des `commit` et des `push` régulièrement. Il vous est de plus demandé de créer un fichier différent par question.

