

Le respect de ces modalités de rendu est obligatoire et leur non-respect entraîne **la nullité du rendu**.

1 Travail en TP

- Le travail en TP se fait seul ou en binôme.
- Les binômes sont constitués **pour l'ensemble des travaux à rendre de cette UE** (pas de changement de binôme et pas de séparation au cours du semestre).

2 Outils de travail et de rendu

Pour ces TPs, seront utilisés :

- la plateforme AMETICE (cours **[25-26]-M1 Fiabilité logicielle**) où vous retrouverez les sujets de TP
- le système de gestion des versions git avec le serveur qui vous est dédié <https://etulab.univ-amu.fr/>, où vous retrouverez le matériel (sources, documentation, ...) et par lequel se fera le rendu des TPs.
- l'environnement de développement IntelliJ.

3 Modalités du travail et du rendu

3.1 Pour le premier TP

Sur le serveur git [etulab](https://etulab.univ-amu.fr/), vous trouverez le dépôt à “forker” à l'url suivante <https://etulab.univ-amu.fr/alaboure/M1-INFO-FSI-TP-template.git>. Vous y trouverez un répertoire nommé TP1.

Voici les étapes à réaliser pour disposer d'une copie de travail et de rendu (**une seule personne du binôme réalisera cette opération**) :

1. Créer un “fork” (duplicata) du projet :
 - cliquez sur le bouton “Fork” (en haut à droite)
 - Modifiez le nom du projet en y ajoutant à la fin votre nom de famille et celui de votre binôme éventuel. Configurez la visibilité de votre projet à *Private* afin que vos camarades non-membres du projet n'aient pas accès à votre code.
 - Dupliquer le projet (“Fork Project”)
2. Placez-vous au niveau du projet dupliqué dont vous êtes maintenant le propriétaire
3. Invitez sur votre projet votre binôme (si vous en avez un) avec les droits *owner* pour qu'il ait les mêmes droits que la personne ayant forké le dépôt ainsi que les deux enseignants de l'UE (François-Xavier Dupé, identifiant `@fdupe` et Arnaud Labourel, identifiant `@alaboure`) avec les droits *reporter* afin qu'ils aient accès au code.
4. Modifiez le fichier `README.md` à la racine du projet afin d'ajouter les noms du ou des membres du projet.

Vous disposez maintenant d'un espace de travail collaboratif pour ce TP qui sera aussi votre espace de rendu.

3.2 Pour les TPs suivants ...

Vont être mis à jour au fur et à mesure dans le projet des répertoires pour chacun des TPs comprenant le sujet et les sources fournies éventuelles.

Afin de mettre à jour le dépôt “forké”, dans la page web [etulab](#) du projet, il existe une mention du dépôt originel et à sa droite un bouton pour le mettre à jour (*update fork*).

4 Exploitation de votre espace

Votre projet est maintenant créé sur le serveur. Mais il vous faut en faire une copie locale. Pour cela, il faut cloner ce projet. La commande `git clone url-depot` vous permet de le faire. Pour trouver `url-depot` de votre projet sur le serveur `etulab`, cliquez sur le bouton `Clone`.

Une autre possibilité (conseillée) est d'importer ce projet sous IntelliJ via “Project from Version Control”. Vous aurez la possibilité de réaliser les opérations usuelles de `git` (`pull`, `add`, `commit`, `push`) depuis IntelliJ.

5 Rendu

Le rendu du TP sera donc le contenu de votre projet (ou de celui de votre binôme) à l'heure à laquelle le rendu est attendu (la date de la dernière poussée (“push”) du projet faisant foi). **Tout retard entraînera une pénalité sur la note du TP rendu.**

Par défaut, le rendu sera à faire la veille du TP suivant à 23:59 (donc le dimanche de la même semaine). Cette date pourra être modifiée au cas par cas.

Chaque rendu comportera **obligatoirement** un compte rendu de TP au format `pdf` présent dans le projet.

Les compte-rendus seront des fichiers nommés respectivement `CR-TP x` dans un répertoire `comptes-rendus` à la racine du projet, du TP x . Ces comptes-rendus pourront être très succincts selon ce qui est demandé et fait en TP. Par ailleurs, le contenu de ce compte rendu est un élément essentiel de l'appréciation du TP.