

Projet V6.0.0



Cette œuvre est mise à disposition selon les termes de la [licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage à l'Identique 3.0 non transposé](#).

Document en ligne : www.mickael-martin-nevot.com

Date de rendu : **04 décembre 2024 à 23h00**
Date de présentation : **05 décembre 2024 à 08h30**
Travail : **groupe de trois**

1 Assistance

Vous pouvez contacter l'enseignant en cas de besoin en formalisant et en ciblant précisément votre demande. Pour ce faire, vous devez respecter les règles de communication et d'envoi (ci-dessous).

2 Communication et envoi (5 %)

2.1 Généralités

En joignant vos coordonnées (*e-mail* et téléphone portable notamment) à un message ou à votre livraison, vous pourrez être joint en cas de problème.

2.2 Communication

Chaque communication devra être faite :

- à destination de votre enseignant responsable :
 - Mickaël Martin Nevot : mmartin.nevot@gmail.com ;
 - Eliesse Hadjem : eliesse.hadjem@gmail.com ;
 - Marina De Vido : marinadevido@gmail.com ;
- en faisant figurer [AMU][IUT][S3] en début de sujet.

2.3 Livraison

Votre livrable devra être :

- nommé de la manière suivante (Nom1, Nom2, Nom3 étant vos noms, par ordre lexicographique de vos noms de famille, et Prénom1, Prénom2, Prénom3 vos prénoms) : Nom1 Prénom1 – Nom2 Prénom2 – Nom3 Prénom3 ;
- compressé dans une seule archive au format ZIP **n'excédant pas 10 Mo** ;
- remis, avant la date de rendu, sur AMeTICE (<http://ametice.univ-amu.fr>), à la section rendu du cours correspondant.

3 Présentation orale (12,5 %)

Vous effectuerez une présentation orale sous forme de démonstration de votre projet respectant les consignes suivantes :

- une durée maximale de **20 minutes** devant un jury ;
- à la **date de présentation** (selon l'ordre de passage que vous sera communiqué par votre enseignant responsable) ;
- sur la réalisation de votre **projet** à laquelle des questions de cours viendront s'ajouter.

4 Sujet

Vous devez écrire la **spécification fonctionnelle technique** et coder l'**implémentation** d'une application en **Java** et ses **tests**, qui doit utiliser les concepts vus lors de cet enseignement.

Le sujet porte sur les réalisations des TD TD3 : Java, cas pratique et TD4 : Algorithmique « avancée ».

La partie « graphique » de votre application peut être minimaliste (affichage en ligne de commande par exemple).

Vous devez utiliser le plus possible d'abstraction et de particularités du langage **Java** (5 %), de **structures de données** (2,5 %) ainsi que **d'algorithmique** (5 %), notamment ceux vus dans cet enseignement. En particulier, votre application doit comporter des **classes abstraites**, des **interfaces**, des **collections** avec des **itérateurs**, de la **généricité**, des **modèles de conception**, des **exceptions**, des **threads** et au moins un algorithme de **tri**.

Vous devez apporter un soin tout particulier à la présentation du **code source**, respect d'une **convention de nommage**, **indentation**, **commentaires** (5 %), à l'**architecture** des paquetages ainsi que des autres répertoires (2,5 %), à la **gestion de version** avec des solutions comme **GitHub** (5%), aux tests unitaires avec des solutions comme **JUnit** et à l'**intégration continue** avec des solutions comme **Travis CI**, **GitLab** ou **Jenkins** (5 %).

Votre application doit être intégralement **fonctionnelle** (20 %), **évolutive**, **modulaire**, **robuste**, **fiable**, professionnelle (5 %) et **perfectionnée** (10 %) avec de **nouvelles fonctionnalités** et/ou une véritable **interface graphique** (avec des solutions comme **JavaFX** ou **Swing**).

5 Livrables

Vous devez envoyer les éléments suivants avant la date de rendu :

- la **spécification** sous forme d'un ou plusieurs documents (5 %) :
 - l'**étude de conception** ;
 - la clôture du travail (**synthèse**, **bilan technique**, **problèmes rencontrés**, **écarts avec les prévisions**, **mesures d'amélioration**) ;
- l'**application** :
 - l'URL vers le projet **git** en ligne (voir Sujet) ;
 - le **code source**, avec les **tests unitaires** (voir Sujet) ;

- le **projet au format Eclipse** (voir Livraison) ;
- le fichier « **lisez-moi** » (2,5 %) ;
- le **manuel d'utilisation** (présentant plusieurs scénarios d'utilisation pas à pas), vidéo ou non (5 %) ;
- la documentation complète générée par la **Javadoc** de l'application (5 %).

6 Conseils

Voici quelques conseils :

- organisez et pilotez votre projet avec des **méthodes agiles** ;
- gérez la répartition de réalisation des fonctionnalités entre les membres du groupe (en se répartissant différents modules par exemple) ;
- pour ceux qui sont à l'aise, vous pouvez mettre en place quelques fonctionnalités intéressantes supplémentaires. À faire **uniquement** après avoir terminé tous les points essentiels notés ci-dessus et après avoir optimisé et approfondi les fonctionnalités demandées ;
- utilisez des **modèles de conception** adaptés à vos problématiques et optimisez votre code source en tentant de diminuer la complexité de vos algorithmes ;
- testez rigoureusement votre application.

Voici quelques conseils sur la présentation orale :

- testez toujours le matériel avant une présentation orale ;
- habillez-vous d'une tenue correcte en respect avec l'exercice ;
- ne vous cachez pas derrière quelqu'un d'autre, et ne renvoyez pas la faute à autrui ;
- ayez une voix qui porte (ce qui ne signifie pas forcément forte) ;
- regardez l'assistance ;
- employez une approche positive mais réelle (valorisante mais sans mensonge) ;
- assurez-vous que ce qui est montré est bien vu.

Voici les compétences relationnelles communément recherchées lors d'une présentation orale :

- supprimer les parasites ;
- avoir une bonne gestuelle ;
- se mouvoir correctement ;
- gérer l'utilisation de son regard.