



Lycée
**TOUCHARD
WASHINGTON**

Section des Techniciens Supérieurs

*Systèmes Numériques option
Informatique et Réseaux*



Projet Horloge

~ TD n°2 ~

Analyse UML et Codage C++

Version 4.0 – septembre 2020

Objectif

- Codage C++
 - Classe
 - Instance
 - Constructeur
 - paramètres par défaut
 - Surcharge d'opération
 - Flux
 - Interaction avec la console avec les codes ANSI

Conditions de réalisation

Ressources utilisées :

Matériel

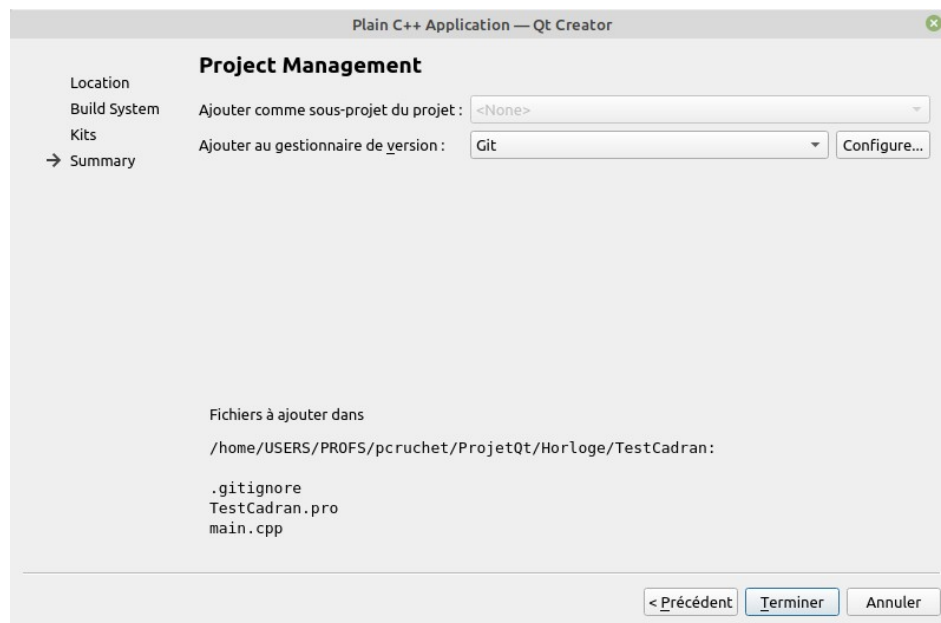
Un ordinateur sous Linux

Logiciel

QtCreator
Projet sans la librairie Qt en
mode console

1. CONSTRUCTION DU PROJET

1. Sous Qt réalisez un projet de type Non Qt en C++ nommé **TestCadran**. Dans un répertoire spécifique nommé **ProjetHorloge**. Ce projet doit faire l'objet d'un suivi de version avec Git



Vous ajoutez à votre projet la classe Cadran, elle doit également faire l'objet du suivi de version avec Git.

2. UTILISATION DE LA CLASSE CADRAN

1. Dans le programme principal, déclarez une instance de la classe **Cadran** nommée **leCadran**, cette instance utilisera les paramètres par défaut du constructeur.

Remarque : Pour que le cadran reste visible ajoutez les lignes suivante pour que le programme ne se termine pas de suite :

```
char touche;  
cin >> touche;
```

2. Quelle est la position du cadran sur l'écran, indiquer le nombre de ligne et la largeur de l'afficheur (nombre de caractères affichables).
3. Utilisez les méthodes **void Cadran::Afficher()** pour faire défiler un compteur variant de 0 à 10 en position 3. Pour ralentir l'affichage on propose d'utiliser la fonction sleep de la librairie `<unistd.h>`. Lorsque l'affichage est terminé, inscrire le mot "Fin" en position 1.

4. Dans votre programme principal, ajoutez une deuxième instance de la classe **Cadran** nommée **unAutreCadran**, cette fois-ci le cadran doit se placer sur la ligne 10 et la colonne 10 de votre écran. Il doit permettre d'afficher 2 lignes de 20 caractères.
5. Complétez les méthodes **Afficher** de la classe **Cadran** afin de pouvoir écrire sur une autre ligne (ici par exemple la deuxième ligne du second cadran). Par défaut (si on ne précise pas le paramètre lors de l'appel) l'écriture se fait sur la première ligne.
6. Complétez la classe **Cadran** en ajoutant une méthode **Effacer** qui vide le contenu du Cadran.
7. Dans le programme principal, réalisez dans la même boucle un décompteur de 10 à 0 en position 5 sur la ligne 2 puis affichage du mot "Fin" en position 1 dans le deuxième cadran. Utilisez la méthode **void Cadran::Effacer()** si besoin pour obtenir un meilleur résultat.
8. Ajoutez au constructeur de la classe, deux nouveaux paramètres permettant de fixer la couleur de fond et la couleur d'écriture de votre cadran
Vous pouvez vous aidez du lien suivant pour le choix des couleurs :
https://fr.qwe.wiki/wiki/ANSI_escape_code
9. Complétez la documentation de la classe **Cadran** et générez la documentation Doxygen dans le dossier **ProjetHorloge/documentation**.
10. Faites remonter sur votre Github l'ensemble du dossier **ProjetHorloge**.