

Exercice 4 : QGIS standalone

Partie 4.1

- QgsApplication (init/exit)
- QgsMapCanvas
- mise en place d'une application QGIS standalone

```
> cd exercices/ex4/part1
> mkdir build && cd build
> cmake .. && make

#include <qgsapplication.h>
      ^
compilation terminated.
```

Question 1 : Récupérer le script FindQGIS.cmake dans le projet QGIS et l'utiliser dans notre contexte pour compiler notre application QGIS standalone.

Partie 4.2

- QgsApplication
- QgsVectorLayer
- QgsFeature / QgsFeatureIterator
- QgsGeometry
- QgsMapLayerRegistry
- QgsProject
- QgsMapCanvas
- QgsMapCanvasLayer
- QgsMapSettings
- QgsMapRendererSequentialJob

Question 1

L'objectif de cet exercice est d'implémenter un remote viewer permettant de lire des fichiers .qgis présent sur une autre machine. Pour cela des sockets TCP sont utilisées pour transmettre les informations.

Le serveur doit être exécuté comme ci-dessous :

```
> ./server [port] [qgisfile]
```

Ainsi, le serveur va attendre une connexion cliente sur le numéro de port [port]. Une fois une connexion établie, le serveur va envoyer le positions/la géométrie des features des couches vectorielles au client de manière periodique.

Le client :

```
> ./client [ipserver] [port]
```

Le client essaie de se connecter au serveur présent sur la machine ayant l'ip [ipserver] et sur le port [port]. Il attend alors les informations de couches vectorielles (features) qu'il devra afficher dans un QgsMapCanvas.

Concernant la partie implémentation, la partie réseau (les sockets) sont déjà mises en place. Il ne s'agit ici que d'utiliser l'API QGIS puis d'utiliser les sockets pour simplement envoyer/recevoir les messages contenant les features.

Question 2

Dans un second temps, ajouter un bouton côté client afin de permettre de capturer le contenu du canvas et ainsi le sauvegarder dans un fichier image .png.