

DURÉE ET TARIF : 5 jours (35 heures) -

SANCTION DE LA FORMATION : Attestation de fin de stage et de présence, Evaluation des acquis

NATURE DE LA FORMATION : Adaptation et développement de compétences

Objectifs

La formation Qt 5 C++/QML aborde les techniques de développement d'applications graphiques, réseau et multitâche, en utilisant la technologie Objet C++ du framework Qt 5.

- Maîtriser les langages C++ et QML de programmation d'interface utilisateur tactile sous Qt 5
- Maîtriser les concepts de la technologie Qt C++ et ceux de Quick 2
- Maîtriser Qt Designer et le SDK Qt 5
- Maîtriser Qt Quick Designer et l'environnement de mise au point QML Debugger
- Mettre en oeuvre des composants graphiques dynamiques complexes
- Développer des applications Qt Quick 2 tactiles en QML
- Concevoir une architecture logicielle avec des composants Qt Quick 2 et des objets C++ Qt 5 à base de plugins
- Développer des applications communicantes sous Qt 5 et Qt Quick
- Développer des applications multitâches sous Qt 5 et Qt Quick
- Développer des applications Hybrides QML/C++

Pré-requis

La formation C++/QML est particulièrement destinée aux développeurs ayant de bonnes notions de programmation Objet, confrontés aux problèmes de développement d'applications graphiques, tactiles, multitâche et communicantes en utilisant la technologie hybride Qt5 C++/QML avec l'environnement Quick 2 sous Qt 5.15 LTS

Introduction à la technologie Qt5

L'offre Qt

- Les outils de développement Qt 5 : Qt-Creator, Qt-Designer, Qt Linguist et Qt Assistant
- Rappel sur les concepts Objets et programmation C++
- Particularités liées à la gestion mémoire sous Qt et QObject parent
- QMetaClass : Les propriétés, l'inspection ...

Programmation des interfaces graphiques

- Architecture d'une application Qt 5,
- Modules QtCore et QtGUI
- Classe QDialog, modalité et Boîtes de dialogue standards
- QWidget et Qt Designer
- Connexions signaux et slots
- Notion de layout et feuille de style
- QLocale et internationalisation des applications Qt 5 avec Qt Linguist

Travaux pratiques

- Construction d'applications de type QDialog sous environnement Qt Creator
- Initiation à la création d'interface et de Widget avec Qt-Designer
- Usage des QStateMachine pour la gestion des IHM

1.

Introduction à la Programmation Réseau sous Qt5

Module QtNetwork sous Qt 5

- QSocket TCP et UDP
- Classe QWebSocket

Programmation Multitâche sous Qt 5

- Affinité de thread et boucle d'événement
- Classe QThread, QEvent, Sémaphore et Mutex
- QTimer
- QProcess

Travaux pratiques

- Réalisation d'une application réseau multitâche client serveur sous Qt entre PC et une cible SBC Mira phyCORE-i.MX 6
- Réalisation d'une application multitâche d'acquisition de données asynchrone sous Qt

Introduction à QtQuick 2 Toolkit

- Éléments graphiques
- Éléments texte
- Layout « ancrage »
- Bindings
- Handlers

3.

Techniques avancées d'implémentation des collections C++ sous Qt5

Classes Containers, collections et sérialisation d'objets

- QString
- Gestion du système de fichiers
- QStreamData, QStreamText, méthode de sérialisation d'objets
- QLocale et internationalisation des applications Qt 5 avec Qt Linguist
- Gestion des réglages de l'application via QSettings
- QtXML : introduction au support XML (SAX et DOM)
- QVariant
- Vector, Lists et Maps
- Méthode de sérialisation d'objets
- QFuture et QConcurrent, parallélisation optimisée des traitements de données de collection

Travaux pratiques

- Exercices de manipulation de collections, traitement et sérialisation de données

2.

Interaction avec l'utilisateur sous Qt Quick

- Interaction à la souris
- Interaction tactile
- Interaction au clavier

Travaux pratiques

- Prise en main du Toolkit QML
- Premières applications QML tactiles

Les Composants

Les Animations

Travaux pratiques

- Mise en oeuvre d'applications graphiques QML animées simples

Présentation des données sous Qt Quick

- Ordonner des éléments graphiques
- Modèles de données simples
- Les vues

Travaux pratiques

- Mise en oeuvre d'applications de présentation/saisie de données simples
- Mise en oeuvre d'applications de graphiques multi-interfaces/multi-contrôles

4.

Intégrer QML avec C++

- Exporter des objets C++ en QML
- Exporter des classes en QML
- Exporter des classes non graphiques
- Exporter des classes graphiques

Travaux pratiques

- Mise en oeuvre d'une application graphique QML de présentation de données sérialisées dans un fichier au travers une collection C++ Qt
- Usage et Analyse des traces du QML Profiler

Programmation avancée des Interfaces graphiques sous Qt5

Intégration de fenêtre QML dans une application C++ sous Qt 5

- Classe QMainWindow
- Gestion des menus et barre de menu
- Traitement des QAction
- Interface multidocuments/vues
- Mise en oeuvre des QQuickWidget, QQuickView, QQuickTextDocument et QQuickWindows dans une QMainWindow

Travaux pratiques

- Réalisation d'application d'affichage graphique de scene QML dans une application Qt 5 C++ QMainWindow classique

5.