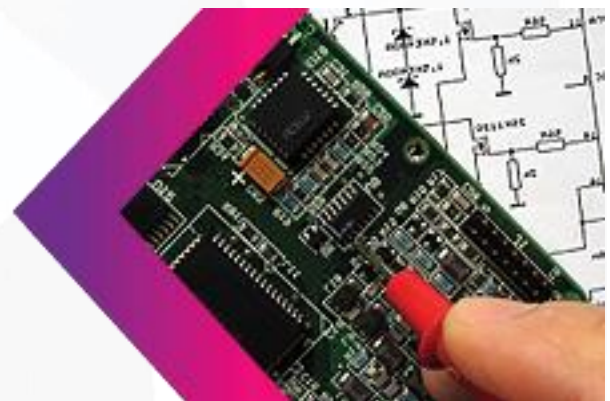
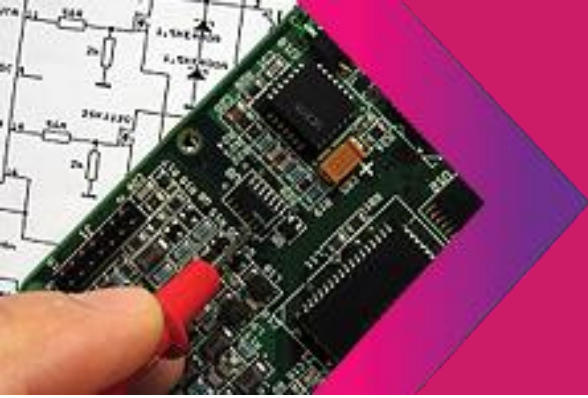




UI Programming using Qt Quick and QML Workshop





NOW FOR SIMPLIFIED SOLUTIONS

UI Programming using Qt Quick and QML - Workshop

- **Objectives:**

- ✓ Become proficient in developing GUI using QML
- ✓ Get familiar with QML concepts - elements, layouts, positioners, properties and signals
- ✓ Learn to structure/modularize QML programs
- ✓ Implement Fluid GUI design
- ✓ Learn to build applications by integrating QML and C++

- **Duration:**

- ✓ 2 days (One weekend)
- ✓ 4 session (Duration of each session: 3 to 3.5 hours)

- **Platform:**

Any Linux Platform supporting QT (Preferably Ubuntu)

- **Delivery method:**

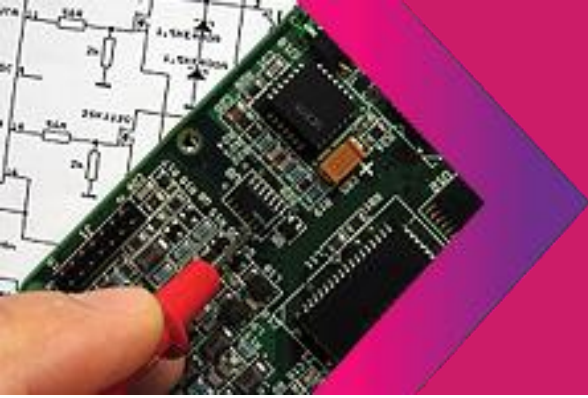
Workshop based approach with demo and hands-on exposure

- **Pre-requisites:**

Knowledge about Qt programming using C++

- **Detailed course contents:**

- ✓ **Kickstart**
 - Using the IDE
 - *Illustration: Hello World*
- ✓ **Qt Quick Overview**
 - Qt Quick Components
 - QML Language Overview
 - Qt Modules Overview
- ✓ **Programming with QML**
 - Basic concepts and GUI elements
 - QML Syntax
 - QML Components
 - Properties
 - Alias properties
 - Signals
 - Functions



NOW FOR SIMPLIFIED SOLUTIONS

- ✓ **Building GUI using QML**
 - QML Elements
 - QML Item
 - Visual Elements
 - Item Layouts
 - Item Coordinates
 - Item anchors
 - Positioners

- ✓ **Structuring QML Programs**
 - QML Components (creation)
 - Component files and import
 - Property scopes
 - Inline components
 - Dynamic Loading

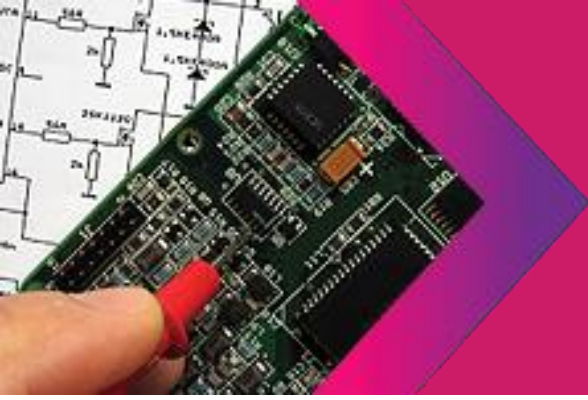
- ✓ **Building Fluid GUI**
 - QML Animations
 - Animation Overview
 - Animation basics
 - Animation types
 - Animation objects
 - Animation grouping
 - Drag and drop

 - GUI states and animated state transitions
 - GUI states in QML
 - State transitions

- ✓ **User Interaction**
 - Mouse and Key Events
 - Mouse Input
 - Mouse drag
 - Keyboard Input

 - Flickable element

- ✓ **Displaying Data**
 - Models, views and delegates
 - Data elements
 - Data models
 - Data views
 - GUI delegates



NOW FOR SIMPLIFIED SOLUTIONS

- ✓ Hybrid programs (Integrating QML with C++)
 - QDeclarativeView
 - QdeclarativeContext
 - Exporting Objects to QML
 - Using exported classes in QML
 - QML object visibility



Emertxe Information Technologies Private Ltd
#83, 1st Floor,
Farah Towers,
MG road,
Bangalore - 560001

T: +91 809 555 7 333 (M), +91 80 4128 9576 (L)

E: training@emertxe.com

www.emertxe.com