

Flutter Application Development Training

COURSE CONTENT

GET IN TOUCH



Multisoft Systems
B - 125, Sector - 2, Noida



(+91) 9810-306-956



info@multisoftsystems.com



www.multisoftsystems.com

About Multisoft

Train yourself with the best and develop valuable in-demand skills with Multisoft Systems. A leading certification training provider, Multisoft collaborates with top technologies to bring world-class one-on-one and certification trainings. With the goal to empower professionals and business across the globe, we offer more than 1500 training courses, which are delivered by Multisoft's global subject matter experts. We offer tailored corporate training; project Based Training, comprehensive learning solution with lifetime e-learning access, after training support and globally recognized training certificates.

About Course

Flutter Application Development training by Multisoft Systems provides a comprehensive curriculum designed for both aspiring and experienced developers looking to master Flutter. This course covers essential aspects of Flutter, including its framework and core concepts, allowing you to build high-quality, cross-platform mobile applications for Android and iOS.

Module 1: Introduction to Flutter and Dart Programming Language

- ✓ Introduction
- ✓ Importance of Flutter
- ✓ Introduction to Dart
- ✓ Writing Dart code
- ✓ DartPad
- ✓ Installing Dart SDK
- ✓ IntelliJ IDEA

Module 2: Dart Programming - Syntax

- ✓ Introduction
- ✓ main() function
- ✓ Dart Variables
- ✓ Dart Data Types
- ✓ Input of Information to Dart Program
- ✓ Writing Comments
- ✓ Dart Conditional Operators
- ✓ If Statement
- ✓ If – Else Statement
- ✓ If...Else and Else...If... Statement
- ✓ If Else and Logical Operators
- ✓ For Loops
- ✓ While Loops
- ✓ Do-while Loops
- ✓ Break Statement
- ✓ Switch Case Statement

Module 3: Dart Functions & Object-Oriented Programming (OOP)

- ✓ Functions
 - ❖ Function Structure
 - ❖ Creating a Function
 - ❖ Function Return Data Types
 - ❖ Void Function
 - ❖ Function Returning Expression
 - ❖ Functions and Variable Scope
- ✓ Object-Oriented Programming (OOP)
 - ❖ Object
 - ❖ Class
 - ❖ Creating a Class
 - ❖ Adding Methods to Classes
 - ❖ Providing Constructors for Your Classes
 - ❖ Class — Getters and Setters
 - ❖ Class Inheritance
 - ❖ Abstract Class
- ✓ Dart Project Structure and Dart Libraries

Module 4: Introduction to Flutter

- ✓ Understanding Flutter
- ✓ Flutter Framework
- ✓ Android Studio
- ✓ What is Android Studio?
- ✓ Android Studio Software Prerequisite
- ✓ Installing Android Studio
- ✓ Flutter SDK

- ✓ Installing and Configuring Flutter SDK
- ✓ Creating a New Flutter Project
- ✓ Setup an Android Virtual Device
- ✓ Run a Flutter App
- ✓ Installing Flutter on Mac
- ✓ Test Your Flutter App on iOS Phone with Windows O.S
- ✓ Android Studio Sugar and Spice
- ✓ Run your Apps on a Hardware Device (Physical Phone)
- ✓ Run your Flutter App on Android Phone
- ✓ Run your Flutter App on iPhone Device
- ✓ Emulator Debug Mode
- ✓ Introduction to Flutter Widgets
- ✓ Creating a Flutter App Using Widgets
- ✓ What is a MaterialApp widget?

Module 5: Flutter Widgets Fundamentals

- ✓ Scaffold Widget
- ✓ Image Widget
- ✓ Container Widget
- ✓ Column and Row Widgets
- ✓ Icon Widget
- ✓ Layouts in Flutter
- ✓ Card Widget
- ✓ App Icons for iOS and Android Apps
- ✓ Hot Reload and Hot Restart
- ✓ Stateful and Stateless Widgets
- ✓ Use a Custom Font

Module 6: Navigation and Routing

- ✓ Button Widget

- ❖ FloatingActionButton
- ❖ RaisedButton, FlatButton, and IconButton
- ❖ DropdownButton
- ❖ OutlineButton
- ❖ ButtonBar
- ❖ PopupMenuButton
- ✓ App Structure and Navigation
- ✓ Navigate to a New Screen and Back
- ✓ Navigate with Named Routes
- ✓ Send and Return Data Among Screens
- ✓ Animate a Widget Across Screens
- ✓ WebView Widget in Flutter

Module 7: Visual, Behavioral, and Motion-Rich Widgets implementing Material Design Guidelines - Part 1

- ✓ Introduction
- ✓ BottomNavigatorBar Widget
- ✓ DefaultTabController, TabBar, and TabBarView Widgets
- ✓ ListTile Widget
- ✓ ListView Widget
- ✓ Drawer Widget
- ✓ DataTable Widget
- ✓ SelectableText Widget
- ✓ Stack Widget

Module 8: Visual, Behavioral, and Motion-Rich Widgets implementing Material Design Guidelines - Part 2

- ✓ Input and Selections
 - ❖ Text Field Widget

- ❖ CheckboxGroup and RadioButtonGroup Widgets
- ❖ Date Picker
- ❖ Time Picker
- ❖ Slider Widget
- ❖ Switch Widget
- ✓ Dialogs, Alerts, and Panels
 - ❖ Alert Dialog Widget
 - ❖ Cupertino Alert Dialog Widget
 - ❖ Bottom Sheet
 - ❖ Modal Bottom Sheet
 - ❖ Persistent Bottom Sheet
 - ❖ Expansion Panel Widget
 - ❖ Snack Bar Widget

Module 9: Firebase

- ✓ Introduction
- ✓ What is the JSON?
- ✓ How does Firebase Database work?
- ✓ Firebase authentication (Signup and Login to Flutter App)
- ✓ Configure Your App to use Firebase Services
 - ❖ Adding Firebase to your Android App
 - ❖ Adding Firebase to your iOS App
- ✓ Configuring Firebase Authentication
 - ❖ Login to an App Using Firebase User Accounts
 - ❖ Logout Configuration
- ✓ Firebase Database
 - ❖ Which database is right for your project?
 - ❖ Real Time Database
 - ❖ Cloud Firestore

Module 10: Location-Aware Apps: Using GPS and Google Maps

- ✓ Introduction
- ✓ What is GPS and how does it work?
- ✓ The Camera Position
- ✓ Adding Google Maps to a Flutter app
 - ❖ Getting a Google API key
 - ❖ Adding Google Maps Flutter plug-in as a dependency
 - ❖ Adding your API key for your Android app
 - ❖ Adding your API key for your iOS app
 - ❖ Adding a Google Map on Your Flutter App Screen
 - ❖ Adding a Google Map Marker
 - ❖ Google Map Types
 - ❖ Moving the Camera (Camera Animation)
 - ❖ Capturing an App User's Location for iOS and Android Apps
 - ❖ Getting a Google API key
 - ❖ Creating an App Interface
 - ❖ Configuring your App to Use Your API Key
 - ❖ Adding a Google Map on your Flutter App Screen
 - ❖ Adding a Google Map Marker
 - ❖ Capturing Users' Location
 - ❖ Configuring User App's Permission

Module 11: App Testing & Publishing

- ✓ Testing and Feedback for Your App
 - ❖ Setting up a Test Environment
 - ❖ Usability Testing by Participants
 - ❖ Starting your Test Session

- ❖ Analyzing your Test
- ✓ Publishing Flutter Apps
- ✓ Publishing Android App on Google Play Store
- ✓ Publishing iOS app on Apple Store