MS40542 - Build Native Cross-Platform Mobile Applications with a Shared UI for iOS, Android and UWP in C# .NET with Xamarin.Forms

Microsoft - Xamarin - Sviluppo

Durata:	Lingue:	Certificazione:
5 Giorni	**Italiano**	**-**

Descrizione del corso

During this course, developers will learn to use a variety of techniques, from beginner to advanced, to develop cross-platform apps with Xamarin.Forms. Topics begin with the basics of mobile application development with C# and .NET and then advance to include data storage with SQLite, using REST-based services, designing a shared UI with XAML, displaying data, and using effects and renderers to create enterprise quality cross-platform applications.

Programma

Module 1: Getting Started with Xamarin (XAM101)
- Define the state of mobile technology today
- Discuss mobile app trends
- Identify approaches to development
- Discover the Xamarin Approach
- Setup your development environment

Module 2: Introduction to Xamarin.Forms (XAM120)
- Create a single screen cross-platform application
- Cross Platform Applications with Xamarin
- Arrange the UI using Layouts
- Use platform-specific features in shared code

Module 3: Introduction to Cross-platform Development (XAM110)
- Add shared components to your application

- Share code using Shared Projects
- Share code using Portable Class Libraries
- Share code using .NET Standard Libraries

Module 4: XAML in Xamarin.Forms (XAM130)
- Examine XAML Syntax
- Add behavior to XAML-based pages
- Explore XAML capabilities

Module 5: Layout in Xamarin.Forms (XAM135)
- Specify the size of a view
- Arrange views with StackLayout
- Apply Attached Properties
- Arrange views with Grid
- Scroll a layout with ScrollView

Module 6: Resources and Styles (XAM140)
- Avoid duplicate XAML with Resources
- Create consistent UI with Styles
- Make your Resources and Styles available across your entire app
- Apply the user's Accessibility choices with built-in Styles

Module 7: Data Binding in Xamarin.Forms (XAM270)
- Use data binding to decouple code-behind from UI
- Use value converters to bind incompatible types

Module 8: Model-View-ViewModel (XAM320)
- Decide where to place code in Model-View-ViewModel
- Define Visual Behavior
- Use Commands
- Test MVVM based apps

Module 9: Patterns for Cross-Platform Development (XAM250)
- Locate dependencies using the Factory Pattern
- Use a Service Locator to register and retrieve dependencies
- Use an IoC container to automatically inject dependencies

Module 10: Consuming REST-based Web Services (XAM150)

- Obtain the device's network capabilities
- Introduce REST
- Consume REST services with Xamarin
- Integrate with platform-specific network features

Module 11: SQLite and Mobile Data (XAM160)
- Choose a data storage strategy
- Store data locally with SQLite
- Use SQLite asynchronously

Module 12: ListView in Xamarin.Forms (XAM280)
- Display a collection with ListView
- Add and remove items dynamically
- Customize ListView rows

Module 13: Customizing ListViews (XAM312)
- Create custom cell definitions
- Add headers and footers
- Separate your data into Groups
- Performance-tune your ListViews

Module 14: Xamarin.Forms Effects (XAM330)
- Customize control appearance
- Apply effects to controls
- Create an effect

Module 15: Xamarin.Forms Renderers (XAM335)
- Embed native controls into Xamarin.Forms
- Customize a renderer for an existing control
- Create a renderer for a custom control
- Send notifications between renderer and element