Since all client applications require some sort of end-user experience, user interface development is the first topic I will dive into after the introduction provided in Chapter 1. Also, for client-application development, it is usually the first aspect of the application that developers start to experiment with in learning a new technology.

The user interface is usually the hallmark of a great application proposal or application prototype. It helps communicate what an application will look like, as well as what it will actually do. How an application looks and feels is just as important as what the application actually does. It is understandable, as users expect more from client-applications, especially non-Web-based client applications such as mobile applications.

First up is an overview of the Windows Phone application design language and design resources. The Windows Phone product team is incredibly passionate about the user interface they created for Windows Phone. In an effort to help jumpstart great user experience (UX) design and development for Windows Phone developers, the Windows Phone design team created detailed resources and guidance, which I cover in this chapter.

In this second edition of the book, I have expanded this section covering design to include a description of what makes a well-designed application. In addition, I provide coverage of Expression Blend 4 SketchFlow, which can help you create design documentation and a working prototype that you can share without having to require a Windows Phone device.

After covering design, the chapter dives into developing UX in Visual Studio 2010, with a quick background on the underlying architecture of the Silverlight programming model and the .NET Framework for Silverlight for mobile developers new to the platform. This book is not an introductory one, but this section provides hopefully enough background to fill in the major blanks as well as help understand the important namespaces and classes if you are new to Silverlight development.

The markup language for Windows Phone UX is Xml Application Markup Language (XAML). XAML development started out in Visual Studio 2005 with a basic design surface for Windows Presentation Foundation. Expression Blend was created to provide a design tool to make it easier for designers to work in XAML, but in a visual way. Both Visual Studio 2010 and Expression Blend XAML tooling has improved greatly, with unique support for phone development with Silverlight. I cover the built-in controls and capabilities using Visual Studio 2010 as the backdrop.

After covering Visual Studio 2010, UX development with Expression Blend 4 is next. As a developer, you need to learn Expression Blend 4 – it is definitely not just a tool for designers, and you will find some tasks are much easier to do in Expression Blend, such as visual data binding and creating sample data. The section covers laying out controls in Blend, sample data, data binding, and animations in Expression Blend as well.

The chapter concludes with an overview of what’s new in Windows Phone OS 7.1 (Mango) as it relates to user interface development, where the biggest change in the update is moving from Silverlight 3 compatibility to Silverlight 4.
UX Performance Improvements

Developers with experience in Windows Phone have probably experienced challenges with UI performance for ListBox scrolling. There were a set of tips and tricks that developers leveraged to improve scroll performance for ListBox controls, data virtualization, etc. With Windows Phone OS 7.1 (Mango), the product team made significant improvements in controls so that developers do not have to do this work themselves. There are two major areas of improvement:

- An input thread has been added to improve performance for things such as ListBox scrolling.
- Images are now decoded on a background thread, which means text and other elements will appear quickly and the images will load when they are available.

WebClient has also been improved to return on the calling thread. I cover WebClient in Chapter 4. In the next section I provide an expanded overview on designing for Windows Phone.

Designing for Windows Phone

This section covers design-oriented content that a developer will find very useful. I phrase it that way because this book in general is not geared toward designers; however, just as end-users were no longer satisfied with character-based UI when graphical UI came into vogue, today’s end-users expect even more from client-based mobile applications when compared to the rich UX available via the desktop. Developers are an integral part to the design/development cycle, which is why it is strongly encouraged that developers embrace design concepts as much as possible to help create better UX beyond design wireframes and visual comps.

Design Approach

The design team at Microsoft refers to the “design language” of Windows Phone, codenamed “Metro,” when describing the approach they took when designing the Windows Phone UI. The Metro codename is inspired by the graphics found in transportation systems worldwide, which rely on big, bold, beautiful graphics and text to communicate to the “end-user” or traveler where they need to go. It is inspired by other sources as well, such as Zune, video games, product packaging, and the like, which focus on these key principles:

- **Light and simple:** Focus on primary tasks, achieving a lot with very little (little user effort and few phone resources, steps, and so on). Use whitespace to its full advantage.
- **Typography:** Focus on beautiful, not just legible, with sensitivity to font weight, balance, and scale to communicate a sense of hierarchy for the content.
- **Motion:** Responsive and alive UI. Transition between UI is just as important as the design of the UI. Create an animation system, not just transitions.
- **Content, not chrome:** Reduce visuals that are not content. Navigate via content itself.