1. Quickly and efficiently create forms and minimize training time.

Create forms to support business processes using the familiar Microsoft Office tools, such as fonts, bullets and numbering, table design, and clip art. The familiar tools and environment help minimize training time and speed forms creation.

The Fluent user interface, introduced in many 2007 Office system applications, is included in InfoPath 2010. Based on extensive usability data and recent advancements in hardware and software, the Fluent user interface is designed to make it easier for you to find and use the full range of features in Office applications—and get the results you want.

2. Easily create sophisticated forms without writing code.

For example, you can:

· Quickly create and deploy forms using improved rules management, quick rules, styles, and dynamic layout sections.

· Set rules for conditional formatting.

· Create forms that include actions, such as alerts, calculated values, and data validation.

Create validations, formatting, or actions based on user roles or create validations and actions that show based on the type of field.

3. Give participants the ability to fill out forms online or offline.

With InfoPath 2010, people can fill out forms online or offline with a native integration of forms in the Microsoft SharePoint Workspace interface. A user can easily take a SharePoint library with InfoPath forms offline in the SharePoint Workspace environment. Information entered in the forms will be automatically synchronized to the SharePoint library once the participant is back online, so people can be productive whether they’re connected to a network or not.

4. Build modular, extensible, and portable SharePoint applications.

Use SharePoint Server 2010 and InfoPath 2010 to create SharePoint applications. SharePoint applications can be built, for use at the departmental or enterprise level, using little or no code. These solutions are modular (you can build from existing list or library), extensible (you can add capabilities with code), and portable (you can publish using .wsp format to move the application from site to site or server to server).

5. Make browser forms conform to standards.

InfoPath 2010 forms on SharePoint Server 2010 are now compliant with Web Content Accessibility Guidelines 2.0 (WCAG 2.0) AA to help you create forms that are accessible to users with disabilities. In addition, forms powered by SharePoint Server 2010 are now fully XHTML 1.0 compliant.

6. Enjoy more powerful Web browser forms.

Improved parity between InfoPath 2010 client forms and InfoPath forms in SharePoint Server 2010 ensures greater consistency for users who are filling out forms. For example, functionality available in both environments include bulleted, numbered, and plain lists; multiple-selection list boxes; combo boxes; picture buttons; hyperlink capabilities; choice group and section; filtering functionality; date and time controls, and people pickers.

7. Embed forms in Web Parts.

In SharePoint Server 2010, it’s easier than ever to host your forms on Web pages using the new InfoPath Form Web Part. In SharePoint Server 2007, users who wanted to host their InfoPath forms on Web pages had to write code in Visual Studio. Now, without writing a single line of code, you can simply add the InfoPath Form Web Part to a Web Part page and point it to your published form.

You can use the Web Part to host any InfoPath browser form that has been published to a SharePoint list or form library. You can also connect it to other Web Parts on the page to send or receive data.

8. Enhance forms with code.

Although you can customize many aspects of InfoPath forms without writing code, you can also enhance InfoPath forms with code if declarative logic does not meet the needs for implementing

your solution functionality.

You can enhance InfoPath 2010 forms using Visual Studio Tools for Applications (VSTA), which ships with InfoPath 2010. VSTA provides a light-weight version of the Visual Studio Integrated Development Environment. You can use VSTA to write managed code in C# or VB.NET. The code is then embedded in the InfoPath form template (.xsn file). For example, if you want to include the ability for calculating interest in a form, you can embed code in the form itself and InfoPath Filler or InfoPath Forms Services will run this code when the form is opened or edited.

9. Connect forms with line-of-business information and REST Web Services.

SharePoint Server 2010 offers an extensible architecture for managing user access to data connections and systems. InfoPath 2010 also integrates with Business Connectivity Services (BCS) of SharePoint Server 2010. BCS enhances Office applications and SharePoint platform capabilities with out-of-box features, services, and tools that streamline development of solutions with deep integration of external data and services. Also, InfoPath 2010 now supports getting XML data from REST Web Services. REST Web Services use input parameters that are passed in through a URL. Form designers can now change the URL parameters dynamically in the InfoPath form without any code using rules to get desired data from the REST Web Service.

10. More easily manage forms on the server.

InfoPath forms can be monitored as a true component of SharePoint Server 2010. New SharePoint Maintenance Engine Rules ensure that InfoPath forms are correctly configured in their farms. You can also manage InfoPath forms using Windows PowerShell command line shell and scripting language, which is included in SharePoint Server 2010.