7/14/2020

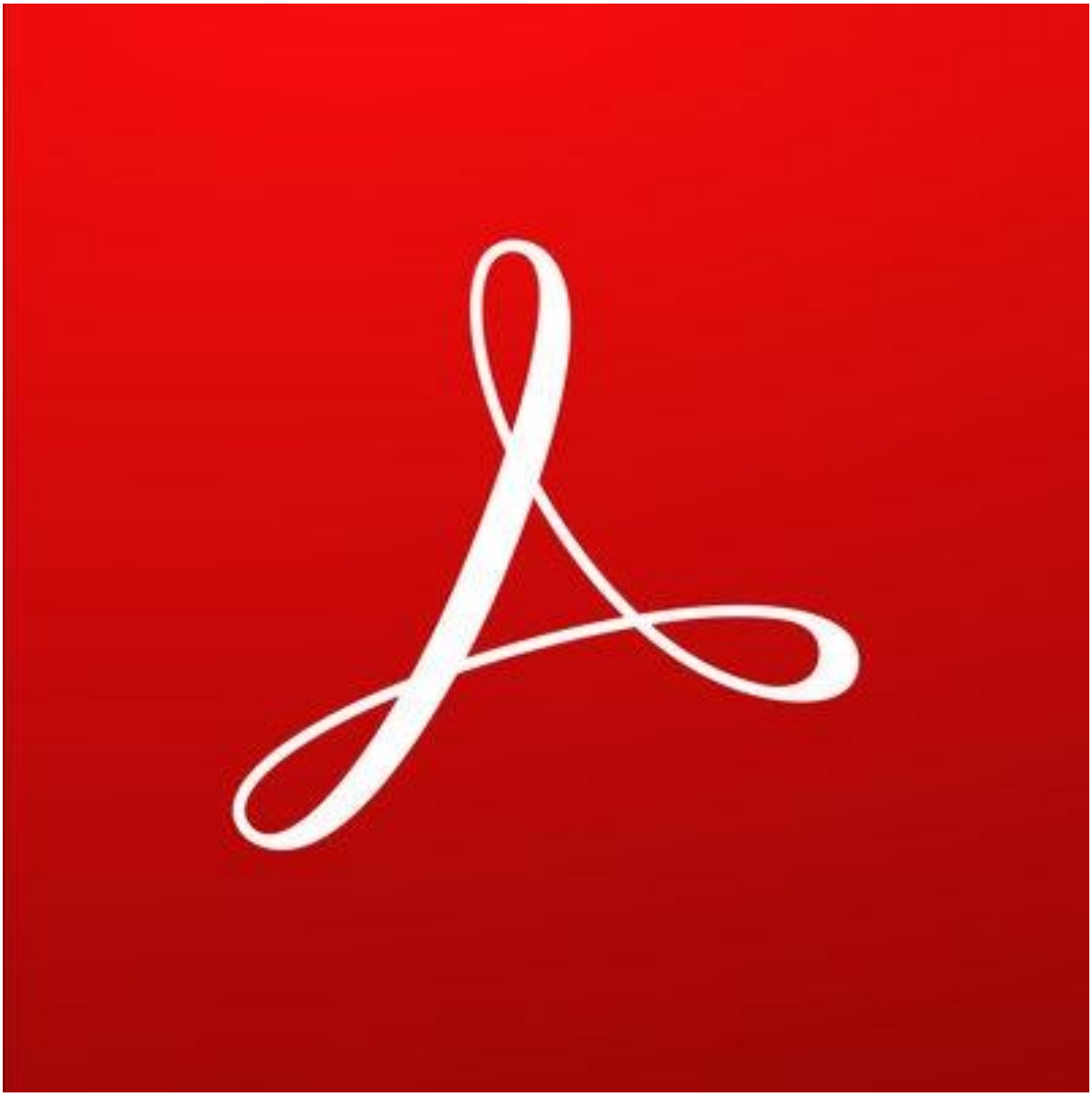
# Software Engineering

## Project: Adobe Reader

By:
MAHRUKH_037
HALA_ALI_KHAN_007
ARSHIYA_SALEEM_004
IZWA_AFZAL_012

**Project:**

# Adobe Reader



## SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC)

As we think adobe acrobat reader is developed under agile methodology. Agile methods are being widely accepted in the software world recently. The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. As we know that in agile model functionality can be developed rapidly and demonstrated, suitable for fixed or changing requirements, same in adobe acrobat reader new functionality is continuously updating as time passes. Also there is online help support forum for customers if they found any query they can ask for solution. Good model for environments that change steadily. Customer Interaction is the backbone of this Agile methodology, and open communication with minimum documentation are the typical features of Agile development environment.

Acrobat Reader DC works with Adobe Document Cloud services to take the world's best PDF viewer to an entirely new level. With an intuitive interface, Acrobat Reader DC delivers powerful new functionality to help you get work done from anywhere on any device. The newly expanded Home view helps you access and manage documents, track the one you have shared with others and provides visibility into documents shared with you to view, review, or sign along with detailed information about the task. Acrobat Reader DC desktop software runs on both Windows and Mac operating systems. Just like Reader on the desktop, Acrobat Reader DC for mobile is free and continues to offer the industry's most reliable mobile experience for viewing and interacting with PDFs. And now, with the addition of Adobe Document Cloud services, you can unlock premium features within the app to do more with PDF.

## FUNCTIONAL REQUIREMENTS

In software Engineering functional requirements tell what a system do and its components. It describes the functions a software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data operation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional requirements can also expressed in the form of use case.

- **Edit ,create and export files:**
  **Acrobat Reader** DC is a free, stand-alone application that you can use to open, view, sign, print, annotate, search, and share **PDF** files. Through Acrobat reader you can open read and print pdf, you can fill out pdf forms. If you want to edit a pdf document, convert a document to pdf, you can do that. You can convert Microsoft Word, Microsoft Power Point in pdf, jpeg in to pdf, excel file in to pdf, png file into pdf, it also provide the function of e-signature and online support forum (electronic signature).
- **Commenting tools:**

Reader makes it easy to annotate PDF documents. With Reader on your desktop, you can: annotate PDFs using sticky notes; type text directly onto the page; highlight, underline, or use strikethrough tools; and draw on the screen with the freehand drawing tool. The same commenting tools are also available in Acrobat Reader for mobile

- **Fill out and save forms:**

  The Fill & Sign tool automatically recognizes whether your PDF has fillable form fields or not and takes you to an in-built filling experience, powered by Adobe Sensei, to get the job done quickly and easily. With a fillable form, it's easy to type your answers or select from drop-down lists, then save your completed form. When working with simple PDFs that haven't been optimized with form fields, you can click anywhere and type answers right onto the form, or you can accept suggestions from your personal autofill collection. When you're done, you can save your changes and send the completed form to others.

- **Export PDF files:**

  When you subscribe to Adobe Acrobat Export PDF you can convert PDF files to Microsoft Word, Excel, or RTF files by choosing Export PDF from the Tools Center in Reader. When you subscribe to Adobe Acrobat PDF Pack, you can also convert PDFs to PowerPoint. In addition, you can also export PDFs from the Acrobat Reader mobile app or your web browser.

- **Create PDF files:**

  When you install to Adobe Acrobat PDF, you can convert files to PDF by choosing the Create PDF tool from the Tools Center in Reader. You can also create PDFs from the Acrobat Reader mobile app or your web browser. You can also use Adobe Scan to create high-quality PDFs from papers and whiteboards with automatic text recognition.

- **Accessibility for people with disabilities:**

  Acrobat Reader DC makes it possible for people with disabilities – such as blindness, low vision, or mobility impairments – to interact with PDF documents and forms. Accommodations include built-in capabilities as well as support for users of assistive technologies. In addition, Adobe Acrobat Pro DC provides authors with a complete set of tools to create and optimize accessible PDF files.

- **Using Acrobat Reader DC on Windows**:

  Users can:

  - Customize accessibility preferences with an easy-to-use wizard
  - Work with magnifiers
  - Read PDFs out loud using built-in functions or working with professional screen-reading software.
  - Set high-contrast colors for better visibility
  - **NEW** Read text easier in high contrast mode with dynamic background adjustment
  - Enable large-type display by setting text to reflow through pages

- **Using Acrobat Reader DC on Mac** :

Users can:

- **NEW** Work with the VoiceOver screen reader on Mac OS X computers
- Work with magnifiers
- Set high-contrast colors for better visibility
- **NEW** Read text easier in high contrast mode with dynamic background adjustment
- Enable large-type display by setting text to reflow through pages.

## NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are the requirements how system behave while performing operations. A non-functional requirement is a specification that describes the system's operation capabilities and limitations that improve its functionality. As every software has some functional as well as non-functional requirements. Adobe reader also have some non-functional requirements. First non-functional requirement is that when you convert any file into pdf it will take few seconds to convert. And it will display the same writing style and same text size, the way original document is it will same copy in pdf document.

- **Protected Mode:**

  To protect you and your organization from spiteful code that attempts to use the PDF format to write to, or read from, a computer's file system, Adobe delivers an implementation of "sandboxing" technology called Protected Mode. Enabled by default in Acrobat Reader DC, Protected Mode protects against attackers who attempt to install malware on your computer system and blocks spiteful individuals from accessing and extracting, sensitive data and intellectual property from your computer or corporate network.

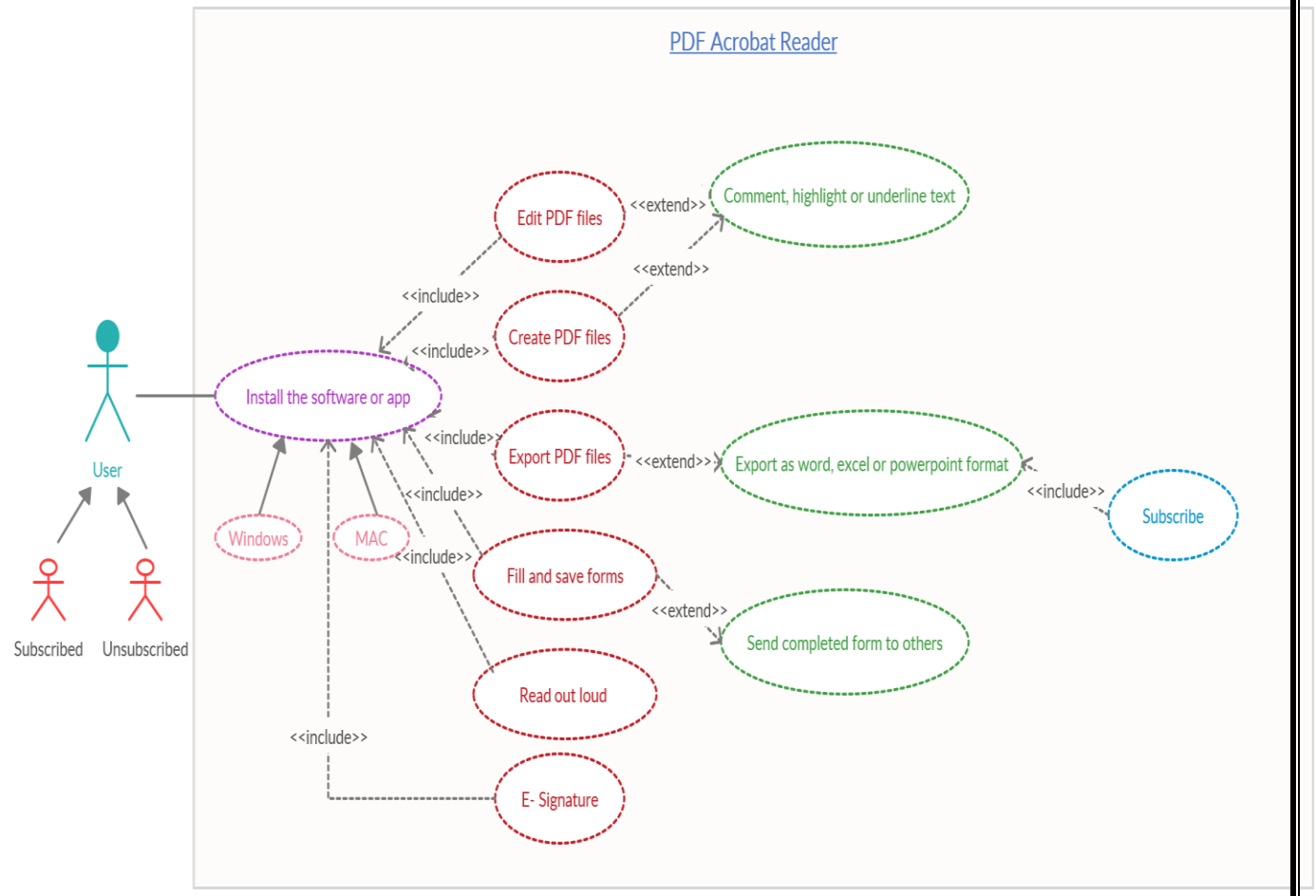
- **Control access of sensitive Information:**

  With Adobe acrobat software in security settings, authors can define who can open, view, print, copy, or modify a document. These capabilities help organizations protect the confidentiality of sensitive information. With access privileges, authors can define a password that users will need to open the document, or they can use a certificate ID to code the document so select recipients can open it by entering their own, unique certificate IDs.

- **Cloud Services secure:**

Adobe documents are cloud services secure. At Adobe, security practices are deeply in-built into our internal culture, software development, as well as service operations processes. Whether related to identity management, data confidentiality, or document integrity, Adobe Document Cloud services employ leading-edge security practices to protect your documents, data, and personal identifiable information to the highest degree possible. The following diagrams represent the use case and class diagrams of PDF Acrobat Reader.

## USE CASE DIAGRAM:




## Actors:

The primary in this case is the user. The user has been generalized as subscribed and unsubscribed user.

## Install app or software:

In order to use the app, the foremost step is to install the suitable version of this app on your PC or mobile phone, which will give you access to a number of useful features.

## Edit PDF Files:

The app allows you to edit the PDF files. The editing includes commenting, highlighting or underlining the text.

## Create PDF Files:

Similarly, anyone can create a PDF file using this app.

## Export PDF Files:

In order to export the PDF file in other format, such as word, excel, power point etc. the user must be a subscribed user. Or it is accessible to the user who has subscribed to adobe acrobat PDF reader.

## Fill & Save Forms:

This app allows you to fill the forms and save them. Moreover it allows you to share that form with others too.

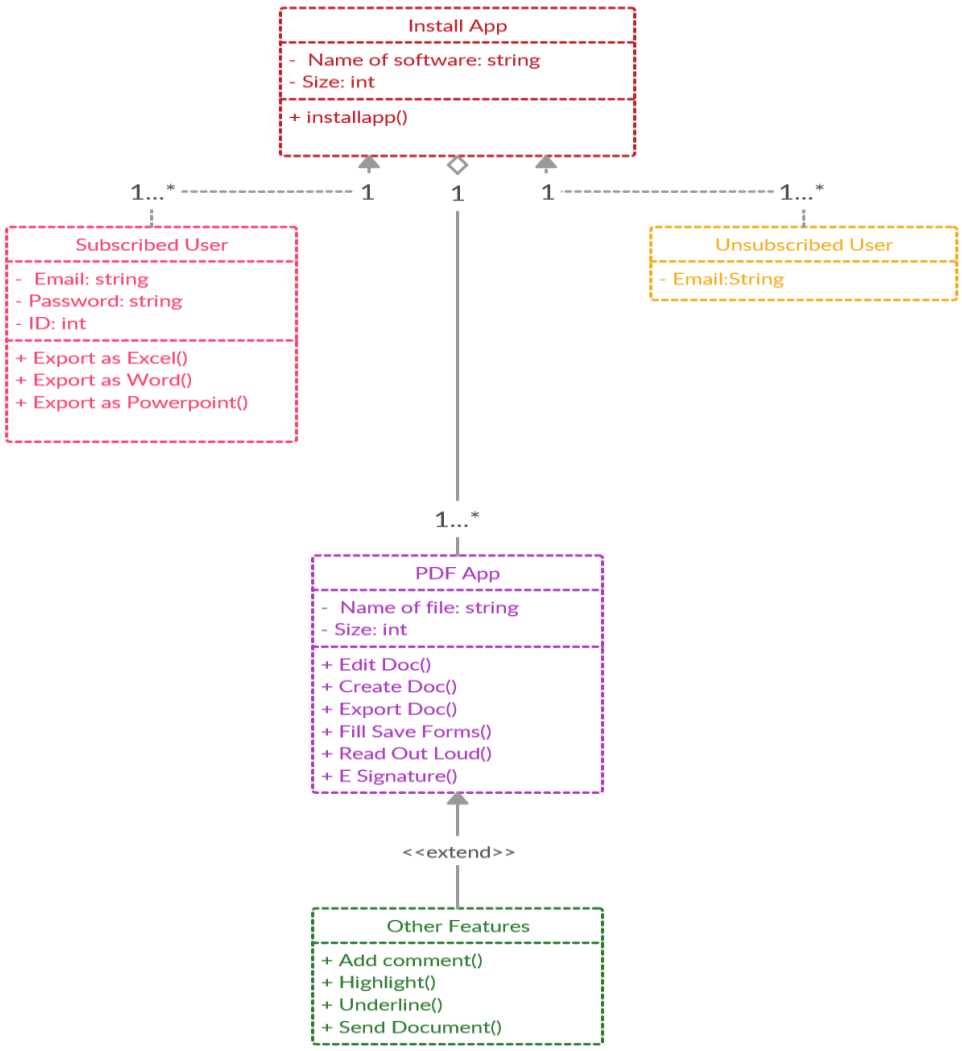
## Read Out Loud:

This feature is the latest feature added to this app. It reads the text on the screen, bringing ease to the user. It can be really beneficial for blind people.

## E-Signature:

E-Signature is a feature that allows you to add your signature as you were doing it directly on the paper.

## CLASS DIAGRAM:




The above diagram is the pictorial representation of functions and operations involved in the app.

## Install App:

The very first class or the parent class is install app, which has the attributes named as software name and the size of the software, the function install app is the only operation performed in the class.

## Types of users:

The parent class has been divided into two types of users subscribed and unsubscribed, who have their own attributes in addition to the base class attributes.

## PDF App:

Next class includes the basic or main features of this app, which are listed as the functions of the class.

## Other Features:

In addition to the main features some more additional features are also mentioned. They are presented as extend function because their use depends upon the desire of the user or in other words they are optional.

## SOFTWARE ARCHITECTURE:

PDF reader works on documents. It's packed with all the tools you need to convert, edit, and sign PDFs. You can use your device camera to capture a document, whiteboard, or receipt and save it as a PDF. We have top five software architectures i.e. **Layered (n-tier) architecture, Event-driven architecture, Microkernel architecture, Micro-services architecture** and **Space-based architecture.** But we only choose one of them on basis of their functionalities. For the whole system, **Microkernel software architecture** is the best choice PDF reader to maintain the whole system. It is because if we see other four architectures for PDF reader, then we will come know that they are not compatible with this software, these architectures having functionality of database, user interfaces, asynchronous data flow, Websites with small components, High-volume data like click streams and user logs etc. are not suitable for this system.

## MICROKERNEL ARCHITECTURE:

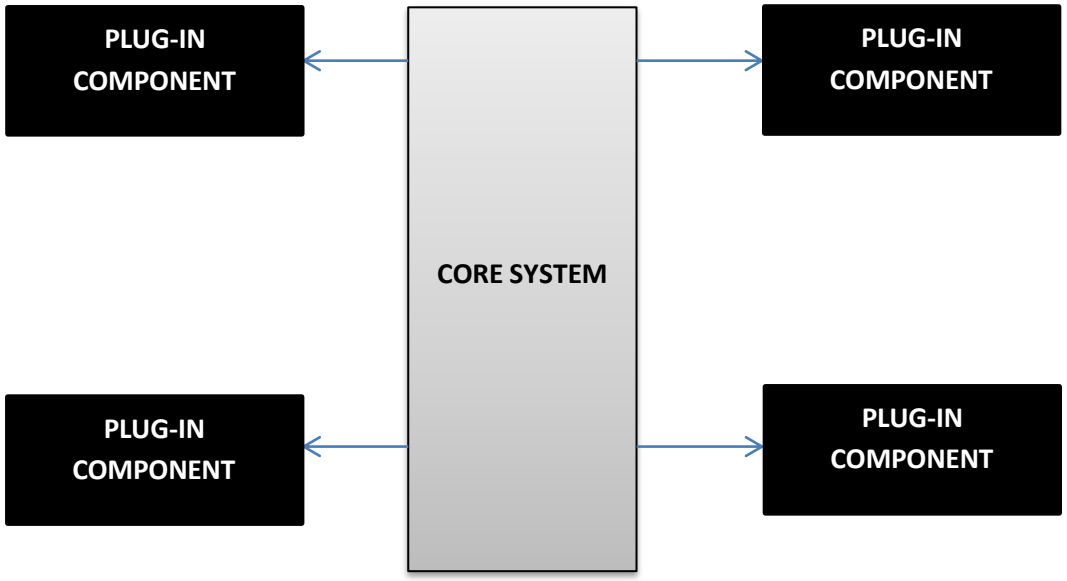
"The microkernel architecture pattern (sometimes referred to as the plug-in architecture pattern) is a natural pattern for implementing product-based applications. A product-based application is one that is packaged and made available for download in versions as a typical third-party product. "

Many applications have a core set of operations that are used again and again in different patterns that depend upon the data and the task at hand. The popular development tool Eclipse, for instance, will open files, annotate them, edit them, and start up background processors. The tool is famous for doing all of these jobs. In this case, the basic routines for displaying a file and editing it are part of the microkernel. However, many companies also develop and release their internal business applications like software products, complete with versions, release notes, and

pluggable features. These are also a natural fit for this pattern. The microkernel architecture pattern allows you to add additional application features as plug-ins to the core application, providing extensibility as well as feature separation and isolation.

## Pattern Description:

The microkernel architecture pattern consists of two types of architecture components: a core system and plug-in modules. Application logic is divided between independent plug-in modules and the basic core system, providing extensibility, flexibility, and isolation of application features and custom processing logic.




### ❖ PLUG-IN COMPONENT:

The plug-in modules are stand-alone, independent components that contain specialized processing, additional features, and custom code that are meant to enhance or extend the core system to produce additional business capabilities. Generally, plug-in modules should be independent of other plug-in modules, but you can certainly design plug-ins that requires other plug-ins to be present. Either way, it is important to keep the communication between plug-ins to a minimum to avoid dependency issues.

### ❖ CORE SYSTEM:

The core system of the microkernel architecture pattern traditionally contains only the minimal functionality required to make the system operational. Many operating systems implement the microkernel architecture pattern, hence the origin of this pattern's name. From a business-

application perspective, the core system is often defined as the general business logic sans custom code for special cases, special rules, or complex conditional processing.

- **CONNECTION BETWEEN PLUG-IN COMPONENT AND CORE SYSTEM:**

Plug-in modules can be connected to the core system through a variety of ways, including OSGi (open service gateway initiative), messaging, web services, or even direct point-to-point binding (i.e., object instantiation). The type of connection you use depends on the type of application you are building (small product or large business application) and your specific needs (e.g., single deploy or distributed deployment). The architecture pattern itself does not specify any of these implementation details, only that the plug-in modules must remain independent from one another.

**The End**