CASE STUDY

Aspose.Total for .NET Competitive Upgrade Case Study

ContinueLink Health Technologies, LLC

Home Care Operations System
Satish Movva, 26 February, 2010

Product Background/Overview

ContinuLink Health Technologies, LLC, is a recent spinoff of a healthcare company in the US.

ContinuLink has built the premier operations platform for Home health care, hospice, private duty and supplemental staffing industries. The product encompasses both operational and clinical components of a home care services agency including smart phone access and point of care devices. It's a fully web bases SaaS platform and is a pioneer in the industry. The product encompasses over 6 million lines of code and is 100% Microsoft ASP.net. The product has been in continuous use since 2004 and is used by over 300 home care services provides agencies in the United States.

The product allows a home care services agency to manage all agency operations including client intake, payer setup, scheduling, billing, payroll, payment posting and collections. In addition, the product fully supports all clinical functions for Medicare, Medicaid and commercial insurance for the provision home healthcare services including assessments, plans of care, medication profiles and interactions analysis and documentation of nurse visit notes.

The product is used by agency personnel to manage day to day operations and by physicians to view and manage their parents' care progress and by nurses to provide the care at the patients' home.

Requirements Scenario

We have a requirement to perform a server side stitch of many clinical records of a patient into a single PDF arranged in chronological order. The clinical records can be composed of PDFs, live data from databases formatted into a PDF form, scanned images Word and Excel documents, electronic faxes in TIFF format and clinical wound pictures. All the documents have to be converted on the server side on a high performance and reliable manner with no server performance degradation. There should be no requirement to install Microsoft Office or other client side products onto the servers to enable conversion of Microsoft Office documents as that typically results in unstable server environments.

Solution Implementation

Documents are either uploaded or scanned into our system and stored in a Microsoft SQL database as BLOBs. The documents can be in Word, Excel, PowerPoint, JPEG, BMP, PNG, TIFF formats among others.

We previously used a combination of libraries for stitching PDF documents together, converting images to PDFs and filling out PDF documents on government mandated forms. While some worked well for conversion of image formats we ran into a problem with conversion of Microsoft Office documents. The library supported conversion if openoffice.org is installed or the Microsoft Office Libraries are installed on the servers, the use of these client side tools on the server led to server instability due to memory leaks as well as performance degradation on the servers. Since these were front end server that also served a large n-tier application, this caused a slowdown of the entire system when one user converted documents.

We used Aspose.Words for .NET to convert Word documents to PDF. We implemented the conversion code into our system very quickly and easily.

Benefits

The product lets us use one product for all of our PDF conversion and manipulation needs, including stitching multiple documents together, barcodes and water marking. It's one product to learn and use seamlessly in Visual Studio.

The Aspose product is high performance and reliable. It does not require Microsoft Office, OpenOffice or other client type applications to be installed on the server to enable access to Microsoft Office documents and their manipulation. We now have the ability to manipulate Microsoft Office documents and to support new functionality such as mail merge that we weren't capable of before. This allows our users to have greater productivity and make more efficient use of our system.

In addition, we now expect to be able to run our application on 64-bit server platforms without enabling 32-bit compatibility mode as previous tools we used were restricted to 32-bit mode.

We now have no more issues of memory leakage due to instantiation of client type application and libraries on the servers when manipulating Microsoft Office documents on he server side.

Future Implementations

We plan to use Aspose.Pdf.Kit and Aspose.Pdf for PDF forms manipulation as well as document stitching so that we can use one product to do all tasks rather than two products from two different vendors.

We plan to use the rest of the features in the Aspose.Total for .NET product as they have opened up our horizons in new functionality we can add to our system. This included enabling ad hoc reports for end users using Aspose.Reports for .NET, using network programming features for automating submission of files to government agencies, using Flask programming for a service area visualization tool using GIS data, using barcode generation and detection for auto indexing of documents into our built-in document management system.

These new features will substantially increase the value of our system to our users and provide us a competitive edge.

Conclusion

We are very glad to have found Aspose as an alternative to using multiple different products from different vendors in our need for document conversion and document manipulation on the server side in a high performance large scale n-tier SaaS platform.

Figure 1: A page where the user selects all possible documents for a patient.

Figure 2: A filled out bill sent to the patient.

Contact

For more information on this case study and to request details on the customer's previous supplier please [contact us](#).

Aspose.Total for .NET Competitive Upgrade
Case Study