Sollutium

Complex web solutions, API design for mobile apps.
.NET, ASP.NET MVC, WCF, NodeJS, AngularJS

What we do

- Web-apps for business, API
- Development and consulting for optimization of architecture of business apps
- E-Library systems
- Document flow systems, CRM, ERP
- Consulting for implementing the system of corporate search Sphinx

Some of our projects

DVS

On-line library with full text search and secure documents view

DVS – web application for processing library collections. With DVS it is possible to find a book or any other document in 20 TB storage. And to view it online afterwards and order printing or recording of relevant document fragments without leaving the browser.

What we've done

- business process audit
- integration with existing systems
- processing of large data amounts
- customer support, complete documentation on implementation, on administration, on the use

Used languages, technologies and tools

- **operation system:** Windows Server 2008 R2
- **server side:** C#, .NET Framework 4.0, IIS 7.0, NLB Cluster, EntityFramework, WCF, REST, MVC, XML, XSLT, JSON
- **web interface:** HTML, Razor, Twitter Bootstrap, JavaScript
- **document viewer:** framework Adobe Flex, language Action Script 3
- **data storage:** MS SQL Server, AppFabric Cache for often used data
- **processing of documents:** SwfTools, completion tools with c++

"Sollutium", http://sollutium.com
Chernihiv, Ukraine

CEO: Igor Kyslyi
igor.kyslyi@sollutium.com

Sollutium

Complex web solutions, API design for mobile apps.
.NET, ASP.NET MVC, WCF, NodeJS, AngularJS

Virtual Hospital

web application for monitoring of patients condition by physicians and their interaction

What we've done

- creating of a MVP (minimum viable product)
- teamwork
- API reuse

Used languages, technologies and tools

- **language:** javascript
- **framework:** AngularJS
- **build system:** grunt
- **tests:** jasmine, karma, protractor
- **package manager:** bower, npm
- **external services:** google maps

For data requiring real-time update SSE (server sent events) is used SSE (server sent events).

As the entire web interface is a set of static files, it was decided to separate it from the API server. CORS is used for cross-domain interaction.

FTS (Sphinx search)

solution for indexing and full text search in storage with more than 20Tb of documents

FTS – Web application for search by digitized books and dissertations. Allows searching not only by bibliographic data, but also through the whole document text. The application processes large text documents, in particular scanned and recognized, including low-quality old scanned copies.

What we've done

- full-text search for large amounts of data (>300Gb of text)
- morphology support
- rapid changes in search base data
- combining search implementations
- documentation for developers

CEO: Igor Kyslyi
igor.kyslyi@sollutium.com

Sollutium

Complex web solutions, API design for mobile apps.
.NET, ASP.NET MVC, WCF, NodeJS, AngularJS

Used languages, technologies and tools

- **operation system:** Ubuntu x64
- **search server:** Sphinx search
- **indexer:** c#, mono runtime.
- **web interface:** programming language php, using php-fpm process manager
- **data storage:** MySQL as primary storage, Redis for frequently used data

The specificity of the input data (recognized documents) complicated both indexing and results formation. Our team has developed the algorithm of preliminary text filter that removes characters that are “garbage”. Since filtering was built on the set of rules we together with customer’s representatives have formed the set of test queries to control each of them. This allowed us to remove the excess without affecting the document body, even if the word was recognized with an error.

During the project implementation a few errors in the use of open source solutions were found. We have implemented patches for some of them to fix errors in our environment.

Secure Document Viewer
control for in-browser viewing of documents

SDViewer - embedded component of secure online document view. Allows viewing the document to which you have access without leaving the browser. Embedded on the customers’ site and supports protection of documents from unauthorized downloading.

What we’ve done

- audit of protected documents display options
- development of embedded solution
- development of flexible architecture (1 server, many servers, cloud)
- on-the-fly documents processing
- documentation for developers on deployment and embedding

Used languages, technologies and tools

- **operation system:** Ubuntu x64
- **conversion utilities:** SwfTools, poppler
- **built-in view component:** Платформа Adobe Flex, язык ActionScript3
- **web server:** Nginx
- **application server for conversion service:** UWSGI
- **conversion service:** язык Python, фреймворк bottle
- **load balancing:** HA Proxy
- **deployment in AWS cloud infrastructure:** SaltStack