

# Scanning Book Services Application

*Client Company is a product Development Company from Nederland, leader in Accounting and Tax management solutions for SME's and Accounting Firms. The company's core product offering is a fully integrated suite of applications, including Online document scanning and uploading, Document workflow based on document type and BPO Management.*

## The Vision

The Project Scanning Book Services application is developed to achieve scanning and uploading of documents (Single or in Batch), integration with existing accounting systems and interface to other third party applications.

## Application Overview

SBS provides B2B solution for SME's and Accounting Firms. It also provide SDK interface to third party systems like Account View system to integrate with Scanning Book Services (SBS).

## Desired features:

The following major features were desired in the Screen Scraping Application:

- Online single and multipage scanning of documents using TWAIN drivers in Portal as well as SDK support.
- User authentication and authorization in Portal as well as Software Development Kit (SDK) support.
- Ability to control user access on document type and functionality level
- Document workflow automatic and manual management with the help of Financial Process Outsourcing (FPO).
- Image improvement, processing, compression and OCRing.
- Complex filter support to view required business information by applying data filters.
- Rich internet application web user interface using EXT – JS.

- Impressive portal implementation to support User registration, password reminder facility, document preview with transaction details, document process queue management, view documents as per process and types, filters on each grid with pagination, searching sorting, Manage Subscriptions, organizations, users, ledger s and sectors.
- Email communication support for various business flow requirement with FPO, SME and Accounting forms.
- Scheduler support to perform automatic scheduling of document improvement, OCRing and compression.
- User state persistence of last view, operations or preferences.
- Multi theme support to select user preferred theme.
- Multi- language and multi-currency support for portal.
- Inter-FPO communication module based on document support.
- Ability to search and fetch organization information from global repository to organizations.
- Web service SDK support for all important modules.
- Global financial standard, XBRL (eXtensible Business Reporting Language) was used for sending invoices to third party system.

## Technologies and Tools

J2EE Architecture was followed as following technologies exactly fit in our development needs. It provides following benefits:

- Large number of packages, tools, techniques, design frameworks, IDEs, testing methodologies gives the architectural flexibility
- Simplicity and long term viability
- Portability and Scalability
- Application Security
- Large number of deployment support tools

## Technologies

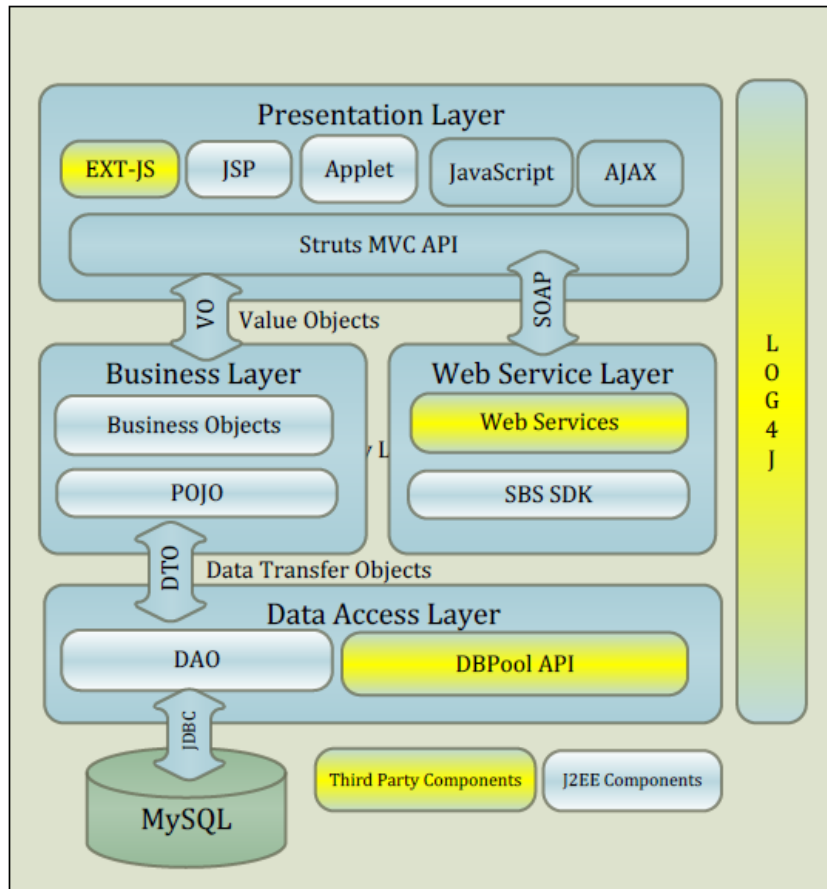
### **Presentation Layer:**

- Ext JS: Ext JS library for providing rich user interface, client has purchased licenses based on our recommendation. Ext JS is a cross -browser JavaScript library to build rich internet applications and in it customizable UI widgets and extensible component model.

In our project we have used following Ext JS components

- Ext JS Editable Grid
  - Ext JS tab structure screens
  - Ext JS Tree
  - Ext JS Toolbar
- JSP: All presentation components are written as Java Server Pages (JSP). These pages are basically HTML pages with embedded JSP functionality. These pages are free to contain JavaScript or any other front-end scripting tool.

- Applet: To scan document in the web page using TWAIN driver, this applet also supports multipage scanning, image rotating and resizing.



### **Business Logic:**

- JAVA: All business logic and back-end processing is written in Java. JavaBeans are used wherever appropriate to enforce data encapsulation. In general a JavaBean contains both the data and the business logic for each entity.
- Struts: Struts application development framework is used to enforce the Model -View- Controller (MVC) approach to development. This approach allows for the separation of presentation and business logic.
- JSON: JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

### **Web Service Layer:**

- Apache AXIS Web Services: To take an advantage of decoupling principle of best design practices and provide easy simple SDK interface to third party system web service layer is introduced using AXIS API.

### ***Data Access Layer:***

- JDBC: Data Access layer using JDBC
- DB Pool: A Java-based database connection pooling utility, which supports time-based expiry and statement caching, connection validation, and easy configuration using a pool manager. Also included is a generic object pool which can be extended to create your own pools of custom types.

### ***Backend:***

- MySQL 5.1

### ***Browser Support:***

- IE 7.0 and above
- Firefox 3.0 and above
- Safari and Google Chrome

### ***Web server:***

- Tomcat

### ***Supported Deployment Platforms:***

- Windows 2000
- Windows 2003

### ***Tools***

- Eclipse: Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. This is very helpful for rapid application development.
- PMD: Automatic code review product is used to minimize code review time and improve development efficiency and reduce defect density per code lines.
- ER Studio and SQLyog: These tools improve the productivity of developers and DBAs by providing an easy-to-use development environment. Using this tool we have managed MySQL database.
- Ant: We have used Ant to automate building application. It facilitates compiling, packaging, and preparing programs for execution.
- VietOCR/ABBYY API: This API's are used to OCR document and extract image data into textual format to analyze document type and accounting purpose information.

## ***Challenges and Solutions***

1. Tightly constrained project schedule

- a. e-Zest had provided accurate estimations and followed highly disciplined project management practices.
- b. e-Zest had provided onsite resources to speed up the requirement gathering process and QA support
- c. e-Zest had provided complete transparency to customer which helped for better reporting, monitoring and controlling of project execution
- d. Look and feel was made consistent with client's other products e. Following are the challenges
  - i. Using ExtJS Technology for UI Development
  - ii. Designing reusable UI components to avoid rework

## Solution Provided

See below some screen shots where e-Zest had made User interface more intuitive and consistent. See some complex screens below, we have made them users friendly. UI shown in different themes.

- Scan/Upload Screen

File Name	Subscription Name	Document Type	Date/Time	File Size
doc.jpg	gggtyst	Purchase Invoice	27/04/2010 11:43	387400
doc.jpg	gggtyst	Purchase Invoice	27/04/2010 11:41	971019
doc.jpg	gggtyst	Purchase Invoice	27/04/2010 11:40	971019
doc.jpg	gggtyst	Purchase Invoice	27/04/2010 11:38	971019

- Subscription Screen

The screenshot displays the 'Subscriptions' management interface. At the top, there's a navigation bar with 'Home', 'Dinesh', 'Settings', 'Theme', 'Language', 'Sign Out', and 'About'. A sidebar on the left contains 'SBS Menu' with options for 'Subscriptions' and 'Users'. The main area features a table of subscriptions with columns: Name, Internal ID, External ID, Address, Postal Code, City, Phone Number, Email Address, and Activated. Below the table, there are 'Edit' and 'View User' buttons. The 'Edit' form includes fields for 'Subscription Name', 'Internal ID', 'External ID', 'Accounting Firm', 'Address', 'Postal Address', 'Communication', 'Reference', 'Bank', 'Subscription Type', 'Activity', and 'Info'. The 'Address' section is expanded, showing fields for 'Address', 'Postal Code', 'City', 'State', and 'Country' (set to Denmark). 'Save' and 'Reset' buttons are at the bottom right.

Name	Internal ID	External ID	Address	Postal Code	City	Phone Number	Email Address	Activated
ffffff	sdsdsd	dsdsd	dasdasd	q3e33223	weqeweqwe	234234234	sdds@ddd.com	22/04/2010
gggtyst	gggtyst	q32242edasdas	sadasdgg	wqeweweqw	sdaasdasd	rwerwerw	23sdfs1@dfdf.in	16/04/2010
sub3	6666666	666666	trtret	42432424	dfgfdg	576575	sas@dfjd.in	26/04/2010

- Transaction panel(Document) View

The screenshot shows the 'Invoice' document view. The top navigation bar is identical to the previous screen. The sidebar on the left includes 'SBS Menu' with 'Subscriptions', 'Documents', 'Scan/Upload', 'Process', 'Vouchers (73)', 'Statements (46)', 'Rejected', and 'Archived'. The main area displays the 'kpn' logo and address: 'Postbus 16566, 2500 BR Den Haag'. To the right, there's a form for document details including 'Date', 'Number', 'Organization', 'Reference', 'Currency', 'Ledger Number', 'Description', 'Net Amount', 'VAT Percentage', 'Net Amount', 'VAT Amount', and 'Gross Amount'. Below this, a 'Documents' table lists documents with columns: ID, Subscription Name, Document Type, Invoice Number, Reference, Date, and Alert. A 'Process Log' table on the right shows a log of actions with columns: Code, Description, Date, and Time.

ID	Subscription Name	Document Type	Invoice Number	Reference	Date	Alert
64	sub3	Purchase Invoice	3339	SADA58	16/04/2010	
67	ffffff	Purchase Invoice				
79	sub4	Purchase Invoice				
84	gggtyst	Purchase Invoice				

This screenshot shows the 'Documents' list view. The top navigation bar and sidebar are consistent with the previous screens. The main area displays a table of documents with columns: ID, Subscription Name, Document Type, and Alert. A 'Process Log' table on the right shows a log of actions with columns: Code, Description, Date, and Time.

ID	Subscription Name	Document Type	Alert
76	sub3	Bank Statement	
77	sub3	Bank Statement	
78	sub3	Cash Statement	
101	sub3	Bank Statement	
102	sub3	Bank Statement	
103	sub3	Bank Statement	
107	sub3	Bank Statement	
116	sub4	Bank Statement	
119	sub4	Bank Statement	
120	sub4	Bank Statement	
121	sub4	Bank Statement	
122	sub4	Bank Statement	
123	sub4	Bank Statement	
124	sub4	Bank Statement	
130	sub4	Cash Statement	
132	sub4	Cash Statement	

- User's Screen

Name	Email Address	Phone Number	Created Date
Abhijeet g ahd	ha@zf.com		14/04/2010
adssa aa	asd@asd.com		14/04/2010
Amol Godbole	dsfdfs@adf.com	32324243	12/04/2010
asd ads gg	ad@g.cp		15/04/2010
assad asda asds	ga@ad.com		14/04/2010
thigh ghjg	tt@tt.com	6767676767	14/04/2010
Bob Colin	testuser@gmail.com	023202333	12/04/2010
dffdsff adsf	sdfhg@saf.cpo		14/04/2010
dfsf dsfdfsdf dsfdfsdf	hadfdfs@assd.com		14/04/2010
Dinesh P Sonsale	sonsale@gmail.com	2323232323	14/04/2010
dsg sdg hs	gh@af.com		15/04/2010
gg ha gag	hga@ad.com		20/04/2010
gha ha jhs	haj@ad.com		15/04/2010
Harry(FPO) Viet	fpouser@yob.nl	0203223233	12/04/2010
Harshal Shripad Ambekar	harshal.ambekar@e-zest.in	66565757	21/04/2010
Hello aaa Kello	sa@ads.com	324232	12/04/2010
hh jk s	hj@ac.com		15/04/2010

- Organization Screen

Name	Address	Postal Code	City	Telephone	Website
4999	dfgd				
Hello	Address1	12345678	Pune	9898989898	www.google.com
namedsadsadadsadasadasda	addr	68768	PN	9890	http://www.google.co
Org e-zest	S addr	S P code	S city	8888888888	http://www.dosco.cc

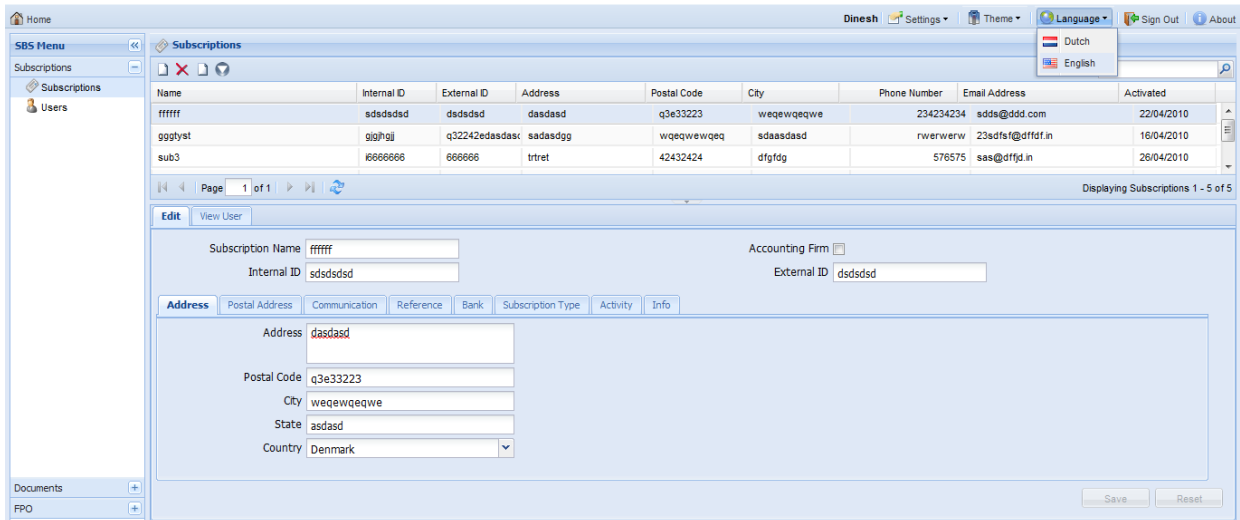
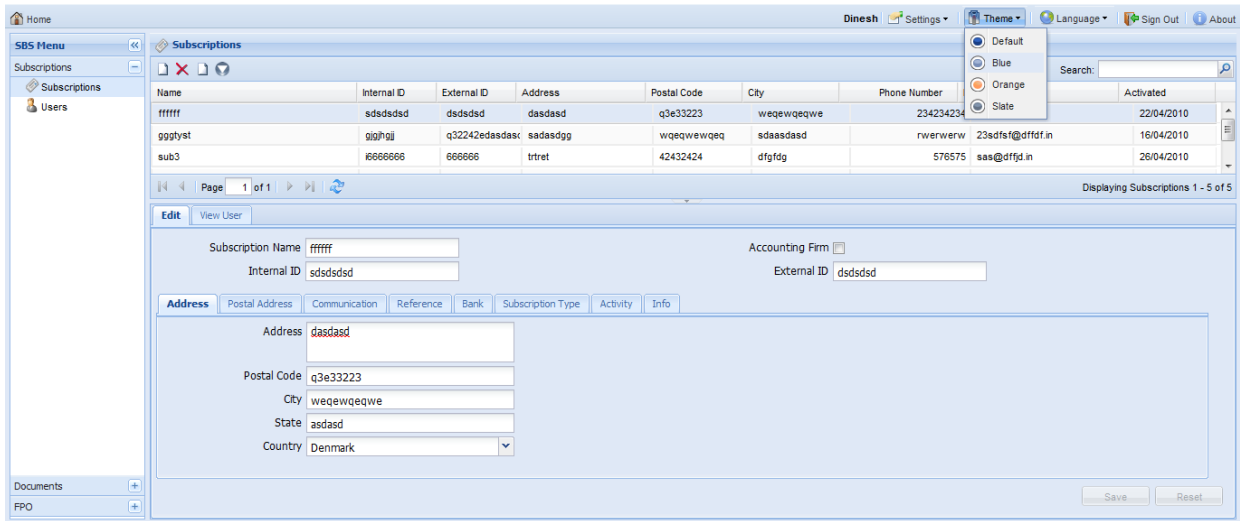
Code	Description
56768	Akkerbouw
56769	Varkenshouderij
56770	70 desc again
56771	Textiel- en lederwarenindustrie
56776	Elektrische apparaten
56778	Bouwnijverheid
56779	Detailhandel food
56783	Zakelijke adviesbureaus

Code:

Description:

Save Reset

- Theme and Language selection options Screens



To assure consistent look and feel through the application, we used Template based approached.

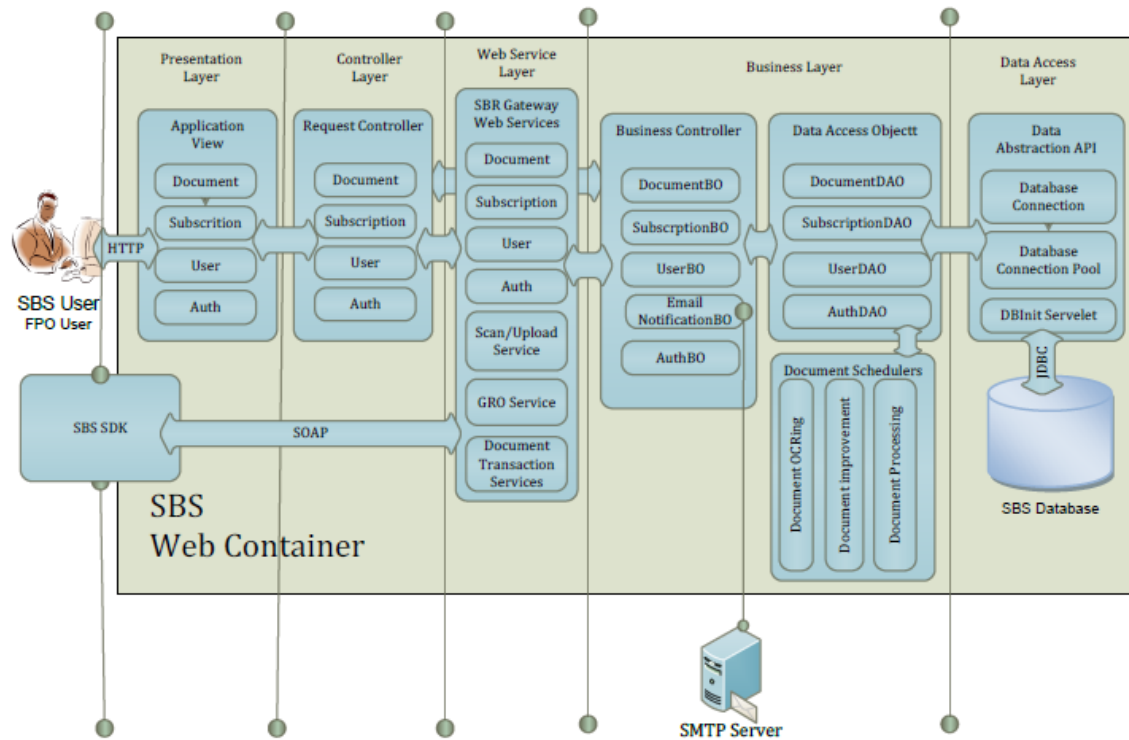
Team had grasped ExtJS very fast and prepared some sample which is ready to use for other team members.

## Solution Architecture

The SBS follows the MVC (Model View Controller) approach to development. This approach separates all of the presentation logic, business logic, and application logic. At e-Zest we used Struts framework from the Apache Software Foundation to enforce this approach. The front-end of the application is written in JSP with ExtJS components and Applet. In addition, JavaScript is used where necessary. The server-side of the application is written using Java 1.6 Standard Edition (J2 SE) and components from Java 2 Enterprise Edition (J2EE). Java Beans are used to store the data for the applications. e-Zest evaluated five open sources and paid OCRing tools and recommended VietOCR APIs for OCRing of applications. The drawing



below illustrates the general structure of the application and the division between the model, view, and controller components.



## Technical Breakthroughs

- Effectively used ExtJS to build rich user interface.
- Web based document Scanning solution using TWAIN Drivers
- OCRing and Image improvement solutions
- BPO support for Document flow management.
- Code optimization to improve auto schedule process performance
- Code consolidation for constraints, configuration rules and preferences so that user can get consistent results.
- Data dictionary implementation to easy support for customer product version migration and product upgrade

## Bottom Line

e-Zest out performed in the project evolution phase carried out by the client at the start of the project to analyze the capabilities in terms of commitments and work quality. This helped to build a strong and long term relationship with the client. Enterprise level product development with in-depth understanding of client's business focus, needs and constraints and complete sense of ownership of project by each team member. This resulted in a mutually profitable and constantly growing relationship bringing in business stability to both client and e-Zest.