May 12, 2015

INDIGO SIGNATURE SERVICE INSTALLATION GUIDE

VERSION 1.0

GLOBAL HEADQUARTERS

41 University Drive, Suite 202
Newtown, PA 18940 USA

EU HEADQUARTERS

Corvin Offices I. Futó Street 47-53
Budapest, H-1082, Hungary

CIS HEADQUARTERS

9th Radialnaya Street, Building 2
Moscow, 115404, Russia

CONTENTS

1. REQUIRED SOFTWARE 1
1.1. Database Server 1
1.2. Application Server 1
1.3. Build Machine 1
1.4. Client Machine 1
2. REQUIRED HARDWARE 1
2.1. Database Server 1
2.2. Application Server 1
3. INSTALLATION 2
3.1. Prerequisites 2
3.2. Installing Package 2
4. CONFIGURATION 3
4.1. Application Settings 3
4.2. Archiving Settings (SFTP) 4
4.3. Parameters Stored in Database 4
4.4. Administrator User Account 5
4.5. Integration with Indigo ELN 5
5. UNINSTALLATION 5
5.1. Undeploying WAR Package 5
5.2. Removing Database Schema 5
6. REST SERVICES API 5

1. REQUIRED SOFTWARE

1.1. DATABASE SERVER

Indigo Signature Service requires one of the following databases:

- Oracle Database 10+;
- Oracle Database Express Edition 10+
- PostgreSQL 9.2

1.2. APPLICATION SERVER

The application server should have the following components installed:

- Java SE 1.6
- JBoss Application Server 7.1.1 OR Apache Tomcat 6+

1.3. BUILD MACHINE

The build machine should have the following components installed:

- Java SE 1.6
- Apache Maven 2+

1.4. CLIENT MACHINE

The only requirement to the client machine is one of the following browsers installed:

- Internet Explorer 9+;
- Google Chrome;
- Mozilla Firefox

2. REQUIRED HARDWARE

2.1. DATABASE SERVER

Component	Supported configuration
Processor	Intel IA-32, Intel EM64T (Xeon), or AMD64 (Opteron) 1 GHz or higher Tip: A faster processor delivers faster performance.
RAM	1 GB or higher Tip: More RAM provides faster performance.
Hard disk space	10 GB
Network	Ethernet

2.2. APPLICATION SERVER

Component	Supported configuration
Processor	Intel IA-32, Intel EM64T (Xeon), or AMD64 (Opteron) 1 GHz or higher Tip: A faster processor delivers faster performance.

Component	Supported configuration
RAM	1 GB or higher Tip: More RAM provides faster performance.
Hard disk space	1 GB
Network	Ethernet

3. INSTALLATION

3.1. PREREQUISITES

Before installing the Service, make sure you meet the following prerequisites:

- Ensure that PostgreSQL and JBoss (Tomcat) servers are up and you have write access to the JBoss folder.
- Check the presence of the following files in the system folder: **mvn**, **java**, **javac**, **sqlplus** (for Oracle), **psql** (for PostgreSQL).

3.2. INSTALLING PACKAGE

To install the package:

1. Unzip the archive to any location. All further instructions in this document refer to subfolders of the location of the extracted archive.
2. Configure Database properties as explained in Section 4.1.
3. Install the database schema. You need to execute the following command in the **database** folder:

For Oracle:

```
$ sqlplus sys/admin@//localhost:1521/xe as sysdba @oracle/schema_install.sql
```

For PostgreSQL:

```
$ psql -h localhost -p 5432 -d postgres -U postgres -f
postgresql/schema_install.sql
```

4. Configure Application properties as explained in Section 4.1.
5. Build the package:

```
$ mvn clean package
```

6. Configure parameters stored in the database as explained in Section 4.3.
7. Deploy the package by copying **target/signatureservice.war** to the JBoss deployments folder or deploying it to Tomcat.

4. CONFIGURATION

4.1. APPLICATION SETTINGS

The **user.properties** file contains the main application settings, see the following table. In the main application settings, specify the signature method to use (local by default).

Database settings	
database-connection-url	Connection string for the database
database-user	Database user for Signature service
database-password	Database password for Signature service
database-driver-class	Driver to use for database connection
Mail settings	
Application sends emails to users in order to notify them about a new document in the queue	
mail-from	Email address for “From” field
mail-username	Username for the email service
mail-password	Password for the email service
mail-smtp-host	Email service host
mail-smtp-port	Email service port
mail-auth	Email service requires authentication
mail-ssl	Email service requires SSL
mail-tls	Email service requires TLS
signature-service-address	Address of the server hosting Signature Service. The Service sends this address to users in "Request to sign document" messages.
Archiving settings	
final-status	Defines the final status of a document: SIGNED – the document has reached the SIGNED status, no more actions are required. ARCHIVED - the document has reached the SIGNED status and has passed to the archiving procedure. When it is done, the document gets the ARCHIVED status. The archiving procedure is specified by the com.chemistry.enotebook.signature.archiver.Archiver setting. See Section 4.2 for information.
com.chemistry.enotebook.signature.archiver.Archiver	Defines the full class name of the implementation of the Archiver interface. It is possible to add a new implementation of the archiving procedure and

	specify its class name here. In an archiver implementation, you can specify the desired location to store the copies of signed documents.
User interface settings	
is-upload-documents-allowed	Allows uploading documents from the Signature Service UI true – UI shows the Upload New Document button; false - UI does not show the Upload New Document button.
Signing method	
signing-method	Defines the signing method to use. You can specify your own signing method and congifure Indigo Signature Service to use it here. Please contact EPAM if you are interested in the signing method implementation that uses the Verizon document signing service. default – use the local document signing service. Indigo Signature requests a certificate for the signature and the password for this certificate. After that, Indigo Signature locally signs the document with the certificate provided.

4.2. ARCHIVING SETTINGS (SFTP)

Indigo Signature Service contains a default Archiver implementation that places signed documents to an SFTP file share. The **sftp.properties** file contains connection settings and a configuration for the uploading schedule. The Service places all archived documents in a queue and transfers them to the SFTP file share according to the schedule.

> ***Note***
> *Indigo Signature Service performs the archiving step only when the final status setting in **user.properties** is **ARCHIVED**. In case the value is **SIGNED**, Indigo Signature Service skips the archiving step.*

Setting	Description
sftp.host	Host for the sftp service
sftp.port	Port for the sftp service
sftp.password	Password for the sftp service
sftp.user	Username for the sftp service
sftp.timeout	Connection timeout
sftp.remote.dir	Folder on the sftp server
poller.cron	Configuration for the uploading schedule

4.3. PARAMETERS STORED IN DATABASE

In Signature Service, when users sign documents, they select a reason for signing. By default, there are two reasons predefined in the database: “I am the Author” and “I am the Witness”. To configure the list of possible reasons for signing documents, make the necessary changes in the **Reason** table of the database.

4.4. ADMINISTRATOR USER ACCOUNT

Creating the database schema, see Section 4.1, automatically creates a default user with Administrator permissions and with the password: **1234**. The Administrator user can create other users in the system.

After the installation of the Service is complete, you must necessarily change the password of the Admin user. See the Indigo Signature Service User Guide.

4.5. INTEGRATION WITH INDIGO ELN

To enable users to sign experiments from inside Indigo ELN, in Signature Service, the Administrator user should create user accounts for those users who need to have this ability.

5. UNINSTALLATION

5.1. UNDEPLOYING WAR PACKAGE

To undeploy the package, you need to remove the .war file from the JBoss deployments folder or undeploy the package from Tomcat.

5.2. REMOVING DATABASE SCHEMA

If you need to remove all information that the services wrote to the database, you can run the following command:

For Oracle:

```
$ sqlplus sys/admin@//localhost:1521/xe as sysdba
@oracle/schema_uninstall.sql
```

For PostgreSQL:

```
$ psql -h localhost -p 5432 -d postgres -U postgres -f
postgresql/schema_uninstall.sql
```

Be careful! This command removes all data from the database, including all uploaded compounds.

6. REST SERVICES API

Signature Service provides a possibility to integrate with other systems. For this purpose, Indigo Signature Service provides an API, containing the following commands:

URL	Http method	Parameters list	Format of result	Description
/loginProcess	POST	username, password	Text	Log in to Signature Service
/api/downloadDocument	GET	document id, session id	Binary	Get document content by ID
/api/getDocumentInfo	GET	document id, session id	JSON	Get document meta-information by ID
/api/getDocuments	GET	username, session id	JSON	Get all the documents to which the user has access

URL	Http method	Parameters list	Format of result	Description
/api/getDocumentsByIds	POST	list of document ids, session id	JSON	Get all the documents that have certain IDs
/api/getFinalStatus	GET	session id	Text	Get signature service property finalStatus
/api/getStatuses	GET	session id	JSON	Get the list of existing statuses
/api/uploadDocument	POST	username, template id, document file, session id	JSON	Submit a new document to Signature Service
/api/getReasons	GET	session id	JSON	Get the list of available reasons for signature. Note: The complete set of reasons is stored in the Reason table in the database.
/api/getTemplates	GET	username, session id	JSON	Get the list of templates available for a given user

Note
EPAM implemented Signature Service integration in Indigo ELN 1.1.3 that uses the Indigo Signature Service API.