

|  |
| --- |
| Design Specification  Indigo ELN v. 2.0 |
|  |

| **Related Artifacts** | |
| --- | --- |
| **Ref.** | **Name** |
| N/A | N/A |

|  |  |
| --- | --- |
| **Abbreviations and Acronyms** | |
| **ACL** | Access Control List |
| **ELN** | Electronic Lab Notebook |

Contents

[1 Introduction 4](#_Toc498451518)

[2 Hardware and software requirements 4](#_Toc498451519)

[2.1 Database tier 4](#_Toc498451520)

[2.1.1 Database Server Hardware Requirements 4](#_Toc498451521)

[2.1.2 Database Server Software Requirements 4](#_Toc498451522)

[2.2 Application Server 4](#_Toc498451523)

[2.2.1 Application Server Hardware Requirements 4](#_Toc498451524)

[2.2.2 Application Server Software Requirements 5](#_Toc498451525)

[2.3 Client Tier 5](#_Toc498451526)

[2.3.1 Client Tier Software Requirements 5](#_Toc498451527)

[2.4 Development Environment Requirements 5](#_Toc498451528)

[3 Installation Overview 5](#_Toc498451529)

[4 Installing BingoDB 5](#_Toc498451530)

[4.1 Build from sources 5](#_Toc498451531)

[4.2 Configuration 6](#_Toc498451532)

[4.3 Run without installation 6](#_Toc498451533)

[4.4 Installation 6](#_Toc498451534)

[4.4.1 Linux 6](#_Toc498451535)

[4.4.2 Windows 6](#_Toc498451536)

[5 Installing Indigo ELN Database 6](#_Toc498451537)

[5.1 Installing Indigo ELN Database 6](#_Toc498451538)

[6 Installing Indigo ELN 7](#_Toc498451539)

[6.1 Build from sources 7](#_Toc498451540)

[6.1.1 Server 7](#_Toc498451541)

[6.1.2 UI 7](#_Toc498451542)

[6.2 Configuration 7](#_Toc498451543)

[6.2.1 Server 7](#_Toc498451544)

[6.2.2 UI 8](#_Toc498451545)

[6.3 Run without installation 8](#_Toc498451546)

[6.3.1 Server 8](#_Toc498451547)

[6.3.2 UI 8](#_Toc498451548)

[6.4 Installation 9](#_Toc498451549)

[6.4.1 Server 9](#_Toc498451550)

[6.4.1.1 Linux 9](#_Toc498451551)

[6.4.1.2 Windows 9](#_Toc498451552)

[6.4.2 UI 9](#_Toc498451553)

# Introduction

This document provides installation details of EPAM Indigo ELN. Indigo ELN is an open-source product.

# Hardware and software requirements

## Database tier

### Database Server Hardware Requirements

|  |  |  |
| --- | --- | --- |
| **Component** | **Linux computer:**  **Supported Configuration** | **Windows computer:**  **Supported Configuration** |
| Processor | x86/x86\_64 processor, at least two cores  Tip: A faster processor delivers better performance. | x86/x86\_64 processor, at least two cores  Tip: A faster processor delivers better performance. |
| RAM | 2GB or higher  Tip: More RAM provides better performance. | 2GB or higher  Tip: More RAM provides better performance. |
| Hard disk space | 2GB of free space | 2GB of free space |

### Database Server Software Requirements

|  |  |
| --- | --- |
| **Linux computer:**  **Supported configuration** | **Windows computer:**  **Supported configuration** |
| CentOS 7.3 or higher | Microsoft Windows Server 2008R2 or higher |

The following software **required** on the database tier:

|  |  |  |
| --- | --- | --- |
| **Product** | **Vendor** | **Supported versions** |
| MongoDB | MongoDB, Inc. | 3.4.17 or higher (tested with 3.4.17) |
| BingoDB | EPAM Systems, Inc. | 1. or higher |

**Note:** BingoDB is shipped with IndigoELN. See installation instructions below.

## Application Server

### Application Server Hardware Requirements

|  |  |  |
| --- | --- | --- |
| **Component** | **Linux computer:**  **Supported Configuration** | **Windows computer:**  **Supported Configuration** |
| Processor | x86/x86\_64 processor, at least two cores  Tip: A faster processor delivers better performance. | x86/x86\_64 processor, at least two cores  Tip: A faster processor delivers better performance. |
| RAM | 2GB or higher  Tip: More RAM provides better performance. | 2GB or higher  Tip: More RAM provides better performance. |
| Hard disk space | 2GB of free space | 2GB of free space |

### Application Server Software Requirements

|  |  |
| --- | --- |
| **Linux computer:**  **Supported configuration** | **Windows computer:**  **Supported configuration** |
| CentOS 7.3 or higher | Microsoft Windows Server 2008R2 or higher |

The following software must be installed on the computer on which the Indigo ELN application server is installed and on which further configuration occurs:

|  |  |  |  |
| --- | --- | --- | --- |
| **Software** | **Product** | **Vendor** | **Supported versions** |
| Environment for IndigoELN and BingoDB | Java Runtime Environment | Oracle/OpenJDK | 1.8 |

## Client Tier

### Client Tier Software Requirements

The following software required on the client tier:

|  |  |  |  |
| --- | --- | --- | --- |
| **Software** | **Product** | **Vendor** | **Supported versions** |
| Web Browser | Google Chrome | Google, Inc. | 61+ |

## Development Environment Requirements

The following software is required to customize Indigo ELN properly in the development environment:

|  |  |  |  |
| --- | --- | --- | --- |
| **Software** | **Vendor** | **Supported Versions** | **Comments** |
| Maven | Apache | 3.1+ | Required to build deployment units |
| JDK | Oracle/OpenJDK | 1.8+ | Environment for Maven |

# Installation Overview

To install and run Indigo ELN at your site:

1. Download the Indigo ELN sources.
2. Install the Indigo ELN database on the database server.
3. Install BingoDB on the database server.
4. (Optional) Customize Indigo ELN source code to use custom services.
5. Build Indigo ELN.
6. Deploy and configure the Indigo ELN application server.

The following sections explain these steps in detail.

# Installing BingoDB

## Build from sources

Execute the following command to build and produce **bingodb.war** package:

mvn clean package -P release

## Configuration

File called **application.properties** can be placed in the same directory with **bingodb.war**.

Common properties which can be placed into **application.properties** (if property is not specified, it will use default value specified here):

# folder where BingoDB will store its data files

bingo.folder=bingo

# port for rest api access

server.port=9999

# username for secured access to rest api

security.user.name=bingo

# password for secured access to rest api

security.user.password=pass

## Run without installation

Execute the following command:

java -jar bingodb.war

## Installation

### Linux

**bingodb.war** is an "executable" archive which means that it already contains necessary startup scripts for Linux OS. To install BingoDB on Linux, just execute the following command as root (or with sudo):

ln -s /path/to/bingodb.war /etc/init.d/bingodb

This command will register integrated startup scripts in the system.

After this, BingoDB can be started with

service bingodb start

or

systemctl start bingodb

Command depends on initialization system installed in Linux distributive.

### Windows

See [winsw project documentation](https://github.com/kohsuke/winsw/blob/master/doc/installation.md) for service installation on Windows.

# Installing Indigo ELN Database

## Installing Indigo ELN Database

There are several initialization scripts, they are managed by Mongobee framework.

These scripts will be applied incrementally during first launch and will perform following steps:

* Create required indexes on collections
* Create a role with all the permissions available in the system
* Create admin user with default password and assign the role to it
* Create all required dictionaries

**Note**: Before scripts initialization ensure that Mongo DB is installed and run. Also database “indigoeln” is created. Ensure that Mongo DB user exists otherwise create user with name “user” and password “pass”. For full list of required actions for Mongo DB connection set see section 6.2.1.

# Installing Indigo ELN

## Build from sources

### Server

Execute the following command in server sources to build and produce **indigoeln.war** package:

mvn clean package -P release

### UI

Execute the following commands in UI sources to build and produce UI files in **dist** folder:

npm install

npm run build

**Note:**env.apiUrl=http://server:port/api/ parameter specifies Indigo ELN API URL in case you are planning to run UI side on different servers. You can change this parameter in package.json.

**Available npm scripts:**

npm build - just build production version

npm start - start server with development version

npm start:prod - start server with production version

## Configuration

### Server

Database schema will be initialized at first application run.

File called **application.properties** can be placed in the same directory with **indigoeln.war**.

Common properties which can be placed into **application.properties** (if property is not specified, it will have default value specified here):

# URL for MongoDB

spring.data.mongodb.uri=mongodb://user:pass@localhost:27017/indigoeln

# URL for BingoDB api

bingodb.api-url=http://localhost:9999/api

# Username to access BingoDB api

bingodb.username=bingo

# Password to access BingoDB api

bingodb.password=pass

# Key for 'Remember me' http header

indigoeln.security.rememberme-key=secure\_key

# BCrypt hash for default 'admin' user's password

default-admin-password=use-bcrypted-pass-here

# [optional] Signature service URL

signatureservice.url=http://localhost/signatureservice

# [optional] Signature service username

signatureservice.username=user

# [optional] Signature service password

signatureservice.password=pass

# [optional] Search service URL

crs.search-service-url=http://localhost:9999/crs/service/SearchService

# [optional] Registration service URL

crs.registration-service-url=http://localhost:9999/crs/service/RegistrationService

# [optional] Registration and Search service username

crs.username=user

# [optional] Registration and Search service password

crs.password=pass

# [optional] Signature service URL for return to UI part

indigoeln.client.signatureservice.url=http://localhost/signatureservice

# [optional] Run profiles (specify **release,cors** if you are planning to run UI on different server

spring.profiles.active=@spring.profiles.active@

# [optional] Allowed cross origin (use in case you are specified **cors** in previous property

cors.origin=http://localhost:9000

**Note:** Indigo ELN disables cross origin by default. Use custom **spring.profiles.active** and **cors.origin** values only in case you are running UI on different servers.

In “**default-admin-password**” parameters use bcrypted value.

### UI

Indigo ELN UI doesn’t have specific configuration files. The only configuration is specifying **apiUrl** property at build procedure.

## Run without installation

### Server

Execute the following command:

java -jar indigoeln.war

### UI

Execute the following commands:

npm install

npm run start

**Note:** env.apiUrl=http://server:port/api/ parameter which specifies Indigo ELN API URL in case you are planning to run UI on different server. You can change it in package.json.

**Available npm scripts:**

npm build - just build production version

npm start - start server with development version

npm start:prod - start server with production version

## Installation

### Server

#### Linux

**indigoeln.war** is an "executable" archive which means that it already contains necessary startup scripts for Linux OS. To install Indigo ELN on Linux, just execute the following command as root (or with sudo):

ln -s /path/to/indigoeln.war /etc/init.d/indigoeln

This command will register integrated startup scripts in the system.

After this, Indigo ELN can be started with

service indigoeln start

or

systemctl start indigoeln

Command depends on initialization system installed in Linux distributive.

#### Windows

See [winsw project documentation](https://github.com/kohsuke/winsw/blob/master/doc/installation.md) for service installation on Windows.

### UI

Indigo ELN UI part can be deployed using two different ways:

1. Place UI files into **static** folder beside **indigoeln.war**. Indigo ELN server part with use its internal server to serve UI
2. Use your preferred static server to serve UI. In this case don’t forget to specify **apiUrl** property during UI build and configure server cross origin properties